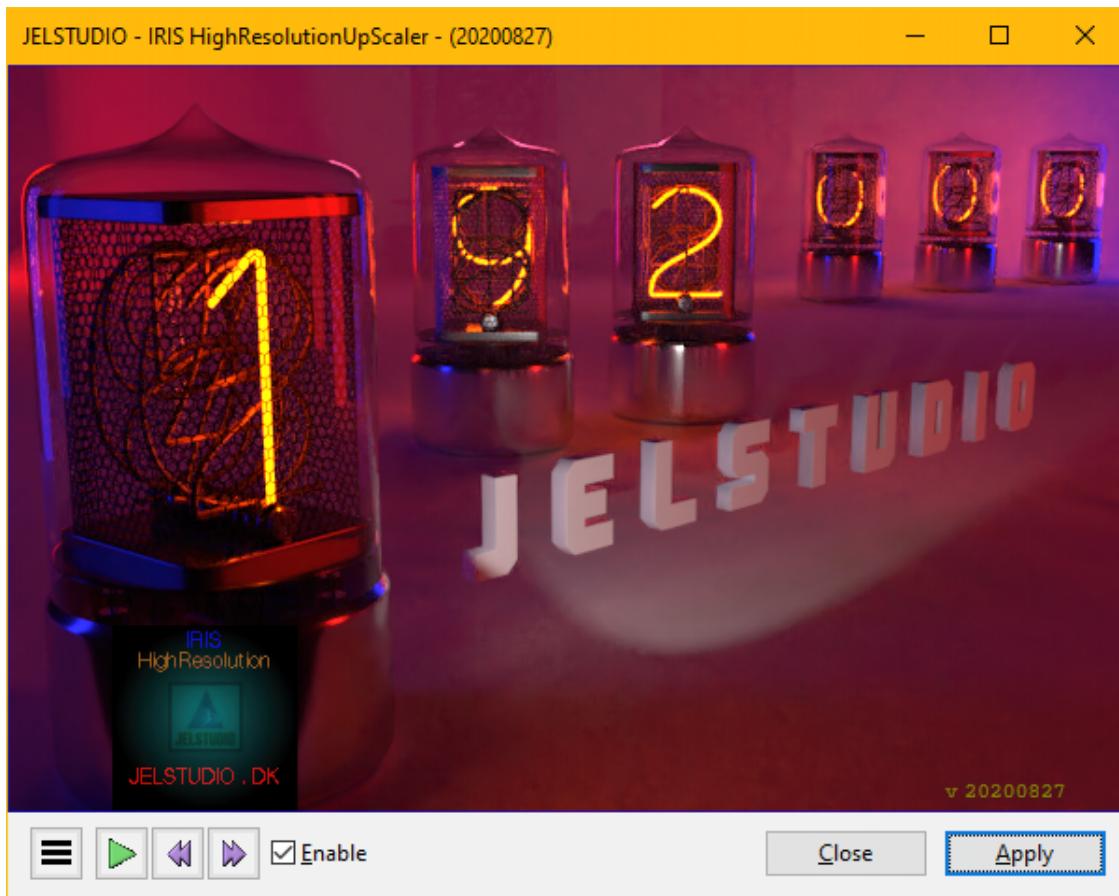


JELSTUDIO's "IRIS" (High Resolution Up-Scaler)

version 20200827



IRIS is a spectrum-resolution up-scaler for pre- or post-processing use.

Use IRIS to up-convert Low-res audio-files or samples to Hi-res audio-files (Tested up to 192 kHz, but has no technical upper-limit so may work at even higher rates)

Sample-based musicians can enhance their sample-library to allow for wider playback-ranges before risking aliasing.

Musicians/singers who recorded/mixed/mastered their music at LoRes can convert it to HiRes to sell on HiRes streaming-websites.

Music-lovers/Sound-engineers can increase resolution of LoRes recordings to HiRes for listening pleasure or further post-processing.

Technically it is a 32/64 bit VST2 audio-effect plugin for use in DAWs, sound/video-editors or VST-hosts.

