Wave Repair User Manual

©2003, Clive Backham

1 What's in this Manual

This user manual is intended to provide an overview of what Wave Repair can do, but does not describe every available feature. The program's online help is not merely a repetition of this manual, and it goes into more depth about the various options.

2 Wave Repair - What It Is & What It Isn't

Wave Repair is an editor designed specifically for ultra-fine level alteration of the waveform shape in WAV files. It is a 32-bit Windows program that runs under Microsoft Windows 95, 98, ME, NT, 2000 and XP. It operates only on 16bit stereo WAV format files sampled at 44.1kHz or 48kHz. I developed Wave Repair for my own use while restoring music signals retrieved from vinyl records. It is specifically targeted at identifying and repairing the clicks, pops and ticks that inevitably plague vinyl replay. I wrote it because I was unable to find an affordable WAV file editor which performed this task in an acceptable manner.

Other editors which claim to perform declicking do so by algorithmic means, usually by performing some kind of spectral analysis. One major barrier to using these tools productively is that they tend to have a number of configuration parameters that can be tried, but the actual declicking is a slow process. The consequence of this is that the typical procedure is: set parameters; declick; listen to results; repeat again and again. Wave Repair includes a real-time declicking function, which allows the user to interactively adjust the click detection parameters while listening to the effects. In this way, the best settings for a particular section of waveform can be found quickly. Despite this facility, my own experience is that any kind of automatic approach often fails to produce acceptable results. This is probably due to the unpredictable nature of vinyl damage.

There are of course professional hardware and hardware/software systems that perform this task (such as CEDAR), but they are very expensive. I am unable to verify their performance since I do not have the funds to try them for myself, but assume that since the professional recording industry is enthusiastic about them, then they probably do a very good job. Therefore, if you are a professional with the funds to use these kinds of tools, Wave Repair is probably not worth your consideration.

I have come to the conclusion that the only reliable way to do high quality declicking on a budget is to actually repair the damage manually on an individual click basis. Some editors do allow the re-drawing of waveform shapes, but are usually somewhat clumsy in this respect, often restricting the zoom factors under which redrawing is possible. In contrast, Wave Repair allows the user to zoom in to any degree in both the time and amplitude domains, and to re-draw waveforms at any degree of zoom to any degree of precision using the mouse as though it were a pen. It also offers a

number of other methods for amending waveforms: interpolation, smoothing, muting, and replacement (from either elsewhere in the same file or from another file).

Wave Repair is not a general purpose digital audio editor. There are other shareware editors available which serve that purpose. I can recommend GoldWave and CoolEdit as two excellent examples. It is unlikely that Wave Repair could be used for all your WAV file editing requirements, but rather might be a useful addition to your toolkit.

Despite that fact that Wave Repair does have a real-time declicking function, for good results it remains primarily a manually operated system. The user scans through the waveforms listening and looking for damage, and fixes any found. (The main use of the automatic declicking is for cleaning up very quiet sections such as fade-outs which can contain thousands of tiny ticks and would be unbearably tedious to fix on an individual tick-by-tick basis). Consequently, using Wave Repair is a time-consuming business: listen to the music; home in on a click by scanning through the waveform; repair the damage; repeat for the next click. Performing a thorough clean up of an entire LP can easily take 10 hours or more of work.

2.1 Should I Use It?

Nice though it would be to sell lots of copies of Wave Repair, it's not in my long term interest for people to use it in inappropriate circumstances.

It will be of use to you if you want to be able to redraw music waveforms at a very fine level of precision. It also provides a number of other easy to use mechanisms for modifying waveforms in a manner that is appropriate for the typical types of damage exhibited by vinyl records. It is much easier to use for these particular tasks than any other shareware WAV file editor I know; in fact I only wrote it because other editors were difficult to use for this purpose.

- Do you want to use your PC to clean up recordings of vinyl records?

  Wave Repair is specifically targeted at removing clicks and pops from LPs. The primary mechanisms for this are manual manipulation of the waveform.

  It also has an automatic declicking function that sometimes works and sometimes doesn't, depending on the characteristics of the source material. Like all other automatic declicking software, Wave Repair's algorithms cannot always make a perfect job of detecting and repairing the largely unpredictable nature of LP damage. Many other packages make bold statements about their capabilities, but my own experience is that they often fail to achieve acceptable results. I acknowledge that the same is true of Wave Repair's automatic declicking.

- Do you want to get really good results when removing clicks and pops?

  If you're not too bothered about the quality of the results, it is possible to get rid of big clicks in a fairly clumsy way using other tools.

- Do you have a huge budget?

If you have tens of thousands of dollars available for your project, there are professional hardware units that will do the job much more easily.

- Are you prepared to expend effort to achieve your goals?

  If you're lazy and want the job done automatically, then Wave Repair can't help you.

2.2 Things You Can't Do With Wave Repair

There are other very good quality shareware editors available, and I saw no reason to attempt to compete with them. Wave Repair offers only those facilities appropriate to cleaning up vinyl records. It therefore lacks some features commonly found in general purpose WAV file editors:

- Have many wave files open at once.
- Arbitrary copy/cut/paste.
- Effects such as distortion, echo, reversal, pitch shift, playback rate change.
- Resampling to different sample rates and/or resolutions.
- Mixing of multiple waveforms into one.

3 Interface

3.1 The Main Window

Working in Wave Repair is largely carried out in its main window:
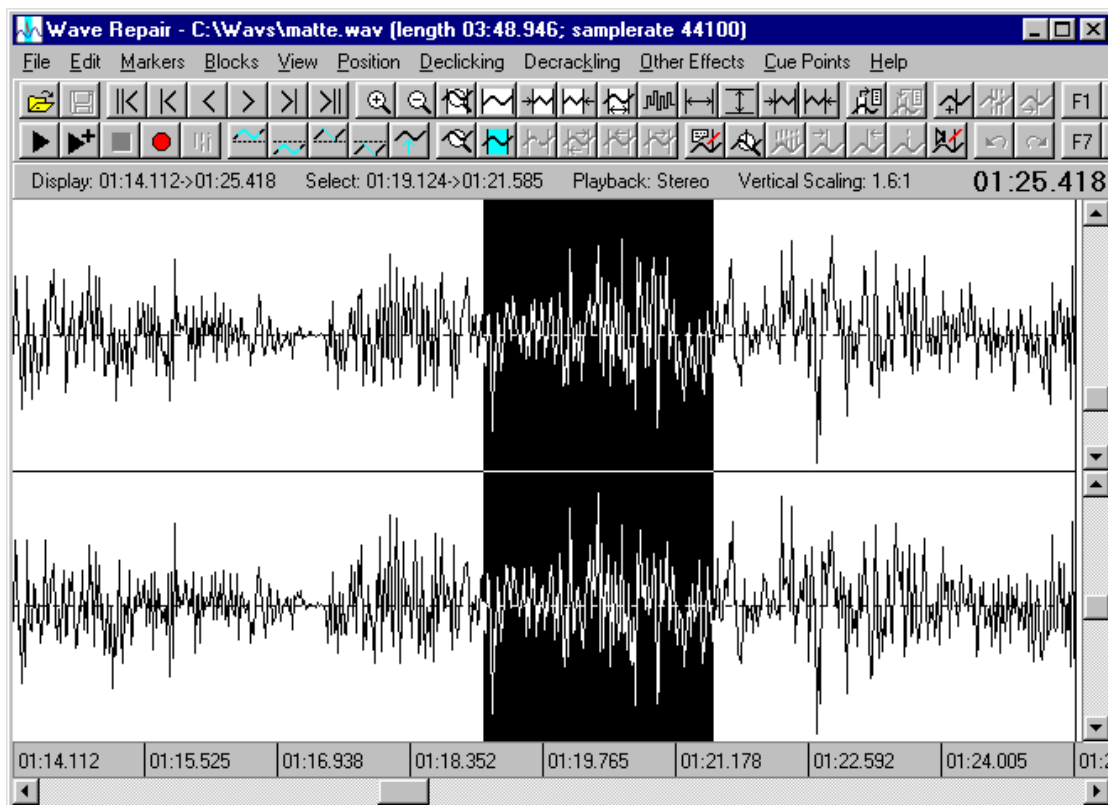
The caption gives the name of the WAV file being processed, its total length as a time, and its sample rate. Beneath the menu bar is a toolbar containing buttons to execute common functions; each button has a tool tip describing its use which pops up if the mouse is held over the button. The toolbar can be switched off if the user prefers not to use it. Underneath the toolbar is a status bar showing various attributes of the WAV file: the part of it currently displayed on the screen, the part of it which is currently selected, which channels are selected for playback, the amount by which the vertical scale is magnified, and at the far right hand end the current position of the mouse.

