

 is Copyrighted, © 1993, 1996, by F. Markus Jönsson of  
✕ FMJ-Software, All rights reserved

# **A**wave v.3.2

Audio & *W*avetable *I*nstrument



Converter, *E*ditor & *P*layer

## *S*hareware

✕

1. Terms of use
2. Background and features
3. Getting started
  - Installation and configuration*
  - Loading and inspecting a file*
  - Editing and saving*
4. Advanced features
  - Menu commands*
  - The property sheets*
  - The AudioPlayer*
  - Using auditioning*
5. File formats
  - Import formats*
  - Export formats*
6. Revision history

✕

### **Written by:**

F. Markus Jönsson

mail: Svedjevägen 7, 147 52 Tumba, Sweden

www site 1: <http://hem.passagen.se/fmj/fmjsoft.html>

www site 2: <http://www.student.nada.kth.se/~f93-maj/fmjsoft.html>

email: [f93-maj@nada.kth.se](mailto:f93-maj@nada.kth.se)

phone: +46-8-530 390 81

✕

# Terms of Use

✕

## *Shareware notice:*

**T**his software is distributed as *Shareware*.

This means that you may try out the program a few times, but if you decide to use it, you must register it and pay a fee to obtain a user license. The license will cover all versions of the software, past present and future, and so is a once per lifetime expense. The enclosed registration form REGISTER.TXT contains the details on how to register.

## *Copyrights:*

**T**he program and its components are copyrighted by the author, *F. Markus Jönsson*, a.k.a. *FMJ-Software*.

The program and its components may not be modified or altered in anyway whatsoever without prior consent from the author.

All trademarks, servicemarks and other registered names, mentioned in this, and accompanying texts are the property of their respective owners.

## *Distribution notice:*

**T**his product may never be bundled, sold or otherwise distributed with, or as a companion to, a commercial product expect by prior consent from the author. Unauthorized reproduction or distribution of this program, or any portion of it, may result in sever civil and criminal penalties. *Relaxations:* You (an individual or an entity) are hereby authorized to freely keep, distribute and multiply copies of the *unregistered distribution archive* for this software with, through and on, BBSs, Internet-sites and equivalent information and on-line services. You are also allowed to store it on personal media like hard disks, floppy disks and networks, for personal use only. You are further allowed to include it on so called '*Shareware collection*' CD-ROMs and equivalents. Note that these relaxations do not affect the actual use of the software whatsoever. Users of such products will still have to pay the registration fee to gain the legal right to use the software as defined in the Shareware statement.

The following files must always be supplied:

FILE_ID.DIZ	<i>Short info file.</i>
AWAVE.EXE	<i>Program executable.</i>
AWAVE.HLP	<i>Program documentation in WinHelp hypertext format.</i>
AW2WF16.EXE	<i>Awave to TB WaveFront 16bit drivers middle layer.</i>
AWAVEFAQ.TXT	<i>Frequently Asked Questions and Troubleshooting text.</i>
README.TXT	<i>Installation instructions.</i>
REGISTER.TXT	<i>Registration form.</i>

## *Disclaimer of warranty:*

**FMJ**-Software exclude any and all implied warranties, including warranties of merchantability and fitness for a particular purpose.

Nor does *FMJ-Software* make any warranty of representation, either express or implied, with respect to this product, its quality, performance, functionality or fitness for a particular purpose.

Nor shall *FMJ-Software* have any liability for special, incidental, or consequential damages arising out of or resulting from any use of this product.

Nor shall *FMJ-Software* have any responsibilities, nor any obligations whatsoever to provide support for this product.

Nor shall *FMJ-Software* have any responsibilities whatsoever to provide upgrades or bugfixes of any kind.

x

## Background and Features

x

### *Background:*

**A***wave* first saw the light of day some three years ago when I needed to extract some samples from an Amiga tracker module and save them as UltraSound instruments. I hacked up a small program called MOD2PAT to fill my needs. As some programs tend to do, this one soon grew beyond bounds. More import formats were continually added (and still are). It was ported to Windows, it got a steadily improving user interface. The name was now something of an understatement so I changed into 2PAT. Now after several major rewrites, it is supporting more than a hundred audio carrying file formats for import, as well as most popular formats for export. And sporting many editing, playback and auditioning functions too boost! And the name was again somewhat inappropriate so it was changed into *Awave*.

### *Features:*

**A***wave* reads a veritable host of audio carrying file formats from different platforms, synthesizers and trackers. It presents the instruments contained in a file as a graphical tree with the bank item at the top, instruments as branches and waveforms as leaves. If you have one of the supported wavetable synthesis sound cards you can just click on an item, and directly 'audition' it using a midi keyboard or the computer keyboard. If you just have a basic soundcard capable of audio playback, you can use it as a crude synthesizer to get an idea of what it should sound like. Saving an item to any of the supported export formats, is as easy as double clicking on the item and selecting export type. Sending and receiving MIDI SDS dumps are also supported for transferring waveforms with professional synthesizers. There are visual editors for most instrument and waveform parameters like loop points and keymaps. Some audio processing and editing functions are also provided. There is also the 'AudioPlayer' feature, a nice audio playback control, sporting both real-time oscilloscope and frequency analyzer graphs. It is available from inside *Awave* as well as integrated into the Windows95 shell in order to easily let you play the sounds of any of the supported file formats! Thus it is also an ideal 'helper app' for your WWW-browser! If you are into any form of sampled audio on the Windows platform, you'll most likely find this program useful!

x

# Getting Started

✕

## *Installation and configuration:*

**P**lease start with reading the 'README.TXT'. If you have trouble with something, you may later want to browse through the 'AWAVEFAQ.TXT' file. If you ever wonder about what a control does, select it and then press F1 to get context sensitive help. Very well, lets get you started!

Choose **Program...** from the **Options** menu.

Select the **AudioPlayer** tab. Under **Primary dev.**, select the primary digital audio device to be used with the **Audio Player** (whenever this primary device is already busy playing something else, **Awave** will automatically search for, and use, any secondary device(s) you may have installed).

If (and only if) you have problems with no sound in the AudioPlayer then select 'Use Windows multimedia services, unbuffered output' else leave it at 'Use Windows multimedia services, buffered output'.

If you have Microsoft DirectX installed on your computer then its is recommended that you select 'Use Microsoft DirectSound services'..

See **Auditioning** for details on setting up and using the auditioning feature in the **Auditioning** tab. Click **Ok**.

## *Loading and inspecting a file:*

**P**ress the right mouse button, select **Open new file** and select a file using the dialog box. **Awave** will try to recognize the correct **import format** of a file even though it may have no or even an incorrect extension. If you want you may select several files and open them all at once.

A tree-view list with a Bank->Instrument->Waveform hierarchy will be constructed from the file loaded and shown. A special 'folder' with a list of all waveforms contained in the loaded file may sometimes also be appended at the bottom of the list. The bank item is represented by the topmost AW icon, the instruments by a keyboard icons, and waveforms by some green things.

Select an instrument or a waveform from the list by clicking on it with the mouse.

You may now be able to play it on your midi keyboard (or computer keyboard if you have enabled the virtual keyboard feature), provided that you have setup auditioning properly. You can quickly turn on and off the auditioning feature by toggling the **Auditioning** checkmark on the **Options** menu. See **Auditioning** for details.

Or, you can press the right mouse button and choose the **Play audio...** menu item to play the sound using the Audio Player with a nice oscilloscope view..

## *Editing and saving:*

**T**o edit the name of an item, do a 'slow double click' on the name or press F2 and edit the name as in a standard text edit control (same as when editing filenames in the Windows 95 shell). Note also that a 'fast double click' executes the default action of the right mouse button menu.

For all other parameters, select **Properties** in the right mouse button menu to bring up the currently selected items property sheet. Try this for all the various item types (banks, instruments, waveforms)

and carefully explore all the tabs! You will find, among other things, a visual loop point editor, a waveform editor, an instrument keymap editor and an audio processor wizard. Note! When auditioning is enabled, click the **Apply** button to download (and hear!) the changes.

**T**o save an item, press the right mouse button and select *Save X as...*, or select the corresponding command from the File menu. *X* will reflect the currently selected item type (if nothing is selected it will default to bank). Ex: To save an instrument, select the instrument and then select what now reads *Save instrument as...*

A standard dialog box will appear and prompt you for the file name to save to. Choose your export format in the **File format:** box. In the **Subformat:** box you can sometimes (depending on the file format selected) select what data format to save it in. Mark the **Mono** and **Stereo** checkboxes to indicate what channel format you want to save it in. If both boxes are checked then it will be saved in the channel format of the original waveform i.e. leave as it is. When saving instruments and banks, these selections apply to all waveforms in the instrument and/or bank. Enter a filename (no extension needed) and press OK.

Note! Many of the export formats only support saving waveforms and not Banks or Instruments.

Note! When converting instruments, **Awave** does not convert envelopes, modulators, or layers. Part of the explanation for this omission is simply incompatibilities between the architectures of the different synth chips. This means that to achieve the highest possible quality instruments for your wavetable synth, you'll probably want to use an instrument editor to manually improve some of the output from **Awave** by slightly editing amplitude envelopes and other synthesizer specific parameters. These things are set to reasonable default values when you save an instrument file.

*Now you should have a general idea of how **Awave** works...  
You may now want to continue with the Advanced Features section.*

✘

### ***Instrument Editors***

For the UltraSound there is the Patch Maker software.

For the Turtle Beach products there is the WavePatch software.

For the SB AWE32 there is the Vienna SF Studio software.

These softwares should be bundled with the latest revisions of the respective soundcards. New software versions are continuously made available from the respective manufacturers.

# Advanced Features

✕  
*The menus:*

## **Left mouse button:**

*Single click* Selects an item,  
*Double-click* Execute default action on item,  
*Slow double-click* Edit name field of item.

## **Right mouse button:**

*Save X as...* Saves the currently selected item,  
*Open file...* Clears the instrument list and prompts you for a new file,  
*Play audio...* Launches the integrated *Awave* Audio Player with the selected item,  
*Split into mono channels* Splits a stereo waveform into two mono waveforms,  
*Properties...* Will pop up a property sheet information and editors for the selected item,  
*Help* Brings up the contents page of this file.

## **Keyboard:**

*Delete* Deletes selected item,  
*F1* Context help.  
*F2* Edit name of selected item.  
*F5* Same as *Save as...*  
*F6* Same as *Open file...*  
*F7* Same as *Play audio...*  
*F8* Same as *Properties...*

## **File menu:**

*Open new...* Clears the instrument list and prompts you for a new file,  
*Save X as...* Saves the current item (of type X),  
*Merge with another file...* Inserts or merges the contents of another file with the current bank,  
*Tip!* An alternative way to merge files is to simply drag them from the Explorer.  
*Save all in multiple files...* Saves all instruments or waveforms depending on output format,  
*Add new instrument...* Creates a new, empty, instrument in current bank,  
*Autocreate instruments* Creates instruments for all 'lonely' waveforms,  
*Exit* Closes the program.

