

AmadeusLite

The intuitive yet powerful wave editor



User reference guide for v2.0

Martin Hairer

Copyright ©2011 by HairerSoft

Contents

1	Introduction	5
2	Basic notions of sound editing	5
2.1	What is a sound?	5
2.2	How does a computer represent a sound?	6
2.3	Sound intensities: dB versus percentages	7
2.4	Overview of an AmadeusLite document	8
2.4.1	Sound overview and waveform	8
2.4.2	Playback / panning envelope	8
2.4.3	Selection	9
2.4.4	Playback head	9
2.4.5	Markers	9
2.4.6	Metadata	10
3	File Formats	10
3.1	The 3GP / 3GPP-2 Audio formats	11
3.2	The AAC ADTS, .mp4, .m4a, iPhone ringtone, and Bookmarkable AAC formats	11
3.3	The AIFF format	11
3.4	The AMR format	12
3.5	The Apple CAF File format	12
3.6	The FLAC file format	12
3.7	The Mp3 file format	12
3.8	The NEXT/Sun format	13
3.9	The Ogg Vorbis format	13
3.10	The QuickTime Movie format	13
3.11	The SoundDesigner II format	14
3.12	The Wave format	14
3.13	Windows Media Audio (WMA) format	14
3.14	Importing raw data	14
3.15	Importing and exporting markers	14
3.16	iTunes support	15
4	Sound recording and playback	15
4.1	Sound recording	15
4.2	Sound playback	16
4.3	Adjusting the playback device	17
5	Editing	17
5.1	Navigation	17
5.2	Basic editing	18
5.2.1	Inserting silence	19
5.2.2	Changing the characteristics of a sound	19
5.3	Applying effects to a sound	19

5.3.1	Echo	20
5.3.2	Amplify	20
5.3.3	Equalizers	20
5.3.4	Fadings	20
5.3.5	Normalizing a sound	21
5.3.6	Reverse	21
5.3.7	Interpolate	21
5.3.8	Denoising	22
5.3.9	Audio Units	22
5.3.10	Change Pitch and Speed	24
5.4	Maintaining a list of commonly used actions	24
5.5	Working with presets	24
6	Key shortcuts	25
7	Preferences	26
7.1	General	26
7.2	Sound	27
7.3	Colors	27
7.4	Temp Files	27
7.5	Units	27
7.6	Advanced preferences	28
8	How to's	28
8.1	How to work with your iTunes music library	28
8.2	How to convert a sound from one file format to another	28
8.3	How to create a CD from a tape or vinyl recording	28
8.4	How to record from the output of another program	29
9	Troubleshooting	30
9.1	I can't hear anything when playing back	30
9.2	I don't hear what I am recording	31
9.3	I can't record anything or the recording is not what I expect	31
9.4	Frequently Asked Questions	31
9.4.1	What are the differences between AmadeusLite and AmadeusPro ?	32
9.4.2	Why are CD track names all wrong on CDs burned by AmadeusLite ?	32
9.4.3	Why does AmadeusLite refuse to create an Mp3/AAC file from my recording?	32
9.4.4	Can AmadeusLite record the output from another program?	32
9.4.5	Could you add a key shortcut for function XYZ?	32
9.4.6	The sliders for the record levels are greyed out and the input level is inappropriate, why?	33
9.4.7	AmadeusLite attempts to make an internet connection at startup! What the heck?	33
9.4.8	I cannot open m4p files purchased on Apple's iTunes Store with AmadeusLite , why?	33
10	Acknowledgment to third party software	33
10.1	Ogg Vorbis	33

10.2 LAME	34
10.3 mpg123	34
10.4 FLAC	35
10.5 AtomicParsley	35
10.6 FFmpeg	35
10.7 MTCoreAudio	35
10.8 AMPreferencePane	36
10.9 ILCrashReporter	36
11 Bug reporting	36
11.1 Disclaimer	37

1 Introduction

Thank you very much for choosing to use **AmadeusLite**. This is a very high quality feature-packed audio editor which is nevertheless easy to use and very reasonably priced.

This manual will guide you through the main features of **AmadeusLite**. It is not intended to be an exhaustive reference text, but it will focus on those features that are maybe not absolutely intuitive at first glance. The best way by far to find out how a feature that is not covered in detail by this manual works is to simply try it out and see what happens. Most of the time, it should be pretty much what you expected from the name of the command and / or from the conventions in use by most Mac OS X applications. You should not be shy about trying functions out, since destructive actions can always be undone, unless a specific warning is given that this is not the case.

This manual also assumes that you are reasonably familiar with computers and know how to use basic Mac OS X applications. (Think of this as your car manufacturer assuming that you actually passed your drivers license test before buying a car from him.) In particular, it assumes that you know about files, windows, the menu bar, etc. If you have little familiarity with sound editing, it is probably a good idea to start by reading Sections 2 and 5 in their entirety, since these sections explain the basic terminology and concepts used in this manual.

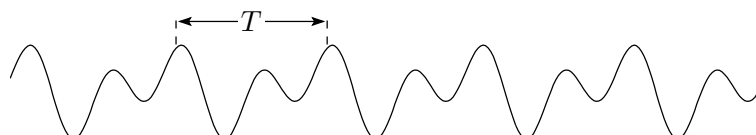
This manual is a PDF file with hyperlinks. All words written in blue (this includes not only URL's but also references to sections and page numbers for example) are clickable. Some terms that may not be familiar to everyone are linked to their [Wikipedia](#) entries. This is why no glossary is provided, since the Wikipedia entries are usually quite well written and more complete than any glossary that could be provided here. It is hoped that these Wikipedia links are stable, but it may of course happen that they break sometime in the future. If you find a broken link, please inform me by sending an email to Martin@HairerSoft.com.

2 Basic notions of sound editing

2.1 What is a sound?

Let's start with the basics since, in order to understand how a sound is represented and can be manipulated by a computer, one should have some understanding of what a sound actually is physically. In a nutshell, a sound is a small rapid variation in the air pressure. The variation of pressure is captured by your internal ear and transmitted to the brain, which interprets this signal as a sound.

A microphone is nothing but an artificial ear. It measures the air pressure and transforms it into an electric impulse. A sound is therefore usually represented as a function representing the air pressure (or equivalently the voltage at the exit of the microphone) as a function of time. A typical representation of a sound signal would be:



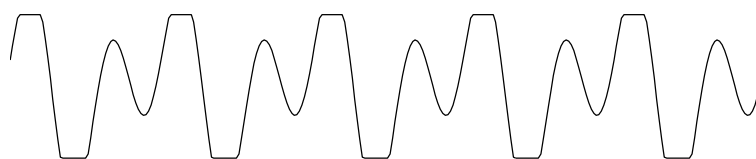
Note that, very often, the signal is (at least approximately) periodic as in the picture above. The time interval between two identical looking chunks of the signal, denoted by T in the picture above, is called the *period*

of the signal and is usually measured in milliseconds. The inverse of the period is called the *frequency* of the signal and is measured in Hertz, also abbreviated by Hz. Some high-frequency sound signals are measured in kiloHertz (that is thousands of Hertz), also abbreviated by kHz. A Hertz is the inverse of a second. So a sound with a period of 0.05 seconds (or equivalently 50 milliseconds) has a frequency of $1/0.05 = 20$ Hertz.

The human ear can hear sounds that have frequencies ranging approximately between 20Hz and 20kHz. These bounds however vary with age, with older people less able to perceive high frequencies. Sounds with frequencies lower than about 20Hz are called *infrasounds* and sounds with frequencies higher than about 20kHz are called *ultrasounds*. As an aside, this is an interesting experiment to perform: generate a sound with a frequency of say 16000Hz with **AmadeusLite**'s **Effects** \Rightarrow **Generators** \Rightarrow **Waveform** tool and play it back. Younger people will be able to hear it without problem and it will be perceived by them as a very annoying sound. Older people will be completely unable to hear it.

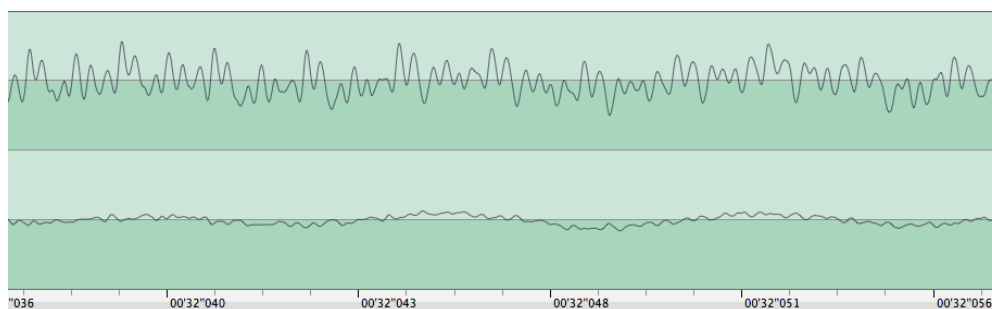
2.2 How does a computer represent a sound?

A computer can only handle discrete pieces of data. A signal like the one depicted above will therefore be transformed into a set of discrete values. Each of these values is called a *sample* and the number of values per second used to represent the signal is called its *sampling rate*. Therefore, if a signal has a sampling rate of 44.1kHz (the rate used in audio CDs), then the signal is measured 44100 times per second. The individual values making up the signal are of course discrete as well. If the signal is stored in the computer by using 2^N different values, it is said to have a *sampling depth* of N . Sound on an audio CD has a sampling depth of 16, meaning that it is stored using $2^{16} = 65,536$ different values. The characteristics (sampling rate and sampling depth) of a sound are displayed in the bottom left corner of the sound window. For practical purposes, it is usually assumed that samples take values between -1 and 1 . In the previous example of a 16bit sound, this means that every sample can take 65,536 values that are equally spaced through the interval $[-1, 1]$. If a sound containing samples that are close to these limits is amplified, a phenomenon known as 'clipping' occurs. This is when, as a result of a sound effect being applied to it, some samples 'would like' to take values outside of the $[-1, 1]$ range and are forced back into it by the computer. If the sound signal represented above was to be amplified too much, it could therefore end up looking like this:



When clipping is severe, it introduces very noticeable artifacts that are quite difficult to get rid of, one should therefore take care to avoid it.