Quick-use:

User interaction:

IRIS has 1 user-function (ByPass effect) controlled by left-clicking anywhere on the GUI with the left mouse-button.

As long as the mouse-button is pressed, the IRIS sound-effect will be by-passed (The incoming raw un-affected audio can then be heard instead) and all the tubes on the GUI will show as OFF (The tubes will be dark)

As soon as the mouse-button is released the IRIS sound-effect will be ON again.

Information display:

The GUI has 6 tubes which show which sample-rate IRIS is currently detecting and using.

The tubes can show the following sample-rates:

“044100” (44.1 kHz)

“048000” (48 kHz)

“088200” (88.2 kHz)

“096000” (96 kHz)

“192000” (192 kHz)

“384000” (384 kHz)

“000000” (X kHz)

When using any other sample-rate than those just listed above here (Such as, for example, 44099, or 48001, or 176400, etc), IRIS will display “000000” (All zeroes).

This does not indicate a problem, but is merely a graphical limitation. The audio-engine in IRIS is technically samplerate-agnostic.

The JELSTUDIO logo is 'breathing' (oscillating in brightness) to give visual indication the plugin is still running correctly and has not 'crashed'/malfunctioned.

The IRIS version-number is shown at the bottom-right side (vYYYYMMDD)

Step-by-step Use in Reaper and Audacity:

Reaper:

- 1: Set your audio-device to the desired output sample-rate (Look to confirm this on the top-right of the Reaper-window. It should read the sample-rate you want to up-scale your music to. If you want to upscale music from 44.1 kHz to 96 kHz, it should read 96 kHz)
- 2: Load IRIS on the track where you have the music you want to up-scale, or on the master-track (You can also load a spectrum-plugin to check what IRIS does)
- 3: Export your music at the desired bit-depth and Sample-rate (You set these in the export-dialog)
- 4: Check your export was done correctly, at least until you get comfortable with the procedure, with software such as "Spek" (<http://spek.cc/>)

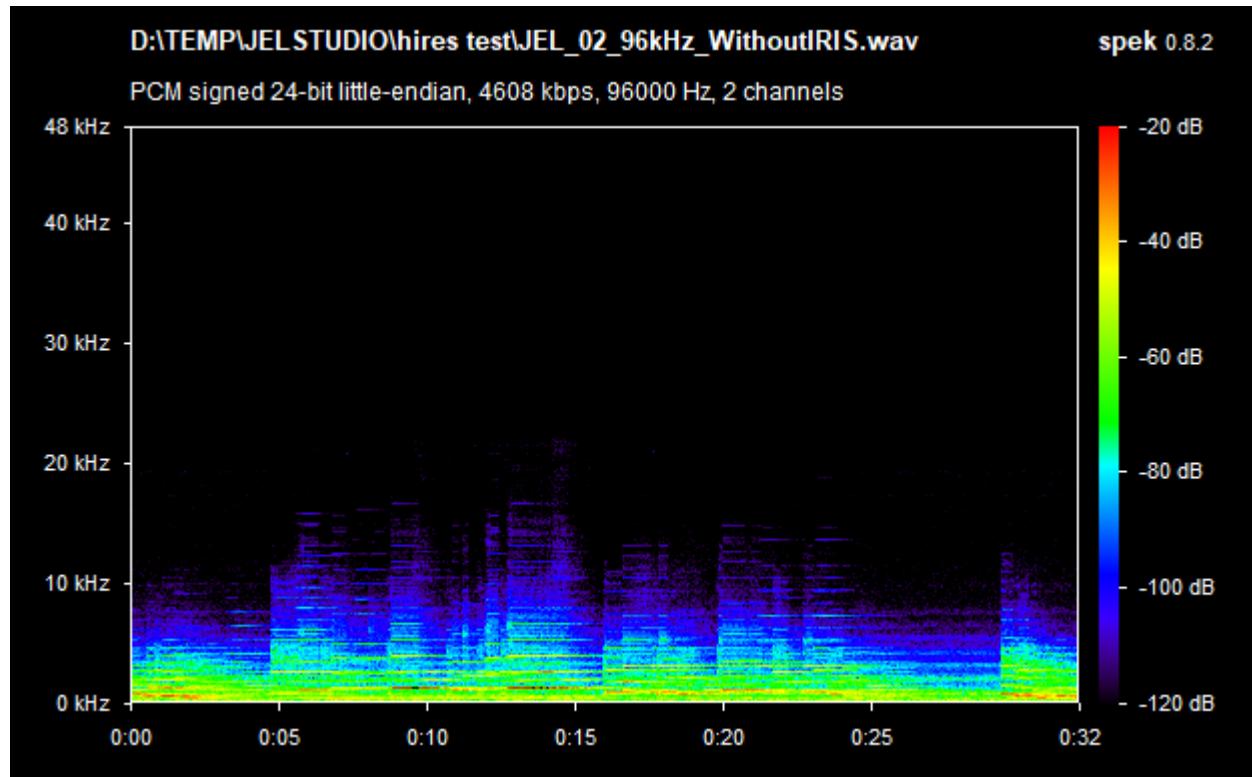
Audacity:

- 1: Import the music you want to upscale.
- 2: Set your 'project-rate' (You set this in the lower left of the window) to the desired output sample-rate. It should be the sample-rate you want to up-scale your music to. If you want to upscale music from 44.1 kHz to 96 kHz, it should be set to 96000 Hz)
- 3: ReSample your track to the desired sample-rate (Click top-menu: 'Tracks' → 'ReSample')
- 4: Load IRIS as an effect (It may initially show an incorrect sample-rate on the GUI. Ignore this.)
- 5: Make sure you 'high-light' the track you want to convert (Either double-click the sample-window or single-click the area to the left of the sample-window)
- 6: Begin playing the track (Press 'spacebar' to start/stop play-back). You only need to play a second or so. This is just done to get IRIS to 'see' the desired sample-rate (And update its internal settings to this sample-rate), which should now be shown on IRIS' GUI (If you want to up-convert to 96 kHz, IRIS' tubes should show 096000 on the GUI)
- 7: Click 'Apply' on IRIS (This button is located at the bottom-right of the window)
- 8: Export your audio from Audacity (Click top-menu: 'File' → 'Export')
- 9: Check your export was done correctly, at least until you get comfortable with the procedure, with software such as "Spek" (<http://spek.cc/>)

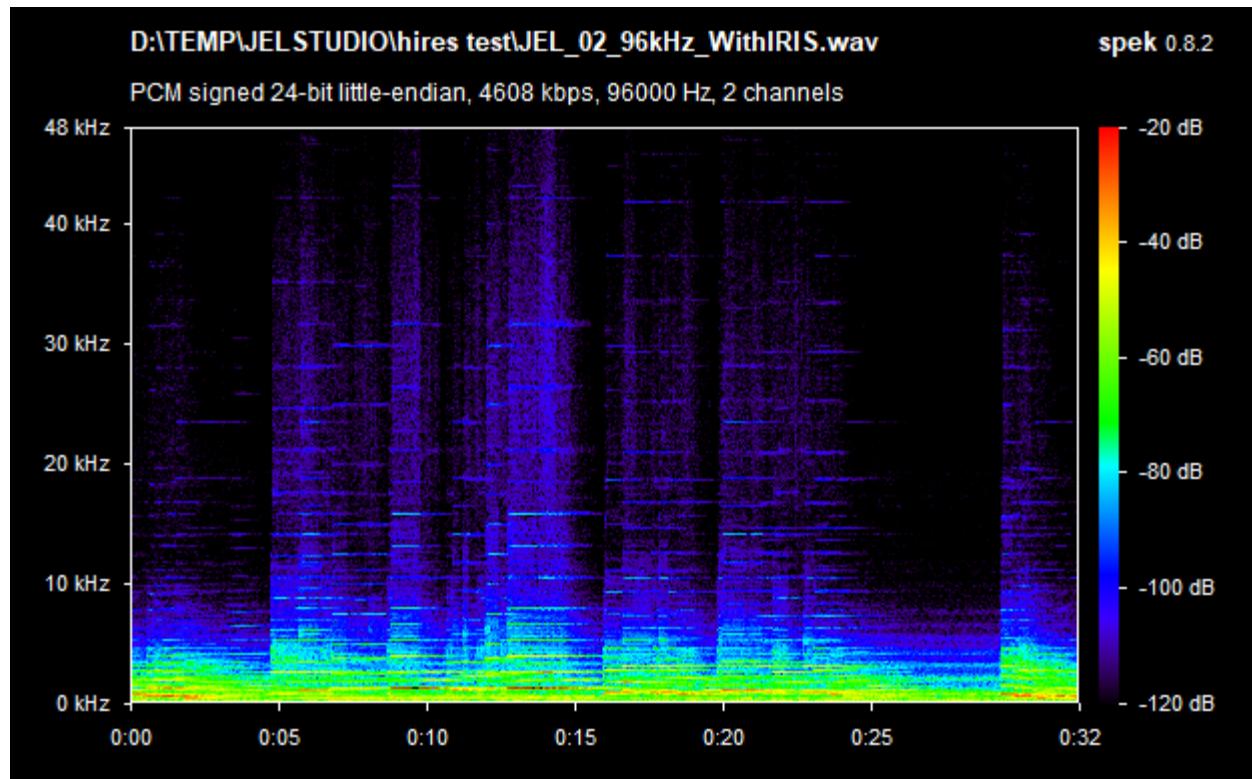
Examples:

Here are a couple of examples of what you should see in “Spek” (<http://spek.cc/>)

First an image of ‘test-sound 02’, converted from 44.1 kHz to 96 kHz **WITHOUT** IRIS, where we can see there is no sound above ~22 kHz:

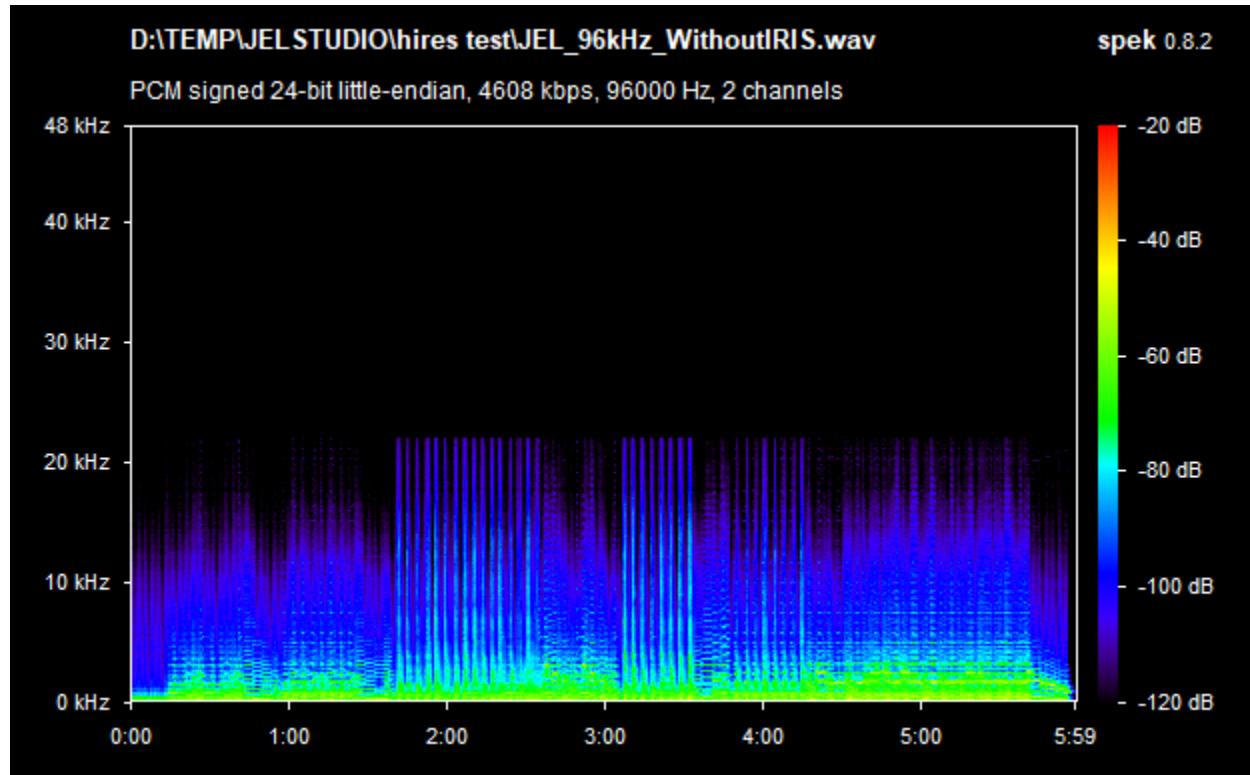