The two waveforms are the left and right channels (left on the top, right on the bottom). The waveform is normally displayed in black-on-white, but the part of the WAV file which is selected is shown reversed (ie. white-on-black). This selected region is important, since most of the functions within Wave Repair operate on the selected region. To set the selected region, click the left mouse button to set the start-of-selection and the right mouse button to set the end-of-selection; alternatively click and drag the mouse through the required selection. To the right of the waveforms are a pair of vertical scroll bars which control the zoom factor of the vertical axis (ie. the amplitude domain). The top scroll bar adjusts the zoom factor, while the bottom scroll bar adjusts the centre-line offset. Since in most cases it is convenient to work with **View | Maximise Amplitude** enabled (whereby the zoom factor in the vertical axis is automatically adjusted for each displayed page so that the waveform is shown in greatest detail), these scroll bars can be switched off if desired.

Beneath the waveforms is a time bar which shows elapsed time into the WAV file for the displayed screen.

Finally, there is a horizontal scroll bar. The entire scroll bar represents the complete WAV file, the size of the scroll bar's thumb indicates what proportion of the file is currently displayed, and the position of the thumb indicates where in the file the currently displayed page is located. This scroll bar can be used to navigate through the WAV file. Alternatively menu items and/or keyboard shortcuts can be used, so the scroll bar can be switched off if it is not required (which frees up a few more pixels for the waveform display).

By default Wave Repair shows times in the form **min:sec.millisecs**, but the user can change this to be **min:sec.frames** (where a frame is 1/75th sec; the standard block size used on CDs).

3.2 Positioning Within WAV Files

The slice of the file displayed in the main window is known as a page. You can move forward or back by a page using the **Position | Next Page** and **Position | Previous Page** options. If you wish to move forward or back by a smaller amount, this can be done with the **Position | Next Fragment** and **Position | Previous Fragment** options. A fragment in this context is a fraction of a page; unless you have re-defined it with the Options dialog, a fragment is 16% (about 1/6th) of a page. Moving forward or back by a fragment is useful if you see something interesting at the very start or end of the page and would like to see it in its surrounding context.

The number of samples displayed in a page is controlled by a variety of options:

1. **View | Standard Detail Scale** sets the number of samples to that defined in **Detail Scale Samples/Page** in the Options dialog. If you have not set this (the usual state of affairs), it defaults to a few samples less than the number of pixels across the main window; this means that it shows one sample per pixel, which is the best scale for visual identification of clicks and pops.
2. **View | Selected Samples** sets the current page to be exactly the current selected region, and the number of samples per page is set accordingly.
3. **View | Zoom In** and **View | Zoom Out** simply halves or doubles the number of samples per page.
4. **View | Entire File** sets the number of samples per page to the number of samples in the file; this means that the whole file is displayed.
5. **Position | Mark Region** sets the current selected region to a start and end time, and also sets the current page to be that region.

Other options do not alter the number of samples per page, but move the display to a different part of the file:

1. **Position | Goto Position** sets the current page such that the time specified appears near the start of the page.
2. **Position | Start of File** shows the page starting at the first sample in the file.
3. **Position | End of File** shows the page ending at the last sample in the file.
4. **Position | Start of Selection** shows the page starting at the first sample in the currently selected region.

3.3 Redraw Mode

You can switch Wave Repair in and out of a special mode called **redraw wave mode**. This allows you to use the mouse to touch up the waveform manually. You can easily tell when the mode is on since the mouse cursor changes from the normal one to an up-arrow or cross (depending on which shape you have chosen), and the background colour changes.

Once in redraw wave mode, by clicking and dragging the mouse, you may repaint the waveform to any desired shape you wish. While dragging, the redrawn waveform is shown in red; when you release the mouse button the redrawn shape replaces what was previously present. Although redrawing can be done at any level of zoom, it is normally only sensible to redraw waveforms at very high zoom levels, where you can clearly see each individual sample.

3.4 Markers

While processing a file, you may notice places of interest to which you would like to return at some later time. Rather than trying to remember where all these places are, you can mark them. The marks are displayed as green lines, which provide a visual clue to the places you wish to revisit.

There are two situations in which you may wish to add markers:

During the course of playing the selected region, you may hear many instances of potential damage to which you would like to return for further investigation. These places can be marked by pressing the space bar as you hear them, which causes the current position in the playback to be marked.

Clearly, there will be some delay between you hearing something to be marked and pressing the space bar, primarily due to simple reaction time. Since each user's reaction time may be different, the Options dialog allows you to set up an appropriate delay (in milliseconds). Some experimentation may be necessary to determine your own reaction time, but around 300 milliseconds is a good place to start.

While you are processing the file manually (ie. not during playback), you may notice suspicious looking waveforms that may be damage of some sort, but you would rather do something else before returning to them. These places can be marked by holding down the control key while clicking on the position with the left mouse button.

3.5 Use Of The Mouse & Keyboard

As well as the standard uses of the mouse (eg. for selecting menu items, pressing buttons, etc), the mouse is used in Wave Repair for these purposes:

While Wave Repair is in **redraw wave mode**:

- Left click and right click don't really do anything, although they may result in a minuscule redraw if you happen to drag the mouse accidentally. Basically, if you can't see that a redraw has occurred, it probably won't be audible; if you can see an accidental redraw, simply undo it.
- Drag from left-to-right (either button, either direction): replaces the current waveform with whatever shape you draw with the mouse.
- Left click while holding down the control key: allows you to redraw without needing to hold down the mouse button at the same time. This may make accurate redrawing easier. The redraw is completed by left clicking once again.

While Wave Repair is not in **redraw wave mode**:

- Left click without holding down the control key: sets the start-of-selection position.
- Left click while holding down the control key: sets a marker at the position clicked.
- Right click without holding down the control key: sets the end-of-selection position.
- Right click while holding down the control key: sets a special "audible marker" at the position clicked. On playback, a distinct "click" is emitted at this point, which some users may find useful in determining the position of a nearby audible defect.
- Right click while holding down the Alt key: begins playback at the position where the mouse click was made. Playback continues for as long as the mouse button is held down. When the button is released, playback ends and a marker is placed at the position where playback stopped.
- Drag (either button, either direction): sets the selection to be the dragged region.

- In the special case where an overlay block has been established (see later in the **Overlay Blocks** section for a discussion of overlay blocks), if a mouse drag is initiated within the displayed overlay block, then the overlay block is moved with the mouse.

In either mode, the mouse scroll wheel (if present and activated) navigates through the WAV file: by a page at a time normally, or a fragment at a time if the control key is also held down.

In normal working, Wave Repair doesn't need any keyboard input, so all keystrokes are available as shortcuts for various menu items. Therefore, rather than requiring you to press key combinations or sequences, the most commonly used accelerators are single letter keystrokes (eg. the **L** key interpolates the left channel). This strategy is very convenient if you need to use the mouse and keyboard together: use the mouse for waveform editing, and your other hand to execute common editing functions without having to constantly move the mouse up to a menu item. All the shortcuts are shown on the corresponding menu items. In addition to those, the following additional shortcuts can be used to move through the WAV file:

- space bar: next fragment (or sets a marker during playback)
- escape: previous fragment (or pauses/resumes during playback)
- return/enter: next page
- '+' above enter: previous page

3.6 Customisation

3.6.1 Macros

Wave Repair includes a facility whereby you can set up macros that execute a sequence of operations. Up to 24 such macros can be created via the Macros dialog. The first 12 macros can be invoked by pressing one of the keyboard's 12 function keys or a corresponding button on the toolbar. Macros 13 to 24 can only be invoked by pressing one of the 12 function keys while holding down the Shift key; in this case the macro which is invoked is 12 plus the function key number (eg. Shift+F3 invokes macro number 15). Note that during macro execution, if an operation involves the display of a dialog box which has a Cancel button, then pressing the Cancel button will abandon the remainder of the macro execution.