## **Edit menu:**

*Properties...* Will pop up a property sheet information and editors,  
*Play audio* Launches the integrated *Awave* audio player,  
*Process all->Resample...* Resample all waveforms to one common sample rate,  
*Process all->Conv.loops...* Convert all waveform loops to a common format,  
*Copy to clipboard* Exports a waveform to the Windows clipboard,  
*Paste from clipboard* Imports a waveform from the Windows clipboard,  
*Midi SDS transmit* Opens the Midi SDS transmit dialog,  
*Midi SDS receive* Opens the Midi SDS receive dialog,  
*Delete* Deletes it from memory,

## **Options menu:**

*Program...* Is the program property sheet.



*Auditioning*                      A quick switch for toggling auditioning on/off.

**Help menu:**

...                                      Brings up various help sections.

*About Awave*                      Brings up some program info.

*The property sheets:*

Almost all of the editing power of *Awave* is hidden in property sheet dialogs, easily accessible through the right mouse button.

**Waveform properties:**

*General tab*,                      Info, sample rate, rootkey, volume, pan,

*Loop tab*,                              Visual loop point editor, loop type,

*Editor tab*,                              Waveform editor and display (power or current selectable),

**Audio processing wizard:**

"Convert bi-directional to forward loop",

"Convert reverse to forward loop",

"Convert to mono channel",

"Convert to stereo channels"

"Cut away selection",

"Fade selection",

"Maximize amplitude",

"Remove silent sections",

"Resample waveform",

"Reverse selection",

"Set loop points to selection",

"Smooth out the selection",

"Statistics",

"Trim waveform to selection"

*Advanced tab*,                      Source data type, file position and size.

**Instrument properties:**

*General tab*,                      Info, volume, program no, comment,

*Keymap tab*,                              Waveform to keyrange assignment,

**Bank properties:**

*General tab*,                      Info, bank no,

*Programs tab*,                              Program no assignment to instruments,

*The AudioPlayer:*

If you start *Awave* with **-AP** as the first option on the command line and follows this with one or more filenames and/or directory names, the Audio Player mode will be entered. You can also launch the Audio Player at any time from the normal main window of the program with the **Play audio** command..

In this mode *Awave* acts as a digital audio player that can play all the import formats supported by *Awave*.

Files will be consecutively loaded and all the sounds contained within them played before the program terminates. When a directory name is specified, all the recognized audio files in that directory will be played. You must press the >>Next>> button to advance from one waveform to the next, closing the Audio Player after the last one (to immediately close it, skipping any remaining waveforms, click the standard close window button [X]). The play/stop button should be obvious. The *off, wf, sp* radio buttons determines the operational mode for the scope (off/hold, waveform oscilloscope, or frequency spectrum analyzer).

The following command line switches are allowed, and must precede the file names.:

- AP Specifies that the AudioPlayer mode shall be entered.
- ONCE Automatically starts playing the next item when the current one is done; closes the AudioPlayer after the last one.
- DELETE Delete the input file(s) from disk after the AudioPlayer is closed.
- NOWIN Runs the AudioPlayer and plays the sound without displaying any window.
- LOOP Repeatedly plays the indicated files and/or directories. Break the loop by pressing the [X] button. Warning if you combine the -NOWIN switch with the -LOOP switch then *Ctrl-Alt-Delete* is the only way to break the loop.

With the Audio Player option, you can use *Awave* for example as a helper app to your favorite Web-browser!

Example: AWAVE -AP QUADRA.AU POPCORN.MOD

Example: AWAVE -AP -ONCE -NOWIN -DELETE filename

Example: start AWAVE -AP -ONCE -LOOP .

The last example repeatedly plays all files in the current directory (the current directory is, as usual, denoted by a single dot (.)). The start command can be used in Windows 95 to start a registered program without having to have it in the path or having to type whole the path explicitly.

### Using auditioning:

If you have a:

- Gravis Ultrasound, or a
- Turtle Beach Maui or Tropez), or a
- Creative Labs SoundBlaster AWE32, or
- Any digital audio card (using the *Awave* software synth)

you can use the auditioning feature of *Awave* to directly hear and play instruments and waveforms by just selecting them in the main treeview list.

To enable auditioning, go to the Options->Program->Auditioning tab. Just click the Enable checkbox and choose appropriate MIDI-Input and WaveSynth devices. If you don't have a midi-in device select the <none> midi input device. If you check the **Enable virtual keyboard** box, you can use the computer keyboard as a midi keyboard as well. Click **OK** and everything should be go.

The virtual keyboard has the following layout:

		S		D			G		H		J			2		3			5		6		7			9		0							
	Z		X		C		V		B		N		M		Q		W		E		R		T		Y		U		I		O		P		
	C-1													C 0																					
	shift/control												keys	-																					inc./dec. attack & release velocity

numeric plus/minus keys      shift keyb. range 1 octave up/down.

If you have a non-standard, or foreign country keyboard layout, there is a way to [edit the key layout](#).

x

### *The Upload bank command:*

The Upload bank command, found under the file menu is used to upload all of the instruments in the bank to the wavesynth. It is only available when auditioning is enabled (except when using the Awave AudioPlayer synth, then it is never available) As it is still in the 'experimental stage', I can not guarantee that it will work as intended. The idea is that, using this command, you'd be able to develop 'cross-soundcard' compatible instrument bank files. You'd then be able to distribute them with your midi songs and use Awave to upload them to any one of the supported wavetable sound cards. If there is enough interest, I might be able to produce freeware, standalone, 'bank loaders' for the GUS, AWE32, Maui/Tropez respectively.

Things to consider when authoring cross-soundcard compatible banks:

- After uploading the bank, selecting a waveform or an instrument in Awave will destroy the uploaded bank (it will be replaced with the new instrument/waveform) so don't touch Awave until you're done testing the bank using some other program.
- For the GUS, upload the bank *after* you have loaded your midi file in a sequencer.
- For the AWE32, upload the bank *before* you start your sequencer.
- Do not use multilayered instruments (i.e. no more than 1 waveform per midi key, you can't do that in Awave anyway...),
- Do not use stereo waveforms,
- Do not use reverse or bi-directional looping,
- The drum channel, ch10, is always supposed to be the standard GM drum bank,
- Do not expect perfect reproduction of envelopes and other modulation parameters,
- Do not use more than 512KB or 1MB of waveforms and no individual waveform > 256KB,
- Use only one 'bank' and that bank must be bank no 120,
- Always insert a GS bank change to bank no 120 on all midi channels (CC0 120),
- Do not use any midi program not defined in the bank,
- Remove all unused waveforms from the bank,
- Use one of the following file formats when saving the bank: .SBK, .SF2, .WFB, .INI+.PAT's

x

If you want to edit the key assignment for the virtual keyboard you can do that with the registry editor (regedit.exe) that comes with *Windows 95*. Open this \_computer->HKEY\_CURRENT\_USER->Software->FMJ-Software->Awave and edit the mtVKeys string. It's just a string of (normally 29, but maximum 36) uppercase characters holding the ascii key to assign to a note, beginning with C3 for the first character and then in increasing order...

*Awave* can use any audio card capable of playing digital audio to emulate a very simple wavetable synth in software. It should be enough to give a rough idea of an instruments or waveform. It supports the following midi messages: note on, note off, pitch bend, all notes off. Also, it is normally monophonic. However if you have selected Use Microsoft DirectSound services in the AudioPlayer property tab then it will be 8 voice polyphonic!

*Awave* supports downloading instruments to the Advanced Gravis UltraSound Classic/Max/Ace soundcards.

Remember that, that a GUS with full memory can handle a maximum of 1M of 8bit samples.

Note! It does *not* support downloading instruments to the new GUS PnP card, use the software synth instead.

*Awave* supports downloading instruments to TB WaveFront synthesizers; Maui, Tropez, ...

You must have installed the WFGATE.DLL file into your Windows directory in order to use auditioning with the WaveFront synth. This file normally is installed with TBs WavePatch software but you may also find this file on my [www](#) site.

Note! Any custom instruments that you may previously have loaded into wavetable RAM will be overwritten. You may also have to reset the synth to restore the GM setup afterwards.



*Awave* supports downloading instruments to the SB AWE32 family of soundcards.

Note! You will need the very latest drivers with 32bit application as well as .SF2 bank file support in order to be able to use auditioning with the AWE32 (these drivers were only available in beta version when this was written). Also make sure that the AWEMAN32.DLL file is correctly installed in your windows directory.

Note! If you run out of memory on the card when trying to load an instrument, try removing any other user banks you may have previously loaded to card memory (by using the standard AWE32 control panel).



# File Formats

*import formats*

*export formats*

✕

When opening files, *Awave* tries to determine the file type, not only by looking at the filename extension, but also by looking inside the file. Thus, even if it has an incorrect or no extension, it can, for file types marked with *\*\*\**, be correctly identified. For formats that are not possible to detect in this way you will have to pay attention to using correct file extensions. If a file is not of a recognized format, it is treated as raw audio data and the waveform properties Advanced tab pops up querying the user for additional info.

*Awave recognizes the following import file formats:*

- 669** 669 tracker module. *\*\*\**
- AIFC** Compressed Audio Interchange File Format.  
*\*\*\**
- AIFF** Audio Interchange File Format.  
*\*\*\**
- AIS** Velvet Studio Instruments.  
*\*\*\**
- ALAW** European telephony format audio.
- AMS** Extreme's Tracker module format.  
*\*\*\**
- AMS** Velvet Studio Module.  
*\*\*\**
- ASE** Velvet Studio Sample.  
*\*\*\**
- AU** Sun/NeXT/DEC Audio file.  
*\*\*\**
- AVI** MS Audio Video Interleave file.  
*\*\*\**
- AVR** Audio Visual Research sound file.  
*\*\*\**
- C01** Typhoon wave files.  
*\*\*\**
- CDR** Raw Audio-CD data.
- DCM** DCM module format.  
*\*\*\**
- DEWF** Macintosh SoundCap/SoundEdit recorded instrument format
- DIG** Digilink format  
*\*\*\**
- DIG** Sound Designer I audio file.
- DMF** Delusion/XTracker Digital Music Fileformat.  
*\*\*\**
- DSF** Delusion/XTracker Digital Sample Fileformat.  
*\*\*\**
- DSM** Digital Sound Module tracker format.  
*\*\*\**
- DTM** DigiTrekker module.  
*\*\*\**
- DWD** DiamondWare Digitized file.  
*\*\*\**
- EDA** Ensoniq ASR disk image.  
*\*\*\**
- EDE** Ensoniq EPS disk image.  
*\*\*\**
- EDK** Ensoniq KT disk image.  
*\*\*\**