Humans perceive sounds with two ears. If one wishes to reconstruct accurately an auditory experience, it is common to use two microphones, one for each ear. This is called a *stereo* sound. When only one microphone is used, it is called a *mono* sound. The following screenshot shows how a stereo sound is displayed in **AmadeusLite**:



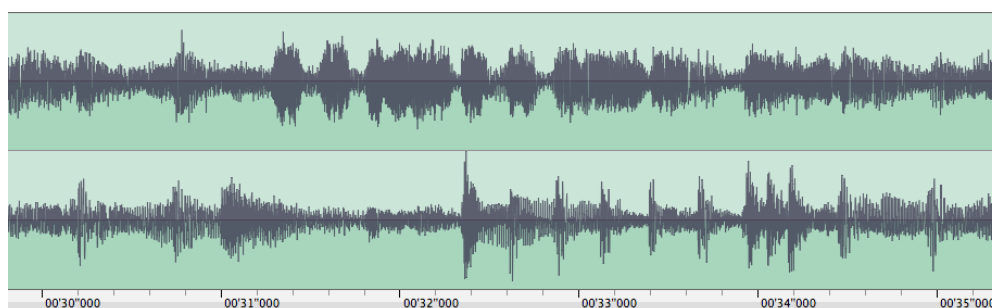
The upper signal corresponds to the left ear and the bottom signal corresponds to the right ear.

Note that the structure of the signal (also commonly referred to as the ‘waveform’) only becomes apparent when one zooms into the signal. For example, in the example above, the width of the whole display only corresponds to about 50 milliseconds (one twentieth of a second). Zooming into the display can be achieved with the **Horizontal zoom** slider:



The number appearing next to the slider when you move it represents the number of samples of sound that fit into one pixel of your screen. Clicking on the text saying “Horizontal zoom” allows you to change this into a “Vertical zoom” slider that controls the vertical scale at which the waveform is displayed. (By default, the maximal possible value for a sound sample corresponds exactly to the height of the display.)

When a larger portion of the sound is shown, a typical display would rather look like this:



Here, you can see that the width of the display is about one minute. At this scale, the structure of the waveform is not apparent anymore. What you can see essentially represents the volume of the sound as a function of time (the fatter the dark blue area, the louder the sound).

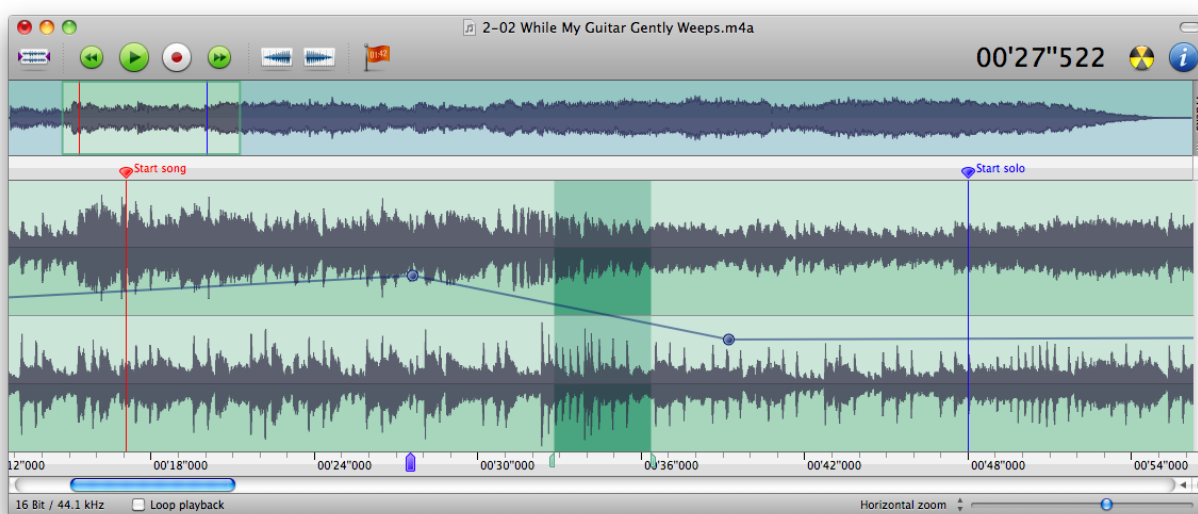
2.3 Sound intensities: dB versus percentages

The intensity of a pure sine wave is defined as the square of its amplitude. It is normalised in such a way that a pure sine wave with the maximal amplitude has an intensity of exactly 1. **Decibels** (abbreviated as dB) are a logarithmic unit that better reflects the fact that our perception of sound intensity is not linear. A *difference* of 10dB corresponds to a *ratio* of 10 in intensity. It is normalised in such a way that a sine wave with maximal intensity 1 corresponds to 0dB and a sine wave with intensity 0.1 corresponds to -10dB . This is why intensity values displayed in dB will always be *negative*.

For ‘physical’ rather than ‘digital’ sounds, the reference is usually chosen such that 0dB corresponds roughly to the threshold of human hearing, so you will often see sound intensity measurements with *positive* dB values. Since a given physical sound can end up at any intensity in the computer, depending on the setting of both the recording and the reproduction equipment, there is no convenient way of converting between the two. However, *differences* in dB values have the same meaning in both cases.

2.4 Overview of an AmadeusLite document

An **AmadeusLite** document consists of one single sound track, that can be either mono (one channel) or stereo (two channels). A typical document will be displayed by **AmadeusLite** in the following way:



The next sections are a short guided tour through the various elements that can be seen in this picture.

2.4.1 Sound overview and waveform

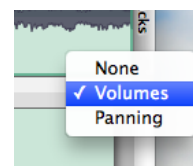
The top part of the window shows an overview of the whole sound document. This part is “optional”: you can show and hide it by clicking on the top left icon of the toolbar or by selection **Show / Hide Whole Sound** in the **View** menu. The overview is convenient for fast navigation within a document: click onto a location to navigate to it. Part of the overview is highlighted: this is the part of the sound that is shown in the main part of the window below.

2.4.2 Playback / panning envelope

You can see on this picture that a sound document can have a playback envelope that is represented in this picture by a thick dark blue line which connects knobs. New knobs can be added to the playback envelope by simply clicking on it. A knob can be removed by dragging it out of the window. The first and the last

knob are fixed and cannot be removed. The playback envelope allows you to change the volumes of different sections of a sound independently.

Similarly, the panning (whether it appears to come from a source placed more to the left or to the right) of a sound document can be adjusted as a function of time in pretty much the same way. Whether the document window displays the playback envelope or the panning can be adjusted by using the **Show Envelope For** submenu of the **View** menu or by using the small pop-up menu located at the top-right corner of the main document window, just below the overview (see the screenshot to the right)



The **Apply Envelopes to Sound** function from the **Sound** menu allows you to apply both the playback envelope and the panning envelope to the open sound. This is a destructive operation that does actually alter the sound data.

2.4.3 Selection

A portion of the sound can be selected by clicking into the sound and dragging the mouse while holding its button down. In our example, the portion of sound between about 32 and 36 seconds from the start is selected.

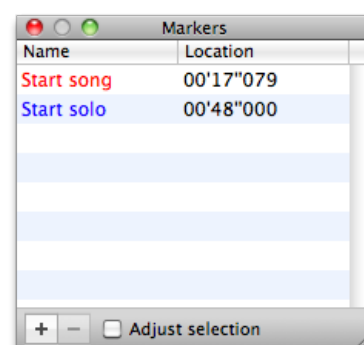
If a sound is stereo, it is possible to select only the left (respectively right) channel by clicking near the top (respectively the bottom) of the track in question. In this case, the 'cross' cursor will be modified to show a small 'L' (respectively 'R'). This allows for example to apply an effect to one channel only.

2.4.4 Playback head

The blue shape located in the timescale at around 27.5 seconds is the playback head. This is where playback would start (resume) if you hit the space bar. For more information on sound playback, see Section 4.2.

2.4.5 Markers

As one can see in the screenshot, a document can have an arbitrary number of markers ('Start song', 'Start solo', etc). These are just handy ways of remembering a particular location in a document. In order to create a marker, simply click in the white / gray area at the top of the window or move the insertion point to the correct location and type 'm'. Markers can also be given colors, for example in order to distinguish between markers that serve different purposes. One convenient way of working with markers is the **Markers** tool that can be accessed from the **Windows** menu. When you click on a marker in this tool, the display of the current window changes so that the clicked marker is moved to the center of the window. If you use shift-click to select several markers, the display is zoomed out (or in) in such a way that all the selected markers fit within the width of the current window. If **Adjust selection** is checked, then the selection of the current track is adjusted so that it selects the portion of sound between the chosen markers. If only one marker is chosen, it moves the insertion point to the location of that marker. The '+' button creates a marker at the center of the



selection of the current track or at the insertion point if the selection is void. The ‘-’ button allows you to delete the selected marker(s).

The four last items in the **Selection** menu are devoted to the management of markers. Since the meaning of the first two should be obvious, let us only concentrate on the function of the remaining two:


- **Generate Markers...** Allows to generate equidistant markers on your sound. This is especially useful in conjunction with the **Split According to Markers...** function if you want to cut a very long sound into pieces of reasonable size.

You can choose **Search for silences** in order to place markers at the location of silences in your sound. The **Maximal level** parameter gives the threshold under which a sound is considered to be silence. The **Minimal length** parameter gives the minimal length a silence must have in order to be taken into account.

If a number is found in the marker name entered in the **First marker** text field, it is automatically incremented for each subsequent marker.

- **Modify Markers...** This item allows to modify all the markers of a file in one go. If **Change name** is checked, **AmadeusLite** will use the corresponding name for the first marker and will try to make a guess on how to increment it for the subsequent markers. For example, if you specify ‘Track 01’, the next markers will be called ‘Track 02’, ‘Track 03’, etc. The **Align position** field allows to align the marker positions on a specified time grid. For example, specifying ‘75 Hz’ for a file sampled at 44.1kHz allows to align markers with sector boundaries on an Audio CD (tracks on Audio CD’s always have a length that is a multiple of 1/75th of a second). This is useful to burn a CD without gaps between songs.