3.6.2 Colours and Menu Shortcuts

The colours used for the various items in the main window display and the keyboard shortcuts assigned to menu items can be changed to suit the user's taste via the Options dialog.

4 The Restoration Process

The process of restoring a vinyl record comprises five steps: recording, trimming, locating damage, repairing damage, and archiving the results.

4.1 <u>Recording</u>

The first step is to record the WAV file onto hard disk. Wave Repair can do this for you (via the Recording dialog), or you might prefer to use another recording package: Wave Repair is quite able to read in a standard WAV file from hard disk. One thing to bear in mind is that Wave Repair only works on 44.1 or 48 kHz 16 bit stereo WAV files, so you should be sure to record in that format.

If your eventual aim is to write the results of the restoration to CDR, remember that CDR must be written at 44.1 kHz, so you should record at that sampling rate. If you already have your analogue sources digitised on some external format (eg. DAT or Minidisk) then it makes sense to transfer to hard disk digitally if you have a soundcard capable of digital I/O: this avoids the possible degradation involved in an extra D/A and A/D conversion stage. However, if you have DAT tapes recorded at 48 kHz and wish to write the results to CDR, then you will need to perform a sample rate conversion at some stage. Wave Repair does not support sample rate conversion; you will need to use another package (such as GoldWave or CoolEdit).

When processing an LP side or lengthy tape, do not be tempted at the recording stage to try and split the tracks into separate WAV files, since you are likely to lose the timing of the inter-track gaps. (You might be surprised at how unnatural the results sound if the inter-track gaps change on an album with which you are familiar). Moreover, if tracks segue into one another, it will be very difficult to arrange for the segues to remain correct if you split the tracks at record time. When the time comes to write a CDR, you may need to split the tracks into separate WAV files: this is simple using **Cue Points | Split Tracks**.

Don't try to cue up the recording to start and stop precisely. Instead, allow a little time before and after the music to be included in the WAV file. It is easy to remove this later by using Deletion Marks, and removal by this method is far more accurate than trying to start and stop a recording at the right moment.

4.2 <u>Hum and Rumble Removal</u>

If it is necessary to treat hum and/or rumble, it is best to do so first.

Hum is generally caused by incorrect grounding (earthing) of the stereo system and/or PC. Often it is due to different ground potentials at the stereo and the PC. Ideally you should aim to eliminate hum before recording. However, if it proves impossible to do this, you may be able to reduce or remove the hum in the recorded WAV file using Wave Repair's filtering facility. Invoke **Other Effects | Filter Preview** to listen to the WAV file while experimenting with the filter settings. Hum is best removed by one or more notch filters, using the steepest possible filter characteristic which removes the hum. There are some presets that ought to work well for hum for the two common mains frequencies of 50Hz and 60Hz. However, you may find that some minor adjustments are necessary. For example, hum is commonly found at both the mains frequency and double that frequency (eg. 60Hz and 120Hz), but I have also seen hum at three times mains frequency, so if the presets don't work, try adjusting the filter frequency accordingly. After finding the appropriate filter settings, invoke **Other Effects | Filter** to actually apply the filtering.

Rumble is a low frequency "grumbling" caused by a less than perfect turntable bearing. Even though it is often almost inaudible due to the extremely low frequencies involved, it soaks up amplifier power and can cause speaker cones to "flap about", sending them into a non-linear part of their operating range. As with hum, use **Other Effects | Filter Preview** to find the correct settings, then invoke **Other Effects | Filter** to actually apply them. Rumble requires a high pass filter, again with as steep a slope as possible. The preset supplied is unlikely to be a perfect setting for all turntables, but it provides a reasonable starting point for experimentation.

4.3 Decrackling

Some records suffer from a constant background of small ticks and crackles. (In many cases it is only a problem in quiet sections). In order to reduce crackle it is necessary to be able to take a "fingerprint" of the background noise on its own, and so this must be done before any "dead space" is trimmed away from the recording:

1. Select a region of the recording which contains just background noise and crackle. Do not include any of the music itself or any large clicks and pops.
2. Invoke **Noise Reduction | Get Fingerprint**. This analyses the region of noise to take a "fingerprint" of what the background crackle is like.
3. The next step is to discover the best settings for the two parameters Wave Repair uses to carry out the decrackling. Due to the complex nature of decrackling, there is no realtime preview available. Therefore it is best to first try decrackling a short section using a variety of different settings:
4. Select a fairly short region (10 seconds or so) which contains music with background crackle.
5. Invoke **Noise Reduction | Decrackle (With Undo)**. A dialog will appear which allows you to set two parameters that control the amount of decrackling. Briefly, you should aim to set the **sensitivity** as low as possible and the **threshold** as high as possible while still achieving a meaningful degree of decrackling. Pressing the **Help** button will provide a more detailed discussion of the use of these parameters.
6. Listen to the results to determine whether the decrackling is acceptable. Bear in mind that decrackling does not remove larger clicks and pops, so you should expect these to remain.
7. If some crackling remains, increase the amount of decrackling by increasing the **sensitivity** and/or decreasing the **threshold**. Which of these should be adjusted depends on the precise nature of the background noise on the individual record and it is difficult to predict how each will respond, so it is best to experiment with both.
8. On the other hand, if the music sounds obviously distorted, it will be necessary to reduce the amount of decrackling by decreasing the **sensitivity** and/or increasing the **threshold**. The aim is to find settings that remove most of the background "hash" without distorting the music signal. Be aware that reed and brass instruments such as saxophone, oboe and trumpet are most susceptible to distortion due to excessive decrackling.
9. Invoke **Edit | Undo**, then repeat steps (b) to (e) with the revised parameter settings until you are happy with the results. Note that it is inevitable that

some records will have a combination of crackle and music which means that you cannot remove all the crackle without damaging the music unacceptably. Once you have determined appropriate settings, select the region to have the crackle reduced and invoke **Noise Reduction | Decrackle**.

4.4 Broadband Noise Reduction

After decrackling, it may be appropriate to reduce constant background noise, such as tape hiss from historic recording. As with decrackling, this is achieved by first taking a "fingerprint" of the background noise. It is important to appreciate that the fingerprint that was taken for decrackling is unlikely to be suitable for broadband noise reduction, as it will contain crackles that will have been removed. Therefore after decrackling, a new fingerprint should be taken if noise reduction is to be performed.

The procedure is very similar to decrackling:

1. Select a region of the recording which contains just background noise and crackle. Do not include any of the music itself or any large clicks and pops.
2. Invoke **Noise Reduction | Get Fingerprint**. This analyses the region of noise to take a "fingerprint" of what the background noise is like.
3. Next, determine the appropriate level of noise reduction. Unfortunately, due to the complex nature of noise reduction, there is no realtime preview available. Select a fairly short region and invoke **Noise Reduction | Reduce Noise (With Undo)**. Try different settings of **sensitivity** and listen to the results. Between each different **sensitivity** setting experiment, use **Edit | Undo** to restore the original music. The aim is to find a degree of noise reduction which achieves a worthwhile reduction in the background noise without introducing unacceptable degradation of the music. There are two basic types of degradation that can occur: (i) excessive removal of parts of the music signal; (ii) the introduction of random "tweeting" noises, which is an inevitable side effect of the way that the noise is subtracted from the signal.
4. Once a suitable setting for **sensitivity** is found, select the region to be de-noised and invoke **Noise Reduction | Reduce Noise**.

4.5 Trimming

The next step is to trim away unwanted leading and trailing sections.

It is a good idea to try and get the music to start very close to the beginning of the WAV file; I recommend that you include no more than 1/5th second before the music starts, and bring the level up from zero to full volume (using **Edit | Fade In**) during this 1/5th second lead-in time.

At the end of the WAV file, don't chop the music off suddenly as it ends. It is usually better to leave a few seconds of trailing space after the music ends, and gradually bring the level down from full volume to zero (using **Edit | Fade Out**) during those few seconds.