**EDQ** Ensoniq SQ1/SQ2/KS32 disk image.  
\*\*\*\*

**EDS** Ensoniq SQ80 disk image.  
\*\*\*\*

**EDT** Ensoniq TS disk image.  
\*\*\*\*

**EDV** Ensoniq VFX-SD disk image.  
\*\*\*\*

**EFA** Ensoniq ASR file.  
\*\*\*\*

**EFE** Ensoniq EPS file.  
\*\*\*\*

**EFK** Ensoniq KT file.  
\*\*\*\*

**EFQ** Ensoniq SQ1/SQ2/KS32 file.  
\*\*\*\*

**EFS** Ensoniq SQ80 file.  
\*\*\*\*

**EFT** Ensoniq TS file.  
\*\*\*\*

**EFV** Ensoniq VFX-SD file.  
\*\*\*\*

**EMD** ABT Extended MoDule.  
\*\*\*\*

**ESPS** ESPS audio files.  
\*\*\*\*

**EUI** Ensoniq EPS family compacted disk image.  
\*\*\*\*

**F2R** Farandoye linear module format.  
\*\*\*\*

**F3R** Farandoye blocked linear module format.  
\*\*\*\*

**FAR** Farandoye tracker module.  
\*\*\*\*

**FFF** GUS PnP bank file format.  
\*\*\*\*

**FNK** FunkTracker module format.  
\*\*\*\*

**FSM** Farandoye Sample format.  
\*\*\*\*

**G721** Raw CCITT G.721 ADPCM format data.  
\*\*\*\*

**GKH** Ensoniq EPS family disk image file.  
\*\*\*\*

**GSM** Raw GSM 6.10 audio stream.  
\*\*\*\*

**GSM** US Robotics voice modems / QuickLink .GSM format.  
\*\*\*\*

**HCOM** Sound Tools HCOM format.  
\*\*\*\*

**IFF** Interchange file format.  
\*\*\*\*

**INI** MWave DSP synths mwsynth.ini GM-setup file.  
\*\*\*\*

**INI** Gravis UltraSound bank setup file.  
\*\*\*\*

**INRS** INRS-Telecommunications audio file.  
\*\*\*\*

**INS** Ensoniq EPS family instrument file.  
\*\*\*\*

**IST** Digitrakker Instrument File.  
\*\*\*\*

**IT** Impulse Tracker module/instrument/sample.  
\*\*\*\*

**KRZ** Kurzweil 2000 sample file (also split files: KR1, KR2).  
\*\*\*\*

**MAT** Matlab variables binary file.  
\*\*\*\*

**MAUD** MAUD sample format.

**MDL** Digitrakker Module File.

**MED** OctaMED tracker module.

**MOD** Amiga and PC tracker module.

**MOV** Apple QuickTime audio.

**MP2** MPEG audio stream, layer I, II or III

**MPA** MPEG audio stream, layer I, II or III

**MTM** MultiTracker Module.

**MUS10** Mus10 audio file.

**NIST** NIST SPHERE file.

**O01** Typhoon vOice file.

**OKT** Oktalyzer tracker module.

**PAC** SB Studio II package file.

**PAT** Advanced Gravis Ultrasound / Forte tech. Patch file.

**PCM** OKI MSM6376 synth chip PCM format.

**PLM** DisorderTracker2 module.

**PLS** DisorderTracker2 sample.

**PRG** WAVmaker program file.

**PSION** PSION alaw file.

**PSM** Protracker Studio Module Format.

**PTM** Poly Tracker module.

**RAW** Raw signed PCM data.

**SAM** Signed 8bit Sample data.

**SB** Raw Signed Byte (8bit) data.

**SBK** Emu SoundFont v1.x Bank files / Creative Labs SB AWE 32.

**SD** Sound Designer I audio file.

**SD2** Sound Designer II flattened file.

**SD2** Sound Designer II resource fork file.

**SDK** Roland S-series floppy disk image.

**SDS** Raw Midi Sample Dump Standard file.

**SDW** Raw Signed DWord (32bit) data.

**SDX** Midi Sample Dump Standard files as compacted by SDX.

**SF** IRCAM SoundFile format.

**SF2** Emu SoundFont v2.0 file.

**SFD** SoundStage Sound File Data file.

**SFI** SoundStage Sound File Info file.

**SFR** Sonic Foundry Sample Resource file.

**SMP** Samplevision file.

**SMP** Ad Lib Gold Sample file.

**SND** Raw unsigned PCM data.

**SND** AKAI MPC3000 sample file.

**SNDR** Sounder sound file.

**SNDT** Sndtool sound file.

**SOU** SB Studio II sound file.

**SPD** Speach Data file.

**SPL** Digitrakker Sample File.

**SPPACK** SPPack sound sample file.

**STM** Scream tracker v2 module.

**S3I** Scream tracker v3 instrument.

**S3M** Scream tracker v3 module.

**SVX** Interchange file format, 8SVX/16SV.

**SW** Raw Signed Word (16bit) data.

**SYW** Yamaha SY-series wave files (really named W??).

**TXT** Ascii Text formatted audio data.

**TXW** Yamaha TX16W wave files (really named .W??).

**UB** Raw Unsigned Byte (8bit) data.

**UDW** Raw Unsigned DWord (32bit) data.

**ULAW** US telephony format (CCITT G.711) audio.

**ULT** UltraTracker modules.

**UNI** MikMod 'UniMod' format.

**UW** Raw Unsigned Word (16bit) data.

**UWF** UltraTracker Wave File.

**V8** Covox 8bit file.

**VAP** Annotated speech file.

**VOC** Creative labs sound file.

**VOX** Dialogic adpcm file

**W??** Yamaha waveform file.

**WAV** Ms Windows PCM wave file.

**WFB** Turtle Beach WaveFront Bank file (Maui/Rio/Monterey).

**WFD** Turtle Beach WaveFront Drum set file (Maui/Rio/Monterey).

**WFP** Turtle Beach WaveFront Program file (Maui/Rio/Monterey).

**XM** Fast Tracker 2 extended module.

**XI** Fast Tracker 2 instrument file.

\*\*\*  
\*\_  
\*\*\*)

Unrecognized formats prompts user to supply basic info. (

✘

# File Formats

*import formats*

*export formats*
































When saving files, select the desired File Type in the appropriate list in the Save As dialog, or just type in a file name with an extension matching one of the supported export types. Note that most of the formats can only handle a subset of the item types as denoted by the symbols below.

*Save can export to the following formats:*

- AIF** Audio Interchange File Format (✱).
- ALAW** Raw A-law data (✱).
- AU** Sun Audio format, linear,  $\mu$ -law or A-law (✱).
- DES** Ascii text parameter description (✱).
- DWD** DiamondWare Digitized file. (✱).
- EFE** Ensoniq EPS family instrument file (✱).
- FFF** GUS PnP bank file format (✱).
- G721** Raw CCITT G.721 ADPCM format data. (✱).
- GSM** Raw GSM 6.10 audio stream (✱).
- GSM** US Robotics voice modems / QuickLink .GSM format (✱).
- IFF** Interchange file format (✱).
- INI** Gravis UltraSound bank setup extract plus patch files (✱).
- MAT** Matlab variables binary file (✱).
- MWS** MWave DSP synths instrument extract (✱).
- PAT** Advanced Gravis UltraSound / Forte tech. patch file (✱).
- PRG** WAVmaker program file (✱).
- S3I** Scream tracker v3 instrument (✱).



).  
**SB** Raw Signed PCM Byte (8bit) data (  
).  
**SBK** Emu SoundFont Bank v1.x file / SB AWE 32 (  
  
,  
).  
**SD** Sound Designer I audio file (  
).  
**SD2** Sound Designer II flattened file (  
).  
**SD2** Sound Designer II resource fork file (  
).  
**SDS** Midi Sample dump standard file (  
).  
**SF2** Emu SoundFont Bank v2.0 file (  
  
,  
).  
**SND** AKAI MPC3000 sample file (  
).  
**SW** Raw Signed PCM Word (16bit) data (  
).  
**TXT** Ascii Text formatted audio data (  
).  
**UB** Raw Unsigned PCM Byte (8bit) data (  
).  
**ULAW** Raw  $\mu$ -law (CCITT G.711) data (  
).  
**UW** Raw Unsigned PCM Word (16bit) data (  
).  
**VOC** Creative labs sound file (  
).  
**VOX** Dialogic adpcm file (  
).  
**W01** Yamaha TX16W wave file (  
).  
**W01** Yamaha SY-series wave files (  
).  
**WAV** Ms Windows PCM wave file, linear,  $\mu$ -law, A-law or GSM (  
).  
**WFB** Turtle Beach WavePatch Bank file (  
).  
**WFP** Turtle Beach WavePatch Program file (  
,  
).  
**XI** Fast Tracker 2 instrument file (  
,  
).

 indicates that the format supports saving waveforms.

📄 indicates that the format supports saving instruments.

🌟 indicates that the format supports saving banks.

✖

**\*: *Unrecognized file types***

If a file can't be determined to be one of the, by this program, supported file types, then the waveform properties box will appear. Select the Advanced tab and supply information on the file's data type.

Note! if you get it wrong, you can always use the waveform properties dialog again and try some other settings.

***Features:***

8 / 16 bits /  $\mu$ -Law / A-Law, mono / stereo.

**.669:**    *669 tracker module (PC)*

*Features:*

8 bits, mono, loop, name, bank.

***.AIF/.AIFF/.AIFC/.C01:      Audio Interchange File Format (Amiga, Atari, SGI,Mac)***

AIFC is an extension of the original AIFF format also supporting compressed sample data. The .C01 files are used by the Typhoon OS for the Yamaha TX16W.

***Features:***

1..32bits / IMA / DWVW compression /  $\mu$ -law encoding, loop, keyrange, mono / stereo, name.

***Unsupported features:***

Apple MACE and ACE compressions can not be supported for patent reasons.

***Internet source of samples:***

*<ftp://ftp.reed.edu/eps/samples/aiff>.*

*.ALAW: European Telephone format (CCITT G.711) audio*

*Features:*

A-Law, mono, 8kHz.

*.AMS: Extreme's Tracker format (PC)*

*Features:*

mono, 8,16 bits, name, bank.

*Unsupported features:*

Compression.

*.AMS/.AIS/.ASE: Velvet Studio Module/Instrument/Sample (PC)*

*Features:*

8,16 bits, mono, name, loop, instrument, bank.



**.AU:**     *Audio file (Sun/NeXT/DEC)*

*Features:*

8,16,24,32 bits /  $\mu$ -law / A-law / CCITT G.721 (4bits) & G.723 (3,5bits) ADPCM / IEEE FP (32, 64bits) FP encodings and compressions, mono / stereo.

*Unsupported features:*

> 2 channel files.  
Other compression schemes.

*Internet source of samples:*

UNIX places, do an ARCHIE search for .au.

***.AVI: MS Audio Video Interleave file (PC)***

***Features:***

Same as for .WAV files

***Unsupported features:***

Of course, only the digital audio streams are read, not video or such...

***.AVR: Audio Visual Research sound file (Macintosh)***

***Features:***

8,16bit, mono / stereo, name, loop.

***.CDR: Raw Audio-CD format data file***

This is raw CD-*format* audio, and not raw CD-*data* which also contains a lot of error correction and formatting information, i.e. you can not read directly from an audio CD but if you have made a raw sound data dump with another program, you can give it an .CDR extension and Awave will read it properly.

***Features:***

16 bits, stereo.

*.DCM: DCM module format*

*Features:*

8,16bits, mono, loop, bank.

*.DEWF: SoundCap / SoundEdit recorded instrument format (Macintosh)*

*Features:*

8bit, mono, loop.

***.DES: Ascii text parameter Description file***

Contains human readable information about items at bank, instrument or waveform levels. Useful if *Awave* can't export directly to your destination format and you'll have to piece together instruments from, say, .WAV files.

***Features:***

Human readable information, no sample data.

***.DIG: Digilink format (Broadcasting)***

*Features:*

16bit, stereo.



***.DMF/.DSF: Delusion/XTracker Digital Music/Sample Fileformat (PC)***

***Features:***

8,16bits, mono, loop, name, instruments, bank.

