Sonic Studio HD™

NoNoise™ Guide

sonic studio

an affiliate of Sonic Solutions

NoNoise User Guide - Sonic Part Number 800170A (9/00)

Written and designed at Sonic Studio LLC, 12813 Industrial Park Blvd., Plymouth, MN 55441-3910, USA

NoNOISE

Sonic Studios' NoNOISE is the world's premier tool for restoring vintage and problematic audio recordings. NoNOISE's advanced processes isolate and eliminate audio artifacts such as hiss, scratches, hum, mechanical and impulsive noise.

NoNOISE is not a single process or software module, but a set of powerful tools to remove bothersome noise without damage to the program material. NoNOISE can be used to restore old recordings, remove unwanted noises from field recordings, and repair audio materials that have suffered damage.

SonicStudio HD offers three forms of NoNOISE processing:

- **Manual Declicking** provides tools for isolating and removing individual clicks in a soundfile within SonicStudio HD. The orders of types B, D and E Manual Declicking have been doubled, yielding superior results when compared with SonicStudio 5.4. The pitch detection and interpolation of type A and C algorithms is also improved over that of SonicStudio.
- **Real-Time Denoise** permits noise removal in real time within SonicStudio HD.
- **AutoSonic™** is a suite of software tools that enable efficient batch processing of soundfiles either by distributed processors on a network or by a single SonicStudio HD workstation. Autosonic performs Sample Rate Conversion, Broadband Denoising, Complex Filtering, Production Declicking and Decrackling.

Manual Declicking

SonicStudio HD's Manual Declicking is designed to assist in removing unwanted noises such as clicks, pops and thumps. It offers five different interpolation algorithms that are capable of correcting even very difficult audio anomalies.

The algorithms analyze audio on either side of the anomaly and, based on this information, synthesize replacement sound. SonicStudio HD substitutes the replacement sound for the original sound, and stores the original sound in a special file called a Restore List. If you don't like the result of the interpolation, you can restore the original audio from this file.

Note: Manual Declicking operates on a single channel, so when working with multichannel material, you must declick each Track separately.

Interpolation Algorithms

There are several interpolators that are available in SonicStudio HD. Each is suited to a particular type of audio problem and context.

Note: The B type interpolator is the most general. The majority of declicking situations can be handled by simply choosing this selection.

TYPE B- GENERAL INTERPOLATION

This is the most general-purpose of the declicking algorithms, and works well on complex musical waveforms (for example, where several signals are combined with instruments that produce non-periodic waveforms, such as sax, strings, etc.).

The basic Type B interpolator examines the audio on either side of the click to determine the context for resynthesizing audio to fill the gap. In most situations this basic interpolator will produce the best results.

There are two variations of the command that ***load*** the context information in a particular way. If, for example, a click occurs just prior to the beginning of the attack of a piano note, the basic Type B interpolator would include part of the piano note in its resynthesis, producing the impression that the piano starts a bit early.

In this case, the B Type, L->R (left-to-right) variation would avoid getting the piano note into the interpolation. Likewise, the R->L (right-to-left) variation might be used in an instance where a click follows immediately after a sudden change in the audio waveform.

TYPES A AND C-PITCHED INTERPOLATION

The Type A and Type C interpolators are pitched waveform interpolators. A waveform interpolator is most useful in dealing with periodic waveforms, such as brass instruments or the human voice. The Type A and C interpolators take context information from six periods (the distance between successive peaks in the waveform) to the left and right of the area identified by the selection.

Note that after interpolation, the Restore Bars extend for a short distance outside the selected area. This is because the waveform interpolator performs a short-duration crossfade of its results back into the original sound. It is the area of the crossfade that extends beyond the selection boundaries.

The difference between these two types is that the Type C has protections built into it for certain cases for which waveform interpolation algorithm produces bad results. Type A lacks these protections. The Type A interpolator will ***usually*** produce an interpolation, but the results may not always be pleasing. Either the A or C Type interpolator can fail to find an acceptable waveform period to interpolate the signal designated, and will alert you to try a different algorithm. If this occurs, try a different interpolation algorithm or try selecting a slightly different section of audio and try the interpolation again.

TYPES D-DECLICKER AND E-DECRACKLER INTERPOLATION

These are very high-order interpolations that may be used to correct problems that elude other interpolation algorithms. Both interpolators use 80-bit precision arithmetic to produce very high quality interpolation.

The Type D interpolation is only capable of replacing up to about 2 milliseconds (0.002 seconds) of sound before it runs out of memory. The Type E Interpolator provides a very similar algorithm that can be used on large sections of audio.

If there is a particularly problematic area, then Type-E manual interpolation can help clean up small regions. As with the other interpolation algorithms, Type E operates directly on the sound file, and it produces restore bars to show where replacement has been performed.

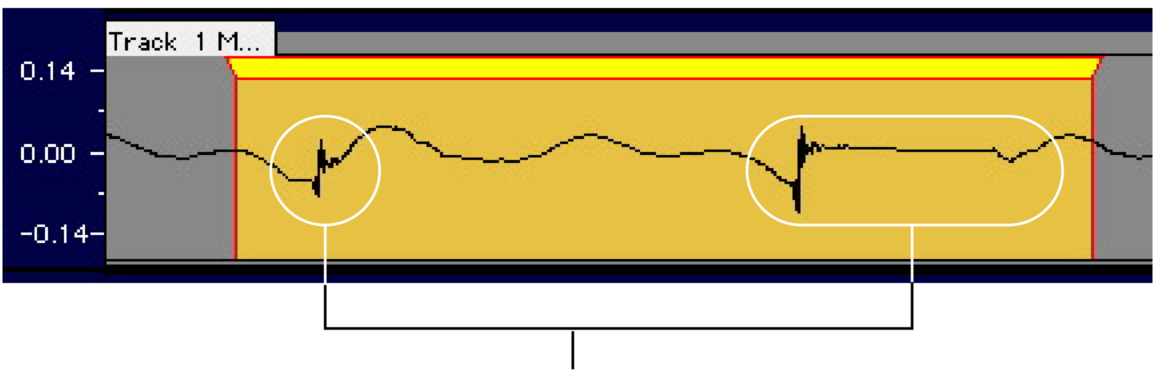
Removing Clicks

To remove clicks:

1 Using the Waveform display, identify the location of a click.

2 Zoom in until you see the click clearly.

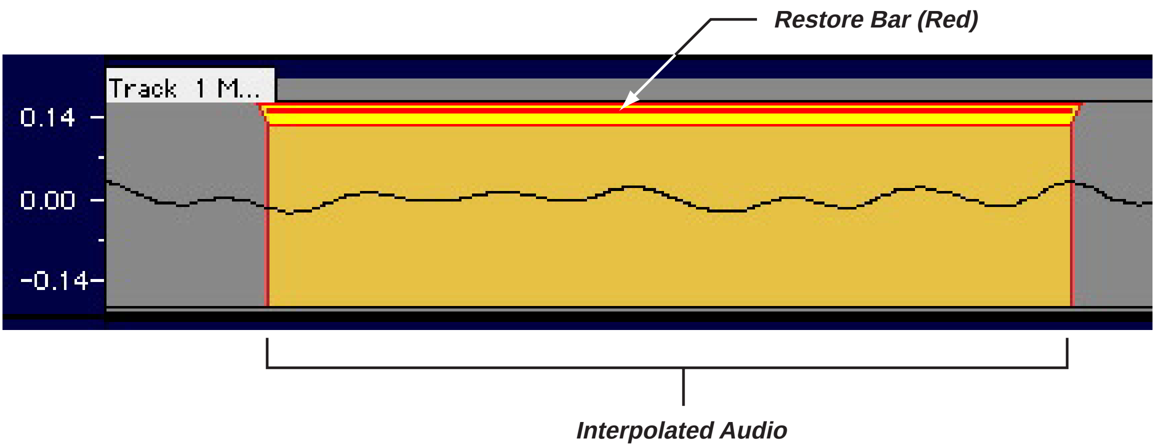
3 Drag in the Track to create a time selection that fully contains the click.



Clicks

4 From the **NoNOISE** menu, select one of the **Manual Declick** types.

SonicStudio HD replaces the click(s) with continuous, interpolated audio and marks the interpolated region with a red Restore Bar.



To restore a click:

1 Zoom in on the soundfile and locate the Restore Bar.

2 Drag to create a time selection that contains the Restore Bar.

3 From the **NoNOISE** menu, choose **Restore Click**.

SonicStudio HD replaces the interpolated audio with the original audio containing the anomaly.

Identifying Clicks

Clicks are impulsive in nature and, unlike a typical music signal, contain substantial high-frequency energy. You can take advantage of this by using SonicStudio HD's filters to create a click-detect file.

To create a click-detect soundfile:

1 With the original soundfile open in an EDL Track, select the entire soundfile (you can use either a time range or a Segment selection).

2 Open the HDSP Manager and drag a high pass filter onto the selection.

3 Double-click the filter in the Desk Events display and set the Frequency to about 10 kHz, the Stop to -40 dB and the Order to 1.

4 Set the Monitor Mode of the Track to **O** (Output) and note its output routing (M1 through M8).

5 Create a target Track for capturing the sound. It's easiest if the new Track is adjacent to the Track you want to work on.

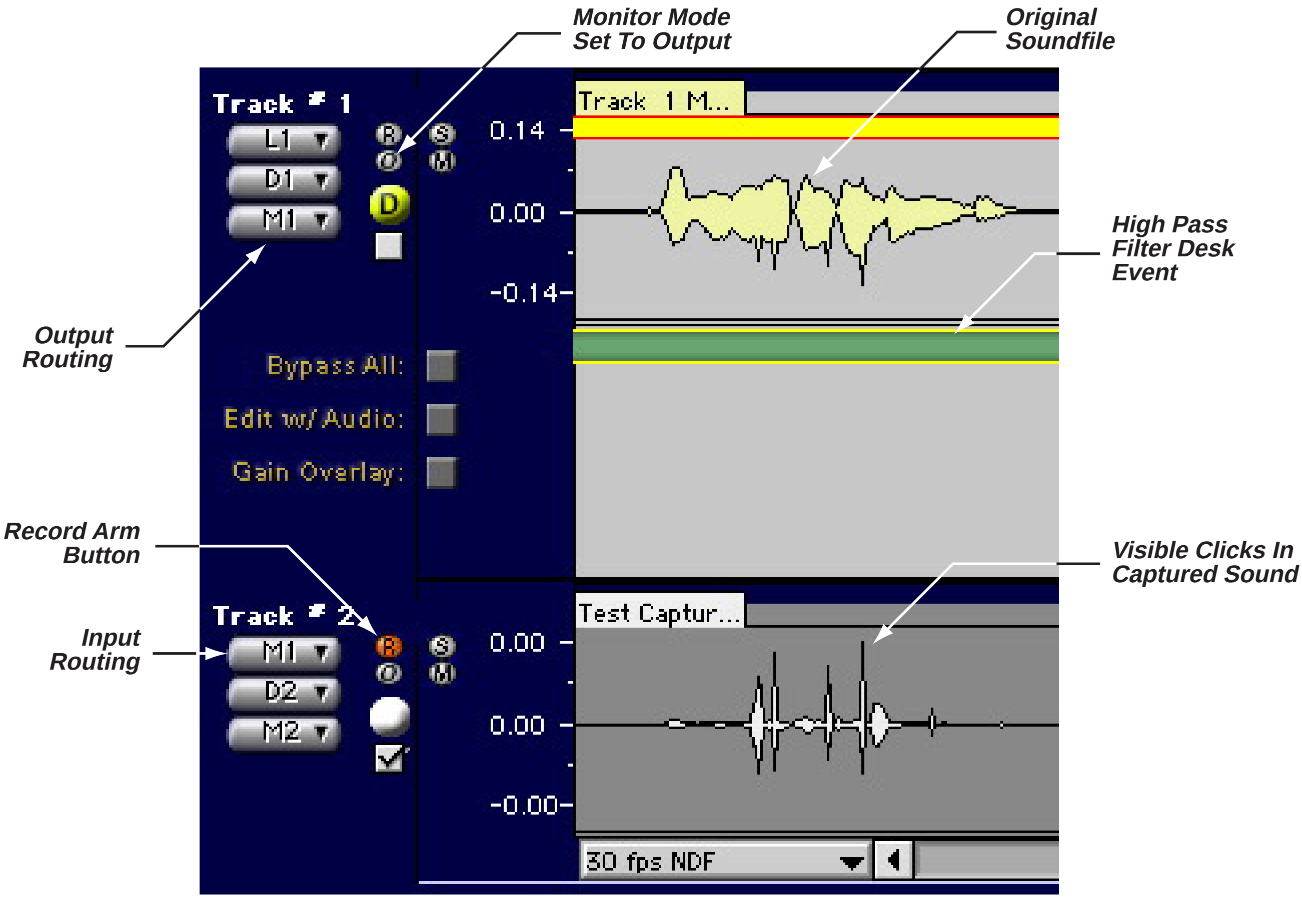
6 Set the new Track's Monitor Mode to **O** (Output) and arm it for recording by clicking the Record button. Set its Input to the output channel of the original soundfile Track.

7 Choose **Windows > New Soundfile Parameters** and select a destination volume for the captured sound.

8 Press **/** (slash).

SonicStudio HD plays the original sound, filters it and records it into the target Track.

When the recording is complete, you will see that clicks in the soundfile are highly visible in the captured sound waveform. You can select these regions and zoom to them by pressing COMMAND+**G** or choosing **View > Zoom to Selection**.


Monitor Mode Set To Output
Original Soundfile
High Pass Filter Desk Event
Output Routing
Record Arm Button
Input Routing
Visible Clicks In Captured Sound
Track # 1
L1
D1
M1
Bypass All:
Edit w/ Audio:
Gain Overlay:
Track 1 M...
0.14
0.00
-0.14
Track # 2
M1
D2
M2
Test Captur...
0.00
0.00
-0.00
30 fps NDF

Realtime Denoising

With a second HDSP Processor card in the system, the denoising algorithm can operate in real time within SonicStudio HD, so that you can adjust the denoising parameters and evaluate the results instantaneously. The parameter and Noise Estimate settings can be stored and used to process the file in AutoSonic. Alternately, the output of the realtime denoising may be transferred directly to an output medium, or to a new sound file using a Capture record.

Note: Real-Time Denoising is also available in single-board systems, but is limited to a single channel of processing.

To use Real-Time Denoising:

1. Open the soundfile that you wish to process into an EDL.
2. Create a time selection or Segment selection to define the portion of the soundfile that you wish to process.
3. Choose **Windows > HDSP Manager** or press COMMAND+**6**.
4. In the HDSP Manager, click on the **Aux** tab.
5. Drag an instance of the **Real-Time Denoise** plug-in onto the selection in the EDL.

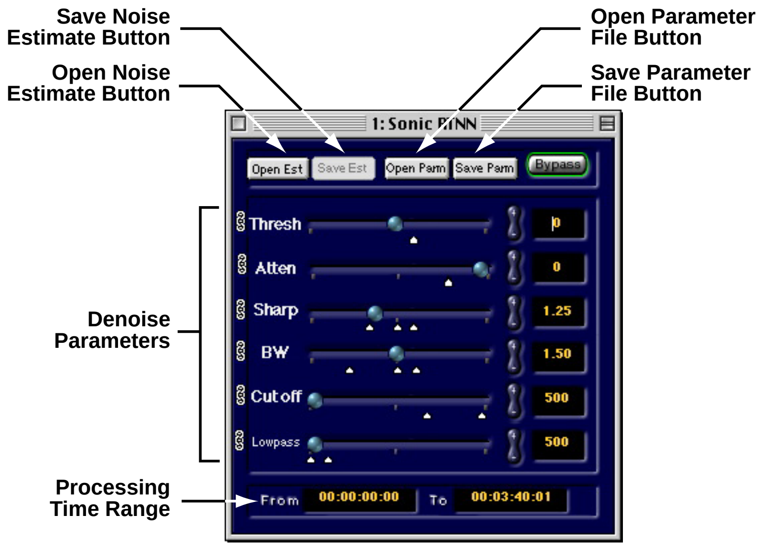
   SonicStudio HD opens the Desk Events panel for the selected Track and displays the Denoise event with default parameters.