Once you've decided which parts of the leading and trailing areas you wish to trim away, there are two basic approaches:

1. Select that part of the WAV file you wish to keep (ie. make the selected region start about 1/5th second before the music starts, and end about 3 seconds after it ends), and use **File | Save Selected Samples** to write the part you want to keep to a new WAV file. Once the new WAV file has been written, you can delete the original file.

2. Mark the leading and ending sections as areas to be removed (using **Edit | Mark Selection for Deletion**) and then use **Edit | Execute Deletions/Insertions** to remove them.

4.5.1 Inserting Samples

As well as trimming unwanted samples out of the WAV file, it may be appropriate to add some (for example to adjust gaps between tracks). Extra silent samples can be added by placing a marker via **Edit | Add Silence Insertion Mark**, which will ask for the length of silence to be inserted. A subsequent invocation of **Edit | Execute Deletions/Insertions** will then physically insert the desired period of silence.

4.6 Declicking: Locating Damage Manually

In this section I will discuss techniques for finding damage by hand. Fixing it is addressed in the next section, and automatic methods are discussed in the section after that.

4.6.1 Listen & Mark

One approach to finding damage is to listen through the WAV file and mark the places where damage is heard. As the file plays, press the space bar every time you hear something which needs to be repaired. This will leave a marker at that place to which you can return later.

4.6.2 Manual Visual Scan

Another method of finding damage is to manually scan through sections of the WAV file. This is best done at a level of zoom which shows one sample per horizontal pixel (which by default corresponds to the degree of zoom selected by **View | Standard Detail Scale**). As you scan through the pages, be on the lookout for obvious discontinuities in the waveform shape. If you find one, you can listen to see if it is an audible glitch by making the selected region just around it and then playing it within its surrounding context using **File | Play Selection With Context**.
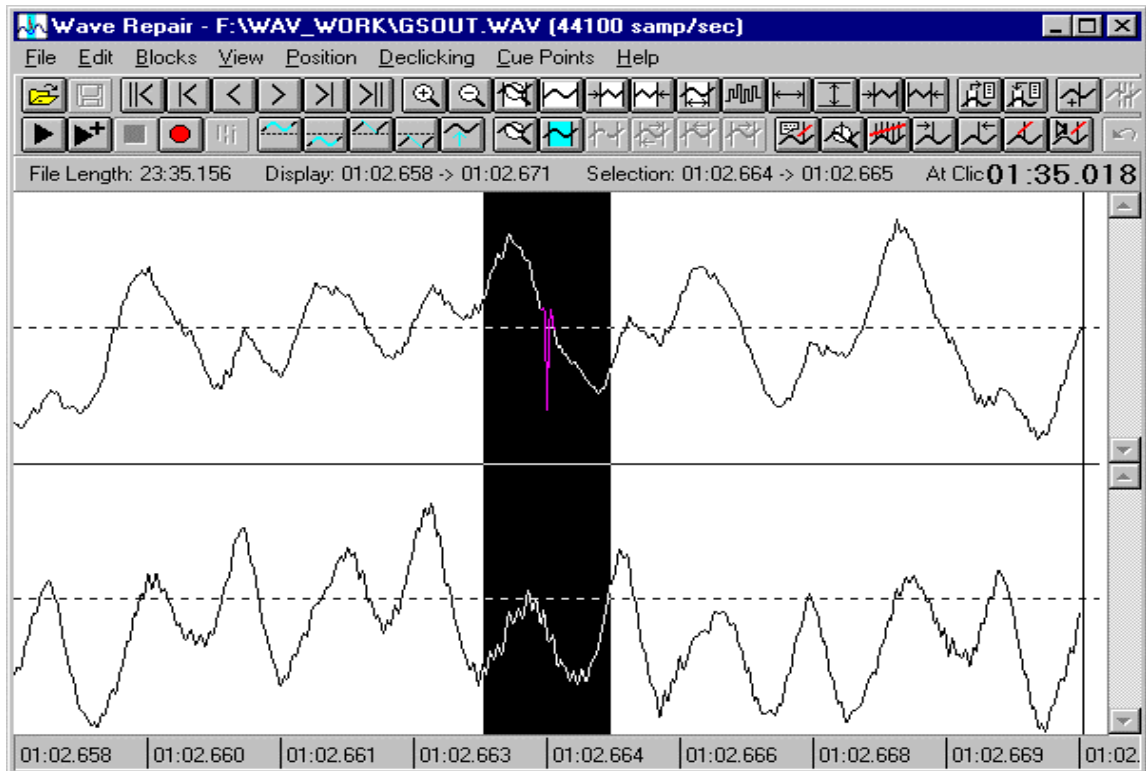
4.6.2.1 Example Screen Shots

This section presents a few screen shots of vinyl damage to give you an idea of the kind of things to look for.

The "classic" LP click is a very obvious spike like this:
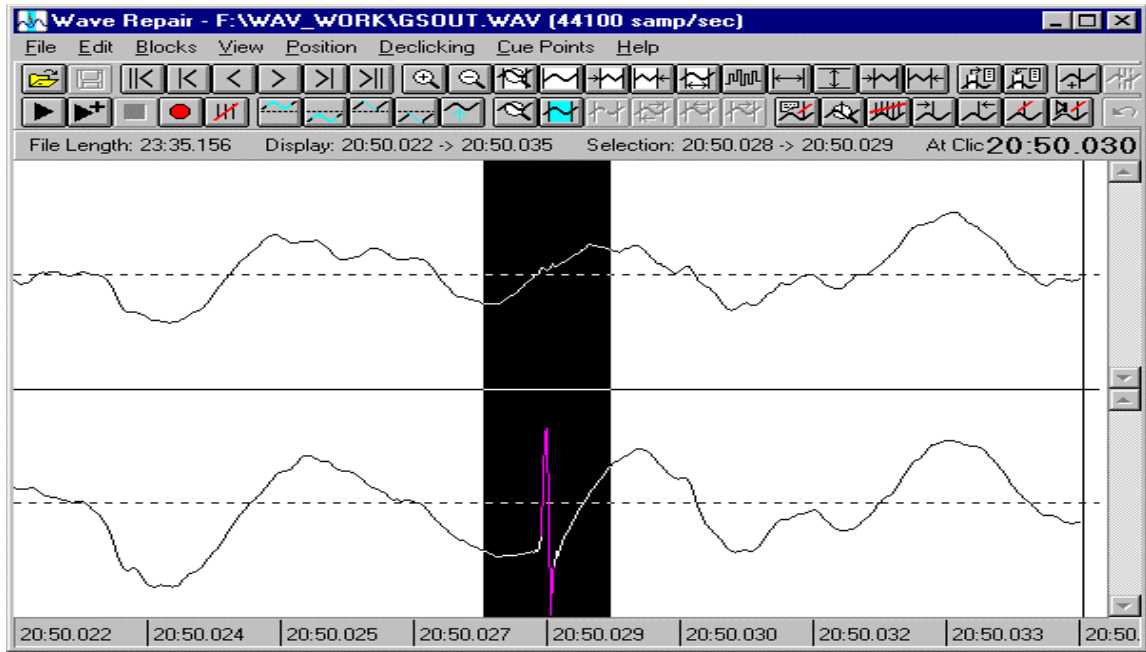
Finding spikes like the above is relatively easy, and they can easily be removed using a Bezier interpolation. Where a spike appears on the side slope of a waveform, such as this:


it can often be repaired using linear interpolation (although a Bezier interpolation will be just as good).