2.4.6 Metadata

An **AmadeusLite** document can furthermore contain arbitrary amounts of metadata, that is data which is not directly relevant to the editing and playback of a sound, but which gives some additional information. Examples of metadata are the song and artist’s name, album cover artwork, etc. 

Metadata can be viewed and edited by clicking on the **Information** button in the toolbar or by selecting **Show Sound Info...** in the **Sound** menu. This will also show some generic information about the sound, like its length, file format, sampling rate, etc.

3 File Formats

AmadeusLite supports a variety of popular sound formats. Some file formats have some limitations that prevent them from being able to represent an **AmadeusLite** document faithfully. The shortcomings of the various formats will be explained below.

As a general fact, sound documents can take up a large amount of disk space, and so many formats include some compression algorithm aimed at minimising the size of the resulting file. There are two families of such compression algorithms: *lossy* algorithms and *lossless* algorithms. Lossy algorithms allow for high compression rates, but this comes at the cost of a loss of information. In other words, every time you decode a document compressed with a lossy algorithm (for example to edit it) and then save it back, the sound will lose a bit of its quality. Lossless algorithms on the other hand preserve the data in its entirety, so that there

is no loss of quality coming from successive edits. The downside of course is that lossless algorithms do not usually achieve very high compression rates.

If you wish to edit a file encoded with a lossy algorithm, it is therefore **strongly recommended** to save intermediate steps of the editing process to an uncompressed format or a format using a lossless algorithm.

Typical lossy algorithms are Mp3, AAC, WMA and Ogg Vorbis. The AIFF and WAVE formats store uncompressed data by default but also allow you to use lossy compression algorithms like μ -Law, a-Law, MACE-3, MACE-6, QDesign Music 2, and AMR Narrowband. It is beyond the scope of this manual to discuss the suitability of these algorithms for various applications. (Please don't send me emails asking for advice. Try to experiment by yourself or look for advice on the internet, there is plenty of information available online.) Typical lossless algorithms are FLAC and Apple Lossless.

Some algorithms compress data by a fixed factor (μ -Law and a-Law compress by a factor 2, MACE-3 and MACE-6 compress by factors 3 and 6 respectively), but most other algorithms support several compression factors. A standard measure of compression is the *bitrate* which measures the amount of disk space (in bits per second; remember that one *byte* equals 8 *bits*) which the resulting file takes. A bitrate of 128kbps thus corresponds to about 16kB per second, which is roughly 10 times less than the amount of disk space taken by uncompressed sound in CD quality (44.1kHz, 16Bits, stereo). Smaller bitrates of course come at the expense of lower sound quality.

Some formats are not suitable to retain all the attributes of an **AmadeusLite** document. If this is the case, a warning will be shown, together with a short explanation of the features that may be lost. The different file formats have specific settings that can be accessed by clicking on the **Settings** button in the file saving dialog. Below is a short description of each of them, roughly in alphabetic order.

3.1 The 3GP / 3GPP-2 Audio formats

These are popular compressed audio formats typically designed for playback on mobile phones. In principle, these container formats support the presence of video tracks, but **AmadeusLite** will not preserve these.

3.2 The AAC ADTS, .mp4, .m4a, iPhone ringtone, and Bookmarkable AAC formats

These are different container formats that are primarily used to save data compressed with the lossy AAC compression algorithm. Metadata is lost when saving to the AAC ADTS format, but most of it is preserved when saving to the three other formats.

Note that the AAC ADTS format is *not* the same as the .mp4 or Apple's .m4a format. These are three different container formats and QuickTime for example will not be able to read AAC ADTS files. On the other hand, the .m4a, Bookmarkable AAC, and iPhone Ringtone formats differ only in the file extension, but are treated differently by iTunes and the iPod.

In most cases, you will probably wish to use the **MPEG Audio (.m4a)** format as this is the format used by Apple's iTunes Music Store.

3.3 The AIFF format

This format is very popular mainly on the Mac OS platform. It is similarly structured to the Wave format and usually contains uncompressed data, but also supports a variety of compressed formats.

The AIFF format supports markers and various metadata. Metadata (artwork, artist name, etc) are saved to AIFF files by appending an ID3 tag at the end of the file. This is the method employed by iTunes and ensures that metadata created with **AmadeusLite** can be read by iTunes.

Due to the nature of the format, AIFF files are limited to 2GB in size. It is possible to encounter AIFF files that are between 2GB and 4GB in size. Such files *do not* comply to the official AIFF specifications and cannot be opened by every program supporting the AIFF format, this is why **AmadeusLite** will refuse to create them. It is however able to open such files and to save them in other file formats.

3.4 The AMR format

This is another popular format for sounds played back by mobile phones.

3.5 The Apple CAF File format

This is a format developed by Apple and designed to replace AIFF in the long term. It removes the 2GB file size limitation of AIFF and supports a wider variety of compression algorithms.

Metadata is lost when saving to the Apple CAF format (this may change in the future, as the format in principle supports the inclusion of metadata).

3.6 The FLAC file format

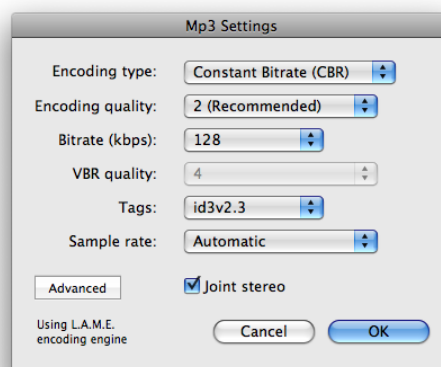
FLAC (Free Lossless Audio Codec) is a compressed lossless sound format popular on the Linux platform. It is possible to choose between different compression levels. Since the format is lossless, there is no downside in principle to choosing the highest possible compression, but it may make file saving operations significantly slower for a minor gain in file size.

Most text-based metadata is preserved in the FLAC file format, but artwork is lost.

3.7 The Mp3 file format

This is an extremely popular lossy sound compression algorithm. **AmadeusLite** reads Mp3 (and also Mp2) files through the tool `mpg123` (see Section 10 below).

Saving Mp3 files is achieved through the tool `lame`. (See also Section 10 below.) It supports a variety of settings (the meaning of the most important one, the bitrate, was explained above). In order to achieve a given file size, it is recommended to use the **ABR** (Average Bit Rate – simple parts of the sound will be compressed more than complex parts, achieving a better overall quality) setting, rather than **CBR** (Constant Bit Rate – all parts of the sound have the same compression rate) in order to achieve maximum quality, but note that some Mp3 players may not support ABR files. The **VBR** (Variable Bit Rate) setting is more flexible and allows you to specify a target quality rather than a target file size. This usually results in better sounding files, but



you have less control over the resulting file size ('complex' sounds will produce larger files than 'simple' sounds for a given quality). Note also that as with ABR files, not every Mp3 player supports VBR files. The screenshot next to this paragraph shows a popular setting that represents a good trade-off between file size and sound quality. Note that the use of id3v2.4 tags is discouraged since their support is broken in iTunes (this is true as of iTunes 7).

The **Sample rate** setting allows you to resample the sound before encoding. In most cases, the **Automatic** setting will yield best results. For example, when encoding a 44.1kHz AIFF file at 32kbps, `lame` will resample it to 16kHz. This will in general yield better result since less bandwidth is wasted on high-frequencies and so the lower frequencies can be described more accurately. By changing this setting to **Original**, this behaviour can be suppressed.

Metadata (including artwork) is preserved in Mp3 files, provided that you allow the creation of an id3v2 tag. If you choose to create only an id3v1 tag (maybe for reasons of compatibility with very old software), then only some text-based metadata is retained.

The **Advanced** settings are only for expert usage. If you are familiar with the command-line tool `lame`, they are self-explanatory. Otherwise, you will probably not need them and can ignore this setting.

3.8 The NEXT/Sun format

A popular sound format on the Linux platform. **AmadeusLite** only supports NEXT/Sun files containing uncompressed sound. It will downmix multitrack documents it to a mono or stereo file, whichever is more suitable. Metadata is lost when saving to the NEXT/Sun format.

3.9 The Ogg Vorbis format

This is a popular alternative to the Mp3 format, especially on the Linux platform. At intermediate bitrates (128–256kbps), most listening tests favour Ogg Vorbis over Mp3 in terms of quality. From an end-user's perspective both formats are very similar in nature and most of what is said for the Mp3 format below also applies to the Ogg Vorbis format. In particular, the settings are almost identical.

Vorbis is the first of a planned family of Ogg multimedia coding formats being developed as part of [Xiphophorus's Ogg multimedia project](#).

3.10 The QuickTime Movie format

This file format supports a wide selection of compression formats. When a multitrack document is saved as a QuickTime Movie file, **AmadeusLite** will downmix it to a mono or stereo file, whichever is more suitable. Metadata is lost when saving to the QuickTime Movie format.

Important: Unlike **AmadeusLite** 1.x, **AmadeusLite** 2.0 *does* preserve video tracks in QuickTime movies. It is therefore possible to simply open a video clip in **AmadeusLite**, edit its sound track, and save it back. However, video data is lost when saving to any other file format, including those formats that would in principle support the presence of a video track.

3.11 The SoundDesigner II format

This used to be a popular format on the Macintosh platform and it is still the format of choice for the software Digital Performer. It can only contain uncompressed sound data, but has the advantage over AIFF and Wave of not having a 2GB file size limitation.

The disadvantage of this format is that all metadata is lost when saving to it.

3.12 The Wave format

This format is very popular mainly on the Windows platform but also on other operating systems. It usually contains uncompressed data, but also supports a variety of compressed formats. **AmadeusLite** only supports Wave files that are either uncompressed or compressed with the (lossy) μ -Law or a-Law algorithms.

Like the AIFF format, the Wave format supports markers and metadata. Metadata (artwork, artist name, etc) are also saved to Wave files by appending an ID3 tag at the end of the file.

Due to the nature of the format, WAVE files are limited to 2GB in size. The same remark as for AIFF files applies to WAVE files of sizes between 2GB and 4GB.