Note: SonicStudio HD applies the Real-Time Denoise plug-in to all Tracks in the Edit Group. By default, it displays the Desk Event panel for the selected Track only. You can display Desk Event panels for another Track in the Group by clicking in the Track to select it and choosing **Track > Show Desk Events** or pressing OPTION+**D**.

To edit the Real-Time Denoise parameters:

1 Double-click the Real-Time Denoise Desk Event.

SonicStudio HD displays the Real-Time Denoise Parameters window.



Use the sliders to adjust the Real-Time Denoise parameters, or select parameter numeric fields and type the desired values (see ***Broadband Denoising Parameters*** on page 44 for a detailed discussion of Denoise parameters).

By default, parameter changes apply to all Tracks in the Edit Group. In denoising multi-track files that incorporate numerous takes with different mic positions and processing, it is necessary to denoise Tracks individually. You can edit one Track's parameters separately by unlinking one or more parameters.

To unlink a parameter:

- Click on the chain-link icon to the left of the parameter name.

SonicStudio HD displays a “broken link” icon. Edits made to the parameter now apply only to the selected Track.

READING A NOISE ESTIMATE

For the denoising process to work meaningfully, it must be supplied with a Noise Estimate. Generally, the Noise Estimate used will be one prepared from the source material itself (see ***Taking An Estimate*** on page 39).

To open a Noise Estimate for real-time denoising:

1 Click the button labeled **Open Est**.

SonicStudio HD displays a Macintosh Open File dialog.

2 Navigate to the Estimate file, click to select it and click **OK**.

SonicStudio HD opens the Noise Estimate file.

SAVING THE PROCESSING PARAMETERS

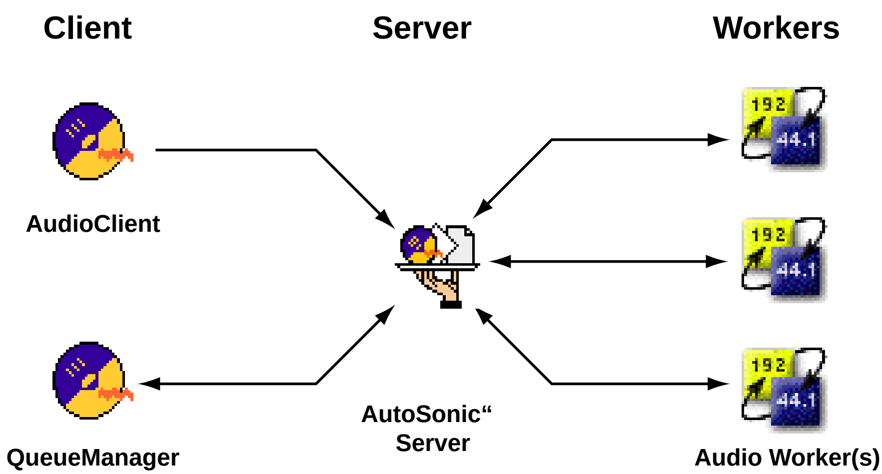
By carefully adjusting the Real-Time Denoise parameters (especially the Threshold and Attenuation controls) and listening to the results, you can optimize the broadband denoising to remove the greatest amount of objectionable noise while avoiding undesirable artifacts. The **Bypass** button can also be used to compare processed versus unprocessed audio. Sets of processing parameters may be saved to disk and recalled by using the **Open Parm** and **Save Parm** buttons.

Parameter settings and Noise Estimates are compatible with those used in Background Denoise operations, so you can read and write settings to be used for background processing. This is convenient because it allows real-time denoising to be used for testing. Once the desired settings are found, the complete processing job can be launched in AutoSonic and the results recovered at your convenience.

AutoSonic

AutoSonic is a suite of software tools that enable efficient batch processing of soundfiles either by distributed processors on a network or by a single SonicStudio HD workstation. Autosonic consists of four applications: AudioClient, QueueManager, AutoSonic Server and Audio Worker.

This diagram illustrates the relationships among the four AutoSonic components:



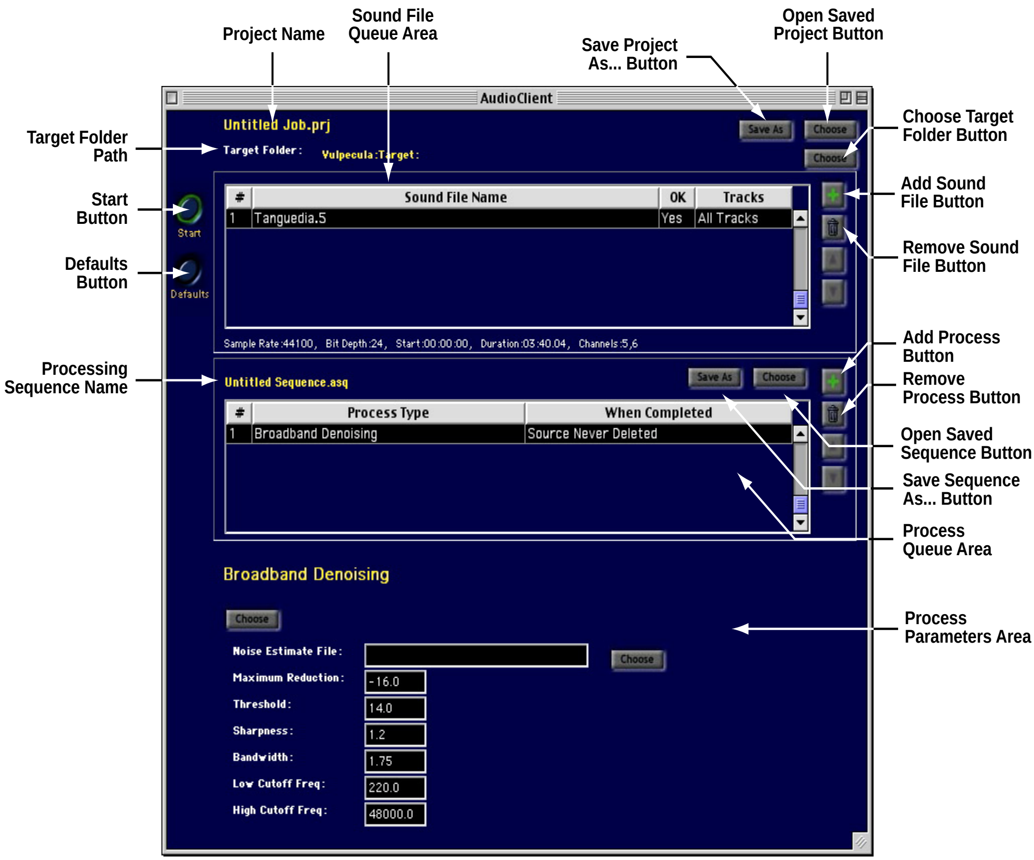
- **AudioClient** defines Projects consisting of soundfiles and associated processes (Declicking, Broadband Denoising, Decrackling and Sample Rate Conversion), and communicates the Projects to AutoSonic Server.
- **AutoSonic Server** directs the execution of Projects defined by AudioClient. AutoSonic Server can mediate Projects from a number of separate AudioClients simultaneously.
- One or more **Audio Workers** perform the required audio processing tasks as directed by AutoSonic Server, reporting the job progress back to the Server.
- **QueueManager** communicates with AutoSonic Server, monitoring the progress of all Projects in the system. QueueManager provides both visual feedback about, and control over, the process queue.

All four applications can run on a single SonicStudio HD workstation, or the Client, Server and Worker(s) may run on separate, networked computers. In the networked case, the Client, Server and Queue Manager do not need an HDSP Processor card. In all cases, Audio Worker must run on a Power Macintosh G3, G4 or better equipped with at least one HDSP Card, and all soundfiles must reside on a hard drive mounted on the desktop of that workstation.

Note: When running AutoSonic on a single workstation, first launch AutoSonic Server, then launch the Client, Worker and Queue Manager applications. To do so automatically, run the application **AutoSonic Startup** and choose which applications to run. AutoSonic Startup will launch them in the correct order.

Using AudioClient

AudioClient assembles Projects that consist of one or more soundfiles and one or more NoNOISE processes that will be applied to all soundfiles in the Project.



The upper panel of the AudioClient window provides information and controls for the current Project. The middle panel provides information and controls for the current Project's processing, and the lower panel displays parameters for the selected process.

Projects and Sequences can be saved and recalled separately:

- Saving the current audio sequence saves the processing sequence in a .asq file. You can then recall that set of processes and apply them to a different set of soundfiles.
- Saving the current Project saves both the soundfile sequence and the processing sequence in a single .prj file.

SETTING UP AUDIOCLIENT

To set up AudioClient:

1 Launch AudioClient.

2 From the **Control** menu, choose **Server** or press COMMAND+**S**.

AudioClient display the Server dialog.

- If you are running AutoSonic Server on the same machine, make sure that the **Check Local First** option is selected, then click **OK**.
- If you are using a remote server, type the server's IP address in the **Server IP Address** field, then click **OK**.

Note: When the **Check Local First** option is selected, AudioClient at launch time always looks for a server on the local machine. If it fails to find one, then it checks at the Server IP Address that you supplied. If you know that you will never use a local server, then clear the **Check Local First** checkbox. This will speed AudioClient's launch, since it will eliminate the step of checking for a local server.

If AudioClient fails to find a server at launch time, it will operate in "local" mode. You will be able to create and save Projects and Sequences, but you won't be able to send them to a server.

SETTING UP A PROJECT

To set up a Project:

1. Launch AudioClient. AudioClient displays an empty Project.

2. Click the **Choose Target Folder** button at the top of the Audio Client window.

3. Navigate to the folder where you wish the output soundfile to be saved, click on the folder to highlight it and click **Choose**.

4. Click the **Add Soundfile** button (plus sign icon). AudioClient displays a Get File dialog.

5. Navigate to the soundfile and select it, then click **Open** or press RETURN.

   AudioClient adds the soundfile to the sequence and displays its name in the upper Soundfile Name area.

Tip: You can also add soundfiles by dragging them from the Finder onto the Process Parameters panel (see ***Drag and Drop Support*** on page 18 for further information).

- You can add additional soundfiles to the sequence. The Project's processing sequence will be applied to all soundfiles in the queue.
- You can remove a soundfile from the sequence by clicking its name to select it and then clicking the **Remove Soundfile** button (trashcan icon).
- You can rearrange the order of soundfiles in the list using the up/down arrow buttons.

The **Tracks** column displays which Tracks of the file will be processed (the default is **All Tracks**). You can direct AutoSonic to process selected Tracks of a multi-track file; see ***File Processing Options*** on page 16 for more information.

6. Click the **Add Process** button (plus sign icon). AudioClient displays a dialog for choosing a process. From the pop-up menu, choose Declicking, Broadband Denoising, Sample Rate Conversion or Decrackle, then click **Add**.

AudioClient adds the process to the sequence and displays its name in the Process Sequence area.

7 Edit the Processing Parameters to the desired values, or click the **Defaults** button to select the default values.

You can add additional processes to the sequence. The Project's Processing Sequence will execute in the order that processes appear in the queue.

You can remove a process from the sequence by clicking its name to select it and then clicking the **Remove Process** button (trashcan icon).

8 To save the Project, click the upper **Save As...** button. AudioClient displays a Save As... dialog. Name the Project and click **Save**.

Optionally, you can save the Processing Sequence separately by pressing the lower **Save As...** button.

9 To execute the Project, click the **Start** button. AudioClient passes the Project information to AutoSonic Server.

FILE PROCESSING OPTIONS

To edit the processing options for a soundfile in the Source queue:

1 Double-click the soundfile's name.

Audio Client displays the **File Options** dialog.

2 Use the pop-up menu to select your preferred time display:

- **Standard** (hour:minutes:seconds)
- **Seconds**
- **Samples**

3 For multi-channel sources, you can select which channels to process by clicking the corresponding checkboxes. You can also click **All On** to select all channels (the default), or **All Off** to deselect all. You must then select at least one channel for processing (this is useful if you wish to process only one channel of an eight-channel file).

4 You can select a specific time range within the soundfle for processing by typing values in the **Time Period To Process** fields. The default setting is the full time range of the file.

5 Click **OK** to apply your changes, or **Cancel** to revert them.

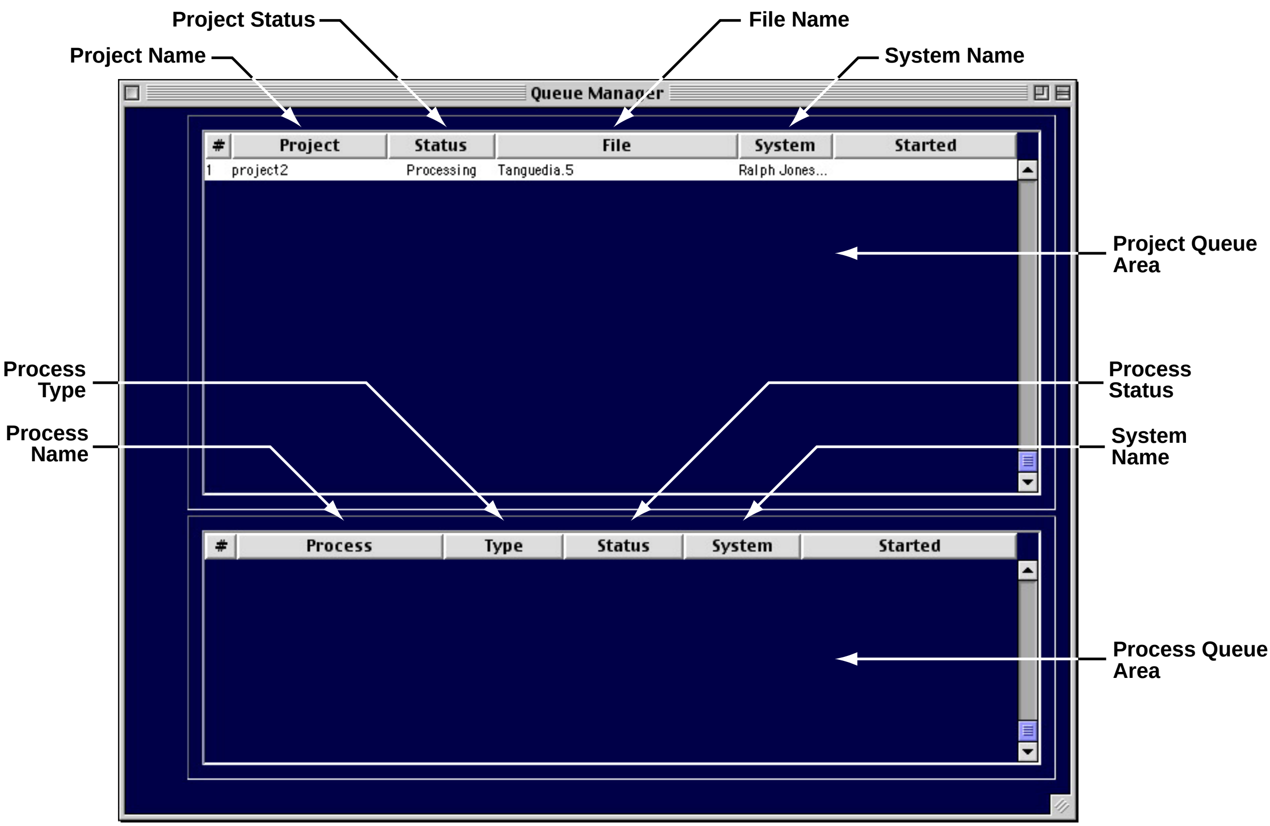
Drag and Drop Support

You can also drag and drop files from the Finder to set up an Audio Client Project. Drag files to the **Process Parameters** area of the window; Audio Client will automatically add them to the Project in the proper place.

- **.prj files** — To run a previously-saved Project, drag and drop the .prj file.
- **.asq files** — To run a previously-saved processing Sequence, drag and drop the .asq file.
- **Soundfiles** — To add soundfiles to the Source queue, drag and drop them in the Parameters area.
- **Filter Spec files** — When you select Declick processing, you can drag in a filter spec file (see **The Filter Spec File** on page 67.).
- **Noise Estimates** — When you select Broadband Denoise processing, you can drag in a Noise Estimate file saved from SonicStudio HD (see **The Noise Estimate** on page 37.).