Some audible clicks and ticks are visually surprisingly small when you finally track them down:
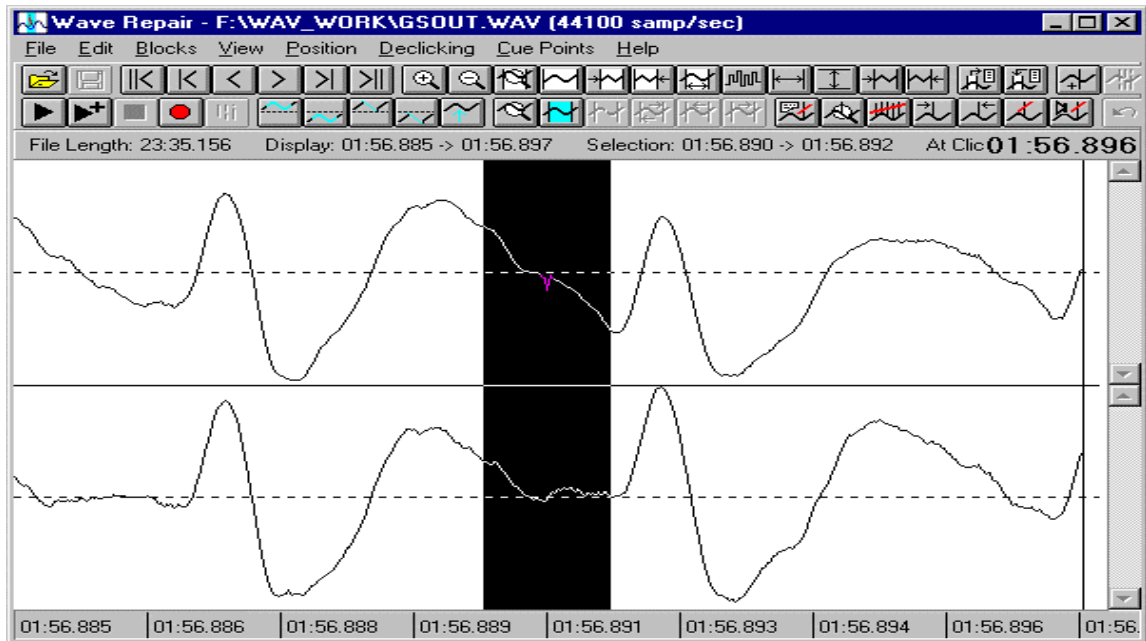
Some damage isn't a distinct click, but more a sort of "scuff" or fleeting "buzz". This is often seen as a brief section of "spikiness" superimposed on an otherwise smooth waveform:
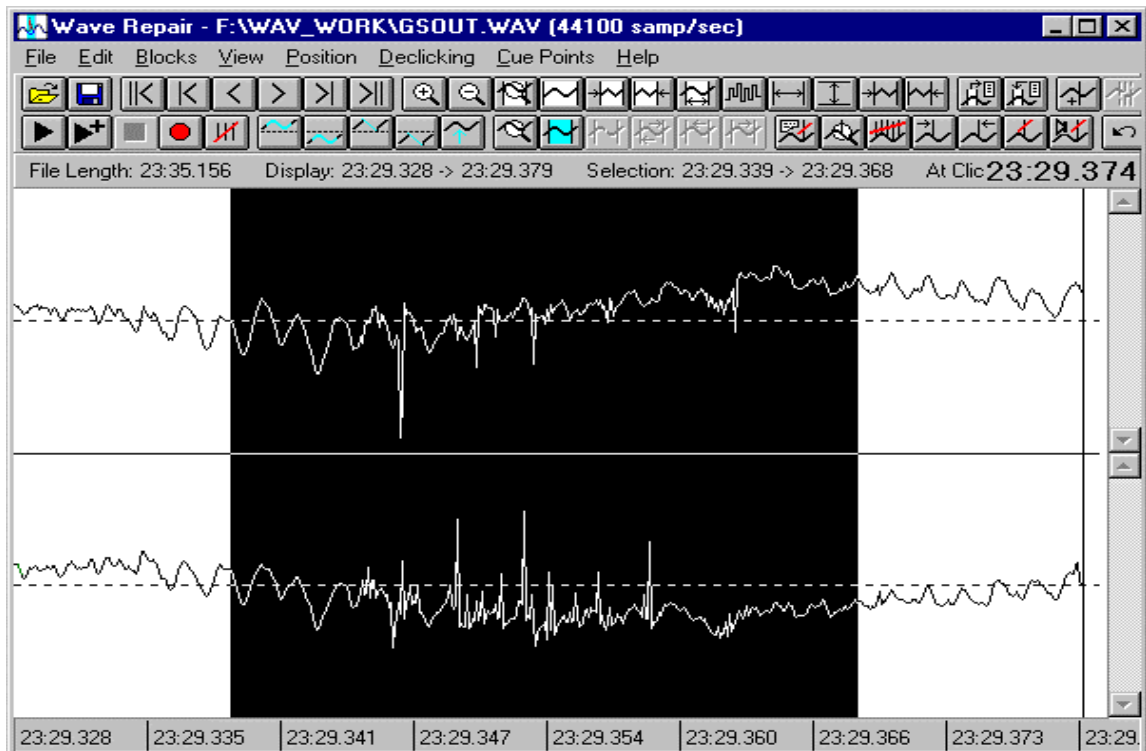
Repairing "scuffs" like this can be tedious. Scanning through the affected region and redrawing the waveform to remove the spikiness is sometimes the only way, but you can experiment with spectral replacement, smoothing the waveform or trying to find a replacement from nearby.

4.6.3Combining Listen/Mark With Manual Scanning

In practice it is often best to combine the two methods described above. It is generally not feasible to listen through the entire WAV file at one sitting, and it is certainly not feasible to manually scan through an entire LP side at the one-pixel-per-sample zoom factor (at an 800 x 600 screen setting, there are over 3000 pages per minute at this level of zoom!)

My own technique is to deal with a short section (about 30 to 60 seconds) at a time. Having selected the next section, I listen to it and mark damage with the space bar as it plays. I then revisit each mark and scan the immediately surrounding area to visually locate the damage.

4.6.3.1Homing In On Damage

Once you've found a glitch of some sort, it's usually easy to isolate a section of about a second which obviously contains the glitch (by simply listening to that section). Homing in further on the damage can be a little tricky. You can't simply play shorter and shorter sections, since below a certain time span it's difficult to hear anything meaningful.

My own technique is to leave a reasonable length after the glitch, and to try and get the start of the selection as close to the glitch as possible. Keep moving the start of selection further forward, playing each time. At some point, the glitch will disappear on playback, which means you've moved the start of selection too far forward. Move it back a touch until you can definitely hear the glitch on playback. This method usually gets to within 1/5th or even 1/10th second of the glitch, at which point you can switch to manual visual scanning of the waveform at one pixel per sample. The glitch is usually found within a few pages of the start of the selected region.

Other users have reported that they have had success a similar procedure, but where they move the end of the selection (rather than the beginning) towards the audible click to see when it disappears.

Finally, you can place a special "audible marker" (indicated by a vertical red line) at any desired position by right clicking with the mouse while holding down the control key. During playback, a distinct "click" is output at the position where the audible marker has been placed. Listening to the relative positions of the audible marker and the glitch being sought can help to get a feel for where the glitch is. The marker can be moved by simply right clicking at a different position.

4.7Repairing Damage Manually

Once you've located the damage, there are a number of options for fixing it: interpolation, redrawing, replacement, deleting, smoothing and muting. Having used one or more of these repair options in an attempt to repair the damage, you can listen to the results before saving the changes. If the results are unsatisfactory, you can discard them using the **Edit | Undo** option.

Warning: only save the changes when you are happy with the results. Remember that in order to achieve good performance, Wave Repair writes the updates directly back over the original WAV file; once you save the changes, the original is overwritten.

4.7.1 Interpolation

If you have a simple spike type click that interrupts an otherwise smooth wave shape (eg. on the up- or down-slope of a clearly defined wave cycle), interpolation will almost certainly do the trick. Simply select the region immediately around the spike, and execute **Edit | Interpolate** or **Edit | Bezier Interpolate** for the appropriate channel.

4.7.1.1 Linear

Linear interpolation replaces the selected region with a straight line joining the waveform up on either side. Linear interpolation is sometimes good enough (eg. when removing a spike contained entirely on the up or down slope of a single wave peak, but in general better results are achieved using Bezier interpolation. Also, if the region to be interpolated extends for more than a few dozen samples, linear interpolation often results in an audibly better repair.

4.7.1.2 Bezier

Bezier interpolation attempts to draw a curve of a suitable shape to replace the selected region. In many cases the curve is a good approximation to how the original waveform would have looked before the damage. Note that the shorter the selected region that is interpolated, the more likely it is that the curve will be a good fit.

4.7.1.3 Spectral Replacement

Strictly speaking, spectral replacement is a form of block replacement (see later) rather than interpolation, but it is worth considering on even quite short sections if Bezier and linear interpolation have failed to produce an accetable fix. To try it out, invoke **Blocks | Spectral Replacement**.