3.13 Windows Media Audio (WMA) format

This is a very popular format on the Windows platform. It is a lossy format consisting of compressed sound at a fixed bitrate. It is possible to choose the bitrate, as well as the version of the WMA format (7 or 9) used for compression. Usually, WMA 9 is preferred over WMA 7, unless the sound needs to be readable by software supporting only the older version of WMA.

The disadvantage of this format is that all metadata is lost when saving to it.

3.14 Importing raw data

AmadeusLite is also able to import sound as raw sound data. Unless you are trying to recover data from corrupt files or know precisely what you are doing, you are unlikely to need this functionality.

In particular, please do *not* use the **Import Raw Data** function to open normal sound files (AIFF, Wave, Mp3, etc) unless you really do know what you are doing. Results are likely to range somewhere between mild annoyances and total garbage.

If you are using the **Import Raw Data** function in order to recover data after a crash of **AmadeusLite**, you should make sure that the quality displayed in the import dialog matches the one of the document that you were working on before the crash.

3.15 Importing and exporting markers

The **Export Markers...** function of the **File** menu allows you to export the markers of the currently open document as a **.cue file**. This is a text-based file format that is commonly used by CD burning programs. This format has the advantage of being editable by any text editor (like for example the free **TextWrangler**), so that it is also possible to create your own **.cue files** 'by hand' or to modify **.cue files**. The **Import Markers...** function allows you to open **.cue files** and to add the corresponding markers to the current document.

3.16 iTunes support

AmadeusLite plays nicely with iTunes. The **File** menu has an **Export to iTunes library...** item that allows you to add the open document to your iTunes music library. It is recommended to set the metadata (song name, artist name, etc) prior to selecting this function, so that the song shows up properly in iTunes.

Next to this, there is a **Export as iPhone Ringtone...** item that does pretty much the same thing, but instead of adding the document to your music library, it adds it to your ringtone library.

Warning: In order to work correctly, both of these functions require the “Copy files to iTunes Media folder when adding to library” option in iTunes’s “Advanced” preferences to be checked. (This is iTunes’s default behaviour.) If this is not checked, the files will not be added to your iTunes library.

Equally, it is very easy to import a song from iTunes: simply open iTunes and drag the song from iTunes onto the **AmadeusLite** icon in the dock. This will open the corresponding file and allow you to edit it!

4 Sound recording and playback

4.1 Sound recording

The first step for any recording is of course to set your hardware up in such a way that the sound gets into your computer. Most Macintosh models have a line-in audio input built in, so you can plug the output of your Hi-Fi system directly into it (you will just need a standard [mini-jack](#) cable). If this is not the case for your computer (check with your local Apple store if you are unsure about which one is the line-in plug) or if you wish to obtain better control and / or better sound quality, you can use one of many third-party audio devices. Any audio device compatible with Mac OS X should be recognized correctly by **AmadeusLite**. Popular brands include [Griffin’s iMic](#), [Motu](#), [Roland / Edirol](#), [Digidesign](#), etc. I am not affiliated with any of these manufacturers and I can therefore not provide any advice on comparing their products or using them.

Once your hardware is set up, the easiest way of recording a sound with **AmadeusLite** is to click on the red **Record** button depicted here. By default, this is exactly the same as choosing **Record...** in the **Sound** menu. This will bring up a standard recording dialog and start the recording immediately. (It is possible to change this behaviour in the preferences.) Once your recording is complete, just hit **Return** or click the **Stop** button and the recorded sound will be inserted at the insertion point in the current document. During the recording, you can pause and restart it at any time by hitting the **Pause** button (see the picture below the record button).



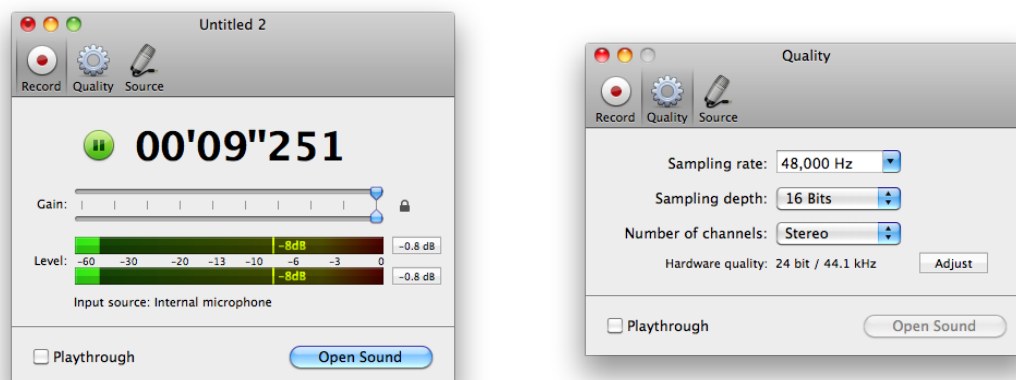
The standard recording dialog only provides control over the volume of the recording. The volumes of both channels (for a stereo recording) can be adjusted individually, but clicking on the little padlock forces **AmadeusLite** to use the same volume for both channels.

Warning: **AmadeusLite** is able to provide volume control only if your hardware allows it. Some sound recording devices do not provide any control over the recording volume and in this case, the corresponding sliders will be grayed out. Please do not report this behaviour, as it does not constitute a bug. **AmadeusLite** does intentionally not provide software volume control at the time of recording since this can only diminish the resulting sound quality. If you are not able to adjust the volume correctly in the recording window, you should make sure that your hardware is set up properly.

One very common mistake is to plug a microphone directly into a [line-in](#) audio input. Without a [preamplifier](#), this will result in extremely low sound volume and poor sound quality. The iMic for example provides a

built-in **preamplifier** that can be controlled with a little black switch. A similarly common mistake is to plug the line-out output of a home Hi-Fi system into a microphone input. This will result in a very saturated sound. Both of these problems can only be solved by setting up the hardware properly and **not** by any software solution. If you are unsure about how to connect your sound source to your computer, please seek advice from your nearest Apple store or from the manufacturer of your audio input device.

Complete control over sound recording can be achieved by choosing **Record to New File...** in the **Sound** menu. This brings up the following dialog:



The **Record** tab provides the same functionality as described above.

The **Quality** tab allows you to specify the sampling rate and sampling depth of the resulting sound. Clicking on the **Adjust** button allows you to attempt to adjust the hardware quality to match the selected quality as closely as possible. If it is not possible to match the selected quality exactly, **AmadeusLite** will automatically convert the recorded sound to the selected quality. This tab also allows you to choose which channels to record from. From the point of view of **AmadeusLite**'s recording function, every sound input device has 2 channels: left and right. Your particular sound input device may have more (or less) input channels than that. It is therefore possible to assign the physical channels of your input device to the 2 channels of **AmadeusLite**'s recording engine by clicking on **Advanced settings** in the **Input Source** tab.

The recording window obeys to a number of handy key shortcuts that are summarised in the following table:

key	action
space	Pauses / restarts the recording
'M'	Creates a marker at the current location of the recording
'm'	Same as 'M', but a dialog comes up, allowing to edit the marker
tab	Same as 'M'.

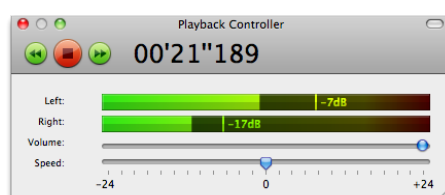
4.2 Sound playback

Once a sound file has been opened in **AmadeusLite**, the most common task is to play it back. In a nutshell, playback is controlled by the spacebar. Hit the spacebar to start it and hit it again to stop it. When no sound is playing, hitting the tab key is equivalent to first hitting the **rewind** button and then hitting the space bar. When a sound is playing, hitting the tabulation key is equivalent to first hitting the space bar and then hitting the **rewind** button.

The current playback position is indicated by the playback head, which is displayed in the timescale of the main document. The playback head is depicted on the right. The location of the playback head can be changed by simply clicking at the corresponding location in the timescale. Whenever you click in the main sound window and the sound is not being played back, the position of the playback head is reset to the position selected in the **General** pane of the **AmadeusLite** preferences dialog. This position would be either the start of the current selection or the start of the document. The precise behaviour of the space bar is the following. If no sound is currently being played back, playback starts at the location of the playback head. If a sound is being played back, playback stops. If **Space bar rewinds sound** is checked in the **General** pane of the **AmadeusLite** preferences dialog, then the position of the playback head is reset, otherwise it stays where it was when the space bar was hit.



More sophisticated control can be achieved through the playback controller:



The **Play** icon has exactly the same functionality as hitting the spacebar. The **Fast forward** and **Fast rewind** icons allow you to scrub through a sound while it is playing back. The **Speed** slider allows you to adjust the playback speed. The scale is measured in semitones (12 semitones make up one **octave**) so that maximal playback speed is double the original speed and minimal playback speed is half. The **Volume** slider allows you to adjust the overall volume of the sound being played back.

4.3 Adjusting the playback device

The sound output device used for playback can be selected in the **Sound** preferences panel accessible by choosing **Preferences...** in the **Amadeus Pro** menu.

Like the situation for recording, from the point of view of **AmadeusLite**'s playback function, every sound input device has 2 'virtual' channels: left and right. It is possible to assign the physical channels of your input device to the 2 virtual channels by clicking on **Advanced settings** in that preferences dialog. On a normal two-channel output device (built-in speakers for example) the assignment will be correct by default. However, if you use a multichannel device for playback, you may have to adjust this.

Note that by default, the output device used for playback is the one selected in the **Sound** system preferences dialog. However, if you set a playback device using **AmadeusLite**'s preferences, this takes precedence over any settings that you may have in the system preferences. The setting will also be remembered from one session to the next.

5 Editing

5.1 Navigation

The first thing anyone would like to do when editing a file is of course to find one's way around the file. The two most obvious ways of navigating through a sound are the **Horizontal zoom** slider that allows you

to change the timescale and the horizontal scrollbar located just below the timescale.