Using QueueManager

QueueManager is a client-side application that allows you to monitor Projects that are in progress. QueueManager communicates with the AutoSonic Server, polling it at regular intervals and receiving status information from it.



To set up QueueManager:

1 Launch QueueManager.

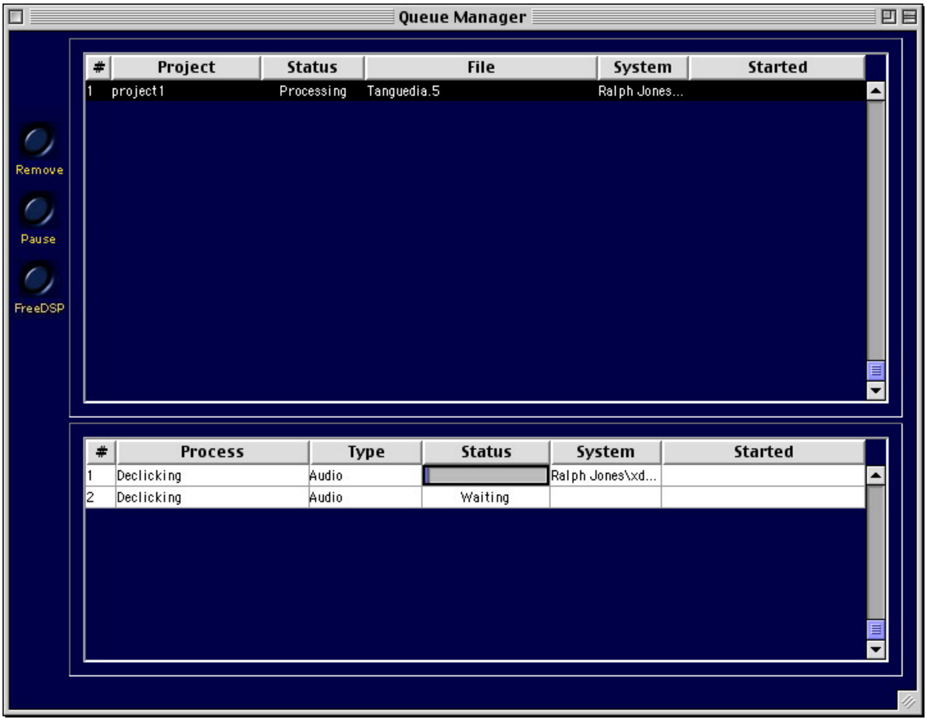
2 From the **Control** menu, choose **Server** or press COMMAND+**S**.

QueueManager displays the Server dialog.

- If you are running AutoSonic Server on the same machine, make sure that the **Check Local First** option is selected, then click **OK**.
- If you are using a remote server, type the server's IP address in the **Server IP Address** field, then click **OK**.

QueueManager connects to AutoSonic Server.

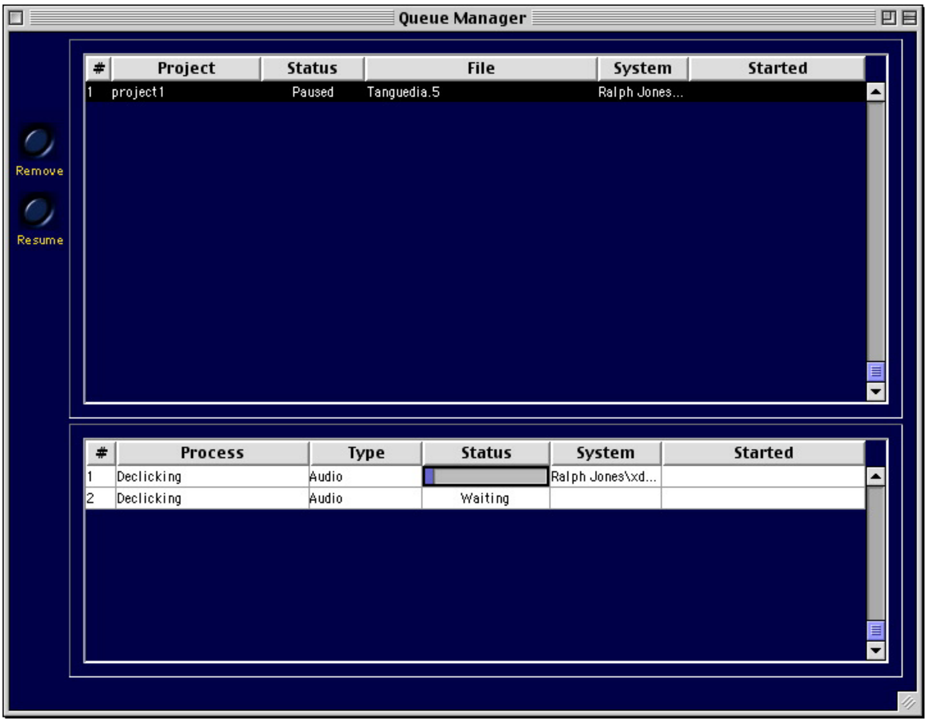
Current Projects display in the upper Project Queue area. If you click on a Project to select it, the Process Queue area displays the processes in the selected Project and their current status:



- Click **Remove** to halt processing and remove the Project from the queue.
- Click **Pause** to temporarily stop processing.

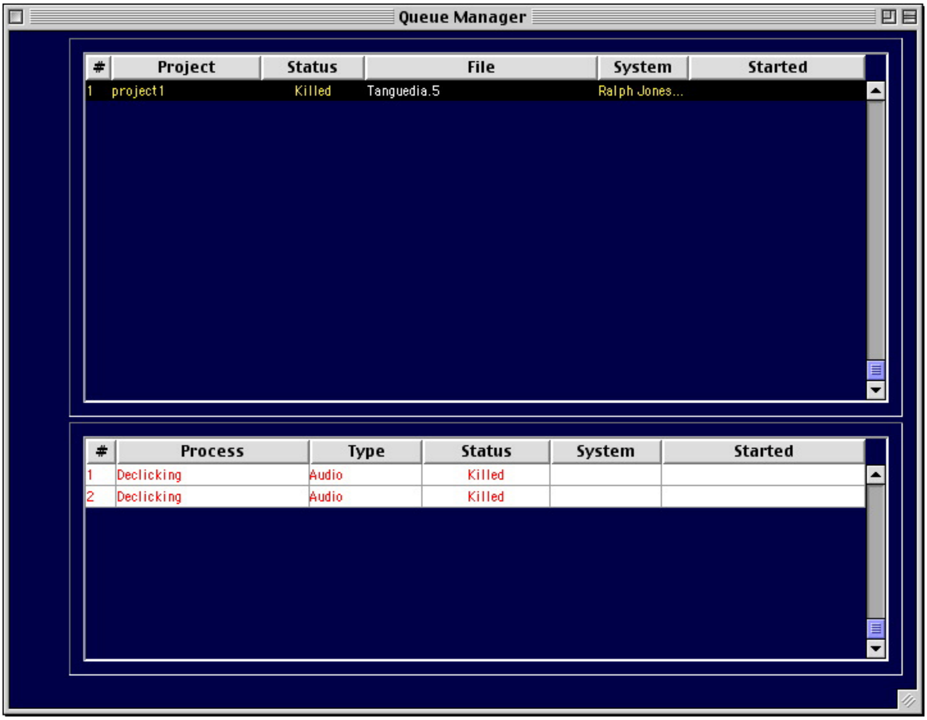
- Click **Free DSP** to pause the Project and make its DSP available to another Project or to SonicStudio HD.

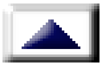
When you Pause a Project, Queue Manager displays its status as **Paused**:



- Click **Remove** to halt processing and remove the paused Project from the queue.
- Click **Resume** to continue processing the paused Project.

When you remove a Project from the queue, Queue Manager displays it Status as "Killed."





Using AutoSonic on a Single Workstation

AutoSonic includes an application called **AutoSonic Startup**. Its purpose is to automate launching of the AutoSonic suite of programs on a stand-alone workstation.

When you start AutoSonic Startup, it presents a dialog:

Choose the AutoSonic applications that you wish to use, then click **Launch**. AutoSonic Startup launches the applications in the correct order.

SPECIAL INSTRUCTIONS REGARDING MACTCP/IP

If you are using AutoSonic on a single, stand-alone workstation, you must create a special TCP/IP configuration.

To set up TCP/IP for stand-alone AutoSonic:

1. From the Apple Menu, choose **Control Panels > TCP/IP**.
2. With the TCP/IP Control Panel in the foreground, choose **File > Configurations...** or press COMMAND+**K**.

Mac OS displays the Configurations dialog.

3 Highlite the Default configuration and click **Duplicate...**

4 Name the new configuration AutoSonic and click **OK**, then click **Make Active.**

5 In the **Connect via:** pop-up menu, choose **AppleTalk (MacIP).**

6 In the **Configure:** pop-up menu, choose **Using MacIP Manually.**

7 Enter a random IP Address in the form xxx.xx.xx.xx (use only numbers smaller than 255).

8 Enter a different, random Router Address in the form xxx.xx.xx.xx (use only numbers smaller than 255).

9 Enter a third, random Name Server Address in the form xxx.xx.xx.xx (use only numbers smaller than 255).

10 Close the TCP/IP Control Panel.

Note: The IP, Router and Name Server addresses must each be unique numbers.

Sample Rate Conversion

AudioClient allows you to choose either Standard sample rate conversion (SRC) or Sonic's High Density conversion.

- Standard SRC accommodates pull-down sample rates and is fast.
- High Density SRC provides the highest possible fidelity and considerable facilities to optimize the conversion. High Density SRC takes somewhat more time than Standard SRC to process a given soundfile.

Select Standard SRC or HD SRC by clicking the corresponding radio button in the Parameters Area of the AudioClient window.

Standard SRC has no adjustable parameters other than the target sample rate. It will perform any sample rate conversion within an octave up or down.

High Density SRC adds some options that allow you to fine-tune the conversion for the highest sound quality:

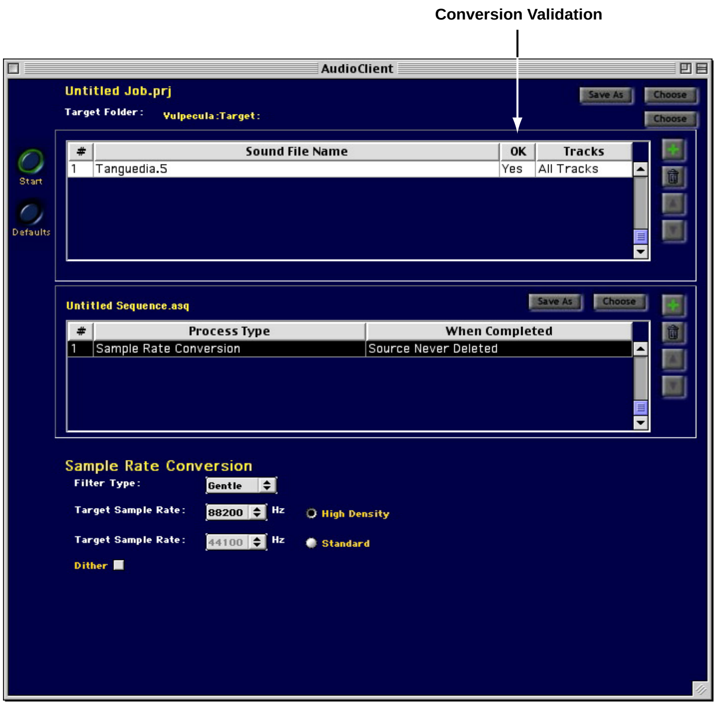
- You can choose from three different decimation/interpolation filter curves.
- You can choose whether or not to add dither.

The following table lists the legal conversions for HDSRC:

Input Sample Rate	Output Sample Rate
192 kHz	176.4 kHz
	96 kHz
	88.2 kHz
	48 kHz
	44.1 kHz

Input Sample Rate	Output Sample Rate
176.4 kHz	96 kHz
	88.2 kHz
	48 kHz
	44.1 kHz
96 kHz	88.2 kHz
	48 kHz
	44.1 kHz
88.2 kHz	96 kHz
	48 kHz
	44.1 kHz
48 kHz	96 kHz
	88.2 kHz
	44.1 kHz
44.1 kHz	96 kHz
	88.2 kHz
	44.1 kHz

The Audio Client soundfile list includes a column labeled **OK**. The values in this column reflect the validity of the selected conversion:



- **Yes** indicates that the conversion is legal (i.e. listed in the above table).
- **No** indicates that the conversion is illegal (not listed in the above table).
- **N/A** indicates that the source soundfile has the same sample rate as the destination.

Selecting a Filter Curve

The heart of the sample rate conversion process is the low pass filter, which is designed to reject aliases (down conversion) or images (up conversion) — both of which are detrimental to sound quality. The filter's response above cutoff must be very steep if it is to be effective at rejecting these artifacts.

Very steep filters can introduce audible distortions, however. Any filter design for sample rate conversion therefore involves tradeoffs among a number of factors: aliasing/imaging, inband/stopband ripple, reduced gain at very high frequencies, and time smear. Each of these factors has effects that can vary with the sonic characteristics of the material: sometimes it may be preferrable, for example, to allow a small amount of aliasing in order to preserve other aspects of signal integrity.

To provide the most flexibility in tailoring the conversion process to different types of program material, HD SRC offers three different filter choices: Steep, Gentle and Gentlest. The Steep option is a Nyquist filter optimized for maximum alias/image rejection. The other options trade a limited amount of aliasing/imaging for reductions in other distortions.

Your choice of a filter curve for a given rate conversion operation should be based on careful audition. For more information, see "About HDSRC" on page 31.

About Dither in HD SRC

HD SRC performs sample rate conversion with 48-bit precision, and the final step in the process of generating an output file is to round the output to produce a 24-bit file. To preserve fine sonic details such as reverberation tails, HD SRC affords the option of applying triangular dither to the output file. To assure the highest sound quality, Sonic recommends that you apply dither whenever you use HD SRC.

About the Output Soundfile

Amplitude — The digital filter that SRC uses exhibits a small amount of amplitude ripple near the cutoff frequency (this is known as the "Gibbs Phenomenon" and is a characteristic of all FIR low-pass filters). Left uncompensated, this ripple could cause source material with high frequencies recorded at high levels to become clipped.

Note: For this reason, SRC reduces the gain at the input of the conversion by 0.1 dB to avoid clipping. Repeatedly converting the same file therefore will gradually reduce its overall level.

Delay — All digital filters introduce propagation delay, and this will be reflected in the output soundfile. Propagation delay is constant with frequency, and does not affect sound quality.

Bit-Depth — Regardless of the input bit-depth, the output soundfile will be a 24-bit file. This preserves the full dynamic range of your soundfile throughout the conversion process. You can use SonicStudio HD's dither processing in the HDSP Manager to change the bit depth if necessary.

Classes of Rate Conversion

AutoSonic performs two classes of upward or downward rate conversion: integer ratio and fractional ratio.

INTEGER RATIO CONVERSION

In integer ratio conversion, the source and output sample rates bear a simple, integer relationship to one another (for example, 44.1 kHz to 88.2 kHz is 1:2). Integer ratio conversions are the most straightforward.

Integer ratio down conversion is achieved by omitting every other sample, in a process known as ***decimation***, and is illustrated in Figure 1. Here, curve A shows the original analog waveform with its higher-sample-rate digital representation superimposed. Curve B is the down conversion result.

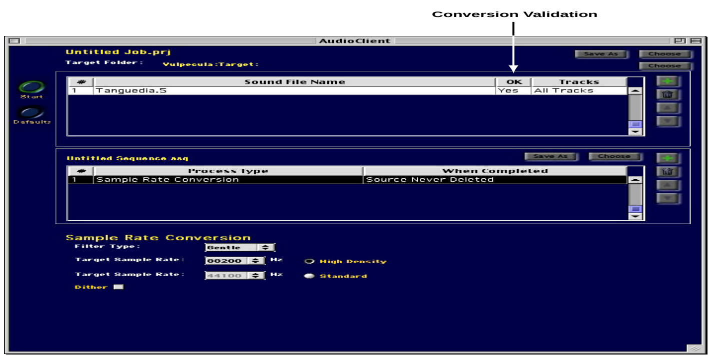


Figure 1 Integer Ratio Down Conversion

Because decimation causes ***aliasing*** (folding of frequencies above the Nyquist frequency back into the audible band), down conversion requires a steep lowpass filter which cuts off at the Nyquist frequency of the new, lower sample rate. (The Nyquist frequency is one-half the sample rate; it determines the usable bandwidth of the system.) For computational efficiency, the decimator and lowpass filter are normally combined in a single processing step.