4.7.2 Redrawing

Redrawing the waveform with the mouse is the most flexible mechanism available. In theory you can of course draw any waveform you like, but in practice it is virtually impossible to predict what a complex drawn waveform will sound like.

Redrawing tends to be useful when the damage is a little more extensive than a simple spike. Having identified the damage to be repaired, switch on **Edit | Redraw Wave Mode**, and by clicking and dragging the mouse, simply draw over the damage with the waveform you regard would have been there had the damage not been present. You will see this new waveform in red as you drag the mouse. When you release the mouse button, the redrawn waveform will replace the damage. You may find that zooming in to a finer level of detail in the time domain will make the redrawing easier. In this case, you'll find yourself switching in and out of redraw wave mode, and swapping between zoom at one pixel per sample v. higher degrees of zoom. This is the reason behind many common options having single keystrokes as shortcuts: you can use the mouse in one hand to define the selected region you're interested in and to redraw waveforms, while using the other hand to press **W** (toggle redraw wave mode), **S** (show selected region) and **D** (show standard detail scale - one pixel per sample).

4.7.3Replacement

Some damage extends for too long a time to be repaired by interpolation or redrawing. Such damage is typically a burst of distortion superimposed over a complex waveform with a lot of high frequency content. Even though it isn't a single click visually, it often sounds like one (although it may sound more like a smeared "splat" than a well-defined click). The problem with these kinds of waveforms is that it's virtually impossible to distinguish the damage from the music. You can try redrawing bits and pieces, but my experience is that you just end up with a different and equally objectionable splat.

4.7.3.1From The Other Channel

If the damage is to one channel only, and the stereo difference between the channels is small, replacement of the damaged section from the other (undamaged) channel may well produce a good repair. To try this, highlight the damaged section and use **Blocks | Copy Left to Right** or **Blocks | Copy Right to Left** as appropriate to copy over the other channel.

4.7.3.2From Nearby In The File

Another option is to replace the damaged region with a copy of some other portion of the same WAV file that looks pretty much the same.

4.7.3.2.1Preceding Block

During the 1970's, analogue devices came onto the market that performed some degree of declicking for record players. The way they worked was to substitute likely clicks with a copy of the immediately preceding signal, on the assumption that it should be pretty much the same. Wave Repair has an equivalent of this very inaccurate approach, via the **Blocks | Copy Preceding Block** option: this simply grabs the same number of samples as are currently selected from the immediately preceding part of the WAV file and copies them over. If you can see that the preceding part of the WAV file looks right, and are careful about defining the selected region, this can work well.

4.7.3.2.2Compatible Blocks

Wave Repair also has a more sophisticated version of the Preceding Block mechanism. It can scan backwards (or forwards) through the WAV file, looking for a block of samples that start and end with sample values that are close to those that start and end the selected region; the hope here is that such a "compatible" block may be suitable to replace a damaged area. Using the **Blocks | Set Compatible Block Params** dialog, you can define the extent of the WAV file to be searched for such a replacement block. It is unlikely that a suitable replacement block will be found more than 1/10th second away from the damaged area; it is also common that searching backwards is more likely to find a suitable replacement. Repeated uses of **Blocks | Find Compatible Block** will search further afield for the next block that qualifies. The current potential block is displayed in green over the top of the selected region that it might replace; if you want to copy it over, use **Blocks | Copy Compatible Block (Left** or **Right**, accordingly). After copying a block over, the edges may need to be tidied up.

4.7.3.2.3Overlay Blocks

Another way to find and copy over a replacement block is via the **Overlay Block** mechanism. In this scheme, rather than having Wave Repair look for what seems a likely replacement section, the original damaged waveform is displayed as a blue overlay which can be slid around the WAV file, allowing the human operator to use his or her own insight in finding a suitable replacement block. To set up the blue overlay for the selected region, use the **Blocks | Setup Overlay Block** option. The blue overlay can now be moved around with other options under the **Blocks** menu or by dragging it with the mouse. Note that you can alter zoom factors and generally move about however you like within the WAV file without affecting the established overlay block. Ultra-fine positioning of the blue overlay is best done with the **Blocks | Back/Forward One Sample** options. Once the overlay has been positioned precisely over the desired replacement segment, the **Blocks | Copy Overlay** option overwrites the original samples (ie. those of which the blue overlay is an image) with the samples underneath the overlay.

In general, the overlay block mechanism gives the best chance of success, but requires a little more patience to use.

4.7.3.2.4 Nudging Blocks

After copying a replacement block, it is possible to "nudge" it up or down if it is offset vertically. This allows the replacement block to fit in with its surrounding context. There are four options in the **Blocks** menu to perform nudging (nudge up or down, left or right channel).

The exact amount by which the block is nudged is that which causes it to move by one pixel on the display (unless the vertical scale is so great that one quantisation level covers more than one pixel, in which case the nudge amount is a single quantisation level). Thus, the vertical scale factor affects the amount of nudge, but this is appropriate since it means the amount of nudge is a constant proportion relative to the surrounding waveform. If you wish to exercise finer control over the nudge amount, you can adjust the vertical scaling factor using the scroll bars along the right hand edge of the display.

4.7.3.3 Spectral Replacement

This replacement method performs a Fourier analysis of the immediately preceding and following blocks, computes an average, and inserts a reconstructed waveform from that average as the replacement. It can sometimes produce an excellent repair of damaged sections that last for several hundred samples. It is limited to a maximum block length of 16,384 samples. To make this type of replacement, invoke **Blocks | Spectral Replacement**.

4.7.3.4 From Another File

In some cases, it may be that you have a suitable replacement for a damaged section in another file. This may have been prepared with a different WAV editor, or perhaps by a re-recording. It is possible to replace a section of the current WAV file with (part of) another, provided the sampling rates of the two files are the same.

To replace a section of the WAV file, proceed as follows:

1. Position the start of the selected region at the exact sample where the replacement is to commence. You can easily check the exact sample number selected via **Position | Selection Details**.
2. Invoke **Edit | Replace From File** to choose the file containing the replacement samples. (It is possible to choose the same file in order to paste a long section, but if the sections overlap, the results will be wrong. I would recommend that another more sophisticated WAV editor should be used to do this kind of update).
3. Once the file has been selected, you will be asked to specify the exact sample number and length of the section to be copied. Note that Wave Repair conventionally denotes the first sample number in a WAV file as 1, not 0.

4.7.4 Deleting

A very short glitch may be successfully repaired simply by removing it (ie. deleting the samples). Provided the number of samples deleted is sufficiently small, the timing inaccuracy introduced into the music's tempo may be completely inaudible. (It is sometimes possible to remove several hundred samples without affected the perceived timing).

To determine whether a deletion will make a suitable repair, select the region you think could be deleted and invoke **File | Play Context Around Selection**. This will play a little before and after the selected region, but miss out the selection itself. By comparing this to the sound of **File | Play Selection With Context**, you should be able to judge whether deleting the region will work. If it sounds right you can then go ahead and delete the selected region. Some care is needed in selecting the precise start and end of the region to be deleted to avoid the introduction of a tick where the ends join up. You should aim to place the ends of the selection such that the waveform immediately preceding the start looks like it will flow seamlessly into the waveform immediately following the end.

To delete the selected region, mark it using **Edit | Mark Selection for Deletion**. At this stage, the samples are not deleted, since to do so would alter the number of samples in the WAV file, which means that the entire file will have to be re-written (a lengthy process). Rather, the marked section is noted for future deletion. You can continue to mark as many other regions to be deleted as you wish, and then finally delete them all in one go using **Edit | Execute Deletions/Insertions**; this means that only one complete re-write of the WAV file is necessary to effect multiple deletions.

While a section is marked for deletion, it will be omitted during any playback so you can hear the effect of making the deletion before actually carrying it out. If you are unhappy with a deletion mark, it can be removed by selecting a region which completely includes it and invoking **Edit | Clear Deletion Mark(s)**.

4.7.5 Smoothing

Where a lengthy section (by "lengthy", we're talking about a few tenths of a second at most) has been damaged by an abrasion, you can try to smooth it out. Select the region to be smoothed, and execute **Edit | Smooth Abrasion**. The general shape of the waveform is retained, but the "spikiness" is reduced. This is somewhat similar to performing low-pass filtering on the selected region (although filtering is not actually the way it is achieved).