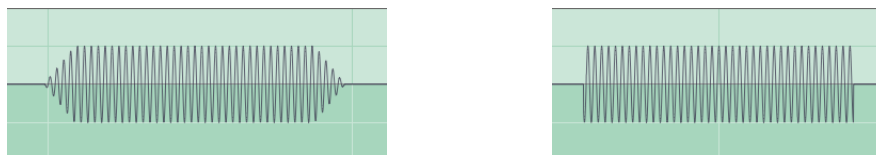
However, there are many other ways to finding your way around a document. These are:

- The **Markers** palette already mentioned in the previous section.
- The ‘+’ and ‘-’ key shortcuts that allow to zoom in and out of the document.
- The **Zoom In / Zoom Out** items in the toolbar (if they are not there and you happen to need them, just choose **Customize Toolbar** in the **View** menu). There are corresponding menu items in the **View** menu.
- The **Zoom to Selection** item in the **View** menu changes the display in such a way that the selected portion of the current track just about fills up the display. If the selection is empty, the waveform is expanded to the maximum (1sample = 1pixel) and it is moved in such a way that the insertion point is located at the center of the display. The **Zoom to Document** item in the same menu has a similar effect, but changes the zoom level so that the entire document neatly fits into the window.
- The scroll wheel of your mouse also allows you to zoom in and out of a document. If your mouse has a horizontal scroll wheel as well (like Apple’s Mighty Mouse or the ‘two finger scrolling’ feature on the trackpads of recent laptops) you can use it to move the document horizontally. There is an advanced preference that allows you to exchange the behaviour of the horizontal and the vertical scroll wheel.
- The **Move to Selection** and **Move to Playback** menu commands move the display (without changing the timescale) in such a way that the center of the selection (or the playback head) is at the center of the display.
- The keys ‘1’ to ‘4’ allow you to set the zoom level to one of four predefined values. You can set the value corresponding to a key to the current zoom level by typing ‘Option-1’, ‘Option-2’, etc.

5.2 Basic editing

Most of the basic editing functions (Copy / Paste / Cut / Delete) work in exactly the same way as in any text editor or image editor. One difference though is that, by default, **AmadeusLite** uses “smart editing”. This means that when you tell it to copy a piece of sound, it actually copies a slightly larger piece. When you then paste this into some other sound, it uses the extra piece to create a smooth crossfade between the copied sound and the recipient. This avoids the appearance of unwanted “cracks” when simply pasting the sound creates a discontinuity in the waveform.

This is maybe best illustrated by what happens when a short pure tone is pasted into a silent sound. The left picture shows the effect when smart editing is turned on, while the right picture shows the effect when it is turned off:

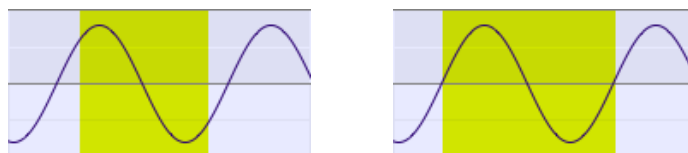


The length of this crossfade can be adjusted in the general preferences (select **Preferences** in the **Amadeus Lite** menu) under **Crossfade for cut / paste**. Setting it to 0 turns smart editing off altogether.

If smart editing is turned off, it is useful to preview the effect of an edit before actually performing it. If the selection is non-empty and one hits the ‘e’ key, **AmadeusLite** will play a short portion around the selected area of the current track, but *omitting* the selection in question. This allows to check for example whether cutting away a piece of sound will introduce a small artefact or not. One way of avoiding the creation of artefacts when smart editing is off is to use the **Extend to Crossing** function in the **Selection** menu. The length of the preview can be adjusted with the **Length of edit preview** function of the general preferences.

The **Paste Over** function allows you to literally paste the content of the clipboard over the current sound, effectively mixing it with the document instead of just inserting it at the insertion point.

The **Extend to Crossing** function in the **Selection** menu allows you to extend the current selection to the nearest points where the waveform crosses the 0-line. This is illustrated by the following two screenshots, where the left hand side shows the selection of a sound before applying this function and the right hand side shows the same sound after applying the **Extend to Crossing** function.



This is again very useful in order to avoid little crackling noises at the points where sounds are cut and pasted when smart editing is turned off.

5.2.1 Inserting silence

To insert a silent portion of a given length, use the **Generate Silence...** function from the **Effects** menu. To silence a portion of a sound, first select the portion you wish to silence and then choose **Amplify** from the **Effects** menu, with a setting of 0%. (Make sure that % is selected and not dB!)

5.2.2 Changing the characteristics of a sound

The characteristics (sampling rate, bit depth and number of channels) of a sound can be changed by choosing **Characteristics...** in the **Sound** menu.

5.3 Applying effects to a sound

AmadeusLite ships with a number of sound effects covering most needs, but its capabilities can be greatly extended by the addition of third-party effects. **AmadeusLite** supports effects in Apple’s Audio Unit format which is the most widely used plug-in format on the Mac platform.

Several Audio Units are shipped by Apple with Mac OS X and should show up in the **Audio Units** submenu without having to take any action. Installing a third-party effect is as simple as dragging the corresponding file into the ‘Library/Audio/Plug-Ins/Components’ folder. A collection of third-party Audio Units can be downloaded [here](#).

In order to apply a sound effect to a portion of a sound, select this portion first and then select the desired sound effect. In order to apply an effect to the whole sound, you can either leave its selection empty or select the whole sound with the **Select All** function from the **Edit** menu.

The remainder of this subsection is devoted to the description of the content of the **Effects** menu.

5.3.1 Echo

Applies a simple echo to the sound (this effect is also called ‘**Delay**’ by some audio programs). The **gain** setting describes the intensity of the echo, it should always be a negative amount in dB or a percentage less than 100%. The **delay** setting gives the delay of the echo, which is the amount of time after which the sound is repeated. The **further** setting, when the corresponding checkbox is ticked, allows to extend the echo further than the selection. To appreciate its effect, try selecting a drumbeat and apply an echo with a delay of 0.2 seconds, a gain of 60% and the **further** setting set to 2 seconds.

5.3.2 Amplify

Allows to amplify (or decrease) the volume of the selection by a given amount. See section 2.3 for a discussion of the meaning of the dB and % units. The **Fading** function allows to make a smooth transition to the selected volume at the start and end of the selection.

5.3.3 Equalizers

The **Equalizers** submenu contains three standard multi-band **equalizers**. They work in pretty much the same way as the graphic equalizers on your hifi equipment.

5.3.4 Fadings

The **Fadings** menu contains various functions that allow to change the volume of a sound. The fading curves used by the **Fade in** and **Fade out** functions can be selected in the palette that opens when selecting the **Settings...** menu item.

The **Symmetric fade** function is equivalent to first applying a **Fade in** to a portion of the desired length at the beginning of the selection, followed by applying a **Fade out** to a portion of the same length at the end of the selection.

The **Ramp** function allows to change the volume of the sound from one level at the start of the selection to another level at the end of the selection.

The **Crossfade to the Left** and **Crossfade to the Right** functions allow you to easily create a transition between two pieces of sound. This is how it works. Let’s say you want to produce a file containing the songs ‘Song A’ and ‘Song B’. Copy them one behind the other into your file and place a marker at the junction of both songs. Then select, say 1 second of ‘Song B’ starting from the marker and select **Crossfade to the Left**. This will first create a fade in of 1 second at the beginning of ‘Song B’ and a fade out of 1 second at the end of ‘Song A’. It will then merge these two parts into one, creating a nice transition. The **Crossfade to the Right** produces the same result if you select 1 second at the end of ‘Song A’.

The **Loop Crossfade** function is similar to what was just described, except that it creates a transition between the end and the start of the selection. This allows the resulting sound to be looped seamlessly.

5.3.5 Normalizing a sound

A commonly performed task is to bring the level of a recording to some standard volume level. This can be performed through the **Normalize...** command. A problem with this is the notion of ‘level’, as there are many different ways of measuring a sound’s level. The **Normalize...** function supports two notions of sound level. The **maximal sample value** simply looks for the sample that has the largest absolute value and then amplifies the whole sound in such a way that this sample is brought to the reference level. This has the advantage that if the level is chosen to be smaller than 0dB, it is guaranteed that no clipping will occur. The disadvantage is that a short outburst of unusual loudness or a crackle in the sound will dominate the picture, so that the achieved effect is not always what was intended.

The **maximal RMS power** setting tries to compensate for this by taking average levels over relatively short time intervals (100ms) and choosing the largest of these average levels as the reference. This has the advantage of being usually closer to the desired effect. However, it does not guarantee that no clipping will occur. Normalizing the **RMS power** to around -10dB or -15dB is quite safe in most situations.

Note that the **Normalize...** function adjusts the global volume of the sound, but does not compensate for variations of volume within a recording. If you wish to obtain several songs sounding very homogeneous throughout (as required for example for background music in a store), you may want to apply dynamical compression before normalizing. One sequence of steps that works well for most situations is to first apply Apple’s **AUDynamicsProcessor** Audio Unit with the ‘hard’ factory preset and then to normalize the RMS power to 25%.

5.3.6 Reverse

This effect reverses the time on the selection, thus playing it backwards.

5.3.7 Interpolate

This function is designed to repair cracks or discontinuities in the waveform. In order to use it, select the (very short; less than 0.2 seconds) piece of sound that needs to be repaired and then choose **Interpolate**. This will replace the selection by a new piece of waveform composed from the surrounding waveform.

The following pictures show an example of a sound containing a crack before and after the **Interpolate** function is called:



5.3.8 Denoising

One common problem of analog recordings is the presence of a continuous background noise. One example is the background hiss on a recording coming from a magnetic tape. Another example is the 50Hz or 60Hz hum coming from the electric grid. (This is commonly picked up by microphones that are somehow connected to a device that runs on a power supply and not on battery.) The **Denoising** menu allows to deal with this type of background noise, but of course only up to a point.

Warning: However good a software package is, it cannot resurrect what is lost. You should therefore always try first to get the best possible quality by choosing appropriate recording equipment and only use software-based restoration functions as a matter of last resort. If a recording is important to you and cannot be repeated, try to make tests in similar conditions first.

