SAC64 Latest Release Info

Introduction

Welcome to SAC64. This 64 Bit environment offers an exciting new vision from Bob Lentini. The interface is built around the idea of a complete Live Virtual Front-Of-House Console and 24 Complete Virtual Monitor Consoles, all wrapped up in one program to offer an entirely new way of manipulating live audio in the virtual world.

Please take some time to explore the built-in HelpFile and also download the .pdf version of the manual, available at ***www.RMLLabs.com***. The navigation capabilities are powerful and complex and will require some getting used to. Reading, exploring and practicing inside the environment will greatly enhance your experience of the interface. The rewards gained for the time invested, will be great. Have fun!

Version History Follows:

>====<

Version 2.1

Enhancements

Bug Fixes

* The VST3 .ini scanning code has been modified to allow and detect plugin names that include a single or multiple periods in the .ini path data line. This helps solve waves loading issues and other plugin installs that may use periods with version numbers in the path data line

>---<

Version 2.0 ($100 Paid Update - Jan 2026)

Enhancements

* The introduction of VST3 Plugin support. It took a while, but its finally here. VST3 Plugins can now be loaded directly into the program with the use of VST3 .ini support files. This code can also support Waves VST3 Shell loading.

  While the use of .ini file links remains optional for VST2x plugins, although recommended, they are now required to use for VST3 plugins. The .ini file is a simple text file that links the actual plugin installed path to the SAWStudio64 interface.

  Note that if you are using Windows 11, the new notepad program no longer creates simple ascii text files by default. I recommend downloading a free notepad replacement called "Notepad++".

The use of the .ini file eliminates the need for the program to scan harddrives looking for and collecting all locations of installed VST3 plugins and also allows you to control which plugins appear in the FX Plugin listboxes in SAWStudio64.

The VST3 Plugin programming interface is much more complex than the older VST 2x versions and for the most part now attempts to control default installation paths in a more restricted manor. In most cases, the default install path in Windows for a VST3 plugin can be found at "C:\Program Files\Common Files\VST3". Within that folder, you may find more sub-folders, depending on the various companies involved, which expand out further until the actual .vst3 plugin progaram file is finally found. Although this is a common default behaviour, VST3 plugins can and many times are installed in many various folders, depending on the companies involved.

For instance, a UAD LA3A default install path might look like this:

C:\Program Files\Common Files\VST3\uaudio_la3a.vst3\Contents\x86_64-win
\uaudio_la3a.vst3

This can get quite complicated to keep track of when you might have dozens or more different VST3 plugins installed from many different companies. The .ini files keep things neatly organized. Each plugin requires its own .ini file that points directly to the actual installation path and VST3 file. The .ini file is a simple text file created in the notepad or similar text editor which has one line of text stating the complete install path, including the actual .vst3 plugin file itself. The .ini file is saved with its name being the actual name that will display in the FXChoices list of all available plgins. This file is saved in the VST_Plugins folder which is under the main SAWStudio64 install folder.

As an example, the UAD LA3A compressor listed above would have an .ini file that could be named "UAD LA3A Compressor.ini" saved in the "C:\SAWStudio64
\VST_Plugins" folder. The .ini file would contain the one line of text as follows:

C:\Program Files\Common Files\VST3\uaudio_la3a.vst3\Contents\x86_64-win
\uaudio_la3a.vst3

In the FX Choices listbox in SAWStudio64, you would see an entry labeled
"UAD LA3A Compressor"

Waves plugins introduced the Shell concept to an already complex system of design. The Waves plugins use a shell file that contains many different separate plugins wrapped inside the shell file. This is handled slightly differently in SAWStudio64, in order to maintain compatibility with the way presets and sessions are saved and recalled.

In general, the shell file will install to the default location as defined above, but the

individual plugin files will install in a Waves independent scheme of folders and locations, generally not directly available to the host interface. To handle this, we use a slightly modified version of the .ini file form. The main shell file can be opened for reading from the FXChoices listbox, but to use the plugins, we need to create a separate .ini file for each plugin we want to access.

So, first you would create an .ini file for the shell itself, which you would probably find at:

C:\Program Files\Common Files\VST3\WaveShell1-VST3 16.0_x64.vst3

Now when you create the .ini file for this shell file, you will change one thing to denote this as a shell and not an actual plugin link. Create the .ini file and name it "Wave Shell v16.ini" or something similar and on its one line of text within the file, add an "s" onto the .vst3 at the end of the line so it would actually look like this:

C:\Program Files\Common Files\VST3\WaveShell1-VST3 16.0_x64.vst3s

This lets SAWStudio64 know that this is a shell file and not a plugin link. When you select it in the FXChoices listbox, it will list out all plugins that are contaned within it. You might find many different versions of each plugin you purchased, such as:

WLM Plus 5.0
WLM Plus 5.1
WLM Plus Mono
WLM Plus Stereo
WLM Meter 5.0
WLM Meter 5.1
WLM Meter Mono
WLM Meter Stereo

This is what gets installed when I only purxhased the WLM Meter. You should make a note of each plugin name that you wish to use. You must match the name exactly as you create a separate .ini file to link to the plugin from the shell file. For instance, I only need access to the WLM Meter Stereo plugin, and do not need to clutter my FXChoices listbox with all the other versions that I might never use. The actual .ini file might be named "Waves WLM Meter Stereo" and would contain the one line text path plus the extra shell information needed to extract the proper plugin from the shell:

C:\Program Files\Common Files\VST3\WaveShell1-VST3 16.0_x64.vst3s WLM Meter Stereo

Notice the "s" added to the shell filename to properly access the shell file and then the actual name as listed inside the shell of the actual plugin file. This .ini file will actually load the plugin listed from within the Waves Shell mechanism. Do this for

each Waves plugin you wish to use from the shell file. I recommend leaving the shell .ini file in the FXChoices as each time you purchase new Waves plugins they could be appended in the same shell file and you will need to list it again to note the actual new plugin installed names. But note that depending on plugin versions, Waves could create a different new shell file that handles only certain plugin versions you purchase, so you may have to create a new Waves shell .ini file for that just so you can see the actual included plugin names.