***Unsupported features:***

Packed samples.

*.DSM: Digital Sound Module tracker format*

*Features:*

8 bits, mono, name, loop, bank.

*.DTM: DigiTrekker module*

*Features:*

8,16bits, mono, loop, name, bank.

***.DWD: DiamondWare Digitized file***

***Features:***

8/16bits, mono/stereo.

## ***.EDx,GKH,EUI:        Ensoniq disk image (MIDI)***

The Ensoniq files are usually encountered in form of .GKH disk image files (with an additional header) as well as in the slightly compacted .EUI disk image format. You may also find the .EDx files, x = A,E,K,Q,S,T,V.

*Awave* can only read contiguous subfiles in Ensoniq disk images. If *Awave* can't handle a certain file (because it's fragmented on the disk), you can extract the individual instrument files to .INS or .EFE files with a utility called EPSDISK (not included with *Awave*). INS files are the same as an EFE files except the latter may also contain EPS sequenced songs, bank info or such, instead of an instrument. Multilayered instruments will have each layer shown as a separate instrument (all have the same name though).

### ***Features:***

16 bits, mono, loop, names, finetune, instruments, bank.

### ***Unsupported features:***

A bunch of parameters et c.

### ***Internet source of samples:***

*ftp://oak.oakland.edu/pub2/eps/samples*  
*ftp://wozzeck.tfo.arizona.edu/pub/sampling/samples/eps*  
*http://www.ensoniq.com/*

***.EFx,INS:    Ensoniq instrument file (MIDI)***

INS files are the same as an EFX files except the latter may also contain EPS sequenced songs, bank info or such, instead of an instrument.

Multilayered instruments will have each layer shown as a separate instrument (all have the same name though).

***Features:***

16 bits, mono, loop, names, finetune, instrument.

***Unsupported features:***

A bunch of parameters et c.

***Internet source of samples:***

*<ftp://oak.oakland.edu/pub2/eps/samples>*

*<ftp://wozzeck.tfo.arizona.edu/pub/sampling/samples/eps>*

*<http://www.ensoniq.com/>*

*.EMD: ABT Extended MoDule format*

*Features:*

16bit, mono, loop, name, bank.

***.ESPS: ESPS audio file***

***Features:***

8,16,32bit int / 32,64bit FP, mono.



*.FAR,.FSM,.F2R,.F3R: Farandoyle tracker format (PC)*

*Features:*

8,16 bits, mono, loop, name, bank.

***.FFF+.DAT: GUS PnP bank file (PC)***

*Note!* The .FFF file contains all instrument parameters and stuff while the accompanying .DAT file contains the actual sample data which is to be downloaded to the synths memory.

***Features:***

8/16/ $\mu$ -law bit, mono, loop, instruments, bank.

*.FNK: FunkTracker module format (PC)*

*Features:*

8 bit, mono, loop, name, panning, bank.

***.G721: Raw CCITT G.721 ADPCM format data***

Note! The first sample is assumed to be in the high-nibble, the second in the low-nibble.

***Features:***

ADPCM 4bit lossy compression, mono.

***.GSM: GSM 6.10 audio stream***

Note! These files cant be autodetected so be sure to use the .GSM extension. The data is assumed to be in 33Bytes/frame packets where the first nibble (4bits) is ignored.

***Features:***

GSM compression, mono.

*.GSM: US Robotics voice modems / QuickLink .GSM format*

*Features:*

GSM compression, mono.

*.HCOM: Sound Tools HCOM format (Macintosh)*

*Features:*

Huffman compression, mono.

***.IFF/.SVX: Interchange File Format: 8SVX or 16SV data (Amiga)***

The 16SV flavor is rather uncommon and is, I think, only used by some PC trackers. Thus if you save 16 bit waveforms as .IFF files, dont expect them to be readable by most .IFF readers.

***Features:***

8,16 bits, loop, name, mono / stereo.



***.INI:     Gravis UltraSnd.Ini bank setup (PC)***

You can read the whole melodic bank 0 setup with all the associated patches from the UltraSnd.Ini file. You can also write extract .INI files for a complete bank that you can then paste into your UltraSnd.Ini file, this will also save all the patches referenced in the bank to .PAT files (they will be automatically named based on the name of the .ini extract file).

***Features:***

Same as .PAT, bank.

***.INRS: INRS-Telecommunications audio file***

***Features:***

16 bits mono.

*.IT: Impulse Tracker module/instrument/sample (PC)*

*Features:*

8,16 bits, mono, stereo, loop, name, instrument, bank.

***.KRZ: Kurzweil K2000 instrument file (MIDI)***

The K2000 file format are considered proprietary by Kurzweil; so any resemblance with the .KRZ file format this program reads are purely accidental :-/

If one, against better knowing should try to use this program to read the former file type, one shall find that they match fairly closely :-D, of course, sometimes things may go a bit wrong here and there...

***Features:***

16 bits, mono, loop, names, finetune, instruments, bank.

***Unsupported features:***

Envelopes, stereo, et c.

***Internet source of samples:***

*<ftp://ftp.uwp.edu/pub/music/lists/kurzweil/sounds>*

*<ftp://bach.nevada.edu/pub/K2000/sounds>*

***.MAT: Matlab variables binary file***

Note! 8bit is saved as unsigned 0..255 and 16bit as signed -32768..32767, however Matlab may think it's too loud when using sound(var,sr) so you may need to divide the sound data vector it by 8 or something.

Note! When reading floats they are expected to be in the range [-1, 1]

Note! Stereo files are read/written as complex vectors.

***Features:***

8,16bits / floats, mono / stereo, bank.

***.MAUD: MAUD sample format***

***Features:***

8,16 bits /  $\mu$ -Law / A-Law, mono / stereo, name.

***.MDL/.IST/.SPL:      Digitracker Module/Instrument/Sample File (PC)***

***Features:***

8,16bits, mono, loop, name, huffpacking, bank.

***.MED: OctaMED tracker module:MMD0|MMD1|MMD2 (Amiga)***

Note! The old MED tracker .MED format is not supported, if you have a description of this format, please send it to me.

***Features:***

8bit, loop, name, bank.



***.MOD: Module file (Amiga, PC)***

Recognizes the following flavors:

M.K., M!K!, 4CHN, 6CHN, 8CHN, 10CH, 12CH, 14CH, 16CH, 24CH, 32CH, FLT4, FLT8, OCTA, CD81, EX08.

***Features:***

8 bits, mono, loop, name, finetune, bank.

***.MOV: Apple QuickTime audio (Macintosh)***

Reads the audio track from QuickTime movies.

***Features:***

8/16 bits/ $\mu$ -law, mono/stereo.

***Unsupported features:***

Compressed formats.

*.MTM: Mtm tracker module (PC)*

*Features:*

8, 16 bits, mono, loop, name, bank.

***.MPA/MP2: MPEG audio stream, layer I, II or III***

Note! These files cant be autodetected so be sure to use any of the above file extensions.

Note! The MPEG decoder is rather slow and the resulting files may eat up a lot of memory...

***Features:***

16 bits, mono / stereo.

***.MUS10: Mus10 audio file (PDP-10)***

*Features:*

12,16,18,20bit, mono/stereo.

***MWSYNTH.INI: MWave DSP synthesizers GM bank setup (PC)***

The software synthesizer module for the MWave DSP uses this file to construct instruments from usual .WAV files.

***Features:***

8,16 bits, loop, name, instruments, bank.

***Internet source of samples:***

*ftp://watson.mbb.sfu.ca/pub/mwave/ibm/audiovation*, the windows install disks contains the patch set

***.MWS: MWave DSP synthesizers MWSYNTH.INI instrument extract (PC)***

The software synthesizer module for the MWave DSP uses the MWSYNTH.INI file to construct instruments from usual .WAV files. When exporting to the MWS format, *Awave* outputs a bunch of .WAV files as well as a .MWS text file containing an instrument extract that can be manually pasted into the MWSYNTH.INI file.

***Features:***

8,16 bits, loop, name, instruments.

*.NIST: NIST SPHERE file*

*Features:*

16 bits /  $\mu$ -Law, mono / stereo.



**.001:** *Typhoon vOice file (Yamaha TX16W)*

Note! The actual waveform data are stored in accompanying .C01 file(s).

*Features:*

DWWW compression, mono, loop, instrument.

***.OKT: Oktalyzer tracker module (Amiga)***

*Features:*

8 bits, mono, loop, name, bank.

***.PAT:   Advanced Gravis Ultrasound GF1 / Forte tech. patch file (PC)***

***Features:***

8,16 bits, mono, loop (incl. fractional), envelopes, pitch scaling, names, instruments.

***Internet source of samples:***

*ftp://wuarchive.wustl.edu/systems/ibmpc/ultrasound*

*ftp://archive.epas.utoronto.ca/pub/pc/ultrasound*

***.PCM: OKI MSM6376 synth chip PCM format***

This format supports the following sample frequencies: 4000, 5300, 6400, 8000, 10600, 12800, 16000, 21200, 32000 Hz. When saving files, the nearest supported frequency will be used and the waveform first resampled if it is not exactly the same frequency. To explicitly save in a certain frequency, first *manually* resample the waveform to that frequency.

***Features:***

16 bits, mono.

*.PLM/.PLS: DisorderTracker2 module/sample (PC)*

*Features:*

8,16 bits, mono, loop, name, bank.

***.PRG: WAVmaker program file (PC)***

Note! when importing the referenced .WAV files must be in the same directory as the .PRG file and when exporting, the .WAV files will also be put in the same directory.

Note! If you want to use Awave to create .PRG instruments, then, for best results when setting loop points, make sure that you first resample the waveform(s) (by waveform->Properties->Waveform->Process->Resample...) into one of the rates supported by WAVmaker: 11025, 22050 and 44100 Hz. If you do not, then they will automatically be resampled when you save the .PRG file..

***Features:***

8,16bits,mono,sterео,loop,name,instrument.

***.PSM: Protracker Studio Module Format***

***Features:***

8,16bit, mono, loop, name, bank.

***.PSION: PSION alaw file***

***Features:***

Alaw, mono.



*.PTM: Poly Tracker module*

*Features:*

8,16bit, mono, loop, name, bank.

*.SAM: Raw signed PCM sample file*

*Features:*

8 bits, mono.

***.SB, .SDW, .SND, .SW, RAW, .UB, .UDW, .UW: Raw PCM data format file***

The file extension has the following meaning:

S Signed data  
U Unsigned data  
B Byte (8bit) data  
W Word (16bit) data  
DW DWord (32bit) data

.SND is the same as :UB.

.RAW is the same as .SB.

Assumes 11khz mono, Intel byte order when reading. Use the waveform properties dialogs advanced tab to change any parameters that are not correct.

***Features:***

8,16,32 bits.

***.SBK/.SF2: EMU Sound Font Bank v1.x / v2.x file / Creative Labs SB AWE 32 (PC)***

Note! When saving a whole bank, all items in the bank will be saved, also unused waveforms. When saving at the waveform level, there will automatically be created an instrument using that single waveform.

The new, improved and generic Sound Font v2.0 (.SF2) format (which just might become a more widely used standard) is supported in addition to v1.x (.SBK). CL says they will support SF v2.0 during the 2nd quarter of 96 so in about a years time it will probably have replaced SF v1.0.