Repeated applications of **Edit | Smooth Abrasion** will make the waveform smoother and smoother. Listen at each stage to determine whether any improvement is achieved. If you execute one smoothing too many, it is simple to **Edit | Undo** the last smoothing.

Due to the nature of the smoothing algorithm, a step may appear at the start and end of the smoothed region. If this happens, you can touch it up by interpolating or redrawing the boundary.

4.7.6 Muting

It is very rare that muting (using **Edit | Silence**) makes a satisfactory repair. However, it may be that a particular piece of damage is particularly objectionable and can't be otherwise fixed; in this case, muting the damage may result in a less objectionable sound, even though it will still be audible.

4.8 Automatic Methods for Finding and Repairing Damage

The automatic click detection process in Wave Repair comprises two aspects: finding clicks, and removing them.

The mechanism which finds clicks can be helpful in locating damage that requires repair. This mechanism is invoked via the **Declicking | Find All Clicks** option, which scans the currently selected region for possible clicks. It basically works as follows. The portion of the file to be declicked is scanned sequentially in chunks (typically each chunk is about 100 samples or less). Within each chunk, an average amplitude change is calculated; this gives a sort of "base level of activity". The aim is then to find amplitude changes within the chunk which exceed the average change by a certain degree. Any such excessive change is regarded as a potential click. The various parameters are definable via the **Click Detection/Repair Parameters** dialog.

Find All Clicks gathers the candidate clicks into a list which can be browsed later by hand using the **Goto Next Click** and/or **Goto Previous Click** options. Each such click can then be considered for manual or automatic repair. Manual repair involves either redrawing the waveform with the mouse, or interpolating the damage out. Automatic repair, invoked via **Declicking | Remove Current Click**, attempts to interpolate a suitable area around the damage, but in many cases it will of course provide inferior results to manual repair. In some cases, you may find that **Remove Current Click** manages a partial repair, but the tail end of the click remains.

In addition to automatic repair of individual clicks using **Remove Current Click**, the option **Declicking | Remove All Clicks** automatically interpolates every click in the current list; the effect is identical to manually stepping through the list and invoking **Remove Current Click** on each one.

Note: If you invoke **Edit | Undo** after performing a **Remove All Clicks**, all the removals are undone; the clicks are not reinstated one at a time.

While **Remove All Clicks** can be effective in some circumstances, it should be viewed as the "lazy" option; manual repair of clicks nearly always yields better results.

Note that if the detection parameters are set aggressively (such that just about every real click is almost certain to be found), there will be a great many "phantom" clicks in the list. On the other hand, if the parameters are set so as to avoid these phantoms, a number of more subtle clicks are bound to be missed. It is best to try experimenting with a few settings to find the best compromise.

I have found through experience that it is all too easy to set aggressive parameters and find that they appear to work well on one section of music. Then, when applied to some other section they result in unpleasant clipping effects. For the very best results, the automatic declicking mechanism is probably best ignored. You are much better off simply listening to the waveform, noting the approximate times where clicks are heard, then going in and manually scanning through the relevant sections zoomed in to one pixel per sample, and with the amplitude scale maximised.

4.8.1 Unattended Declicking

The optimum declicking parameters will nearly always vary within a long WAV file. Therefore for best results, you should declick a long WAV file in sections. This is of course a time consuming task. If you are prepared to accept less-than-ideal declicking in order to save time, the option **Declicking | Find and Remove All Clicks** will process the selected region in one go: finding the clicks and removing them according to the currently set parameters. The best way to use this option is to establish the best compromise of declicking parameters using the real-time declick preview feature.

Warning: Note that in order that this operation can run quickly, it directly amends the original WAV file on hard disk, and its effects cannot be undone. You are strongly advised to make a backup copy of the original WAV file before proceeding in case the results are unacceptable.

Note that in an unregistered copy of Wave Repair, this operation works only on the left channel (this allows an evaluation of its effectiveness while preventing it being of any practical use).

4.8.2 Realtime Declick Preview

Wave Repair has a mechanism known as **Declick Preview** that looks for clicks while playing (a section of) the wavefile. The purpose is to allow the user to adjust the click detection parameters while listening to their effects. This makes automatic declicking a much more useable process than by setting parameters, running a declick/remove pass, listening to the results, and then starting over again.

Note: This real-time mechanism does not actually create amendments to the wavefile in the way that the **Declicking | Remove All Clicks** option does (although the algorithms used to detect and repair the clicks is the same); rather it monitors the samples and removes them on-the-fly in the output buffers which are being sent to the soundcard. So, although what you hear is the result of clicks having been removed, no

actual updates are made; they can be made later in the normal way using the parameter settings selected during Declick Preview.

To use this function, select the range of samples you wish to declick and run **Declicking | Declick Preview**. The Declick Preview modeless dialog appears and the selected sample range will play over and over again in loop mode. While it is playing, you can adjust the parameters in the dialog box. You can also re-select a different region in the normal manner and, starting with the next pass of the playback loop, the new selected region will be played. Once you are satisfied with the parameter settings, press the **Stop** button to end Declick Preview. Since the last parameter settings you chose will have been retained, you are now in a position to run **Declicking | Find All Clicks** and then **Declicking | Remove All Clicks** to actually perform the declicking should you be happy with the results you heard while in Declick Preview. As a shortcut, the Declick Preview dialog has a button labelled **Remove** which automatically invokes **Find All Clicks** followed by **Remove All Clicks**.

Note that Declick Preview requires quite a lot of CPU power to run successfully. I have verified that a 133MHz Pentium is fast enough in most circumstances and that a 100MHz 486/DX4 is marginal. Note however that detection of "hill" shape clicks requires rather more CPU, and depending on the particular combination of parameters may not work on a P133; it certainly doesn't work on a 486.

4.8.2.1 Some Notes on Using Declick Preview

Declick Preview is most useful when you wish to clean up a section of quiet music (eg. the fade out of a song) that has a great number of low level ticks which would be very tedious to find and repair by hand. For removal of odd clicks here and there, manual homing in, scanning and redrawing remains the best option in many cases.

Work on fairly short sections (a few seconds) while you try to find the best settings. This allows you to adjust the settings and hear their effect while you still have a good memory of what the music sounded like on the previous pass, and thus be able to judge if the change is an improvement.

Low level ticks have different characteristics to bigger clicks and pops, and it is futile to try and fix them both in one pass. Try to get rid of the more obvious clicks first, then make another pass to deal with the little ticks. If you try to find settings that do them both together, it is likely that serious distortion will be introduced. (This is because big clicks typically need large click width & relative amplitude settings, while small ticks need small width & amplitude settings: to find both in one pass needs large width and small amplitude settings, resulting in the removal of lots of low level musical peaks).

There are of course a fairly large number of parameters you can fiddle with during Declick Preview:

1. The fundamental goal is to remove clicks with the minimum "collateral damage". Once you've found settings that remove the clicks, you should try to find the

largest settings for **relative amplitude** and the smallest settings for **click width &
repair width** that still get rid of the clicks.

2. If you hear clicks during Declick Preview, the first adjustment to make is to
reduce the **relative amplitude** setting. Keep bringing it down until the clicks start
to disappear. If, when they do, they are replaced by a roughness to the sound, try
increasing the **click width** setting and bring the **relative amplitude** back up a bit.

3. In general the most worthwhile click shapes to look for are **spike** and **instant**.
Although very big pops probably won't be found except as **hill** shapes, setting that
shape will usually find too many musical peaks that aren't clicks or pops.
(Basically, it is best to fix big pops manually). The **step** shape sometimes finds the
odd discontinuity that is missed by **instant**, but if you do try **step**, don't set the
relative amplitude too low or else literally thousands of completely innocuous
samples will be singled out for elimination. Only try **hill** or **step** if you've
exhausted the other adjustments first.

4. If your CPU can handle it, it's usually best to leave **repair with bezier curves**
switched on; the cases when a straight line interpolation gives better results are
extremely rare.