For upward sample rate conversion, the converter needs to create new sample values that are evenly spaced between the source samples. This process is called ***interpolation***, and is illustrated in Figure 2.

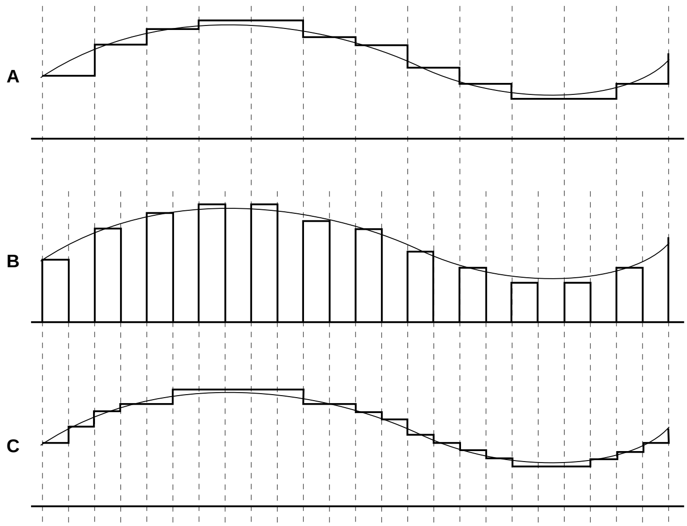


Figure 2 Integer Ratio Up Conversion

You can think of up conversion as occurring in two steps:

- First, zero-value samples are added between input samples at the new, higher sample rate (Figure 2 B).
- Then, the signal is filtered using a steep lowpass which cuts off at the Nyquist frequency of the old, lower sample rate.

The up conversion filter is called an ***interpolator***. Besides providing correct values for the new samples, it also prevents ***images*** of the input spectrum from appearing in the new, wider bandwidth.

FRACTIONAL RATIO CONVERSION

Fractional rate conversion takes advantage of the fact that, where the source and output sample rates are not related by a simple integer, there is nonetheless a higher clock rate to which both ***will*** have an integer relationship. This is illustrated in Figure 3.

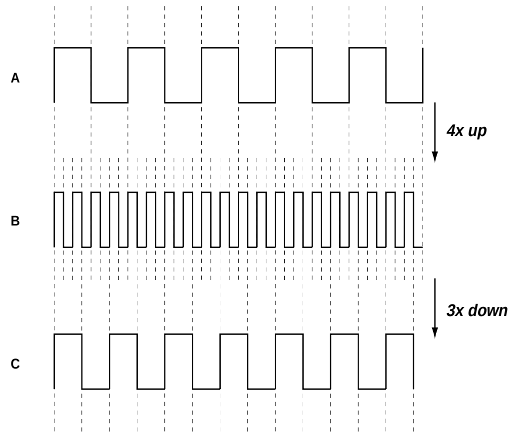


Figure 3 Fractional Ratio Conversion

Here, the ratio between clock rate A and intermediate rate B is 1:4. In turn, the ratio between clock rate B and rate C is 3:1. Oversampling at clock rate B therefore enables conversion between signals sampled at rate A and those sampled at rate C.

In practice, fractional-rate conversion is performed by an FIR lowpass filter which cuts off at the Nyquist frequency of the lower of the two sample rates (source or output) and is oversampled at some high frequency to which both sample rates have an integer relationship. The Nyquist filter prevents aliases (down conversion) or images (up conversion).

About HDSRC

The double precision sample rate algorithms in AutoSonic's HD SRC mode provide conversions between each of the commonly-used sample rates, and produce 24-bit output in all cases (which should be dithered to the appropriate release resolution if other than 24 bits). The algorithms are designed with careful attention to filter parameters and computation precision in order to preserve sound quality to the highest extent possible.

Because the multiphase filters needed for the noninteger conversions (e.g. 96 kHz:44.1kHz) require very large numbers of coefficients and customizing of dsp resources, the filters cannot be designed "on-the-fly." In order to offer choice in the sound of the filters, three versions of each filter are offered; the characteristics each version are described in this section.

SAMPLE RATE CONVERSION IN BRIEF

All rate conversions rely on a relatively simple process, in which the source data is filtered with a low pass filter (lpf) with bandwidth determined nominally by the Nyquist frequency of the lower of the two frequencies (source frequency, destination frequency).

For noninteger conversions, the filter is oversampled, usually at a high rate, to produce a set of coefficients which are then used to filter the data, but only at the output rate determined by the destination frequency. The upconversion and downconversion factors are defined by the equation:

$T = (M/L)T'$

where the integers M and L provide the smallest whole integers relating the new (T') and old (T) sample periods. The oversampling, or upconversion, rate is M/T.

The ratios 96:44.1 and 44.1:96 require high oversampling (M=147 and M=320, respectively) and hence substantially more complicated implementation than the simple decimation/interpolation ratios 88.2:44.1 and 44.1:88.2 (M=1 and M=2, respectively). However, the conversions are based on the same low pass filters, and as long as the same filter design approach is used and high precision is maintained in the computations, the sonic performance should be comparable[1].

SONIC ISSUES

Data Precision

Filter coefficients in these algorithms are calculated using 64-bit floating point arithmetic, converted by rounding to 48-bit integer, and downloaded as 48-bit integers to dsp processors. The audio data (internal word length 24 bits) is convolved with the filter coefficients using a double precision multiply algorithm. The output is rounded (not dithered) from 48-bit to 24-bit integer under the assumption that the roundoff error spectrum is below audibility and that the output will be dithered to the final bit depth. This implementation preserves 24-bit precision in the output data, even with very long (~300 coefficient) filters.

Low pass Filter Parameters

On mathematical grounds, low pass filters requiring steep rolloff in order to attenuate aliases or images just outside the audio band will be fairly long. Depending on the design, a strictly Nyquist filter for 96 kHz:44.1 kHz downconversion having flat inband gain to 20 kHz and stopband attenuation at 22.05 kHz of -120 to -145 dB, has about 220-300 coefficients (or equivalently a length of 2.3 ms - 3.1 ms in time).

Other things being equal, the longer the filter, the greater the potential for smearing of the natural time relationships in the audio source. The filter can be shortened by relaxing the steepness of the rolloff, but at the expense either of reducing gain at the upper edge of the audio band or by allowing aliasing.

From a sonic standpoint, aliasing, reduced gain at 20 kHz, and increased inband/stopband ripple all create audible distortions which, at significant levels, may be greater factors than the less objectionable effects of filter length. On the other hand, small amounts of certain distortions may be tolerated with various audio materials in order to minimize the effects of filter length.

1.Apart from ease of implementation, the advantage of simple decimation ratios, if these are the only ratios of concern, is that optimal filter design approaches such as the Remez discrete Chebyschev approximation can be used to compute the filter.

The filter options below are given in order to permit adaptability to different source data and converter/playback systems. Filters in each group have gain at 20 kHz < 0.2 dB and stopband attenuations about 130-135 dB below full scale. The A option is the Nyquist filter, while the B and C options typically trade a limited amount of aliasing for shortened length.

In many cases, the shorter filters offer clear benefits and should be judged by audition. The rationales for the levels and choices are discussed in a preprint published by the 105th AES Convention, San Francisco, Sept, l998 ("Multiphase Filters for Sample Rate Conversion of High Resolution Audio").

PARAMETERS OF SAMPLE RATE CONVERSION FILTERS

Conversion	Filter	Length (ms)	Gain at 20 kHz (dB)	Gain at Nyquist (dB)	Gain (dB), Frequency (Hz) at Stopband
96kHz > 44.1 kHz	A	2.93	-0.34	-124	22.05 k
	B	2.59	-0.19	-50	-134 @ 22.6 k
	C	2.18	-0.2	-30	-134 @ 23.1 k
88.2 kHz > 44.1 kHz	A	3.14	-0.16	-126	22.05 k
	B	2.82	-0.13	-57	-135 @ 22.5 k
	C	2.32	-0.08	-28	-135 @ 23.1 k
96 kHz > 48 kHz	A	1.80	-0.08	-138	24 k
	B	1.43	-0.1	-50	-137 @ 25 k
	C	1.22	-0.08	-28	-136 @ 26 k
88.2 kHz > 48 kHz	A	1.87	-0.06	-137	24 k
	B	1.51	-0.05	-50	-137 @ 25 k
	C	1.28	-0.03	-29	-137 @ 25.9 k
48 kHz > 44.1 kHz	A	3.19	-0.2	-125	22.05 k
	B	2.85	-0.1	-55	-135 @ 22.5 k
	C	2.27	-0.14	-30	-135 @ 23.05
96 kHz > 88.2 kHz	A	0.34	-0.03	-144	44.1 kHz
96 kHz > 88.2 kHz	B	0.26	-0.04	-44	-144 @ 51.5 k
	C	0.22	-0.01	-21	-144 @ 59.1 k
44.1 kHz > 96 kHz	A	3.47	-0.16	-140	22.05 k

Conversion	Filter	Length (ms)	Gain at 20 kHz (dB)	Gain at Nyquist (dB)	Gain (dB), Frequency (Hz) at Stopband
	B	3.11	-0.1	-62	-145 @ 22.5 k
	C	2.56	-0.1	-35	-145 @ 23 k
44.1 kHz > 88.2 kHz	A	3.47	-0.16	-140	22.05 k
	B	3.11	-0.1	-62	-145 @ 22.5 k
	C	2.56	-0.1	-35	-145 @ 23 k
48 kHz > 96 kHz	A	1.85	-0.12	-141	24 k
	B	1.52	-0.1	-55	-141 @ 25 k
	C	1.27	-0.08	-30	-141 @ 26 k
48 kHz > 88.2 kHz	A	1.85	-0.12	-141	24 k
	B	1.52	-0.1	-55	-141 @ 25 k
	C	1.27	-0.08	-30	-141 @ 26 k
44.1 kHz > 48 kHz	A	3.47	-0.16	-140	22.05 k
	B	3.11	-0.1	-62	-145 @ 22.5 k
	C	2.56	-0.1	-34	-145 @ 23 k
88.2 kHz > 96 kHz	A	0.33	-0.06	-143	44.1 k
	B	0.28	-0.05	-61	-142 @ 48.7 k
	C	0.19	-0.1	-21	-140 @ 61 k

Peak inband ripple for all filters varies from 0.000005 dB to 0.0000005 dB.

Broadband Denoising

Broadband noise, or hiss, is one of the most common forms of audio degradation. Noise can be introduced from any of a number of sources, including the noise floor inherent in analog tape recording and thermal noise from microphones, preamps, and other processing equipment. To eliminate hiss and other noises, it is necessary to analyze noise content and adapt the denoising operation to the characteristics of the material.

Broadband Denoising operates by means of analysis and resynthesis. A Fast Fourier Transform (FFT) frequency analysis is performed on a sample of noise from the material to be processed. The level of noise in each of 2048 individual frequency bands is determined. The output of this analysis is a ***Noise Estimate.***

In actual denoising, the source material is also subjected to a 2048-point FFT analysis. The level of signal in each frequency band (or ***bin,*** in Sonic terminology) is compared against a threshold level determined by the Noise Estimate. Based on this comparison, the processing algorithm determines whether a given band at that particular instant contains audio signal or only noise.

If a frequency bin is found to include elements of the desired signal, it is left untouched. If it is determined that the signal in that band is only noise, the level of that band is reduced by an amount determined by the Attenuation processing parameter. The results of this comparison and adjustment for all bands is a modified version of the original FFT frequency analysis.

A ***reverse FFT*** is then performed using the new, adjusted version of the signal analysis, reconstituting the audio signal with noise attenuated by the specified amount. Because the Denoiser operates in more than 2,000 individual bands, the removal of noise is precise and can leave the original audio signal unaffected.

Note: Broadband Denoise can process only **one** soundfile at a time. If you try to create an AutoSonic Denoise Project with several soundfiles in the Queue, only the first file in the list will be processed.

The Noise Estimate

The first step in denoising is to derive a Noise Estimate from the recording to be denoised. The Noise Estimate is the fingerprint of the noise as analyzed from the sound waveform display. It determines local threshold values for each frequency band, or bin.

SonicStudio HD's **NoNOISE** menu provides a group of commands for creating, processing, editing and storing the Noise Estimate. The Noise Estimate determines the result of the entire denoising process, so it is important to ensure that the estimate taken is valid and represents the true noise floor of the source sound file.

The procedure for taking a usable Noise Estimate has several steps:

1 Open the source sound file into an Edit Decision List.

2 Identify a short (0.3 to about 0.5 seconds) section of audio where there is only noise. Drag to create a time selection of the noise.

3 From the **NoNOISE** menu, select the command **Take Noise Estimate**.

- You can also choose **Take and Interpolate Estimate** to smooth the raw frequency analysis data.

4 If necessary, edit the interpolated Noise Estimate.

5 Select the command **Write Estimate...** to save the estimate as a file that can be used by AutoSonic Broadband Denoise.

Selecting Where to Take the Noise Estimate

Once the source sound file is opened into a panel of an Edit Decision List, the first step is to identify a suitable location from which to take the Noise Estimate.

Note: The denoising algorithm depends on a constant level and spectrum in the noise floor. Noise floors are seldom constant except within single pieces of recorded music (and sometimes even not then).

It is usually necessary to derive a separate Noise Estimate for each cut or take. If these are contained in a single sound file, then that file should be denoised in sections, so that the optimal set of estimates and parameter settings can be used for each cut.

Unless there is strong reason to believe that each cut in a compilation was:

- recorded in the same session
- with the same equipment
- at precisely the same levels
- onto the same media
- stored in the same way
- transferred to the same intermediate media
- in precisely the same way
- converted to digital samples in the same way

then it is advisable to take separate estimates for each cut or take. When denoising stereo material, it is also recommended to derive a separate Noise Estimate for each channel.

Likewise, if the character or level of the noise floor can be heard to change **at all** during the recording, then the best results are be obtained by dividing that piece into sections to be denoised individually.

After denoising, the individual sections can edited together to create a seamless whole.

The Noise Estimate should be taken from a relatively short segment in a quiet part of the recording to be denoised.

Optimum results are obtained when the Noise Estimate is taken from a section of **pure noise** between about 0.3 and 0.5 seconds in length, with a minimum of around 100 milliseconds (0.1 seconds).

The first concern in selecting a segment for analysis is that it represent the noise floor throughout the recording. In many cases, there is an apparently clean segment of noise prior to the start of program. Beware of such segments, as they may not represent the noise in the remainder of the recording.

It is not uncommon for recording engineers to fade up or punch in prior to the start of music, leaving an early clean segment containing only a portion of the real noise floor, leading to an inaccurate estimate.

In some instances, there may be no section where the noise can be measured without signal. Under such circumstances, select a section that is relatively quiet and free of non-harmonic sounds such as cymbals or bells.

When forced to take a Noise Estimate in the presence of signal, you must correct the situation by editing the Estimate as described later on. If the signal present is harmonic in nature, such as a sustained note or chord, or a vowel sound in the case of spoken word, it is often much easier for the operator to identify the frequency components that represent the source signal among the noise.

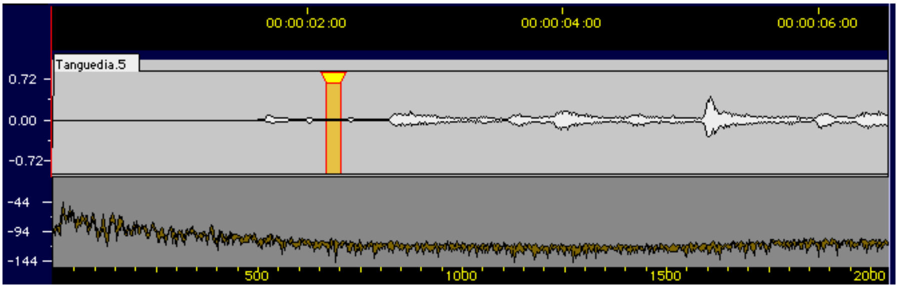
Taking An Estimate

Once a suitable portion of the sound file is identified, it is time to take the Noise Estimate.