The **Denoising** function in the **Effects** menu provides a very efficient algorithm for suppressing background noise. The easiest way to use it is the following:

- Select a piece of your sound containing *only noise*.
- Select the menu item **Sample noise**. The plugin will scan the selected noise and construct a noise profile.
- Select the whole piece of sound.
- Select the menu item **Suppress noise**. This will remove the background noise from the selected part of the sound, according to the noise profile constructed previously.

Sometimes, there is no piece of sound containing only noise available. In this case, you may want use the **Suppress White Noise** function. The easiest way to proceed is to use the **Preview** function to play the selected sound and then to adjust the **Noise RMS power** with the slider provided until the result is satisfactory. Note that choosing too large a value of the **Noise RMS power** will result in a significant loss of quality in the clean signal, whereas choosing a too small value will leave some background hiss remaining.

The **Denoising** function is customizable with the **Settings...** menu item. The **Noise Type** allows you to choose the type of noise you want to remove. Choosing a peaked noise is suited for background noise which is very well localised in frequency space. A typical example is the removal of the 50Hz or 60Hz humming produced by the power supply. The **smooth noise** function is adapted to the removal of noise that is spread over the whole frequency range, like for example a tape hiss. If the **Adaptive Filtering** option is disabled, the program will compute one global filter and apply this filter to the whole sound. This sometimes yields better results when the signal-to-noise ratio is very low. The **Sensitivity Enhancement** option tells the program the amount of background noise to remove. Usually, values between 30% and 70% yield the best results. For lower values, some background noise may remain after the application of the filter. For higher values, the signal may be substantially altered.

The **Algorithm** pop-up menu allows you to choose which **algorithm** should be used for the denoising. Currently, two algorithms are available: **Short FFT** and **Long FFT**. Generally, the **Long FFT** algorithm produces better results, but the **Short FFT** algorithm is slightly faster.

5.3.9 Audio Units

As explained above, **Audio Units** are plug-ins developed by third parties that allow to extend **AmadeusLite**'s (and any other Audio Units host's) abilities. The **Audio Units Rack** allows to apply several Audio Units

in one go. Simply drag the desired effects from the **Available Audio Units** section to the **Applied Audio Units** section of the rack window. In order to change the settings of an effect, either double-click on it or select it and click on the button with the little pen. You can remove an effect from the list of applied Audio Units by selecting it and hitting the ‘backspace’ key or clicking on the button with the minus sign.

The following list of effects ship with Mac OS X 10.4. Unfortunately, Apple doesn’t ship any documentation whatsoever for their settings so your bet is as good as anyone’s. For some of the more esoteric settings, experimenting is probably the best option.

- **AUBandpass** Suppresses all frequencies outside of a specified band.
- **AUDelay** Same as the **Echo** effect. It includes a dry/wet mix setting that allows to choose the proportion between the original sound and the echo. At 0%, it makes no modification to the sound, at 100% it keeps only the echo and removes the original sound. The default value of 50% corresponds to the behaviour of **AmadeusLite**’s **Echo** effect.
- **AUDynamicsProcessor** This allows to expand or compress the dynamical range of a sound. The dynamical range is the ratio in volume between the quiet parts and the loud parts. Compressing the dynamical range results in the volume becoming more uniform throughout the sound. When normalized afterwards, the quieter parts will therefore sound louder. This effect is often used (and abused) by record labels in order to make a CD sound louder. The different presets shipping with this effect give good results.
- **AUFilter** This is basically a combination of an AUHipass and an AULowpass filter, as well as three parametric equalizers.
- **AUGraphicEQ** Standard 10 or 30-band graphic equalizer.
- **AUHighShelfFilter** Reduces drastically frequencies below a cut-off frequency. The gain setting allows to adjust the strength of this reduction.
- **AULowShelfFilter** The opposite of AUHighShelfFilter, it reduces drastically frequencies above a cut-off frequency. The gain setting allows again to adjust the strength of this reduction. Useful to remove a low rumble in a recording.
- **AUHipass** Pretty much the same as AULowShelfFilter.
- **AULowPass** Pretty much the same as AUHighShelfFilter.
- **AUMatrixReverb** Very sophisticated reverb effect. It takes stereo effects into account to create the impression of a large enclosed space and can therefore not be applied on mono tracks. You will probably want at least to lower the dry/wet mix to 50% from its default value of 100%.
- **AUMultibandCompressor** A compressor (like the AUDynamicsProcessor effect) that however first divides the sound into four separate frequency bands, determined by the ‘crossover frequencies’. It then allows to have different settings for each of the four frequency bands.
- **AUNetSend** Allows to send the sound over a network. It doesn’t perform any processing and is therefore only useful in **Preview** mode.
- **AUParametricEQ** Allows to boost or decrease the volume in a frequency band centred around the ‘center frequency’. The ‘Q’ value determines the width of the frequency band.

- **AUPeakLimiter** Amplifies the sound by ‘pre-gain’, but makes sure that it does not get clipped. Works essentially like a compressor.
- **AUPitch** Allows to change the pitch of a sound without changing its speed. The change in pitch is given in ‘cents’. There are 100 cents in a semitone and therefore 1200 cents in one octave. The remaining parameters allow to fine-tune the effect.
- **AUSampleDelay** Simply shifts the sound forward by the specified number of samples. Probably useless in **AmadeusLite**’s context of off-line processing.

5.3.10 Change Pitch and Speed

This effect allows you to change the pitch and the speed of a sound independently. It is very useful for example to constrain a recording to a certain length. The algorithm used is Apple’s AUPitch Audio Unit, which provides excellent quality and is included with every copy of Mac OS X.

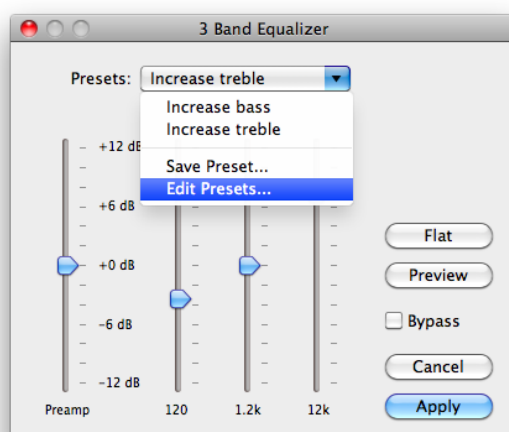
Increasing the quality results in a deterioration of processing speed and vice-versa. The adjustment of the pitch is done in semitones, *i.e.* +12 corresponds to increasing the pitch by one octave and -12 to decreasing it by one octave. If **lock** is selected, the speed increase/decrease exactly matches the pitch increase/decrease in such a way that the result is the same as what would be obtained by just playing the sound back at a different speed. In that case, the sound is simply resampled and does not go through the AUPitch effect.

5.4 Maintaining a list of commonly used actions

The first item of the **Effects** menu allows you to perform again the most recently performed action. The **Actions** tool accessible from the **Windows** menu maintains a list of the five most recently performed actions. It also allows you to save some favourite actions by simply dragging them from the **Last actions** list into the **Favorite actions** list. Favorite actions can be given any name by option-clicking on them. They can be deleted by moving them into the little trash bin in the lower right corner of the window. The **Clear last actions** button allows you to clear the list of last actions. It also clears the state of the first item of the **Effects** menu. Favorite actions are remembered from one launch of **AmadeusLite** to the next.

5.5 Working with presets

Most of the functions in **AmadeusLite** do not only remember their last settings, but allow you to maintain a list of presets which would typically be commonly used settings. For example, the window managing the 3-band equalizer is shown below.



The **Presets** menu contains the list of presets with the name of the currently used preset being displayed inside the menu button. If these don't match any of the presets or if no preset exists, **Not a Preset** is displayed instead.

When you select **Save Preset...**, a dialog allows you to choose the name of the new preset, which will then be a copy of the current preset. The **Edit Presets...** function allows to manage the list of presets. In particular, it allows you to rename, duplicate, and delete them. Some (but by no means all) functions do have a default set of presets built into **AmadeusLite**. If this is the case, the presets menu also contains a **Revert to Defaults...** item which allows you to revert to the factory defaults in case you destroyed them by inadvertance. **Warning:** Your own presets will be lost in this case.

6 Key shortcuts

First of all, note that it is possible to change the key shortcuts for menu items in the **Keyboard Shortcuts** section of the **Keyboard & Mouse** system preferences pane. This is an extremely useful feature that is often overlooked by people and works in every application on Mac OS X.

Here is a list of the key shortcuts that can be used in the main sound window of **AmadeusLite**:

key	action
space bar	Starts sound playback
tab key	Rewinds and then starts playback
option 1-4	Stores the current zoom level into the corresponding register
1-4	Sets the current zoom level the value stored in the corresponding register
e	Previews the effect of deleting the current selection
a	Extends the selection by one pixel to the left
s	Shrinks the selection by one pixel at the left
d	Shrinks the selection by one pixel at the right
f	Extends the selection by one pixel to the right
g	Toggles the display of the grid behind the waveform
h	Toggles the display of the selection in the timescale
m	Creates a marker at the insertion point
r	Interpolates the selection (select less than 0.2 seconds)
t	Interpolates the top (left) channel of the selection (select less than 0.2 seconds)
b	Interpolates the bottom (right) channel of the selection (select less than 0.2 seconds)
p	Creates a marker at the playback head
down arrow	Plays back half an octave lower
up arrow	Plays back half an octave higher
left arrow	Places the insertion point at the left end of the selection
right arrow	Places the insertion point at the right end of the selection
[Places the insertion point at the left end of the selection. If nothing is selected, moves the insertion point to the left by one pixel.
]	Places the insertion point at the right end of the selection. If nothing is selected, moves the insertion point to the right by one pixel.
option-left arrow	Moves insertion point to the previous marker
option-right arrow	Moves insertion point to the next marker
command-left arrow	Extends the selection to the previous marker
command-right arrow	Extends the selection to the next marker
home	Jumps to the beginning of the sound
end	Jumps to the end of the sound
page up	Moves to the left by one screen
page down	Moves to the right by one screen
+ or =	Zooms deeper into the sound
-	Zooms out of the sound
option- '+' or '='	Zooms vertically into the sound
option- '-'	Zooms vertically out of the sound
command-click	Places a marker at the clicked location
shift-click	Extends / shrinks the selection to the clicked location
double-click	Starts playing from the clicked location

7 Preferences

7.1 General

This pane contains the most commonly used settings of the program. The settings in the **Playback behavior** section are hopefully self-contained and those in the **Editing** section are covered in Section 5.2. Here is a

short explanation of the remaining settings:

- **Extend selection by clicking on boundary** allows to change the size of a sound selection by clicking on its boundaries.
- **Don't create new sound at startup** prevents **AmadeusLite** from opening an empty document at launch.
- **Start recording immediately** causes **AmadeusLite** to start recording as soon as **Record** is selected in the **Sound** menu, instead of waiting until the user then clicks onto the recording button in the dialog that comes up.
- **Warn before losing document attributes** causes the program to generate a warning when the user tries to save a document to a file format and the document contains data (for example some markers or other metadata) that cannot be reliably encoded in this format.
- **Use Growl notifications** causes **AmadeusLite** to generate a Growl notification (see [this link](#)) when a file saving / reading operation or a sound processing operation is completed.
- The **Length of edit preview** setting allows to set the length of the piece of sound that is being played back when pressing the 'e' key.
- The **Crossfade for cut / paste** setting allows to set the length of the crossfade used for smart edits (see also Section 5.2). If this is set to 0, then smart editing is turned off altogether.