4.8.3 Analysis of Manually Found Clicks

Wave Repair has another facility that can help in setting up suitable declick
parameters. Having manually found an isolated click, you can ask Wave Repair to
analyse it (using the **Declicking | Analyse Click** option). Here, you place the start of
selection right on the found click, and then Wave Repair attempts to find the least
aggressive declick parameters which successfully find that click (while at the same
time finding no other clicks nearby; for this reason the click must be on its own with
no other clicks nearby).

4.9 Equalisation

This process adjusts the tonal balance of the music. Wave Repair provides a
mechanism that operates very much like the graphic equalisers that are found in some
home stereo systems, with ten frequency bands offering up to 12dB of boost or cut.
The procedure to follow is to use **Other Effects | Equalise Preview** to listen as you
adjust the equalisation settings, and once you are happy with them apply the
equalisation over the desired region by invoking **Other Effects | Equalise**.

When restoring vinyl LPs the most likely requirement for equalisation will be to add
a bit of extra "sparkle" at the top end. There are two common reasons why vinyl LPs
can sound a little dull:

- A small amount of wear is inevitable every time a record is played. This wear
tends to remove high frequencies more than others.
- Some records (especially cheap reissues) are made using "high mileage" stampers
that are frankly beginning to wear out; again, it is the high frequencies that wear
out first.

It is often best to perform equalisation before declicking. The reasoning behind this is
that equalisation may change how audible some clicks are; therefore if you equalise
after declicking, it may be necessary to do another declick pass afterwards.

4.10 <u>Mixing Channels</u>

This facility can be used to adjust the contents of the two stereo channels. Various proportions of the two channels may be mixed together and sent back to the channels. Typical uses for this would be:
- Swapping the channels (if they were accidentally recorded the wrong way round)
- Converting stereo to mono (in fact, to "double mono" where both channels are identical)
- Narrowing or widening the stereo image
- Switching the phase (to correct a recording that was accidentally made out of phase).

To set up the channel mixer, first invoke **Other Effects | Channel Mix Preview**, and listen to the effect as you adjust the settings. Once the desired settings have been established, invoke **Other Effects | Channel Mix** to actually apply the mixing.

4.11 <u>Normalisation</u>

This is used to bring the level of the recording up to a specified maximum. It is a simple procedure, whereby the WAV file is analysed, and then an amplification factor is applied so that the loudest peak(s) are at the level given (which as a general rule would be specified as 0db). If you have managed to record the WAV file so that the peak record level was close to 0dB, then normalisation will have little effect; it is more relevant if the WAV file was recorded at a low level.

It may not always be appropriate to normalise a recording. For example, if the recording is of a quiet instrument, such as flute, then the listener would not expect it to be particularly loud on playback.

There are some disadvantages to normalisation. Firstly, any noise (such as hiss or vinyl surface noise) is amplified along with the music. Secondly, there is a very minor reduction in resolution due to rounding errors in the arithmetic, although when dealing with recordings of analogue sources this effect is likely to be unnoticeable.

An option in Wave Repair's normalisation allows you to remove any DC offset there may be (ie. to correct an off-centre waveform).

Normalisation is a straightforward process: select the region you wish to normalise (usually an entire track or WAV file) and invoke **Other Effects | Normalise**.

4.12 <u>Amplification</u>

To adjust the volume of a file to an arbitrary value, use the amplification section of the **Other Effects | Amplify/Compress Preview** option to establish the desired degree of volume increase or decrease. (Leave the compression factor set to 1.0 to adjust only the peak volume). If you are increasing the volume, take care to avoid introducing clipping distortion (Wave Repair warns you if this happens). Once the desired degree of amplification is found, the **Other Effects | Amplify/Compress** option can be used to actually apply the update.

4.13 <u>Compression</u>

Another optional step, compression reduces the dynamic range of the louder parts of the music so that the overall volume can be increased. The more compressed music becomes, the higher its perceived volume (even though its peak volume is no greater than before it was compressed). Compressed music tends to sound more "punchy", so it is particularly suitable for rock and pop music. Ironically, many people perceive compressed music to have greater dynamics, when in fact the opposite is the case. If you wish to make a faithful restoration of vinyl LPs, then it is best not to apply any compression, although of course the choice is entirely yours. You might, for example, wish to make your CDR transfers of LPs sound as loud as commercial CD releases, which in the rock & pop field tend to be very highly compressed. The main situation where compression is useful is in balancing the perceived volume of a variety of tracks from different sources which you are putting together into a compilation. This can apply to digital as well as analogue sources.

Intuitively, you might expect to apply compression first and then normalisation, but in fact it is easier to do it the other way round. Once the WAV file is normalised, compression can be applied in one step without the need to worry about whether further normalisation is necessary. This is because Wave Repair's compression operates in a manner which maintains the peak volume regardless of the amount of compression. Were you to apply compression first, you would not know how much more volume would be added by a subsequent normalisation, thus complicating the whole process.

Compression should be used with discretion; it is a rather clumsy tool which can make music sound initially impressive while actually robbing it of the genuine dynamics it may have originally possessed.

Excessive compression will certainly make hiss and vinyl surface noise much more noticeable. After compression you may find that tiny clicks and ticks which were previously unimportant or even inaudible become quite noticeable, so a further declick pass may be necessary. As with normalisation, rounding errors in the arithmetic will cause a very slight reduction in resolution, but this will be swamped by the much grosser effects of the actual compression.

Before committing any compression to your WAV files, you should experiment with **Other Effects | Amplify/Compress Preview** to establish what settings are appropriate. (Leave the amplification gain set to 0.0dB to adjust only the compression). Once you have found the required settings, invoke **Other Effects | Amplify/Compress** to actually perform the compression.

Choosing the right degree of compression can only be done with experience; there is no rule you can follow which will work well at all times, since the effects of any particular compression settings depend greatly of the characteristics of the music to start with. However, if you are trying to balance the perceived volume of a compilation of tracks, the following procedure may help:

1. Normalise all the tracks independently of one another.
2. Choose the loudest track as a reference; this track will remain uncompressed.

3. For every other track, use Amplify/Compress Preview to discover the settings which match its perceived volume to the reference track chosen in step 2, and then compress the track with those settings.

The easiest way to compare the reference track with the one you're experimenting with is to have the tracks in separate WAV files, and to have two copies of Wave Repair running (one with the reference track loaded; the other with the track to be compressed) so that you can easily switch back and forth to listen to their relative volumes.

4.14 Archiving to CDR

Once you're happy with the results of restoration, you might very well wish to write a CDR. The CD writing packages on the market fall into two main camps:

1. The majority require each track of an audio CDR to be in a separate WAV file. (Adaptec's Easy CD Creator is the most widespread example). In order to use a package of this type, you will need to split the tracks in your long WAV file into separate WAV files.
2. Other packages can write multiple tracks from a single WAV file. (Goldenhawk's CDRWin is a common example). In order to use a package of this type, you need to write a text file (known as a "cue sheet") which defines where within the WAV file each track begins.

Wave Repair has a facility to manage track markers within the WAV file. These markers are known as **cue points**. To add a one at a suitable place, use the **Cue Points | Add Cue Point** option. This places a cue point at, or close to, the start of selection position. Since CD tracks are required to start on exact block boundaries (a block is 1/75th of a second), Wave Repair ensures that cue points are placed on such boundaries; this is why a cue point may not be placed exactly where the start of selection is.

Once you have placed all the appropriate cue points, they can be used to split the tracks into separate WAV files (using **Cue Points | Split Tracks**) suitable for use by packages like Easy CD Creator, or to write a CDRWin cue sheet (using **Cue Points | Write Cue Sheet**). Note that Wave Repair automatically assumes that track 1, index 1 begins at the start of the WAV file, so you should not place a cue point there. Also note that index cue points are only of use if you wish to write a cue sheet; when splitting tracks Wave Repair uses only the track cue points.

Wave Repair also has a facility which searches for possible track split points by looking for places where a minimum period of time is quieter than a certain volume. This is invoked by **Cue Points | Find Tracks**. You can modify the minimum period and the volume level via the **Options** dialog. It must be stressed that any kind of automatic search of this type is bound to fail in some cases: either missing track boundaries (because the inter-track gap is too short), or finding "phantom" track breaks (because there is a section of music that is very quiet). Please also bear in mind that the exact position where the track boundaries are found will only be approximate. The best way to place the cue points is to do so manually.