To take an initial Noise Estimate:

1. Create a time selection of the soundfile region from which the Noise Estimate is to be derived.
2. Select the command **Take Noise Estimate** from the **NoNOISE** menu.

After processing, the panel changes to show an Estimate display.



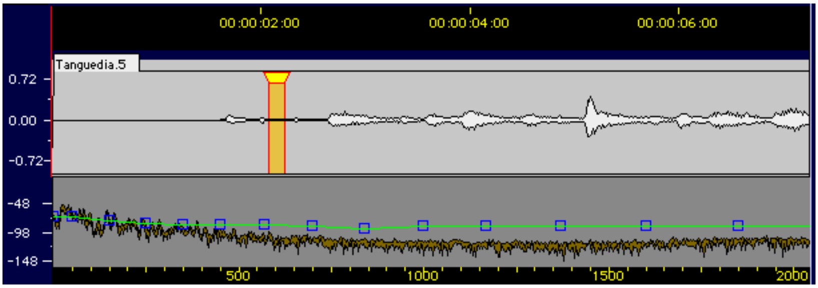
The Noise Estimate divides the area framed by the time selection into windows of 2048 samples each. An FFT analysis is performed on each of these windows, and the results of the FFTs are averaged to produce a kind of composite FFT.

Before the Noise Estimate can be used to provide threshold data for denoising, it must be additionally smoothed and, perhaps, adjusted to account for the presence of desired signal within the noise. There are two ways to accomplish this smoothing.

To smooth the Noise Estimate:

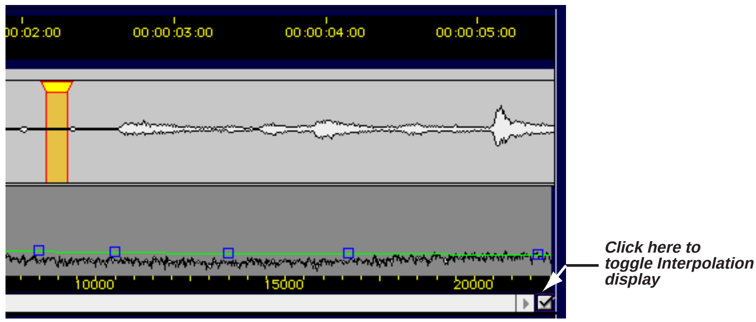
1 After deriving the Noise Estimate, select the command **Show/Hide Interpolation**.

SonicStudio HD performs a best fit approximation from the composite data, displaying it as a smooth Estimate Line connecting a series of boxes drawn over the composite FFT. If the Noise Estimate is derived from a clean sample of noise, this method produces an accurate measurement of the underlying noise floor.



Note: You can also choose **Take And Interpolate Estimate** to derive an interpolated noise estimate in one step.

Tip: You can toggle the Interpolation display by clicking the checkbox located in the lower right corner of the Noise Estimate display:



2 In cases where a clean (completely free of audio signal) sample of the noise floor is not available, a smooth Estimate Line can be derived from just a portion of the FFT display **provided that the noise in question is purely analog tape hiss.**

- Make a time selection of a portion of the Estimate display that appears to be outside the range of the audio signal itself.
- Choose **Fit Estimate From Selection** from the NoNOISE menu.

Analog tape hiss has a characteristic noise curve. **Fit Estimate From Selection** takes advantage of this property to produce a smooth Estimate Line from any individual section of the Estimate display. SonicStudio HD displays a characteristic tape hiss curve using the selected region of the composite FFT.

Editing and Saving the Noise Estimate

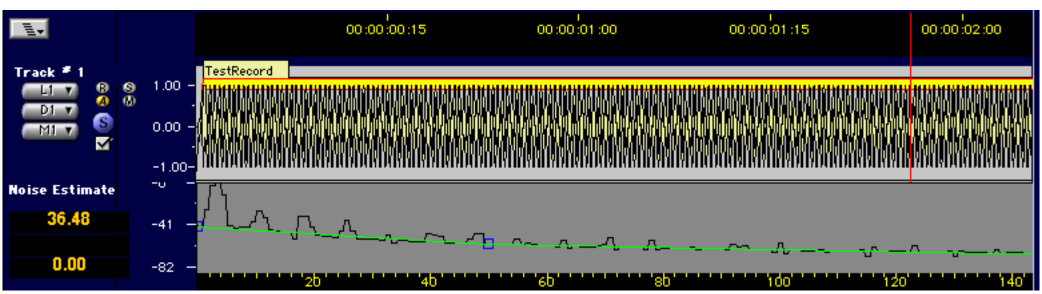
The **Fit Estimate From Selection** technique works well for recordings containing typical tape hiss, but many recordings have other noise sources. In such cases, it is often necessary to 'sculpt' the estimate manually, differentiating between the program signal and the noise floor.

To edit the Noise Estimate:

1 Zoom in to the portion of the display where the audio signal is found by making a time selection and choosing **View > Zoom To Selection** or pressing COMMAND+**G**.

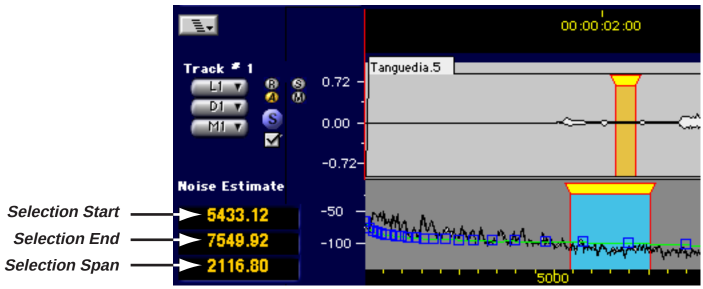
The components of the audio signal usually are visible as prominent, more or less evenly spaced spikes in the analysis display. (This is why, if an Estimate **must** be taken in presence of signal, it is advisable to avoid sections of overly-dense or non-harmonic material.)

2 Edit the Noise Estimate Line by pressing OPTION and using the mouse to drag the boxes superimposed over the line. The boxes can be moved only on the vertical axis.



If, for example, a number of piano harmonics are infecting the estimate, the boxes can be moved so that the Noise Estimate Line follows the valleys that represent the underlying noise floor.

3 You can edit multiple boxes at once. To do so, drag in the Noise Estimate display to create a time selection that contains the nodes. The numeric display in the Track header area shows the selection start frequency, selection end frequency and selection span in Hertz.



To edit the boxes within the selection, press OPTION+SHIFT and drag a box. All boxes within the selection move together.

Note: Noise Estimate editing requires you to exercise skill and judgment. The ability to extrapolate a useful Noise Estimate Line from a contaminated noise sample will grow with your experience.

To save a Noise Estimate file:

4 Choose **NoNOISE > Write Estimate...**

SonicStudio HD displays a Save File dialog.

5 Choose a location for the Noise Estimate file, rename it if you wish, then click **OK**.

SonicStudio HD saves the file to disk.

Broadband Denoising Parameters

The following parameters control Broadband Denoising:

Parameter	Range	Default Value	Recommended	Extreme
Attenuation	-120 to 0	-16	-10 to -30	
Threshold	-60 to 60	14	8 to 25	30
Sharpness	.5 to 5	1	1 to 1.5	> 1.75
Bandwidth	.5 to 3	1.8	.8 to 2.4	
Low Cutoff	0 to 22,050	50	50 to 500	25, 12,000
High Cutoff	0 to 22,050	22,050	22,050	12,000

ATTENUATION

This value in decibels sets the maximum attenuation to be applied in any frequency band. (A setting of 0 dB produces exactly no noise reduction.) The higher (more negative) this value is set, the greater the reduction in noise, but with increasing danger of producing audible artifacts in the audio signal.

The amount of noise reduction perceived is normally about half the maximum attenuation +-3 dB depending on the material and the other denoise parameters. A good starting point for the maximum attenuation is to take the amount of perceived noise reduction you wish to obtain and then double it. If, for example, you wish to obtain a perceived noise reduction of around -8 dB, then start with a maximum reduction setting of -16 dB.

Typical values for this parameter range between -10 (mild), -20 (moderately aggressive), and -30 (extreme). If the maximum attenuation setting is too extreme, ambience or high frequency response may be lost.

THRESHOLD

The Noise Estimate defines the curve of the thresholds that apply to each of the over 2,000 individual frequency bands used by the denoise process. The Threshold parameter allows the curve as a whole to be moved up or down. Together with the Maximum Reduction parameter, this provides the basic throttle that determines how aggressively denoising is applied.

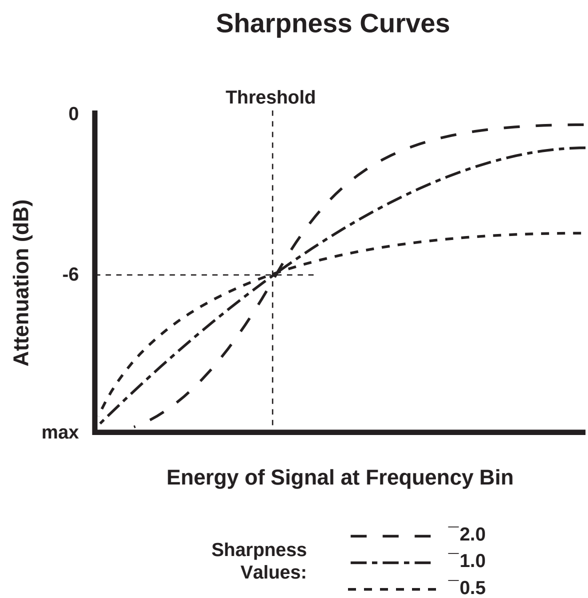
As the Threshold value is raised, more and more of the signal is processed. At extremely high settings, a distinctive watery sound may be heard on the audio signal. If the Threshold is set too low, little or no noise reduction is obtained.

The Threshold can be thought of as the fine line between noise and music, globally raising or lowering the entire Noise Estimate curve relative to its original position. Local adjustment of the threshold according to frequency bands is effected by adjusting the Noise Estimate curve.

The Threshold point is set (somewhat arbitrarily) at -6 dB. Threshold settings and maximum reduction settings should generally be adjusted together for best results. Typical values for this parameter range between 8 (mild), 16 (moderately aggressive), and 25 (extreme).

SHARPNESS

The denoising process works much like a multiband downward expander. As signal level in a particular band drops, the process reduces the gain in that band even further, using an internal attenuation curve.



The Sharpness parameter sets the slope of this curve. Higher values cause quicker attenuation as instantaneous energy falls off from the Threshold value, resulting in a response similar to that of a noise-gate. If the Sharpness is too low there may be no reasonable amount of noise reduction possible despite what the other parameters are set to.

Generally this parameter should be set as high as possible without audible ill effects on the program. A value of 1.0 is recommended for common tape hiss problems, while a value of about 1.2 has been found useful for standard 78 RPM type recordings.

If the sharpness is too high, you may hear a phasing-like problem in the music sometimes described as an underwater effect. The noise that remains may also become more unstable, producing a rapid fluttering of the noise floor.

BANDWIDTH

The Denoiser has been described as a multiband downward expander with many individual bands. Actually, there is a bit more involved. Denoising with each bin adjusted separately produces an unnatural-sounding result. In NoNOISE, individual bins share information for more natural sound.

Bandwidth governs this process. Higher values produce more sharing. Impressionistically, a higher value (in most cases) creates a more natural sounding result, but with the risk of audible pumping of the residual noise floor.

A low value for Bandwidth eliminates the possibility of noise pumping, but may sound less realistic and more muffled. Typical values to use here might be .8 (little sharing), 1.8 (good standard setting), 2.4 (a lot of sharing).

In setting the Bandwidth parameter, look for the best compromise setting. This depends entirely on the program material. It should be high enough to retain high frequency response but low enough to avoid pumping or distortion caused by noise being modulated by the high harmonics of a signal.

HIGH FREQUENCY CUTOFF

The denoise function lets signal in frequency bins above the high frequency cutoff point pass through the process untouched (that is, they are not processed). This limits processing to frequencies below the high cutoff point. This can be used in situations where noise is not objectionable above a certain frequency, but in most cases, this parameter is left set at 22050 Hz.

In extreme cases, good results may be obtained by processing the upper and lower frequencies separately, using different parameter values. Run the audio through the denoiser twice, using the low and high cutoff frequencies to define the area to be denoised.

LOW FREQUENCY CUTOFF

This parameter allows signal below the specified frequency to pass through unchanged. This can be useful if noise is not objectionable below a certain frequency and you wish to leave it alone.

Low Frequency Cutoff is usually left set around 50-100 Hz. If cutoff is below 25 Hz, there may be artifacts because wavelength exceeds the analysis window.

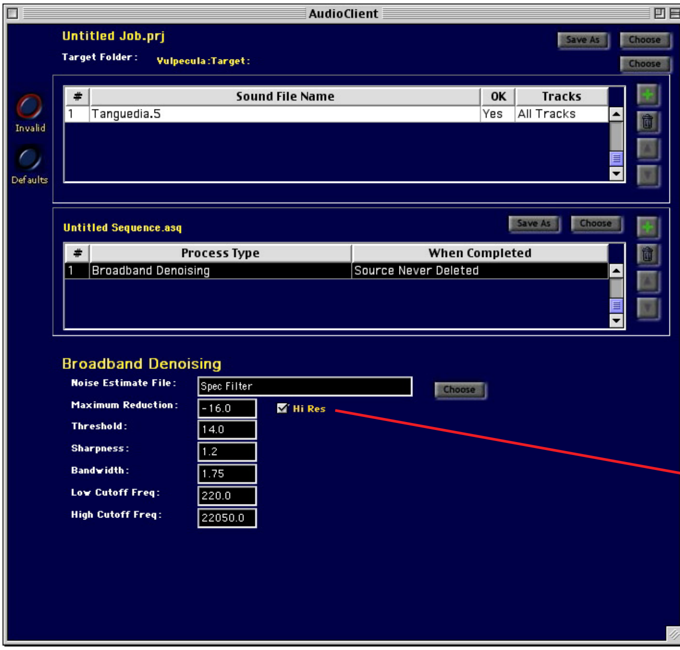
High-Resolution Denoise

AutoSonic incorporates a new, High-Resolution Denoise function. While standard Denoise achieves a maximum dynamic range of about 85 dB, High-Resolution Denoise can achieve 130 dB dynamic range. High-Resolution Denoise is appropriate for processing recordings that are already fairly quiet: for example, you might use it to remove mic preamp noise from a digital recording.

A High-Resolution Denoise pass takes about three times as long to process as standard Denoise.

To use High-Resolution Denoise:

- In the Denoise parameters panel, check **Hi-Res**.



Check **Hi Res** to enable High-Resolution Denoise

Complex Filtering

NoNOISE Complex Filtering can apply up to 192 separate filters in a single processing pass. Background filtering is useful for removing hums and buzzes. In some cases, these problems may be attacked by using the real-time filters in SonicStudio HD, but Complex Filtering offers four advantages over this approach:

- Far more filters (>900) can be applied in a single pass. This is especially useful when multiple notch filters are used to remove the fundamental and harmonics of a pitched noise (such as hum).
- Aggressive, high-order filters can be used without running out of DSP processing power.
- Long files can be processed in the background.
- AutoSonic filters are double-precision (calculated with 48-bit accuracy) for superior sonic quality.