7.2 Sound

This pane allows to choose the standard sound input / output devices used by **AmadeusLite** for recording / playback.

7.3 Colors

This pane allows to change between different color schemes and even to create your own personalised color scheme!

7.4 Temp Files

When handling large files or when performing a recording, **AmadeusLite** stores data on disk in temporary files. By default, it uses the non-networked disk with the largest available space for this. However, you may want to change this, for example in order to use a disk with a faster data transfer rate.

7.5 Units

This preferences pane allows to customize the time display in the sound window and wherever a time interval is being displayed in **AmadeusLite**. For most users, one of the presets will be suitable. More advanced users may want to tinker with the **Format** setting that allows for full customization.

To see how the time format works, let us have a look at the standard format. In this format, a duration of 2 minutes, 5 seconds and 34 milliseconds is displayed as **02'05"034**. The corresponding format reads

`[60:2]'|1:2'"|1/1000:3|`, where corresponding sections are colored identically. We see that every unit (minutes / seconds / milliseconds) is delimited by vertical bars | and consists of two numbers (the first of which can be a fraction) delimited by a semicolon. The first number determines how the unit relates to the base unit (in this example, a minutes consists of 60 seconds, while a millisecond consists of 1/1000th of a second), while the second number determines how many digits will be used to display it (in our example, 5 milliseconds display as 005, while 5 minutes display as 05).

If the base unit is 'beats', then the 'default rythm' allows to specify how many beats there are per minute.

7.6 Advanced preferences

Clicking on any of the items in the list of **Advanced Preferences** displays an explanation for it. In order to change any of the numerical values (for example the maximal RAM usage), hold down the 'Option' while clicking onto the value in question.

8 How to's

8.1 How to work with your iTunes music library

There is nothing easier than editing a song from your iTunes music library: simply drag it from iTunes onto the **AmadeusLite** icon in the dock. In order to add a new file edited with **AmadeusLite** to your iTunes library, the easiest way is to either use the dedicated **Export to iTunes Library** function from the **File** menu or to drag the file from the Finder onto the iTunes library.

Warning: Apple does not allow third parties to read protected AAC files. Songs purchased on Apple's iTunes Store can therefore **not** be edited with **AmadeusLite**, unless they were purchased from **iTunes Plus**. The easiest workaround to this situation is to burn the song to a CD with iTunes and then to open it from the CD with **AmadeusLite**.

8.2 How to convert a sound from one file format to another

A very common operation is to convert a sound from one format to another one, for example to encode a song from a CD to either Mp3 or M4a or to create FLAC files for archiving. Just open the sound into **AmadeusLite** by using the **Open** command from the **File** menu or by dragging the icon of the sound you wish to convert from the Finder onto the **AmadeusLite** icon in the dock.

Then select **Save As...** from the **File** menu and select the desired destination format in the pop-up menu located at the bottom of that dialog. You may optionally wish to tweak the settings associated to that particular format by clicking on the **Settings...** button. See Section 3 for a description of the various formats and the settings that go with them.

8.3 How to create a CD from a tape or vinyl recording

It is possible to achieve this without ever leaving **AmadeusLite** and without installing any additional software by following these steps:

- Record the tape or vinyl to a new document, as explained in Section 4.1. Make sure that the recording is performed in stereo with a sampling rate of 44.1kHz.
- Optionally remove any background hiss as explained in Section 5.3.8.
- Place markers at the junctions between songs. In most cases, this can be done automatically with the **Generate Markers...** function in the **Selection** menu. If this doesn't work (for example in a live recording), you can place them manually. Markers mark the beginning of a track, so that in most instances you would want to place a marker at the very beginning of your file.
- Choose **Burn to CD...** from the **Sound** menu. If you wish to create a CD with gapless playback (when there are smooth transitions between songs), you should choose 0 as the gap size. Otherwise, 2 seconds is a standard value. You can create a CD with CD-TEXT information which will then display the name of the currently played song and the title of the album on certain CD players. The title of a song will be given by the name of the marker which marks the start of the song. You should therefore make sure that there is a marker located at the very beginning of the document as well.

If the document contains songs by different performers, it is possible to store this information within CD-TEXT as well. You should choose either 'Performer:Title' or 'Title (Performer)' in the 'Markers' menu and name your markers accordingly. For example, if your document contains the song 'Breathe' performed by Pink Floyd, you should name the corresponding marker either 'Pink Floyd: Breathe' or 'Breathe (Pink Floyd)' depending on which naming convention you choose.

- After clicking **OK**, the standard system CD burning dialog will give you the choice of which CD burner to use and at which speed to burn the CD. Insert a blank disk and hit **OK** again. You are done.

Important: Before you can burn an **AmadeusLite** document to CD, you need to make sure that it consists of a stereo track and that its quality is 16 Bit / 44.1 kHz. If the quality is different from 16 Bit / 44.1 kHz, use the **Characteristics...** function from the **Sound** menu to adjust it.

8.4 How to record from the output of another program

Sometimes, you may wish to record the output of another program (for example **iTunes** or **Safari**). One possible way of achieving this is to connect the sound output of your computer (the headphone port) back into the line in input. This is however not very elegant and results in a slight loss of quality due to the two analog / digital conversions that take place.

Solving this problem in a more elegant way is not possible without the help of some additional software. One possible solution is to use the freeware **Jack** in conjunction with **AmadeusLite**. After downloading Jack from <http://www.jackosx.com/> and installing it, you can record sound from the output of, say iTunes, by following the steps below. This procedure has been tested with iTunes 7 and JackPilot 0.74; other versions may behave in a slightly different way.

Important: Do not use this procedure to import a sound from your iTunes library into **AmadeusLite**, unless it is a protected AAC file. For unprotected files, use the procedure described in Section 8.1.

1. Make sure that **AmadeusLite** and iTunes are both closed.
2. Open the JackPilot program.

3. Go to the JackPilot preferences and choose **Built-In Output** (or whatever your preferred output device is) in **Interface**. Also make sure that the option **Auto-Connect with physical ports** is unchecked.
4. Click on **Save** in the preferences window and then start the Jack daemon by clicking on **Start** in the **JackPilot** window.
5. Open the **Sound** panel in the system preferences and choose **Jack Router** as the default output.
6. Start iTunes and **AmadeusLite**.
7. Open the **AmadeusLite Sound** preferences and choose your preferred output device (for example **Internal Speakers**) in the **Output** panel and **Jack Router** in the **Input** panel.
8. Choose **Record to New File...** in the **Sound** menu (or option-click on the record button).
9. Bring the JackPilot application to the front and click on **Routing**.
10. Click on the little arrow next to iTunes in the **Send Ports** column in order to expand it and select the first port (labelled **out1**).
11. Click on the little arrow next to **AmadeusLite** in the **Receive Ports** column in order to expand it and double-click on the first port (labelled **in1**). It should turn red and an entry should appear in the **Connections** column.
12. Repeat steps 10 and 11 for the second port (**out2** and **in2**).
13. Start your recording in **AmadeusLite**. Enable the **Playthrough** function in the recording window in order to hear the sound you are going to play.
14. Start playing a song or a radio station in iTunes.

Note that it is also possible for example to create a virtual four-channel device with Jack. In this way, you can for example route the output of iTunes into the first two channels and your microphone into the last two channels. It is then possible to record all four channels simultaneously with **AmadeusLite** in order to create a document with a vocal and a music track.

9 Troubleshooting

The most common causes for problems concern the recording / playback functions since these are the ones where it is easiest to change the settings by mistake in such a way that the resulting behavior appears strange. This section lists a few common problems and their remedies.

9.1 I can't hear anything when playing back

1. Check that you have selected the correct output device in the **Output** tab of the **Sound** preference panel. (We are talking about **AmadeusLite**'s preference panel here, not the system one! The settings there override those in the system preferences.)
3. Reboot your computer. It happens (rarely, but still) that the audio driver gets into a strange state and refuses to work properly.

4. If you are using external hardware for playback, check that it is connected properly to your computer. Can you get iTunes to play back through it?
5. If your sound output device has more than two channels, it might be that you are not directing sound to the correct hardware channels. Change the channel association in the **Advanced Settings** dialog accessible from the **Output** pane of the **Sound** preferences panel.

9.2 I don't hear what I am recording

1. Make sure that the **Playthrough** checkbox is ticked in the recording window.
2. Follow the steps above to make sure that playback works.
3. Are the **vu-meters** moving? If not, follow the steps below to make sure that recording works.