Hang in there getting thngs all configured and setup. In the end, you will have more control in what gets listed in your FXChoices Window and you can control exactly what and how it displays in a less cluttered Plugin selection view.

* Added two new ASIO options under the *Options/Audio Driver Model Menu* selection called *Asio Alternate Out* and *Asio Alternate Out Driver Setup*. The Asio Alternate Out option allows you to select a second Asio Device Driver to route all output signals to. This now allows you to accept inputs from one Asio Device and send outputs to a different Asio Device. Extremely useful for Live Streaming setups where you might not have access to assign your output monitor and Live Stream mixes to physical output connectors on a FOH mixing console which is feeding your input signals through an Asio driver. When connecting to physical consoles with Dante, Madi or AES50 network audio drivers, you generally can only receive and send audio through one common driver. You must first select a primary Asio driver which would normally handle all your input and output audio. Then you will be able to select a secondary Alternate Output Asio driver to handle just the audio output signals. You should also attempt to sync both devices to the same hardware clock, otherwise there could be drift between them if the two clocks are not exactlty synchronized.

* Added an EQ Graph display to the Touch Mixer View Eq Section. This now displays the eq settings as graph curves showing freq, level and bandwidth of each band plus a composite overall resulting curve.

Bug Fixes

* Fixed a bug in the Midi Control Templates that could cause a crash with certain templates when using the "Jump To Prev/Next Section function.

* Fixed a bug that could cause a crash when touching the rotary 8 control with a Mackie Midi Control Template engaged.

* Fixed code to correct a possible crash when connecting a remote to the host with an empty session in the host.

* Fixed Win8 MultiTouch code that was broken when I changed my development compiler version.

>---<

Version 1.4

Enhancements

Bug Fixes

* Fixed Channel Key trigger to properly key from any channel selection.

>---<

Version 1.3

Enhancements

* Added Soft-Clipping code into the Attenuator routines to prevent wrap-around clipping when recording in SAWStudio with the SACLink and adding gain to the channel input with the attenuator control.

Bug Fixes

* More work done to the main engine threads to compensate for current asio driver changes from various manufacturers that could break the asio driver compatibility with the previous engine thread modifications.

>---<

Version 1.2

Enhancements

* Modified the main engine threads to make the engine more immune to slipped buffers. The new design makes it harder for Windows background processes to step on the main engine threads, therefore making the code more stable and less susceptable to buffer slippage.

Bug Fixes

* Fixed the Browse For Folder dialog to correctly enable and disable the combobox options.

>---<

Version 1.1b

Enhancements

* Expanded the max Output Channels for VST Plugins to 64 which should help with compatibility for plugins like Kontakt 6 and others.

Bug Fixes

* Adjusted code to handle even larger sizes for VST plugins data storage.

>---<

Version 1.1a

Enhancements

Bug Fixes

* Fixed code to allow vst compatibility with plugins that depend on accurate vstTimeInfo data.

* Fixed code that caused Soft-Clipping to fail.

* Adjusted code to compensate for VST plugins that store very large amounts of data with their settings in edls and other file saves. This code should now fix some plugins ability to save settings with session and other template files.

>---<

Version 1.1

Enhancements

* The Touch View EQ section now displays the EQ graph curves. Each band is represented as well as a composite curve that displays the interaction between overlapping bands and shows the resulting curve.

* Changed code to drastically improve performance on Win 7, 8 and 10 for all shared SACLink operations between SAC and SAWStudio. CPU loads when using the SACLink are no longer heavily increased as data is passed between the two programs during recording and playback. This now matches, or even exceeds the SACLink performance that was achieved in XP.

* The helpfile now includes specific Win 10 tweaks.

Bug Fixes

* Fixed code that could overrun a few array variables when Max Count input and output devices are used.

>---<

Version 1.0d
Enhancements

Bug Fixes
* Fixed code to correct Output Group Out Assignment to direct the audio output from the group to the proper Master Out.

>---<

Version 1.0c
Enhancements

Bug Fixes
* Fixed code to correct Soft Clipping.

* Fixed code to correct certain internal Data Structure Offsets.

* Fixed code in Remote to correctly display the current Load Percentage.

* Fixed code in Remote to correctly handle Meter Peak clearing.

EQ Plugin Ver 1.0b
* Fixed code to correct an asm string routine.

Echo Plugin Ver 1.0a
* Fixed code to correct an asm string routine.

>---<

Version 1.0b
Enhancements

Bug Fixes
* Fixed code which could prevent Monitor Mixer output under certain conditions.

>---<

Version 1.0a

Enhancements

Bug Fixes

EQ Plugin Ver 1.0a

* Fixed code to correct Band 7 from shifting audio data left.

>---<

Version 1.0

Enhancements

* This first release includes all features and fixes from the 32 bit version thru version 4.5.

Bug Fixes

>---<

>====<