Creating the Filter List

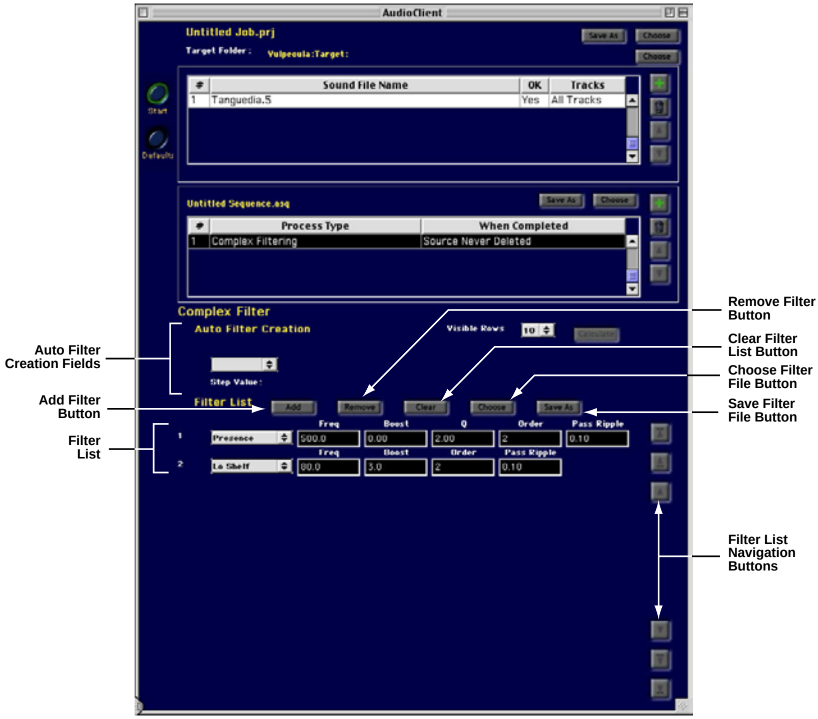
Once you have determined the frequency components to be removed or attenuated, you can create a list of filters and filter parameters in Audio Client.

To create a filter list:

1 Launch Audio Client and specify your source soundfile(s).

2 Click the **Add Process** button and select **Complex Filters** from the pop-up menu.

Audio Client displays the Complex Filters parameter set.



To add a filter to the list:

1 Click the **Add Filter** button.

Audio Client adds a blank filter to the list.

2 Use the pop-up menu to select a filter type.

3 Edit the filter's parameter values.

USING AUTO FILTER CREATION

Audio Client can automatically calculate a set of filters according to parameters that you specify. You could use this function, for example, to remove line frequency hum and all of its audible harmonics.

To calculate a filter series:

1. Select a filter type from the pop-up menu in the **Auto Filter Creation** area of the Parameters display.
2. Edit the filter parameters.
3. Enter **Step** values for the calculation. These will determine the interval increment for each parameter.
4. Click the **# Steps** parameter and type the number of filters you wish to create.
5. Use the **Visible Rows** pop-up to select the number of filters you wish to display in the Parameters area.
6. Click **Calculate**.

   Audio Client calculates the filter series and displays it.

SAVING THE FILTER LIST

Once you have entered a set of filters and parameters into the filter list, you can save the completed list to a file if you wish.

To save the completed list as a Filter Specification File:

1. Click the **Save As...** button.
2. Assign a name for the file, or use the default that appears in the Save As dialog.
3. Click **OK**.

READING A PREVIOUSLY-SAVED FILE

Previously-stored filter lists may be reopened and edited to change the configuration of filters or the parameter values.

To read in a filter file list:

1. Click the **Choose** button.

   Audio Client displays a standard Macintosh Open File dialog.

2. Navigate to the file, click to highlight it and click **OK**.

   Audio Client loads the selected filter file and displays the filter list.

APPLYING THE FILTERS TO A SOUND FILE

To launch an AutoSonic job:

1. Click the **Choose Target Folder** button at the top of the Audio Client window.
2. Navigate to the folder where you wish the output soundfile to be saved, click on the folder to highlight it and click **Choose**.
3. Click **Start**.

   Audio Client sends the job to AutoSonic Server for processing.

Filter Types and Parameters

The filter types available in Complex Filtering are the same as the real-time filters provided for SonicStudio HD.

CENTER FREQUENCY (CF)

The center frequency is the reference point of the filter. There are two different interpretations of the center frequency in the Sonic System Complex Filters, depending on the filter type used:

In filters that have a bandwidth or Q parameter, the center frequency references the midpoint of the affected region. Usually the center frequency is the most affected frequency of these types of filters.

Filter types that affect frequencies above or below a particular frequency reference the center frequency (cutoff frequency) as the -3 dB point from the boost or cut specified. A Hi Shelf filter designed to give a 6 dB boost above a crossover frequency (center frequency) of 10 kHz would have a boost of 3 dB at 10 kHz.

BANDWIDTH (BW) AND Q

Bandwidth and Q are two different ways of specifying the width of the filter. The width of the filter is measured from the -3 dB down points on either side of the center frequency. Bandwidth represents this width in absolute Hz. A bandwidth of 1000 means that the filter is 1000 Hz wide between the -3 dB points.

Q represents the width of the filter relative to the way that we hear. A Q setting of 2 always has a half octave bandwidth regardless of the center frequency. In mathematical terms, the Q is equal to the center frequency divided by the bandwidth.

Q = Center Frequency / Bandwidth

Also: Bandwidth = Center Frequency / Q

A Q of 1 = a one octave filter width. A Q of 2 = a half-octave filter width. A Q of 4 = a quarter-octave filter width. A Q of 0.5 equals a 2 octave filter width.

BOOST/CUT

The boost/cut represents the maximum affect of the filter on the program material. Boost/Cut is expressed in positive or negative decibels (dB).

ORDER

The order of the filter sets the slope of the filter's transition area. A first order filter usually means that the filter has a transition slope of 6 dB/octave. Each increase in the order adds another 6 dB/octave to the transition. Thus, a 3rd order filter would have a transition slope of 18 dB/octave, etc.

BandPass and BandStop filters require an even-numbered order. A second order BandPass/BandStop has transition slopes of 6 dB/octave and a fourth order BandPass/BandStop has transition slopes of 12 dB/octave.

STOPRIPPLE

StopRipple expresses how the filter affects audio in the stopband. The interpretation of Stopband and Stopripple depends on the filter type.

For Hi Pass, Low Pass, Band Pass and Band Stop filters:

- **Stopband** - the portion of audio eliminated or attenuated.
- **Stopripple** - minimum attenuation in the stopband. A setting of -40 dB means the signal will be at least 40 dB down in the stopband.

For High Shelf, Low Shelf and Parametric filters:

- **Stopband** - the portion of audio that is to be unaffected by the filter.
- **Stopripple** - the maximum change in dB as a result of the filter. A setting of 0.5 dB means a maximum EQ alteration of 0.5 dB. 0.5 dB EQ change is approximate minimum ripple audible by the human ear. The default stopripple value for EDL Filter Events is 0.1 dB.

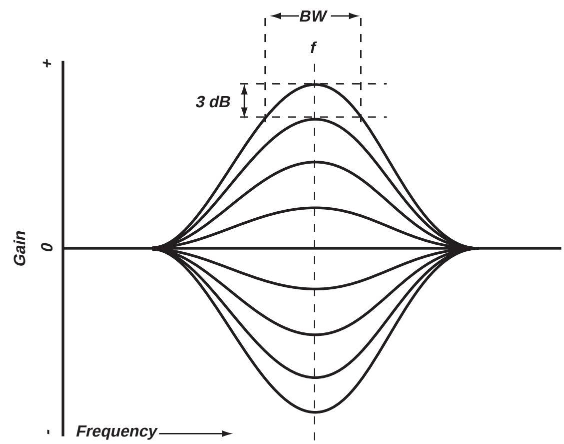
PASSRIPPLE

Passripple expresses the effect of the filter on audio in the passband. Passband has two interpretations in the Sonic System, depending on the filter:

For Hi Pass, Low Pass, Band Pass and Band Stop filters, **Passband** represents the portion of audio that the filter is letting pass through unchanged. For High Shelf, Low and Parametric, **Passband** represents the portion of audio that is affected by the filter. In all cases, passripple is the maximum change in dB in the passband as a result of the filter. A setting of 0.5 dB means that the stopband will have a maximum EQ alteration of 0.5 dB due to the filter.

PRESENCE FILTER

The Presence filter boosts or attenuates a particular region of the audio spectrum.



There are three parameters that define the response of the Parametric filter type:

Center Frequency (cf)

The Center Frequency is the mid-point of the band affected. The Presence filter's center frequency may be selected over a range of 1.0 Hz to 22.050 kHz

Bandwidth (bw)

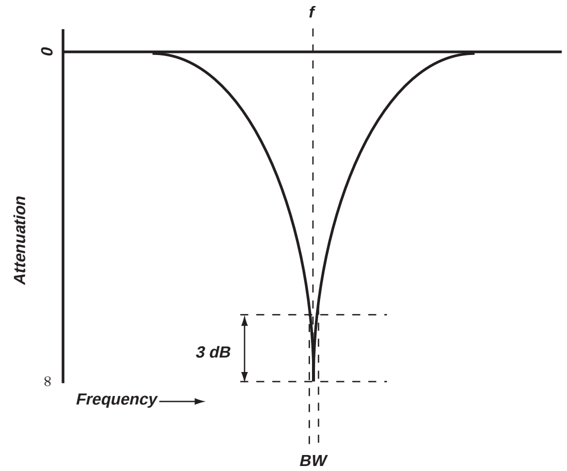
The Presence filter is a resonance-type filter. Its bandwidth in Hertz may be translated to filter Q by the formula: Q = cf/bw.

Boost

Boost indicates the gain applied at the Center Frequency. The Presence filter can supply boost (cut) of ±24 dB.

NOTCH FILTER

The Notch filter is a special case of the Presence filter, in which the gain at the center frequency is set to minus infinity, effectively eliminating all signal at that frequency.



Specifying a notch filter requires only two parameters: center frequency and bandwidth. The gain at the center frequency of a notch filter is fixed at -∞ dB, so that the center frequency is eliminated completely.

Center Frequency (cf)

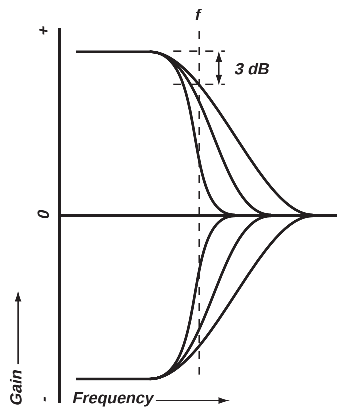
The range of the Notch filter's center frequency is the same as for other filters on the Sonic system, from 0.1 Hz to 1/2 of the Nyquist frequency.

Bandwidth (bw)

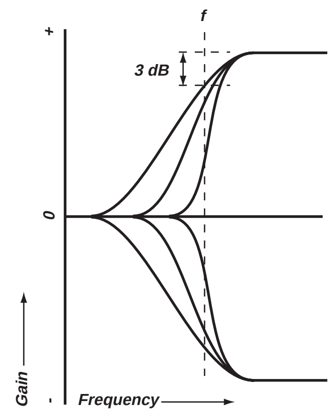
The Notch bandwidth is specified in Hertz, with a range from 1 Hz to 22 kHz.

HIGH AND LOW SHELVING FILTERS (HISHELF AND LOSHELF)

Shelving filters apply a fixed boost or cut to all frequencies beyond the cutoff frequency. Shelving filters have two variable parameters. These are different from those of Presence filters.



Low Shelf



High Shelf

Cutoff Frequency (cf)

Cutoff Frequency is the point where the signal is boosted or cut by 3 dB, **or** by 1/2 the specific boost/cut (if less than 3 dB).

Boost

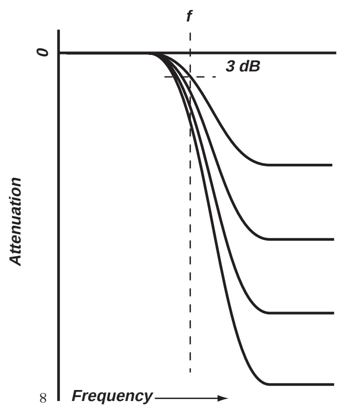
Boost (or Cut) applies to signal above (in the case of HiShelf) or below (in the case of LoShelf) the cutoff frequency. The range of boost or cut is ±24 dB.

HIGH-PASS AND LOW-PASS FILTERS (HIPASS AND LOWPASS)

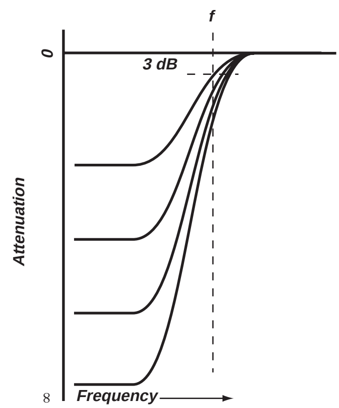
The high-pass and low-pass filters operate in three regions: A passband where signal is minimally altered; stopband where signal is attenuated; and the transition band that separates the two.

In the passband, there is an amplitude fluctuation called passband ripple. Generally, anything less than about 0.5 dB of ripple is inaudible. Passband ripple defaults to 0.1 dB. There is also some variation in the stopband, called stopband ripple. It represents the maximum value that the filter gain attains in the stopband.

For a high pass filter, the frequency parameter refers to the highest frequency at which the gain attains the minimum value in the passband. A low pass filter has a gain of 1.0 (less the pass ripple) below *f*, and falls off smoothly to stop ripple at some point above *f*.



Low Pass



High Pass

Cutoff Frequency (cf)

At the Cutoff Frequency, the signal level is reduced by 3 decibels. The range of cutoffs for both high- and low-pass filters is from 1 Hz to 22.05 kHz.

Filter Order (n)

Filter order, an integer from 1 to 4, controls the steepness of the transition from stopband to passband. The transition band drops off by roughly 6 dB per octave for each unit increase in order. The higher the order, the greater the chance of audible ringing at the cutoff frequency.

Passband Ripple (passripple)

In the Low and Highpass filter, the Passband Ripple is the area that is **not** affected by the filter.

Stopband Ripple (stopripple)

For Low and Highpass filters, the Stopband Ripple is the maximum gain reached in the area after the cutoff frequency.

UTILITY FILTERS (DC, EMPHASIS, RIAA)

Complex Filters includes several types of utility filters for common functions of DC removal, pre- and de-emphasis, and application or removal of the RIAA curve for vinyl records.

These filters have no variable parameters. They are simply on or off.

No DC

The No DC filter is a simple DC reject filter. The No DC filter provides 1 dB of cut at 34 Hz and 3 dB of cut at 18 Hz.

Emphasis and De-Emphasis

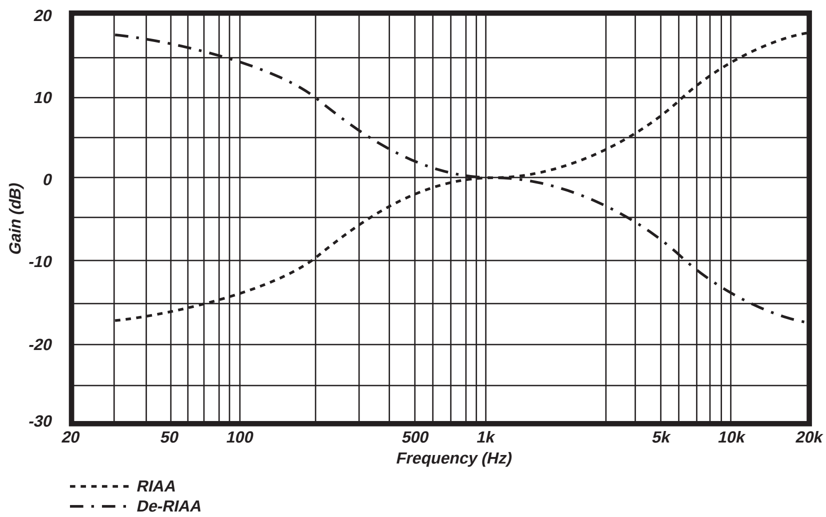
The Emphasis filter is a 15/50 microsecond curve, as defined as an option for Compact Disc masters. The De-Emphasis filter provides for removal of this high-frequency boost from material that is previously emphasized.

DC/De-Emphasis

No DC/De-Emphasis combines this with a filter to remove the Sony F1 (EIAJ digital audio adapter) 15/50 microsecond emphasis curve.

RIAA and De-RIAA

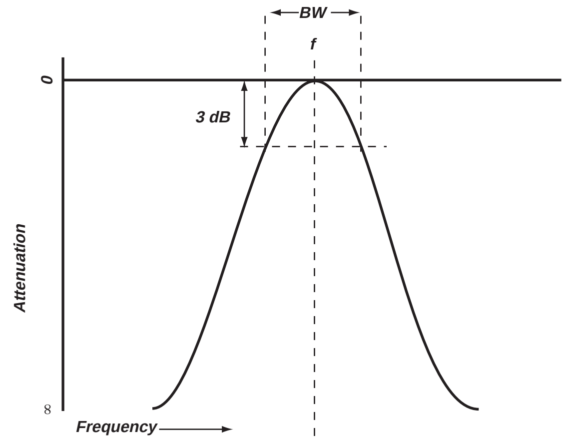
The Sonic system supports RIAA and De-RIAA filters. The RIAA filter imposes the standard RIAA characteristic normally applied, in LP mastering, at the input to a disk cutting lathe. The De-RIAA filter removes the effect of a RIAA filter.