9.3 I can't record anything or the recording is not what I expect

1. Check that you have selected the correct input device in the **Input** pane of the **Sound** preference panel. (We are still talking about **AmadeusLite**'s preference panel here, not the system one.)
2. Make sure that the **Gain** sliders are sufficiently far to the right.
3. If you are using external hardware for recording, check that it is connected properly to your computer. Do the vu-meters move if you select that device in the **Input** tab of the **Sound** system preferences panel?
4. If you are using a microphone, make sure that it either includes a preamplifier or that it is connected to a 'microphone' input. Most models of Macintosh have a 'line-in' input instead which does *not* include a preamplifier. One of the cheapest preamplifiers around is the iMic from Griffin's. (Make sure that the little black switch on it is in the correct location.)
5. If you are using the **Record to new Document...** function, make sure that you have selected the desired quality in the **Quality** tab of the recording window.
6. If you are using the **Record** function, it will create a recording with the same characteristics as the current sound. Make sure that this is what you want.
7. As above, it might be that you are not recording sound from the correct hardware channels. Change the channel association in the **Advanced Settings** dialog accessible from the **Input** pane of the **Sound** preferences panel or equivalently from the **Input** pane of the **Record to new file** window.
8. Reboot your computer, it might be that the audio drivers got into a 'bad' state.

9.4 Frequently Asked Questions

A list of frequently asked questions is maintained at the following adress:

<http://www.hairersoft.com/AmadeusLite/FAQ.html>

Furthermore, tips and discussions on using **AmadeusLite** can be found on the user's forum at

<http://www.hairersoft.com/forum/>

The remainder of this section contains the state at the time of "printing" of the FAQ:

9.4.1 What are the differences between AmadeusLite and AmadeusPro?

The main differences are:

- **AmadeusPro** supports multiple tracks.
- **AmadeusPro** includes a batch processor that allows to automate repetitive tasks.
- **AmadeusPro** includes a repair center for crack detection and repair.
- **AmadeusPro** includes a large number of high-quality sound analysis tools.
- **AmadeusPro** allows to schedule recordings.

Besides this, several functions in **AmadeusPro** have more options giving finer control.

9.4.2 Why are CD track names all wrong on CDs burned by AmadeusLite?

They only appear so in iTunes. The reason is that instead of correctly reading the CD-TEXT information contained on the CD (which is what a CD player does), iTunes tries to be smart and queries an online database (Gracenote) that tries to recognise the CD based on the lengths of the tracks. Chances are that your home-made CD will be matched with some completely unrelated commercial CD that just happens to have the same (or similar) track lengths, thus the resulting garbage. Apart from lobbying Apple to fix that bug (feature?) of iTunes, there's not much that one can do. To be fair, one possibility would be to burn the CD as a data CD instead of an Audio CD, but then most CD players wouldn't play it.

As you can guess from the explanation, this problem is not specific to **AmadeusLite** at all. It is just the same if you burn the CD with iTunes or any other other CD burning software.

9.4.3 Why does AmadeusLite refuse to create an Mp3/AAC file from my recording?

This can happen if you try to create a high-quality compressed file from low-quality uncompressed audio, which is something that you would not want to do on purpose. Just decrease the bitrate in the Mp3/AAC settings to a lower value.

9.4.4 Can AmadeusLite record the output from another program?

Not without any third-party software. The free programs SoundFlower and Jack OS X allow you to route the output of any program (like iTunes for example) into Amadeus Pro for recording. Please refer to the corresponding programs for instructions or look at Section 13.4 in the manual.

9.4.5 Could you add a key shortcut for function XYZ?

Go to the "Keyboard and Mouse" system preferences panel. On Mac OS 10.3 and later, it allows you to assign key shortcuts to arbitrary menu items.

9.4.6 The sliders for the record levels are greyed out and the input level is inappropriate, why?

These sliders directly control the hardware. Therefore, they are only active if your hardware actually does provide gain control and the driver of your hardware implements it correctly. If your particular hardware doesn't support it, there's nothing that one can do about it. You should be careful to adjust the level of the signal to the correct level before it reaches your hardware (or using the controls of the hardware if it has some). Also, make sure that you are actually using the correct sound input device. You can check whether a particular device supports gain adjustment with the "Audio MIDI Setup" application in your Applications/Utilities folder (select the device and check whether the input sliders there are greyed out).

Note that the iMic has a little black switch that allows you to switch between microphone level input and line-in level input. Make sure that this switch is in the appropriate position.

9.4.7 AmadeusLite attempts to make an internet connection at startup! What the heck?

Occasionally, Amadeus makes an http query to see whether a new version of the manual is available. What happens is exactly the same as what happens when you click on [this link](#) (i.e. it just downloads a piece of text).

9.4.8 I cannot open m4p files purchased on Apple's iTunes Store with AmadeusLite, why?

With QuickTime 6.5.1 (the update coming with iTunes 4.5), Apple decided without prior notice to forbid third-party applications the access to m4p files. One workaround is to play the files in iTunes and to use [Jack](#) to record the output from iTunes. Another workaround is to burn the file on an Audio CD and to open it from there.

10 Acknowledgment to third party software

AmadeusLite relies on a number of third-party open source software packages. Its functionality would be nowhere near what it is today without these fabulous gifts to the community. They are listed in this section, together with the corresponding internet links and licensing information.

10.1 Ogg Vorbis

The libraries VorbisLib and VorbisFileLib are licensed under a BSD-type license. Its terms are:

Copyright (c) 2002, [Xiph.org](#) Foundation

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the Xiph.org Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The libraries OggLib and VorbisEncLib are licensed under the LGPL (GNU Library General Public License). For a copy of the LGPL, see the link

<http://www.HairerSoft.com/LGPL.txt> .

All the source code used to build these libraries can be obtained at

<http://www.xiph.org/ogg/vorbis/download.html> .

10.2 LAME

LAME is the package used to provide Mp3 encoding in **AmadeusLite**. LAME is licensed under the LGPL (GNU Library General Public License). For a copy of the LGPL, go to the link

<http://www.HairerSoft.com/LGPL.txt> .

The LAME source code can be obtained at

<http://www.mp3dev.org/mp3/> .

The copy of lame used in **AmadeusLite** uses a patch allowing to add a specified number of null bytes at the beginning of a file. This is so that **AmadeusLite** can then write the metadata to the file by itself. This patch can be obtained at

<http://www.hairersoft.com/Downloads/lamepatch> .

10.3 mpg123

mpg123 is the package used to provide Mp2 and Mp3 decoding in **AmadeusLite**. mpg123 is licensed under the LGPL (GNU Library General Public License). For a copy of the LGPL, see the link

<http://www.HairerSoft.com/LGPL.txt> .

The mpg123 source code can be obtained at

<http://www.mpg123.de/> .

10.4 FLAC

FLAC is licensed under the LGPL (GNU Library General Public License). For a copy of the LGPL, see the link

<http://www.HairerSoft.com/LGPL.txt> .

The FLAC source code can be obtained at

<http://flac.sourceforge.net/download.html> .

10.5 AtomicParsley

AtomicParsley is the software used to read and write metadata from and to Mp4 files. AtomicParsley is licensed under the LGPL (GNU Library General Public License). For a copy of the LGPL, see the link

<http://www.HairerSoft.com/LGPL.txt> .

The AtomicParsley source code can be obtained at

<http://atomicparsley.sourceforge.net/> .

10.6 FFmpeg

FFmpeg is the software used to read and write WMA files. FFmpeg is licensed under the LGPL (GNU Library General Public License). For a copy of the LGPL, see the link

<http://www.HairerSoft.com/LGPL.txt> .

The FFmpeg source code can be obtained at

<http://ffmpeg.mplayerhq.hu/download.html> .

10.7 MTCoreAudio

The MTCoreAudio framework is licensed under the following terms:

MTCoreAudio.framework is Copyright (c) 2004 by Michael C. Thornburgh zenomt@armory.com. All rights reserved.

MTCoreAudio.framework and its source code may be used, modified, incorporated, et ceterated for free by anyone for any purpose. I request, but do not require, that credit for any use of the code is given. I request, but do not require, to be notified of the incorporation of this code into other works.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR

PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MTCoreAudio source code can be obtained at

<http://aldebaran.armory.com/zenomt/macosex/MTCoreAudio/> .

10.8 AMPreferencePane

AMPreferencePane is free software available from

<http://www.harmless.de/cocoa.php> .

It is distributed under the [BSD license](#).

Copyright (c) 2007, Andreas Mayer (andreas@harmless.de)

10.9 ILCrashReporter

ILCrashReporter is free software available from

<http://www.infinite-loop.dk/developer/> .

It is Copyright (c) 2004 Claus Broch, Infinite Loop. All rights reserved.

The version used by **AmadeusLite** has been modified to fix a number of minor glitches. The source code for the modified version can be found [here](#).

11 Bug reporting

Even though I make every possible effort to track down bugs before a new version of **AmadeusLite** is released, the complexity of the project makes it unavoidable that some of them slip through. If you believe that you have found a bug in **AmadeusLite**, please report it to Martin@HairerSoft.com and I will make every attempt to correct it. However, please make sure to respect the following guidelines that will make things so much easier for both of us since I will be able to provide you with a more useful reply and there is a much higher chance that I will actually be able to fix things:

- Make sure that you are using the latest version of **AmadeusLite**. It may well be that the bug has already been fixed!
- Please check the [FAQ](#) and read this manual to make sure that your bug is not a well-known problem. **Example:** Please don't report the fact that **AmadeusLite** cannot read protected AAC files!
- Describe the bug in a short but precise way and, if possible, include detailed step by step instructions on how to reproduce it. State precisely what the observed consequences of your manipulations are and what consequences you expected instead. Include as much relevant information as possible. This

usually comprises at least the version of your operating system and the characteristics of the sound you were working with when the bug occurred. **Example:** If the bug concerns the recording functions, state precisely what hardware you are using for recording.

- If the bug leads to a crash of **AmadeusLite**, there should be a dialog appearing that gives you the possibility of sending a crash log back to HairerSoft. Please do so, this will help improving future versions. Adding information on what you were doing when the crash occurred greatly helps in tracking down the problem.

11.1 Disclaimer

Just to avoid getting into any sort of legal trouble, I have to mention that I am not responsible for any damage or loss of data that may (however unlikely this is) be caused by this or any other version of **AmadeusLite** to your computer, whether this damage is caused by incorrect operation or by a bug in the program.