***Features:***

16 bits, pitch, loop, name, instruments, bank.

***Unsupported features:***

A bunch of parameters.

***Internet source of samples:***

*ftp://ftp.cyberworks.net/pub/AWE32*

*.SD, .DIG:                   Sound Designer I audio file (Macintosh)*

*Features:*

8,16 bits, mono, name.

***.SD2: Sound Designer II flattened/resource fork file (Macintosh)***

On the Macintosh these files are really two file forks. When they are transferred to a PC these two forks may be flattened (preserving all data and parameters) or only the resource fork (containing the raw sound data) may be copied. if only the resource fork has been copied, then it is assumed to be 16bit mono data.

***Features:***

8,16 bits, mono/stereo.

***.SDK: Roland S-series floppy disk image file (MIDI)***

They might go under another extension but SDK seems to be most common.

Note! The instrument key assignments may not always be read correctly so you may have to do some manual editing on that.

***Features:***

12/16 bits, mono, loop name, finetune, instrument, bank.

***Unsupported features:***

A zillion different parameters.

***Internet source of samples:***

*<ftp://lotus.uwaterloo.ca/pub/sgroup/samples>*

***.SDS, .SDX: Midi Sample Dump Standard file (MIDI)***

SDS is a raw midi data dump of an SDS (Sample Dump Standard) transfer and can be up to twice as large as the actual waveform. SDX is a 'compacted' form that the program SDX uses to save SDS samples in order to avoid this problem. It also contains a sample name which SDS does not.

***Features:***

8..32 bits, mono, loop, (SDX only: name , note).

***Internet source of samples:***

*ftp://sweaty.palm.cri.nz/sds/samples*  
*ftp://alf.uib.no/pub/midi/sds/samples*



***.SF:      IRCAM SoundFile format***

*Features:*

16 bits, name, note, mono / stereo.

*Unsupported features:*

Other bits/sample.

SFISFI      SoundStage Sound File Info (.SFI) + Data (.SFD) files.

***.SFI, .SFD: SoundStage Sound File Info + Data file.***

Note! These files must always come in a .SFI + .SFD pair! You should only load the .SFI file.

***Features:***

16 bits, name, note, mono / stereo.

***Unsupported features:***

Other bits/sample.

SoundStage Sound File Info (.SFI) + Data (.SFD) files.

***.SFR: Sonic Foundry Sample Resource file.***

***Features:***

8,16 bits, mono, stereo.

*.SMP: Samplevision file (PC)*

*Features:*

16 bits, mono, loop, name, note.

*Unsupported features:*

Only the first of the 8 possible loop points are considered.

*Internet source of samples:*

*<ftp://ftp-ls7.informatik.uni-dortmund.de/pub/samplevision>*

*.SMP: Ad Lib Gold Sample file (PC)*

*Features:*

8,16 bits, mono, stereo.

*Unsupported features:*

Yamaha ADPCM compression.

***.SND: AKAI MPC3000 sample file***

The AKAI program (.PGM) files defining instruments (and referring to various .SND waves) are currently not supported. Any info on this format would be welcome.

***Features:***

16 bits, mono, loop, name.

***Internet source of samples:***

*<ftp://ftp.mfi.com/pub/keyboard/AKAI/>*

*.SND: Sounder sound file*

*Features:*

8 bits, mono.

*.SNDT: SndTools sound file*

*Features:*

8 bits, mono.



*.SOU,.PAC: SB Studio II sound/package file (PC)*

*Features:*

8,16 bits, mono, loop, name, bank.

***.SPD: Speach Data file (UNIX)***

***Features:***

16 bits / mono.

***.SPPACK:***     *SPPack sound sample file*

*Features:*

8,16 bits /  $\mu$ -Law / A-Law, mono.

*.STM: ScreamTracker module file (PC)*

*Features:*

8 bits, mono, loop, name, bank.

***.S3M, .S3I:   ScreamTracker v3 modules / Digiplayer sample (PC)***

The .S3I files are usually named .SMP but they will be saved as .S3I in order to avoid confusion with Samplevision files..

***Features:***

8,16 bits, mono / stereo, loop, name, note, bank.

***.SYW/.W??: Yamaha SY-series wave file (MIDI)***

The Yamaha 16-bit SY-series wave sample files are really named *.W??*. There's also *.T??* (All Data Files, often contains waves and are much more common than *.W??*; but aren't supported at the moment), *.J??* (Patch parameters?) and *.K??* (Sequenced songs). See also the [12bit format](#).

*Note!* Before reading a directory of files on your synthesizer, make sure they have all unique extensions like *.W01*, *.W02*, *.W02* et c.

***Features:***

16 bits, mono, loop.

***Internet source of samples:***

*ftp://louie.udel.edu/pub/midi/patches/SY99*

***.TXT: Ascii Text formatted audio data***

Ascii text file with integer sample values in base 10. Samples are delimited by either , (ascii no 44) or newline (ascii no 10). Comments and lines not containing sample values must begin with a ; or a %. For stereo files (with interleaved samples), you must manually set stereo in the waveforms properties box advanced tab.

Ex 1:

```
;signed 8bit, 3 samples  
127 ; first sample  
-3  
15 ; last sample
```

Ex 2:

```
% This will be detected as unsigned 15bit, 5 samples  
9,0,32767,1000,20001
```

***Features:***

1..32 bits, signed or unsigned (autodetect).

### ***.TXW/.W??: Yamaha TX16W wave file (MIDI)***

The Yamaha TX16W (and probably also the other Yamaha 12bit samplers) waveform files are really named *.W??* (where ?? is a number). There's also *.F??* (Filter), *.V??* (Voice to midi no assign?), *.U??* (Performance settings) and *.S??* (System setup?) files which are of no use to you. See also the 16bit format.

*Note!* The format only supports 16, 33 or 50kHz sample frequency and no tuning or rootkey parameters so when saving to TX16W *.W01* files, the closest of the three rates will be selected and you must manually tune it when you're using it in an instrument on your synth... Alternatively you may resample it in *Awave* to either of 16, 33, 50kHz before saving it.

*Note!* Before reading a directory of files on your synthesizer, make sure they have all unique extensions like *.W01*, *.W02*, *.W02* et c.

*Note!* When saving files, be aware that the TX16W only supports a maximum of 128K samples in the attack plus 128K samples in the loop, if the waveform is longer then it will be truncated by *Awave*.

#### ***Features:***

12 bits, mono, loop.

#### ***Internet source of samples:***

*ftp://ftp-ls7.informatik.uni-dortmund.de/pub/tx16w/samples*



***.ULAW: US Telephone format (CCITT G.711) audio***

This is also the format used by the MIME mail encoding standards audio/basic attachments..

***Features:***

μ-Law, mono, 8kHz.

***.ULT, .UWF: UltraTracker module / wave file (PC)***

***Features:***

8,16 bits, mono, loop, name, bank.

***.UNI: MikMod 'UniMod' tracker format (PC)***

***Features:***

8,16bit, mono, name, loop, instruments, bank.

**.V8: Covox 8bit file (PC)**

*Features:*

8bit, mono.

***.VAP: VAP annotated speech file***

***Features:***

8bit, mono, name.

***.VOC: Creative Labs sample file (PC)***

Note! When exporting to the .VOC format, files will be written in a newer and much more capable subformat that may not be recognized by some old programs.

***Features:***

8,16 bits /  $\mu$ -Law / A-Law, mono / stereo, name.

***Unsupported features:***

Other compression and packing schemes.

***.VOX: Dialogic adpcm file***

***Features:***

Adpcm compression (4bit), mono, 8kHz.

***.WAV: Microsoft Wave file (PC, Windows, OS/2, NT)***

Supports the Sample (smp1) chunk (used by Turtle Beach Wave SE, WavePatch, Gravis PatchMaker and others), the Instrument (inst) chunk (used by IBM?) as well as cue +labl chunks to support loop. Also supports some info chunks (INAM, ICMT, ISFT).

***Features:***

1..32 bits PCM /  $\mu$ -Law / A-Law, MS ADPCM (4bit) / Intel DVI/IMA ADPCM (2,3,4bit) / CCITT G.721 (4bit) & G.723 (3,5bit) ADPCM / GSM 6.10, MPEG audio encodings and compressions, mono / stereo, note, loop, rootkey, finetune.

***Unsupported features:***

Other compression schemes and formats.

***Internet source of samples:***

*<ftp://ftp.funet.fi/pub/sounds/wav/samples>*  
*<ftp://doc.ntu.ac.uk/pub/sounds/samples>*



***.WFB, .WFP, .WFD: Turtle Beach WaveFront synthesizers / WavePatch file (PC)***

WaveFront based synthesizers are currently the Maui, Tropez, Rio and Monterey.

.WFB files contains a whole Midi Bank, .WFP a single Program and .WFD a complete drum set.

Instruments using ROM samples, of course, cant be converted.

If the files doesnt use embedded waveforms, but external .WAV files instead, you must have the .WAV files in either the exact path they where when the .WF? file was created, or, in the same directory as the WF? file.

Currently *Awave* only saves files with embedded waveforms.

Note! TB Rio users should only save .WFP files as the .WFB exported by this program are Maui style and may not be loadable on the Rio.

Note! When saving to .WFB bank files, waveforms not contained in an instrument will not be saved.

***Features:***

8/16/ $\mu$ -Law bits, name, note, loop (incl. fractional), instruments, bank.

***Unsupported features:***

Everything is read on patch level and not on Program/Drumkit level, i.e. multilayered instruments and drumkits appear as multiple instruments.

Pre v1.2 files.

***Internet source of samples:***

*ftp://ftp.hawaii.edu/outgoing/Maui/*

*.XM, .XI: FastTracker 2 eXtende Module/Instrument (PC)*

*Features:*

8,16 bits, mono, loop, name, instruments, panning, bank.

*Unsupported features:*

A bunch of ramping parameters (envelope, pan, vibrato...).

***'Clipboard' virtual export format:      Copy the audio data to the Windows clipboard.***

The data will be copied as a RIFF-WAVE, the standard clipboard audio format.

***Features:***

Same as for .WAV files.

***'MidiSDSDump' virtual export format: Transfer waveforms over Midi using SDS.***

Pops up a dialog box before transferring each waveform in the selected item using the Midi sample Dump Standard protocol. You must choose destination midi output device, midi channel, and sample slot in the dialog box.

***Features:***

8..32bits, mono, loops.

***'AudioPlayer' virtual export format: Plays the sound on a digital (waveform) audio playback device.***

If you have a basic soundcard capable of playing digital audio, you can use this 'export' format to play the sounds contained in a waveform, instrument or bank.

Set the output device to be used for playback in the Options->Properties dialog. If you have more than one digital audio card installed in your system, choose the most capable one.

***Features:***

8,16 bits, mono / stereo.





MISC:



✘ *FMJ-Software!*

If you enjoy this program, please  
support my efforts by registering it.

PGP:

Shows if the program is registered or unregistered.

Click button to locate your personal registration keycode and remove all imposed limitations of the unregistered version of *Awave*.

Click button to uninstall this program.

This will clean up the registry entries for the program.

If you have a www-browser installed on your computer, then clicking here will get you to FMJ-Softwares www site. There may find the latest news and updates to this program among other things.