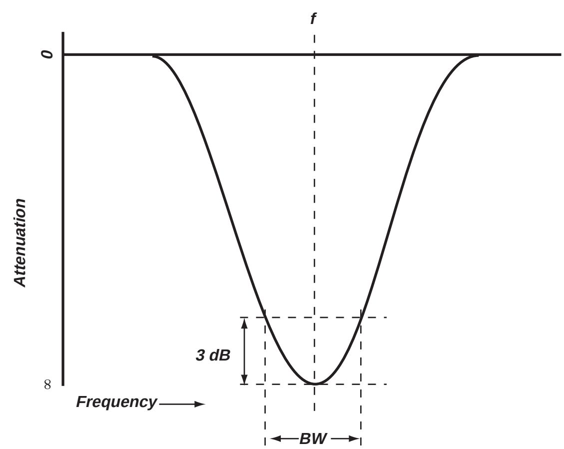
BAND PASS AND BAND STOP FILTERS

Band Pass and Band Stop are like putting together a high pass and a low pass filter.

Band Pass allows only certain frequencies to be admitted and rejects all others that are out of the range.



Band Stop eliminates a certain range of frequencies and passes all the rest.



Center Frequency (cf)

The range of the BandPass and BandStop filter's center frequency is the same as for other filters on the Sonic system, from 1 Hz to 22.050 kHz.

Bandwidth (bw)

Bandwidth is specified in Hertz, with a range from 1 Hz to 22 kHz.

Filter Order (n)

The order of a filter, an integer value from 1 to 4, determines the slope of the filter's response curve. A first-order filter has a slope of 6 dB per octave. Slope increases by 6 dB per octave for each increment of 1 to the filter's order.

PASSBAND RIPPLE (passripple)

In a Shelving filter, the passband is the portion of the signal affected (boost or cut) by the filter Passripple is the maximum variation in dB in the passband. A setting of 0.5 dB means a maximum EQ alteration of 0.5 dB due to the filter.

STOPBAND RIPPLE (stopripple)

The stopband is the portion of the signal ***not*** affected by the filter.

Declicking

Declicking detects and eliminate clicks automatically in two steps. First, an optional ***click detect*** pass reads through the sound file and produces a click list with the location and description of each click. The click list is then used to guide the ***declick pass***. The declick pass makes corrections directly on the source sound file, maintaining a separate hidden file of the original clicks, which can be used to undo any portion of the declicking.

You must specify parameters for both the click detection and the declicking. You can also specify a custom Filter Spec File for the detection pass.

Click Detection Parameters

The following parameters control click detection:

Parameter	Range	Default Value	Recommended	Extreme
Initial Threshold	1 to 5000	5	5 to 10	1, >100
Center Width	.00006 to .04644	.001	.0005 to .0025	>.005
Wing Width	.0001 to .04644	.002	.002 - .01	>.01
Wind Weight	0 to 1	1	1	0

INITIAL THRESHOLD

When the detection algorithm identifies a candidate click site, it measures and assigns a value for total energy. This is compared against the Initial Threshold, and against the energy in the Wings to determine if the site is an actual click.

Initial Threshold is the lowest value that is recognized as a click. Click Detect accepts values from 1 to 5000 for Initial Threshold, but it takes a **very** large click to produce a total energy value greater than 200. Higher values have the effect of excluding virtually all clicks.

Initial Threshold has the effect of limiting the number of clicks detected. This helps to avoid detection of spurious click sites and limits the size of the click list so that it remains manageable within the system's memory.

For 78 RPM phonograph recordings, an Initial Threshold value between 5 and 10 is recommended. To detect only loud clicks, use a range of 100-200. To find all clicks, use the minimum setting of 1, but be aware that it is possible for the click list to become very large and processing will be slow.

To avoid overly large click lists, you can run multiple detect/declick passes with progressively lower thresholds, allowing even the longest pieces to be declicked. This also helps to maintain your control, by allowing you to evaluate and modify the results at each stage.

CENTER WIDTH

Center Width is the length of the frame that the detector analyzes. It represents the duration of a typical click. If a click is much shorter than the Center Width, the click detect bar may not be properly centered over the click.

If a click is **longer** than the Center Width, the system will most likely still detect it, but the site will be listed shorter than the actual click. You may replace the entire length of the clicks during the click removal process by setting a wide Replacement Width setting.

Typical clicks range from about 0.5 to 2.5 milliseconds in length. We recommend keeping the Center Width within this range. The initial default value is 1 millisecond. For special cases, the value might be set as high as 5 milliseconds.

If a recording contains several distinct kinds of clicks exist, for example if a broken record contains loud and long clicks near a break, and also has intermittent normal, short clicks, it is best to process the recording in two passes. Perform the first pass using a large value of Center Width; after declicking and removing those clicks, perform another pass using a shorter value of Center Width.

WING WIDTH

Wings are the sections that precede and follow a candidate click site. They provide the context information for recognizing a valid click. The click detector calculates total spectral energy in the wings and in the site candidate, then subtracts energy in the wings from that in the site candidate. Click amplitude is the difference between the center and the wings.

Wing Width is the time in seconds on each side of the site candidate that is used for analysis. The minimum value is 0.0001 seconds (0.1 milliseconds). The maximum value of Wing Width is 2048 samples corresponding to approximately 42.7 milliseconds at 48kHz sample rate, or 46.4 ms at 44.1kHz sample rate.

The initial default value of Wing Width is 0.002 seconds (2 milliseconds). When using the recommended click detect algorithm (that is, with no filter spec file) this type of narrow Wing Width is preferred.

When using the pre-filter detect algorithm, by entering the name of a filter spec file in the dialog box, Wing Width should be at least two times longer than Center Width. In this algorithm, Wing Width also corresponds to the minimum spacing of clicks that will be detected. If two clicks of roughly similar amplitudes are closer together than the Wing Width, neither will be detected.

If the Wing Width is made too small, however, it may interact with signal, causing normal transients to be detected and eliminated as clicks. Normal music waveforms (such as male voice, trumpet, and trombone sounds) exhibit impulsive behavior that can be mistaken for clicks.

If the Wing Width is much less than one waveform period, the detector will sometimes list a spurious click at the beginning of each period. If the source material includes strong low frequencies, setting Wing Width to 10, 15, or even 20 milliseconds will eliminate this problem, but at the risk of not detecting some clicks separated by less time.

WING WEIGHT

The NoNOISE click detector subtracts total spectral energy in the wings surrounding a possible click site from the energy within the site area as defined from the Center Width. This difference value is used to determine valid clicks to be eliminated.

Wing Weight is a coefficient between 0 and 1, applied to the wings energy value before it is subtracted from that of the candidate click site. In essence, it tells the system how much to value the wings in determining the validity of a click.

The amplitude of a detected click is lowered when the wing amplitude is subtracted from the click amplitude. Lowering the Wing Weight could be thought of as decreasing the contrast between the site candidate and its wings.

This is useful in determining the validity of a click, but can sometimes result in valid clicks being rejected. Lowering Wing Weight compensates for this, but increases the risk of falsely detecting sites that are not real clicks.

Optimum settings for Wing Weight varies with program material. The initial default of 1.0 (Wing Weight has full effect) is suitable for the greatest range of program material.

If the program material is more band-limited than the clicks, as with clicks on 78 RPM discs, Wing Weight may be set to a lower value.

At a setting of 0, the criterion for click site identification is based strictly on the magnitude of the filtered signal. This is useful when the audio material is severely band-limited and the click's principal energy is out of the program's frequency range.

The Filter Spec File

Use Complex Filtering to create a filter spec file and save it to disk (see ***Complex Filtering*** on page 50 for more information). You may then select the filter in the Declicking parameters area of AutoSonic Client.

Declick Parameters

The following parameters control Declicking:

Parameter	Range	Default Value	Recommended
Replacement Width	.000091 to .092880	.0025	.0025 to.003
Context Width	.001 to .081270	.03	.03
Replacement Order	3 to 1024	400	400
Replacement Center	0 to 99.9	40	40
Click Rate/ Threshold	0 to 500	30	30

REPLACEMENT WIDTH

The declick algorithm is capable of filling a gap ranging from 4 to 4096 audio samples. Replacement Width determines the number of samples that will be replaced when removing the click.

Its value is expressed in seconds, from 0.00009 to 0.092 seconds (0.09 to 92.0 milliseconds) at 48 kHz sample rate. At 44.1 kHz, the maximum value is slightly higher. The default value is 0.0025 seconds, or 2.5 milliseconds.

The default value of 2.5 milliseconds is reasonable for normal disk recordings. You should always try to replace the minimum amount of sound necessary to eliminate the clicks. It's generally best to replace just a little bit more than the actual length of the click.

A good rule is to examine the source material to determine the length of the average click, then add approximately 20 percent. The typical click from a 78 RPM phonograph record is 0.5 to 2.0 milliseconds, making the setting of 0.0025 to 0.003 a good default.

Note: The length of sound replaced is not related to Center Width in the click detect pass, nor to the width of the bar that appears over the click site when the Click List is read in.

If the Replacement Width is set too low, then partial clicks may be left in the audio. If Replacement Width is set too high, the chances of generating low frequency thunks increase.

CONTEXT WIDTH

Context Width is the length of audio to either side of the click that is used in resynthesis. Its value is expressed in seconds.

Larger values of Context Width require longer time to process, but are less likely to produce artifacts. The default value of 0.030000 seconds (30 milliseconds) provides a reasonable compromise between speed and accuracy.

The minimum and maximum Context Width depends on the settings of Replacement Order and sample rate:

- Minimum Context Width = ([Replacement Order+1], sample rate (44100 or 48000))
- Maximum Context Width = f([4096 - 256 - Replacement Order],sample rate (44100 or 48000))

For best results, set the Context Width to **at least** twice the minimum value. Using the maximum Replacement Order of 512 at 44.1 kHz sample rate, the minimum context is about 0.012 seconds (12 milliseconds), and the Context Width should be set to at least 0.024000 seconds.

Setting Context Width too low will produce artifacts sounding like anything from low frequency thunks to bursts of noise, depending on the context. Setting the Context Width too high, however, substantially increases processing time.

REPLACEMENT ORDER

Replacement Order sets the precision of the resynthesis calculation, and concerns the power of interpolation. The default value for Replacement Order is 400, with a minimum value of 3 and a maximum of 512.

In general, the larger the Replacement Order the better the interpolation but the slower the processing. If either Replacement Order or Context Width are increased, the processing time increases in proportion to the sum of the increase in these two numbers. If Replacement Order is increased by 10 percent and Context Width by 10 percent, then total processing time increases by about 20 percent.

The default of 400 is a good value to use in most situations. If low-frequency artifacts (thunks) are experienced, then the value of Replacement Order should be increased. The tradeoff is that it will take longer to replace each click.

The default Context Width and Replacement Order have been found suitable for about 80 percent of Declicking projects. For extremely large clicks, Context Width may be raised to about 35 milliseconds (0.0350000 seconds) and Replacement Order to 512.

Note: To use a large value Replacement Width and avoid artifacts, increase Replacement Order. Keep in mind that using a maximum Replacement Order of 512 limits Context Width at 44.1 kHz sample rate to:
[4096 - 256 - 512] / 44100 = 75.5 milliseconds

REPLACEMENT CENTER

Replacement Center specifies an offset (as a percentage) that shifts the area of interpolation in relation to the marked click site.

Clicks are often followed by some amount of ***ringing***, making it necessary to continue replacement for some time after the click itself. In transcribing a phonograph record, for example, the stylus and tonearm resonate in response to the impulse of a scratch on the record, producing a damped oscillation that extends for some time after the click. This ringing is not detected by the click detect functions.

Replacement Center shifts the replacement area to the left or right in relation to the click center according to a percentage from the left edge. A setting of 0 percent positions the click at the left edge of the replacement area, while a setting of 50 percent places the click directly in the middle. A setting of 100 percent positions the click at the very right edge of the replacement area.

If Replacement Center is set so that the ringing of the click extends beyond the Replacement Width, the ringing ***is not interpolated*** and will be heard in the program. Not only that, the ringing will be used as context information, causing erroneous interpolation.

Adjust this parameter to ensure interpolation of any ringing. Large clicks (more than 0.010 seconds in length) can exhibit a lot of ringing. Setting the Replacement Center between 5 and 25 percent will ensure that the ringing will fall within the replacement area.

For 78 RPM record clicks, 40 percent is a good value to use. Replacement Center is almost never set higher than 50 percent, as this would shift the interpolation **forward**, ahead of any ringing.

The Replacement Center parameter can be of use in extraordinary situations. For example, one project at Sonic Solutions involved a record that had been broken and then glued back together, resulting in clicks of about 2.5 milliseconds in duration, followed by over 10 milliseconds of ringing. A Replacement Width of 15 milliseconds and a centering of 20 percent turned out to be the best setting in this case.

RATE AND THRESHOLD

Rate (or Threshold) determines how many interpolations are actually performed out of the click sites marked in the click detect pass. The Initial Threshold parameter in click detect determines which candidate sites are marked. After that, the Threshold (Rate) parameter can be used to enable and disable clicks from the list.

When a declick pass is launched, you must specify a value for Threshold. This value controls the action of the declick pass, regardless of the value specified for the List.

Rate and Threshold limit the number of clicks actually interpolated. A Threshold of 100 means that only clicks that have a total energy value higher than 100 (arbitrary units) are interpolated.

It may be more intuitive to use Rate rather than Threshold. A Rate of 3.0 clicks per second means that the system will calculate a Threshold setting to yield an average of 3 interpolations per second. If the source file is 100 seconds long and the Rate is set to 3, the declicker will interpolate the 300 loudest clicks.

The default value of 30 (for either Rate **or** Threshold) is a suitable starting point. Rate and Threshold are different ways to set the same parameter. Threshold lets you set a specific value that defines the amount of click energy required to trigger interpolation. Using Rate makes the computer calculate a Threshold based on a some factor that may be easier for you to visualize.

If Threshold is set too high (or Rate is too low) then the declicker will not remove as many clicks as expected. If Threshold is too low (or Rate too high) the number of interpolations increases, along with processing time and the possibility of a bad interpolation.

Generating and Editing a Click List

A click list is produced automatically by the click detect pass. (A click list can also be generated manually, but is not recommended for the normal case.) The list resulting from a detect pass may be edited. New click sites may be added, and existing sites disabled or deleted.

To launch a click detect pass:

1 In Audio Client, click the **Add Soundfile** button.

Audio Client displays an Open File dialog. Navigate to the soundfile you wish to process, click to highlight it and click **OK** or press RETURN.

The soundfile name appears in the Soundfile Name list.

2 Click the **Add Process** button and select Declicking from the pop-up menu, the click **OK**. The Declicking parameters appear in the Process Parameters area of the Audio Client window.

3 Edit the Click Detect parameters as you wish.

4 Make sure that the **Enable Click Detect** box is checked and the **Enable Declick** box is **not** checked.

5 Click **Start**.

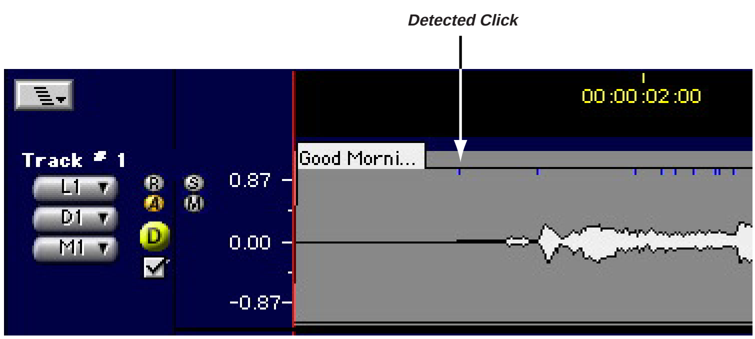
Audio Client sends the job to AutoSonic Server for processing. You can monitor its progress by launching Queue Manager.