Check this box if you want ***Awave*** to associate all its supported files with itself so that you may easily access ***Awave*** through *play audio* and *convert* commands on the Windows Explorers right mouse button menu.

*Note!* If any other program is already associated with a file type, then that association will *not* be overwritten.

*Note!* if you ever uninstall ***Awave*** then these associations will be removed automatically.

PAP:



Check to enable auditioning.

Auditioning works by routing midi messages from a keyboard or other midi-in device to a wavetable synth, combined with automatic creation and loading of instruments and samples to the same wavetable synth.

Select the wavetable synth you want to use for the auditioning feature.

Select the midi input device you want to use for the auditioning feature.  
Data from this device will be routed to your selected wavetable synth, and automatically translated to use the correct bank, program and channel.

Check to enable the virtual keyboard.

When enabled you can play on your computer keyboard instead of (or as a complement to) an external midi keyboard.

PPP:

Select the primary digital audio playback device you wish to use with *Awave* AudioPlayer. If this device is used when the AudioPlayer is launched then any other free device(s) will be used.

The AudioPlayer has an built in oscilloscope graph.

Selects the refresh rate of the graph. Higher numbers gives smoother look but taxes the CPU more. If your computer is up to the task, set it to the monitor refresh rate (usually about 70Hz) or else set it to half or even a forth of it.



Use Microsoft DirectSound services for waveform playback.

Selecting this will let you have multiple simultaneous AudioPlayers and the Awave software synth will be polyphonic rather than monophonic. It will also reduce latency (the time it takes before a sound starts playing) as well as much reduce the memory usage.

This option is only available if you have installed Microsoft DirectX v2 on your computer.

Use the old Windows multimedia services for waveform playback using an audio buffer queue.

Use the old Windows multimedia services for waveform playback without using a audio buffer queue. Select this only if you have trouble getting any sound from the AudioPlayer otherwise. Looping support in the AudioPlayer synth will also be disabled when this box is checked.

BGP:

Source file (import type) description.

Source file name and path.

Info on the number of top level items in the bank and their total size.

The Midi GS Bank Change number intended for the bank.



BPP:

A list of what instruments have been assigned to what midi program numbers in the current bank.  
You can change the program number for an instrument in the instruments property sheets general tab.

IGP:

Info on the number of waveforms the instrument and their total size.

Place comment, copyright, or creator name here.

Relative volume setting.

Range is 0..127. (min. .. max.)

The Midi Program Change number intended for the instrument.

LKP:



The key assignment map for the instrument. Press down the mouse button and drag with the mouse over a range of keys to assign a new sample or to clear the range.

ASSIGN

CLEAR

CANCEL

## WAVEFORMS

LOKEY

HIKEY

WGP:



## Panning.

This specifies the positioning of the waveform in the stereo field.

Range: 0..127, 0 = far left, 127 = far right.

Enter the sampling frequency in Hz (number of recorded wavesamples per second)

Info on the waveform:

SourceBitsPerSample : ExpandedToBitsPerSample, NoOfWavesamples, Signformat, Channelformat

Relative volume setting.

Range is 0..127. (min. .. max.)

Root key.

This is the midi-key number at which the waveform is played back at its original sampling frequency.

Fine-tune.

Adjust the Root Key in steps of 1/1200th of an octave.  
Range is -99..99 cents.

WFP:

Controls the position (offset) of the part of the waveform that is displayed.



Play the waveform with the Audio Player.

Cut away the selection and place on the clipboard.

Copy the selection to the clipboard.

Insert waveform from the clipboard. The insertion point will be the start of the current selection.

Paste waveform from the clipboard. It will be pasted over the current selection.

Select the whole waveform.

Displays the waveform. Red is True RMS (Root of Mean of Squares) display, Light blue dots is maximums. Blue bars is loop points.

Controls the zoom in state of the waveform window.



Bring up the audio processing dialog.

Selects the Power/t graph.

Selects the Current/t graph.

WLP:

How the waveform look like when its looping! They dark parts are just before the waveform first enters resp. leaves the looping portion. The light part is the actual looping section with the loop point break in the middle (this should be much easier to understand by looking at the graph and trying moving the loop point sliders than trying to explain it in words )

Fine-tunes the position of the loop start point.

Fine-tunes the position of the loop end point.

Zooms the two loop split displays in the time domain.



Zooms the two loop split displays in the amplitude domain.

Loop start point.

Also called repeat from point. The loop start point is the first wavesample *in* the loop range.

Note that fractional loop points is supported. This is used by synthesizers that can interpolate between samples.

The point is expressed in number of wavesamples from the beginning of the waveform, starting at zero.

Loop end point.

Also called repeat point. The loop end point is the first wavesample *after* the loop range.

Note that fractional loop points is supported. This is used by synthesizers that can interpolate between samples.

The point is expressed in number of wavesamples from the beginning of the waveform, starting at zero. It may not be less than the loop start point.

Check to enable sampled release.

This means that the waveform loops while the key is pressed down, but stops looping and plays to the end of the sample (remember there may be sample data past the loop end point) when it is released. When this is not checked, it loops until the amplitude envelope has faded it out upon key release.

Check to enable reverse looping.

This makes the loop go backwards instead of forwards.

Check to enable looping.

When looping is on, the waveform is looped between the loop start and loop end points, as long as the midi key is pressed.

Check to enable bi-directional looping.

Bi-directional looping means the loop direction alternates between forwards and backwards.

WEP:



Check to enable the amplitude envelope. When de-checked a default envelope will be used instead.

This graph depicts the amplitude envelope.

Zoom the amplitude envelope graph in the time domain.

Amplitude levels for the nodes in the envelope.

Times between nodes in the envelope.

WAP:

Check if signed wavesamples.

Check if stereo wavesamples.



Check to perform a byte swap when reading 16bit data (necessary when reading Motorola byte order data).

Select the wavesample data type.

Enter the size of the file header (the number of Bytes preceding the actual sample data)..

Enter waveform size (number of Bytes of sample data).

The value is automatically truncated if it extends beyond the file length.

If all data, after the (optional) file header, is sample data, then you need not change this.

Note that 16bit data is 2 Bytes/Sample/Channel.

PLAY:

Name of the waveform.

Info about the waveform.

The length in seconds of the waveform.



The current playing position.

Press to start or stop playing the waveform.

Go to the next waveform in a sequence. Closes the Audio Player if this is the last waveform.

Select to pause the oscilloscope.

Select to set waveform display mode.

Select to set spectrum analyzer mode.

An oscilloscope!

FX:



Select audio processing algorithm to apply to selected data range..

OF:

Press to get help on file formats...

Displays what file format type the currently selected file has been determined to be.

The % value indicates how certain the file format type detection was with 100% being very certain, and 0% being quite unsure...

Select the subformat or dataformat to use when saving waveforms.

Check to allow saving in mono format. If both the mono and the stereo boxes are checked then the most suitable alternative will automatically be chosen for every waveform in the file.

Check to allow saving in stereo format. If both the mono and the stereo boxes are checked then the most suitable alternative will automatically be chosen for every waveform in the file.





# Revision History

✘

*A big thanx should go to those people who have supported me with information and beta testing; Honourable mentions must also go to Fred Warren, who donated a Maui card, and to Ari Laakkonen, who donated an AWE32 card.*

✘

## *Revisions:*

### future versions

version 3.2, <96-09-17>  
version 3.1, <96-06-06>  
version 3.0, <96-04-21>  
version 2.7, <96-03-10>  
version 2.5, <96-01-21>  
version 2.0, <95-09-09>  
version 1.5, <95-04-14>  
version 1.4, <94-10-14>  
version 1.3, <94-08-02>  
version 1.2, <94-07-28>  
version 1.1, <94-06-08>  
version 1.0, <94-04-27>  
version 0.9β, <94-03-30>  
version 0.8β, <94-03-08>  
version 0.7β, <94-01-07>  
version 0.6β, <94-01-05>  
version 0.5β, <93-12-23>  
version 0.4β, <93-12-20>  
version 0.3β, <93-11-26>  
version 0.2β, <93-11-21>  
version 0.1β, <93-11-13>

✘

## *Future versions,*

Things that are being considered for future versions of *Awave* are:

- DirectMusic, DLS 1.0
- More import and export formats (just send me any format descriptions you can find),
- A generic waveform synthesizer...
- Multilayered instrument support...
- Amplitude envelope support and editor...
- What else? Send me a note if you have any great idea!

Remember, the registration fee is the key to further development of this program!

✘

## **version 3.2**

You can now explicitly select subformat type when saving to an export format.

Added US Robotics voice modems / QuickLink format (.GSM) import and export.  
Added Sound Designed II flattened import and export.,+ resource fork only import (.SD2).  
Added QuickTime (.MOV) audio import.  
Added GSM subformat to .WAVs export.  
Added Fast Tracker instrument (.XI) export.  
Added speach data (.SPD) import format, 16b mono, 8000hz, noauto.  
Added Akai MPC3000 wavefile (.SND) export.  
Added  $\mu$ -law support when writing .FFF files.  
Added  $\mu$ -law support when writing .WF? files.

Can now have multiple property sheets and AudioPlayers as long as no-one is for the same 'item'.

Added a 'Use Windows multimedia services, unbuffered output to the AudioPlayer property tab for those that has had problems with no sound in the AudioPlayer.

The Awave AudioPlayer synth has been rename into 'Awave software synth'.  
Fixed the worst 'popping' problems in the software synth.  
The software synth now supports pitch bend and all notes off midi messages.

On the Edit menu there is now a submenu 'Process all' with two subitems:

- Resample..., resamples all waveform to a given sample rate.
- Convert loops..., converts all loops to common format.

Added support for Microsoft DirectSound (optional, select 'Use Microsoft DirectSound services' in the AudioPlayer property tab).

It's benefits are:

- Lower memory usage for the AudioPlayer and the software synth.
- Lower latency for the AudioPlayer and the software synth.
- Allows multiple simultaneous AudioPlayers as long as they do not play the same item.
- The software synth is 8 note polyphonic.

Auditioning GUS & WaveFront: Will load as 16b if it fits in memory, else it will load as 8b if that fits, else it fails...

Improved the "Convert to stereo" selection in the audio processing wizard so that you can now combine two waveforms into one stereo waveform.  
Added Statistics selection to the audio processing wizard.

Added option to enable and disable association of file formats.

Fixed bug with .EFE files (hanged when editing a waveform on an Ensoniq)  
Fixed problem with recognizing .PAT files from 'Howling Dog Systems PowerPatch series' CD-ROMs.

Various minor bugfixes and improvements.

✘

### version 3.1

Added GUS PnP .FFF format import/export.  
Added Yamaha SY-series waveform (.W01) export.

Added Dialogic adpcm file (.VOX) import & export.  
Added Sound Designer I format (.SD) export.  
Added DiamondWare Digitized file (.DWD) import & export support.  
Added OKI MSM6376 synth chip format (.PCM) import.  
Added raw GSM 6.10 audio stream (.GSM) export.  
Added Digilink (.DIG) format import.

It's now possible to select multiple files for opening in the file open dialog.

Added 'Midi SDS receive' to Edit menu.  
Added 'Send all' to the MIDI SDS transmit dialog. This frees you from having to press the 'Start' button for every waveform when downloading banks or instruments; usefull if you want to dump a lot of samples...

The 'AudioPlayer synth' now supports forward looping.

Added 'Convert reverse to forward' loop to audio processing wizard.

The AudioPlayer command line interface now accepts directory names as well as file names, when a directory name is given, all recognized audio files in that directory will be played in turn.  
Added -LOOP switch to the AudioPlayer command line interface to repeatedly play the indicated file(s). Break the loop by closing the AudioPlayer with the [x] button. Ex: To repeatedly play all audio files in the current directory, type on the command line: start AWAVE -AP -ONCE -LOOP .

Various minor fixes and improvements:

Fixed bug with TB WaveFront cards auditioning that made it require wfgate.dll v1.3.0.6, should now work with other versions as well,  
Reading EDE files bug fixed,  
Saving TX16W W01 files sometimes crashed, fixed that,  
In the instrument keymap editor, you can now click on an 'assigned range' (rather than drag with the mouse over a range) to change that waveform assignment,  
Fixed problem with Ensoniq instruments larger than 1MB,  
et c...

✘

### version 3.0

100 file format barrier broken => I must be crazy :-)

Added auditioning for all digital sound cards! Emulates a very simple monophonic synthesizer (no loop support though...).

Added cut, copy, insert and paste to what is now the waveform 'editor' in the wavform property sheet.  
Added Power / Current graph toggle in the waveform properties Editor tab.  
Added (time domain) resampling algorithms (nearest neighbour, linear interpolation, sin(x)/x filter) to the Audio processing wizard.  
Added Bidir to forward loop conversion to the Audio processing wizard.  
Added fade in, fade out, trim, cut, reverse, remove silence, maximize amplitude, set loop point, and smoot out selections to the Audio processing wizard.  
Added 'Split into mono channels' command to the right mouse button menu for stereo waveforms.

For non-standard playback rates, the Audio player Always now resamples the waveform to an appropriate standard rate.

Decreased the need for (physical) memory when using the Audio Player. Should reduce the disk crunshing for large files...

Improved the accuracy of the AudioPlayers frequency spectrum graph.

The Audioplayer positon bar in now draggable, both when stopped and when playing.

Added a -NOWIN command line switch to launch the AudioPlayer and play the sound without displaying the AudioPlayer window.

Tweaked the file open dialog to show detection info on selected files. Also increased the visible length of file formats combobox list.

Fixed what I hope is the last two bugs in my .SBK writing code :-)- (bugs occured whenever there were names longer than 19 chars or when there were 'shared' or 'lonely' waveforms...)

When loading an .SBK which uses hundreds of ROM-samples you no longer have to press OK a hundred times...

Fixed incompatibility between my .AIF writing code and a few programs.

Fixed bug when writing .WFB's.

Added log/status window when auditioning using the Maui/Tropez.

No longer automatically creates instruments for 'lonely' waveforms when saving bank items. You have to specifically use the new 'Autocreate instruments' command (on the File menu) before you save if you want it.

If you want to edit the key assignment for the virtual keyboard you can now do that with the registry editor (regedit.exe) that comes with Windows 95. Open this \_computer->HKEY\_CURRENT\_USER->Software->FMJ-Software->Awave and edit the mtVKeys string. It's just a string of (normally 29, but maximum 36) uppercase characters holding the ascii key to assign to a note, beginning with C3 for the first character and then in increasing order...

On request, I have added a few 'keycuts' (F5 - Save as..., F6 - Open new file... F7 - Play audio... F8 - Properties...).

Added WAVmaker program files (.PRG) import and export.

Added Yamaha TX16W waveform export (.W01).

Added support for 16bit Roland samplers floppy disk images (.SDK). Now also tries to convert the instruments for .SDKs (both 12bit and now 16bit) files although it may not always get it right.

Added Ditrakker instrument (.IST) format as well att v1.x module (.MDL) import.

Added Extreme's Tracker (.AMS) format.

Added FunkTracker module format (.FNK).

Added Impulse Tracker modules/instruments/samples (.IT).

Added DisorderTracker2 module (.PLM) and sample (.PLS) files.

Added Poly Tracker module (.PTM).

Added ABT Extended MoDule (.EMD) format.

Added Velvet Studio Modules (.AMS) / Instruments (.AIS) / Samples (.ASE).

Added DigiTrekker module (.DTM).

Added DCM module format (.DCM).

Added Typhoon vOice files (.O01)

Added Typhoon wave files (.C01)

Now reads more Ensoniq synthesizer files (.Efx) and disk images (.EDx, x=A,E,K,Q,S,T,V)

Added raw CCITT G.721 ADPCM data (.g721) read and write.

Added an experimental 'Upload bank ' command, related to the auditioning feature. You can use it to upload all of the instruments in the bank to the wavesynth. As it is still in the 'experimental stage ', I can not guarantee that it will work as intended. The idea is that, using this command, you'd be able to develop 'cross-soundcard' compatible instrument bank files. You'd then be able to distribute them with your midi songs and use Awave to upload them to any one of the supported wavetable sound cards. If there is enough interest, I might be able to produce freeware, standalone, 'bank loaders ' for the GUS, AWE32, Maui/Tropez respectively.

Various minor fixes and improvements.

✘

## version 2.7

Fixed the SB AWE32 auditioning to work with the new Win95 drivers. The AWE32 auditioning now uses the new .SF2 file format so in fact these latest, .SF2 compatible, drivers are now required (for auditioning with the AWE32). Note! They had only been released in beta (although public beta) versions at the time this was written.

Changed the loop editor to a nice 'dual layer' view.

Added an 'Audio Processor wizard' connected to the 'Process' button in the waveform properties box's waveform tab. Only 16->8bit, 8->16bit, mono->stereo and stereo->mono conversion algorithms are available for the moment.

Added spectrum analyzer option to the Audio Player (uses a Fast Hartley Transform (FHT) algorithm on 256 samples to compute a 128 band power spectrum, all in real time).

Added 'Virtual Keyboard' feature, i.e. when using the auditioning feature (with one of the supported wavetable cards) you can now play on the computer keyboard (if you do not have an external midi device).

Added a 'programs' tab to the bank property sheet where you can easily view (and edit) the instrument to midi program number assignments for the bank.

Can now save whole instruments to Ensoniq .EFE files in addition to single waveforms (i.e. more than one waveform/instrument are now supported).

Added Windows Clipboard import.

Added .AVI import support (audio part only...)

Added ultrasound initialization file (ULTRASND.INI) melodic bank import as well as 'export' (.INI extract + bunch of .PAT files).

Added support for writing  $\mu$ -law and A-law format .AU and .WAV files.

Added Matlab variables binary file (.MAT) import and export.

Added .VAP annotated speech files import.

Added .UB, .UW, .UDW, .SB, .SW, .SDW, .ULAW, .ALAW import formats and changed .RAW, .SND export formats into .UB, .UW, .SB, .SW, .ULAW, .ALAW. Here U means unsigned, S signed and B, W & DW means byte (8bit), word (16bit) & dword (32bit) respectively.

Added Covox 8 Bit File (.V8) import.

Added Ad Lib Gold Sample (.SMP) import

Added Sonic Foundry Sample Resource (.SFR) import

Fixed AudioPlayer scope bug that made it go faster than the sound on some soundcards (read Sound Blasters when playing 16bit and/or stereo data).

Whenever the selected WaveOut device is busy playing something else, the Audio Player will now look for, and use, any other waveout device(s) you may have installed.

Fixed SBK/SF2 bug that made it hang sometimes when saving.

Fixed MS ADPCM (.WAV) format decoder bug.

The mandatory small fixes and improvements.

✘

## version 2.5

Changed the name from 2PAT into the more appropriate *Awave*. All registered users of 2PAT are, of course, also registered to *Awave*.

Waveforms are now cached in memory. This eats memory but is necessary for some of the new as well as for future editing functions. It also makes auditioning more responsive.

A visual loop point editor page has been added to the waveform property sheet.

A waveform display page has been added to the waveform property sheet (true RMS display, plus peak indication)

Auditioning feedback is now available when editing stuff in the property sheets (i.e. if you have one of the supported wavetable synthesizers). Just hit the Apply button to download the changes to the synth.

An oscilloscope display has been added to the AudioPlayer.

Added option to force the AudioPlayer to use only standard sample playback rates (some buggy drivers says they can play things that they cant!).

Added -ONCE (closes the AudioPlayer when finished playing a waveform) and -DELETE (deletes the sourcefile when finished) command line options in addition to the old -AP (launches in AudioPlayer mode) switch.

Auditioning with the SB AWE32 no longer uses a stand alone program, but now requires the latest drivers with 32bit application support, more specifically, with the AWEMAN32.DLL file.

Added an Insert File command that lets you merge several files.

Added a Save All As command to allow you to easily save all instruments or waveforms.

Added MidiSDSDump virtual export format to allow Midi transfers of waveforms to external synthesizers.

Added Ensoniq EPS instruments, .EFE, export (for single waveforms only).

Added Sound Font v2.0 (.SF2 files) export. CL sais they will support SF v2.0 during the 2nd quarter of 96 so in about a years time it will probably have repleced SF v1.0 and .SBK export support will then be dropped from this program.

Added .VOC export format (note! files are written using the new and much more capable type 9 subformat (well its not that new, but theres still a few programs that do not support it)).

Added .S3I, Digiplayer/ST3 samplefileformat export

Added text formatted audio, .TXT export (the old text description format renamed to .DES)

Added MWave DSPs software synthesizers MWSYNTH.INI bank read support as well as .MWS text file export, containing an instrument extract that can be manually pasted into mwsynth.ini.

Added MPEG audio layer I, I and III import, .MP2, .MPA (very slow and memory hungry though).  
Added GSM 6.10 audio stream import (.GSM)  
Added SoundStage .SFI/.SFD import.  
Added Farandole .F3R blocked linear module format (1.0).  
Added SBStudio II, .SOU, .PAC sound & package files import.  
Added .EPSP audio file import.  
Added .NIST SPHERE import.  
Added .SPPack sound sample file import.  
Added .INRS Telecommunications audio file import.  
Added: MAUD sample format import.  
Added .SD1 Sound Designer I file import.  
Added PSION alaw file import  
Added Mus10 audio file import  
Added Sound Tools HCOM format import.  
Added  $\mu$ -law and IMA compression support for .AIFC files.  
Added CCITT G.721 & G.723 ADPCM as well as IEEE 32bit and 64bit floating point support for .AU.  
Added CCITT G.721 & G.723 ADPCM, GSM 6.10 and MPEG audio support for .WAV.  
Added support for the new type 9 .VOC blocks as well a  $\mu$ -law and A-law compression.

Some corrections/bugfixes for loop points of .KRZ and a few other file types.  
Fixed bug with many .GKH files appearing empty. Now also automatically removes the duplicate but noisy wavesamples that appeared from multilayered instruments..

Added fast auditioning on/off switch to the Options menu.

Much more transparent file type associating behaviour. No longer will you be prompted if you want to replace types that have already been associated with other programs. You'll have to do that yourself if you want to. Registers 'convert' and 'play' command verbs in the new Windows 95 shell extension fashion (so multiple programs can cooperatively assign different commands for a given type). This doesnt work very well under Window 3.1 so this feature has been disabled when running under Windows 3.1. You can of course still manually associate file types using the File Manager.