To view the Click List:

1 Open the soundfile in SonicStudio HD.

2 Choose **NoNOISE > Click List Mgr. > Read Click List.**

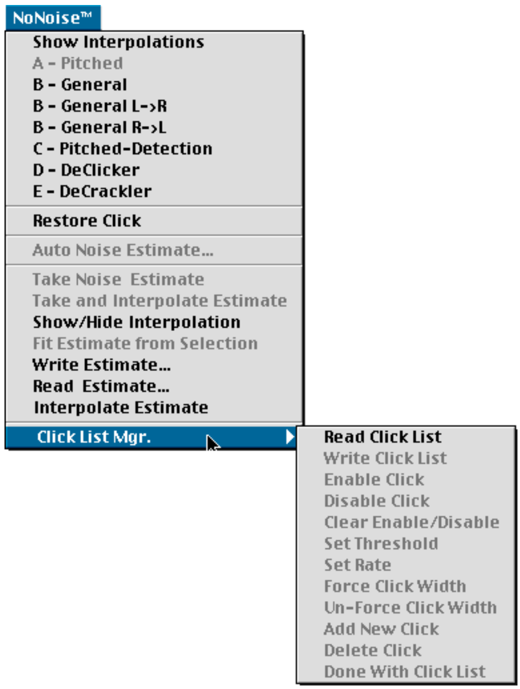
SonicStudio HD displays each detected click as a blue bar above the soundfile waveform.



To edit a Click:

1 Make a time selection containing the click.

2 Choose NoNOISE > Click List Mgr. and select an option.



Threshold and Rate (Enable/Disable)

Common declick sources (phonograph records, optical soundtracks, etc.) may contain hundreds of individual clicks. It is frequently useful to limit the number of clicks actually detected and corrected. For one thing, the click list may grow so large as to not be useful.

Click detection uses a variable threshold to determine which clicks to mark and which to leave untouched. This value may be entered either as a Rate (clicks per second) or directly as Threshold, and may be used to throttle back the declick process.

THRESHOLD

A click is a short section of sound whose spectrum differs markedly from that of surrounding audio. Generally, a click contains far more high-frequency energy than the adjacent signal. Each candidate anomaly is assigned a value for total spectral energy. Sites that read below the specified Threshold are disabled.

The Threshold parameter may be changed by using the command **Set Threshold** from the click list editing section of the **NoNOISE** menu. Any change will cause some clicks to be disabled, while others become enabled in the EDL display.

To view and change the Threshold parameter:

1. Select **Set Threshold** from the NoNOISE menu.
2. Click the parameter value field in the **Click Detect Amplitude** dialog box. Drag with the mouse to select all digits, or use the Tab key.
3. Press Delete.
4. Enter the value desired.

Note: The dialog box field will accept values from 0 to 2000, but only a range of approximately 1 to 500 produces a generally useful effect.

RATE

Amplitude threshold may also be specified in terms of a rate of click removal. When a Rate value is entered, Audio Worker sorts clicks by amplitude, and determines a Threshold setting that will eliminate exactly the specified number of clicks per second.

The effect of the Rate parameter can be seen by using the **Set Rate** command in the NoNOISE menu. The procedure of selecting, deleting, then setting the parameter are the same as in Set Threshold.

The implicit Threshold is indicated in two ways:

- Click sites whose values exceed the threshold show filled-in click detect bars. Other sites show open (unfilled) detect bars.
- The Threshold value in the Click Detect Amplitude dialog box automatically adjusts to correspond to the specified Rate.

The chart below shows representative statistics for older phonograph records. These provide an indication of click Rates for various contexts.

Degree of Difficulty	Total Clicks (3 min.)	Clicks per Second
Easy	50	0.25
Average	200	1
Difficult	600	3
Worst	2000	11

Enable Click

The Threshold (Rate) parameter causes various click sites to be enabled or disabled automatically. You can also force the state of individual clicks.

To enable a click site for correction:

1. Make a time selection that contains the click site.
2. Select **Enable Click** from the NoNOISE menu.

The bar over the click site will appear as filled, and the designated click will be processed, regardless of the Threshold setting.

Disable Click

Similarly, you may force Audio Worker to ignore a particular site.

To disable a click site:

1 Make a time selection that contains the click site.

2 Select **Disable Click** from the NoNOISE menu.

The bar over the click site now appears as hollow or unfilled.

Clear Enable/Disable

Enable Click and **Disable Click** commands force a given click site to be processed or ignored. To revert to the normal condition, in which the value assigned is compared to the Threshold, make a time selection that contains the click site, then select **Clear Enable/Disable.**

Audio Worker thereafter treats the click site based on its energy in relation to the Threshold (Rate) setting.

Add New Click

To add a new site to the click list:

1 Make a time selection that contains the click site.

Make sure that the entire anomaly is included. The declick algorithm uses the areas outside the selection as the basis for resynthesis. If a portion of the click is left outside the selection, correction may not be satisfactory.

2 Select **Add New Click** from the NoNOISE menu.

The area marked by the selection will be added to the click list, with the click enabled.

Delete Click

Spurious click sites can easily be deleted.

To delete one or more click sites from the list:

1 Make a time selection that contains the click site.

2 Select **Delete Click**.

All clicks within the selection are eliminated from the click list.

Replacement Width

During declicking, each click site is replaced. The actual length of the sound replaced is defined by the Replacement Width parameter specified when the declick pass is launched.

You have the option overriding the Replacement Width parameter by editing the click list. For each click, the system can be instructed to use the width of the detect bar instead of the Replacement Width value.

Force Width

Normally, the same Replacement Width is applied to all enabled sites. **Force Click Width** overrides Replacement Width for a particular site.

To override Replacement Width for a particular click:

1 Make a time selection that contains the click site.

2 Select **Force Click Width** from the NoNOISE menu.

After forcing, the length of the click detect bar for that click will be used as its Replacement Width.

To change the width of a click site:

3 Make a time selection that contains the click site.

4 Select **Delete Click** from the NoNOISE menu.

5 Adjust your selection to the exact beginning and end points desired by holding SHIFT and dragging.

6 Select **Add New Click.**

Un-Force Click Width

The **Un-Force Click Width** command reverses the effect of **Force Click Width**. Replacement Width again defines the length of reconstituted audio inserted to replace the click.

Note: When you use the **Add Click** command to add a new click, the new click defaults to Force Width on, so ensure that your time selection encompasses the area you want Declick to replace when you add a click. Alternately, you can select the new click and choose **Unforce Width**; this will tell Audio Worker to use the Declick parameter settings.

Storing and Closing the List

Before the edited list can be used for click removal, it must be stored to disk.

To update the click list on the disk:

- Select **Write Click List** from the NoNOISE menu.

Until the click list is written to disk, any changes have not really taken effect and will not affect the behavior of the declicking pass.

Done With Click List

When click list editing is complete, you can clear the detect bars from the display by closing the list.

To close the list:

- Select **Done With Click List**.

This will close the open list, and remove the detect bars from the waveform display.

If the click list has been changed at all since it was last written to disk, SonicStudio HD prompts you to save the changed version.

Click List File Format

The click list is stored in text form as a normal Macintosh file, and users who are reasonably savvy about Macintosh operations can open this file to view, and even edit, its contents. The click list is found in the same folder as the sound file, with the name of the sound file followed by '.cl.' The Macintosh file type is set to 'DCLK,' while the creator is 'SYSS.'

The click list can be opened from an ordinary word processor program by changing the file type from 'DCLK' to 'TEXT.' (If the list is changed, the file type must be set back to 'DCLK' before the file can be used for declicking.) Many word processors include an option to open any file, no matter what the file type.

These can be used to open the file, but the edited file **cannot** then be used for declicking, because it is no longer of the correct type. A Macintosh utility program (such as FileTyper) may be used to change the edited file's type and creator so that it can be used by AutoSonic.

The click list consists of a program in a tab and comma delimited format convenient for editing in a spreadsheet program. It has three required arguments, with a number of optional arguments. The three mandatory arguments are:

- **Position** — click location (first sample) in samples
- **Length** — length of the click in samples
- **Size —** amplitude of the click (from 0 to 8191 inclusive)

The three optional arguments, accessed by name, are:

- FORCEDON
- FORCEDOFF
- FORCEDWIDTH

If a click site is Enabled using the **Enable Click** command, a "1" is placed as the fourth argument in the click list.

Similarly, **Disable Click** and **Force Click Width** change their respective columns to "1."

Comments may be added to any click statement by placing the comment after the last comma of any click statement.

Examples of Valid Click Site Statements

Argument	Comment
POSITION,LENGTH,SIZE,FORCEDON, FORCEDOFF,FORCEDWIDTH,THRESHOLD	Click List file header; shows the order of the Declick parameters
44100,50,100,,,	Click is one second in (at 44.1 kHz SR), 50 samples wide, with amplitude of 100

Argument	Comment
44100,50,100,1,,,	Same, with replacement width forced to be exactly 50 samples
88200,65,0,1,1,	Click is two seconds in, 65 samples wide, amplitude unspecified so FORCEDON is set to make sure it is replaced; replacement width is forced to 65 samples

Click sites may be placed in any order. They need not be in order of increasing time, or increasing amplitude, or anything else. All arguments **must** be given in exactly the order shown.

Production Declicking

To declick a soundfile using the edited Click List:

1. In Audio Client, click the **Add Soundfile** button.

   Audio Client displays an Open File dialog. Navigate to the soundfile , click to highlight it and click **OK** or press RETURN.

   The soundfile name appears in the Soundfile list.

2. Click the **Add Process** button and select Declicking from the pop-up menu, the click **OK**. The Declicking parameters appear in the Process Parameters area of the Audio Client window.

3. Edit the Declicking parameters as you wish.

4. Make sure that the **Enable Declick** box is checked and the **Enable Click Detect** box is **not** checked.

5. Click **Start**.

Audio Client sends the job to AutoSonic Server for processing. You can monitor its progress by launching Queue Manager.

Decrackle

Impulsive noises in recordings come in two general varieties. Clicks, pops, ticks, and spikes are sizable impulses that break the flow of audio in a way that is comparatively easy to recognize and isolate. Usually, such glitches are spaced far enough from one another so that the audio on either side can be used to reconstitute the area of the click. The Manual and AutoSonic Declicking processes are effective in attacking this type of problem.

The other common type of impulse noise is crackle, in which small impulses crowd against one another, producing a nearly continuous noise, like bacon frying in a pan. This type of artifact requires a different processing approach.

The Decrackle process is effective for correcting this type of dense, impulsive noise by performing a type of sliding interpolation that isolates good audio between impulses and uses it as the basis for resynthesis.

Usually, the Decrackler is used together with Declicking. The general procedure is to perform a light Declicking pass and take out the large, conspicuous clicks. It is acceptable to leave a click here and there. Then run a Decrackle pass, as described in this chapter.

Decrackle does not distinguish between pure crackle and isolated clicks: it tries to eliminate both. The point of running a Declicking pass first is that Decrackle has only a certain amount of processing 'ammunition' for each frame of data. If it has to spend too much getting the larger clicks out, then it will not have enough left for the crackle.

Decrackle Parameters

The following parameters control Decrackling:

Parameter	Range	Default Value	Recommended	Extreme
Clip Fraction	.5 to 1	.9	.75 to .9	< .75
Synthesis Order	10 to 256	75	75 to 128	65, > 150

Parameter	Range	Default Value	Recommended	Extreme
Damping Factor	0 to 1	.001	.001 to .015	> .02
Amplitude Weighting	-1 to 1	0	-.5 to .5	-1, 1

The four processing parameters provide considerable control over the results of the decrackle process, but the effects of these parameters are a bit difficult to describe in common audio terms.

We recommend that you experiment with the different parameters by decrackling short portions of a file. This provides a good sense of each parameter's effect in a reasonable amount of time.

CLIP FRACTION

This number affects the **detection** part of the Decrackle algorithm, in which audio samples are separated into good and bad categories. The synthesis process then replaces the samples that are deemed bad with synthesized material that matches the surrounding sound.

The Clip Fraction is the percentage, or fraction of the samples that will be left in the good category. The range of the parameter is from 0 to 1.00. These samples will pass through the Decrackle process unchanged. The higher the Clip Fraction, the less aggressive the decrackling. If the Clip Fraction is set to 1.0, the output file will be a copy of the input file.

For heavier decrackling, make the Clip Fraction smaller, to 0.85, 0.80, or even as low as 0.75. Any lower is not recommended, as there will be too little of the original signal to provide a basis for resynthesis. Lowering the Clip Fraction by only 0.05 is enough to make a noticeable difference.

If the Clip Fraction is set too high then there will be some amount of crackle remaining in the program. If the Clip Fraction is set too low you may begin to decrackle good audio, which could result in low frequency artifacts (audible thumps).

SYNTHESIS ORDER

The Synthesis Order determines the precision of the Decrackler's resynthesis. In general, the larger this number, the cleaner and more artifact-free the output. The default value of 75 is suitable for the majority of source materials.

If low frequency thunks occur in the processed output, try raising the value of the Synthesis Order to 100 or even 128. However, raising the Synthesis Order markedly increases the amount of time required to process the soundfile. Synthesis Order should generally be left at 75 unless the process is producing unacceptable numbers of artifacts.

Paradoxically, processing artifacts tend to occur ***more*** with clean recordings that have high signal-to-noise ratios. For (relatively) recent material, such as tape recordings from the early 1950s, it is common to set the Clip Fraction to.98 and the Synthesis Order to 128. For 78s from the 1930s, however, it is common to set the Clip Fraction to 0.75 and the Synthesis Order to 75.

DAMPING FACTOR

The Damping Factor affects the way that the Decrackle algorithm tracks high-level transient information. The higher the Damping Factor, the more the process will tend to smooth transients in the source material.

Large transients sometimes produce low-frequency thumps or thunks in the output. A small amount of damping, such as the default value of 0.001 will (in most cases) smooth the material just enough to prevent artifacts without adversely affecting transient response. Although the range of the Damping Factor extends as high as 1.0, the highest value recommended for normal work is about 0.015.

If Damping Factor is set too high, there may be a loss of transient response and, in extreme cases, loss of overall dynamic range.

AMPLITUDE WEIGHTING

While Clip Fraction determines the overall percentage of samples retained unaltered in the processed output, Amplitude Weighting determines how these are distributed between high and low amplitude sections of the source. At the default value of 0.0, all portions of the source file are processed equally.

As the value of Amplitude Weighting is increased (more positive), the processing becomes concentrated in higher amplitude sections. One can think of it as decreasing the Clip Fraction in proportion to signal amplitude. Negative values may also be entered for Amplitude Weighting, in which case processing is concentrated in the sections of ***lower*** amplitude. The maximum range of the Amplitude Weighting parameter is from +1.0 to -1.0. In practice, values less than plus or minus 0.5 are used, with possibly higher values for special purposes such as distortion removal (see next section).

Note: As Amplitude Weighting diverges (positively ***or*** negatively) from zero, it is recommended that the Clip Fraction be reduced by some percentage as well. Otherwise, there may be no processing at all in some regions of the signal.

Removing Peak Distortion

To some extent, the Decrackle function can be used to ameliorate or remove breakup and distortion associated with high signal levels. By using the Amplitude Weighting factor, decrackling becomes concentrated entirely in the highest peaks of the waveform, using the good portions of the wave to reconstruct the portions that are flat-topped or otherwise distorted. In many cases, this approach is able to restore the distorted portions enough to reduce the audible distortion significantly.

To use Decrackle to remove clipping and other high-level distortions:

1 Open the (clipped) soundfile in SonicStudio HD and capture it in a new Track with gain reduction of between 6 and 10 dB.

This ensures that headroom exists for correction. If the source audio is clipped, then it is to be expected that the reconstruction will extend beyond the original top of the waveform. Headroom must be above the clipping level for this reconstruction to take place.

2 Run a Decrackle pass on the resulting soundfile with the Amplitude Weighting parameter set high (perhaps 0.7 or 0.8), and the Clip Fraction somewhat reduced (0.80 or 0.75)

The range of variation in source materials and possible distortion types is huge. Experimentation, using a short section of the source file, is recommended to determine the optimal settings for distortion removal.