And, as always, a whole bunch of bug fixes and minor enhancements .



## version 2.0

User interface enhanced based on the Windows 95 UI guidelines. Runs under Windows 95, Windows NT 3.51 or Windows 3.1 + Win32s v1.3.

The really stupid couldn't register app class bug fixed (occured on many Win32s systems).

Now writes converted stuff to disk first when you actually saves or auditions anything, instead of doing everything in one sweep when opening a file, as was done in previous versions. This change has made possible a number of the enhancements below.

Tree list view with nodes and leaves for bank file, instrument and wavesample levels. You can now save stuff from any of the 'levels' (Note only a few of the export formats support all levels). There's also Property sheets for the items of the various levels. Most stuff is editable!



You no longer have to decide on export format before you load a file. Just select from a list in the Save dialog.

Auditioning now works regardless of what youre going to export to.

SoundBlaster AWE32 auditioning support added.

A virtual '.DAC' export format added. Saving to this format will simply play the item on a digital audio output device. So users that doesn't have a GUS, Maui or AWE32 can now play sounds as well, provided you have some sort of basic audio card. An 'Audio Player' item in the right mouse button menu acts as a shortcut for saving to this format.

Audio Player mode and -AP command line switch added. Lets 2PAT be used as a multiformat digital audio player. Great as a helper app for your favourite Web-browser.

A virtual '.CLIP' export format added. Saving to this format will simply copy a wavesample to the Windows clipboard.

.AU export format added.

.IFF export format added.

.SBK export format added as well as improved import support..

Smart file format recognition, Tries to recognize the format of a file correctly, even if it has no, or even an incorrect, file type extension.

FastTracker 2 .XM (module) and .XI (instrument) files read supported.

Farandoyle .FAR tracker modules read support.

Oktalyzer .OKT tracker module read support.

Sndtool sound file, .SNDT read support.

Souder sound file, .SNDR read support.

IRCAM SoundFile format, .SF read support.

Raw Audio-CD format data, .CDR read support

MikMod 'UniMod' format, .UNI read support

Digital Sound Module tracker format, .DSM read support

Text formatted audio data, .TXT read support

Delusion/XTracker Digital Music/Sample Fileformat, .DMF, .DSF read support

Digitrakker Module/Sample File, .MDL, .SPL read support.

Signed 8bit Sample data, .SAM read support.

Farandoyle Linear module (2.0) format, .F2R read support.

Machintosh SoundCap/SoundEdit recorded instrument format, .DEWF read support.

Protracker Studio Module Format, -PSM read support.

OctaMED tracker modules (MMD0|MMD1|MMD2), .MED read support.

Machintosh Audio Visual Research sound files, .AVR read support.

Ensoniq, .EUI disk images read support, somewhat like .GKH, wich now also reads subdirectories.

US telephony audio format, .UL read support.

AKAI MPC3000 sample files, .SND, read support

Stereo support for a number of formats (WAV,SND,RAW,S3M,S3I,IFF,AU,AIFF)

A-law and  $\mu$ -law encoded .AU files now supported.

A-law,  $\mu$ -law, MS ADPCM and DVI/IMA ADPCM encoded .WAV files now supported (read).

16bit 'SV16' flavour of IFF files now supported (read/write).

A bunch of additional .MOD flavours now recognized.  
Added cue-points support for looping in .WAV files (read/write).  
Fixed ugly bug with .SDS import and export.  
Reads and writes 'inst' chunks in .WAV files.

A 'Dummy' midi in device now added so people without a midi-keyboard can use the audition feature of 2PAT by using any onscreen keyboard program.

Added 'Uninstall' button to the program properties box that can be used to clean up the registry.

Imposed a somewhat annoying save times limit. Registered users receive a personal keycode that removes this.

✘

#### version 1.5

Reworked user interface for Windows 95. 'Right mouse button' based, uses the registry, associates file types, uses long filenames, new icons, resizable window frame et c. No longer uses CTL3D32.DLL since thats no longer necessary for the 3D look. Will still run under Windows 3.11 with Win32s v1.1 and up but will probably not look very good; Win32s v1.3 will probably fix that when its out.

Instrument info box with info corresponding to a .DES file.

Improved Kurzweil *KRZ* support; fixed the sample rate and a few other things. Now creates instruments from both the sample and the keymap level, since the keymap, i.e. instrument, level doesnt always work....

Added *AIF* as export format.  
Added *SND* as export format.  
Added *RAW* as export format.

Fixed problem with not beeing able to launch WinHelp from arbitrary directories.  
Assignes unique names for 'unnamed' samples and instruments (fixes problem with WaveFront based cards).  
Fixed problem with unnamed instruments in .PAT files.

✘

#### version 1.4

Improved support for Turtle Beach WaveFront based cards! (Maui / Rio / Monterey);  
Reads .WavePatches.WFB, WFP and WFP files (v1.2 or later only).  
Exports .WFP files.  
Supports downloading instruments to the Maui card, using the WF GateKeeper (WFGATE.DLL).

Reads ScreamTracker 3 instruments (S3I).

✘

#### version 1.3

Now a Win32s (v1.1 or later) program.  
CTL3D32.DLL replaces CTL3DV2.DLL.

FMJMTHRU.DLL no longer needed.  
Saves config in the Win32 registry instead of in an .INI file.

Now reads the GKH disk image files in addition to INS/EFE for Ensoniq EPS.

Barebone .SBK import support (EMU SoundFonts Banks / SB AWE32); at wavesample level.

✘

#### version 1.2

Restructured for multiple export formats, currently:

- Ultrasound patches (.PAT)
- Turtle Beach Maui (.WAV+.WFS+.DES)
- Ms Windows wave files (.WAV+-DES)
- Midi Sample Dump Standard files (.SDS+.DES)

Config dialog window where you can choose export type and midi in device.  
Saves config in 2PAT.INI.

You can start 2PAT with a filename on the command line and that file will be loaded at once; so now you can associate various file extensions with 2PAT in the File Manager and launch 2PAT by double clicking on a file.

Reads and writes .WAV files with the (new Ms standard) smpl (root key, finetune, loop points) and ICMT (name) chunks.

✘

#### version 1.1

Uses Microsofts 3D controls extensions, CTL3DV2.DLL instead of Borlands BWCC.DLL as earlier.

Improved Ensoniq EPS support (.INS, .EFE, .GKH).  
Now treats the sample rate correctly as well as doing a coarse decay rate conversion.

Now supports finetune for a few formats.

Better at cleaning up temporary files after itself...

✘

#### version 1.0

Now also able to write WAV files in addition to PATches. Typing in an extension of '.wav' instead of '.pat' in the Save dialog box, will save a .DES text file with into about loop points and such, as well as .WAV files with the samples in the patch.

Converted the manual into WinHelp Hypertext format.

Reads INS and EFE files (Ensoniq EPS instrument files). These can be extracted from GKH files with a utility called EPSDISK.

Reads SDS files (raw Midi Sample Dump Standard files).

Reads SDX files ('compacted' Sds file + sample name, as saved by the program SDX).

Reads TXW (Yamaha TX16W wave files). Really named .W??.

Reads SYW (Yamaha SY-series wave files). Really named .W??.

Has not been tested (Please send me some files!)

Note! The much more common all-data format, .T?? are not supported (yet).

Reads SDK (Roland S-550, S-50, S-330 disk images).

Reads AIF files (Audio Interchange File Format).

Reads IFF: 8SVX files (Interchange File Format).

Reads UWF files (UltraTracker Wave files).

Reads PAT files (Ultrasound patch files).

Reads SMP files (Samplevision).

Reads VOC (Creative labs) files.

Reads AU files (Sun/NeXT); only uncompressed mono files supported.

Reads RAW signed data files; assumed 8bit, 11khz.

KRZ files now get their loop point. Now also supports split files (KR1 + KR2). No longer throws out samples if the instrument is too big but rather lets you do it yourself in the patch editor of your choice.

Correctly reads 16bit WAV files now.

Correctly reads ULT's with 16bit samples now.

Looped samples with the end loop point at or near the sample end now get the sample at the loop start point duplicated at the loop end point + 1, in order to keep the GUS interpolation happy (can cause clicks otherwise).

The envelopes now corresponds to the ones Patchmaker use for looped and unlooped waves, except that looped samples have a quite long decay but not at all as long as the long decay setting in Patchmaker.

Now using FMJULTRA.DLL to provide ULTRASND.INI manipulation for loading custom patches; email me if you'd like to use it in your own programs and I'll send you FMJULTRA.H.

Renamed MIDITHRU.DLL into FMJMTHRU.DLL. In order to avoid possible conflicts with other programs.

Various minor changes and improvements.

✘

version 0.9B

Changed program name to 2PAT (from MOD2PAT, after all it reads a lot more stuff than just MOD files).

Reads Kurzweil K2000 sample files (.KRZ); multisampled supported. Since the spec. is 'secret' I have only been able to find out where the actual samples are stored, how long they are, what pitch and what keys they are assigned to and names and such, but haven't been able to determine loop points, envelopes et c.

Reads .SND, raw unsigned data files. Assumes they are 11KHz, 8bit.

Hopefully the problem with samples > 64KB is fixed.

Saves temporary patch-files in the directory specified by the TEMP= environment variable or, if not available, in the program directory.

Faster updating of ULTRASND.INI.

Nicer look with BWCC.

✘

#### version 0.8B

Now a Windows hosted program (Win16).

The user interface is naturally quite different in this incarnation...

The .INI extract feature is no longer implemented, nor is reading raw files.

Patch play feature. If you've got an Ultrasound card (and you have, haven't you?) and a midi in device, you can play the patches from within mod2pat.

✘

#### version 0.7B

Removed the signed to unsigned conversion; Since we've now got Patch Maker Lite, which hasn't got the bug of the old PATCH.EXE.

The .INI extract now looks a bit differently.

I've introduced the concept of "BankName=Whatever" to give the banks names; it will be set to the songname.

Added my bird logo to the text font.

✘

#### version 0.6B

Fixed another embarrassing bug, when converting from signed to unsigned data I now add 80h and not 7Fh as previously (got a little noise...).

✘

#### version 0.5B

Yes, now you can 'hear-out' the samples before you decide whether to save them. Of course, an ultrasound card is required for that!

Fixed a rather embarrassing bug with MOD loop points.

✘

#### version 0.4B

Reads 669, STM and S3M.

Always saves as unsigned, 'this because Gravis PATCH.EXE always uses unsigned data and it's convert command doesn't convert the last byte...

Converted source code into C.

✘

#### version 0.3B

Reads MTM, ULT, WAV, FSM and raw files.

Ability to edit internal patch names (Use TAB to toggle between DOS-name and Patch-name edit).

Added GUSPATCH.TXT; Ultrasound patch format description.

Any MOD that isn't a M.K., 4CHN, 6CHN, 8CHN, FLT4 or FLT8 are now treated as being of the older 15 (versus 31) sample format.

✘

#### version 0.2B

Reads 4,6,8 ch MOD's 'M.K.', '4CHN', '8CHN'.

Basic user interface.

Smart 'auto filename'.

✘

#### version 0.1B

Nothing fancy, reads 4ch MOD's, writes PAT's, ULTRASND.INI extract.

✘