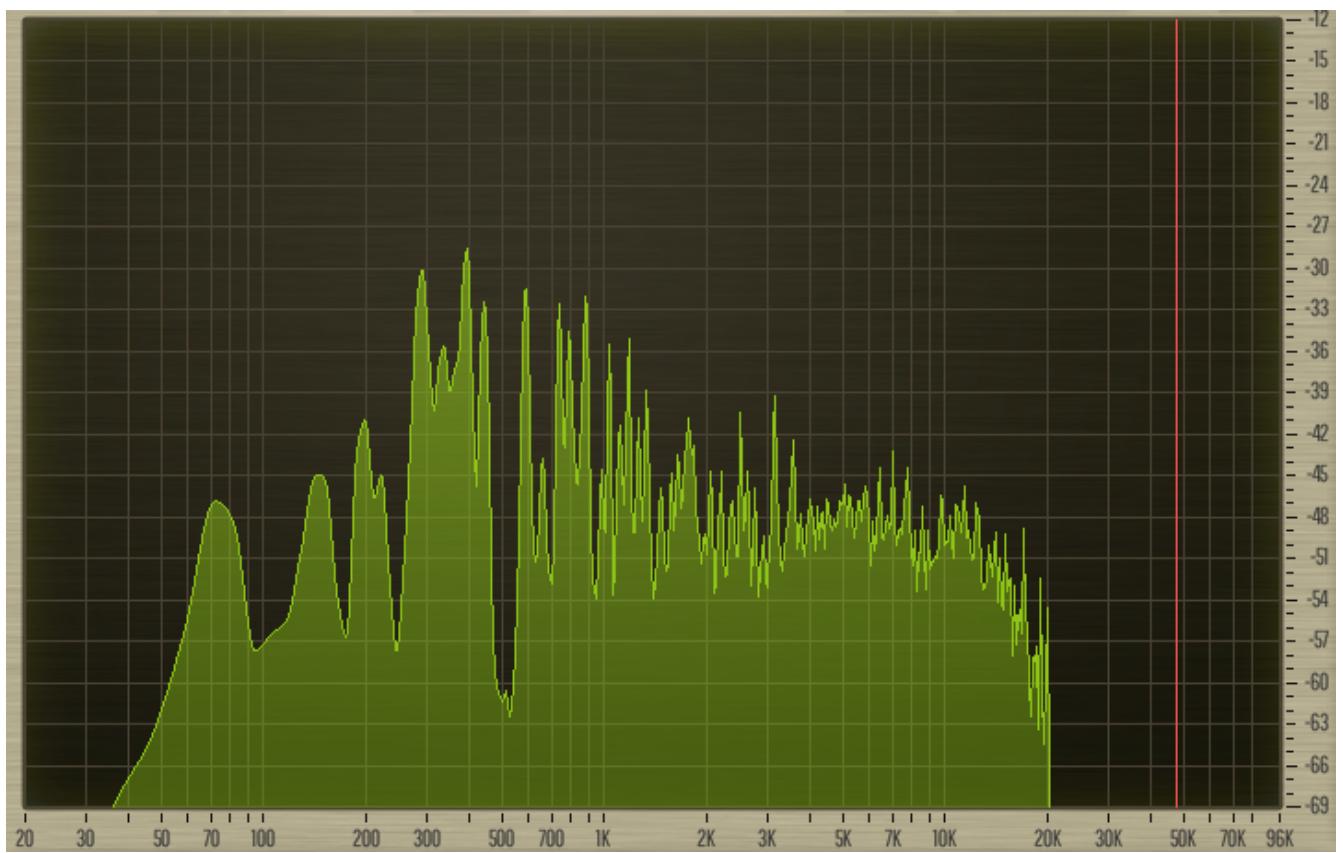
Then an image of the same ‘test-sound 02’, converted from 44.1 kHz to 96 kHz but this time **WITH** IRIS, where we can see there is now sound (Both tonal and transients) above ~22 kHz:



Second we will take a look at two images of ‘test-sound 01’, converted from 44.1 kHz to 96 kHz **WITHOUT** IRIS, where we can see there is no sound above ~22 kHz.

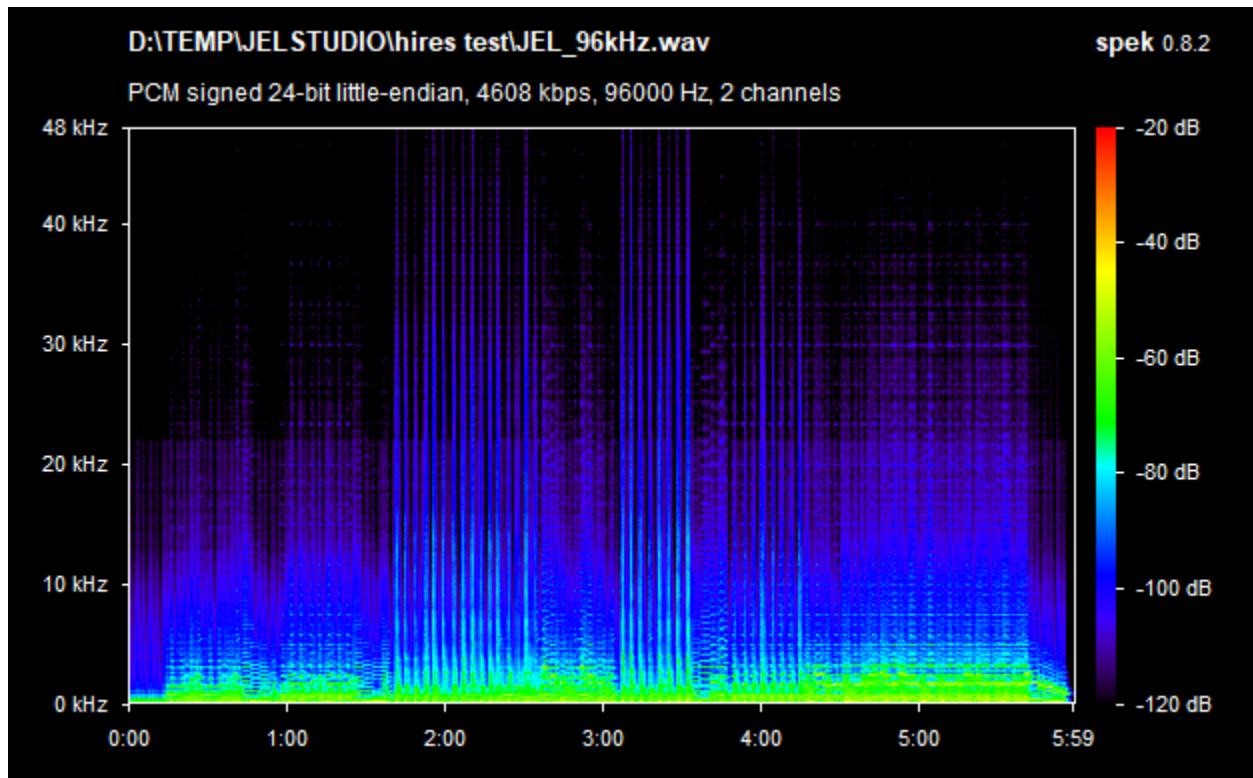
We can also see this song is compressed quite hard (It is always better to use IRIS on non-compressed music, preferably on the initial low-res recordings/samples before the mixing and mastering stages, but if the final low-res master is all we have we obviously have no choice):

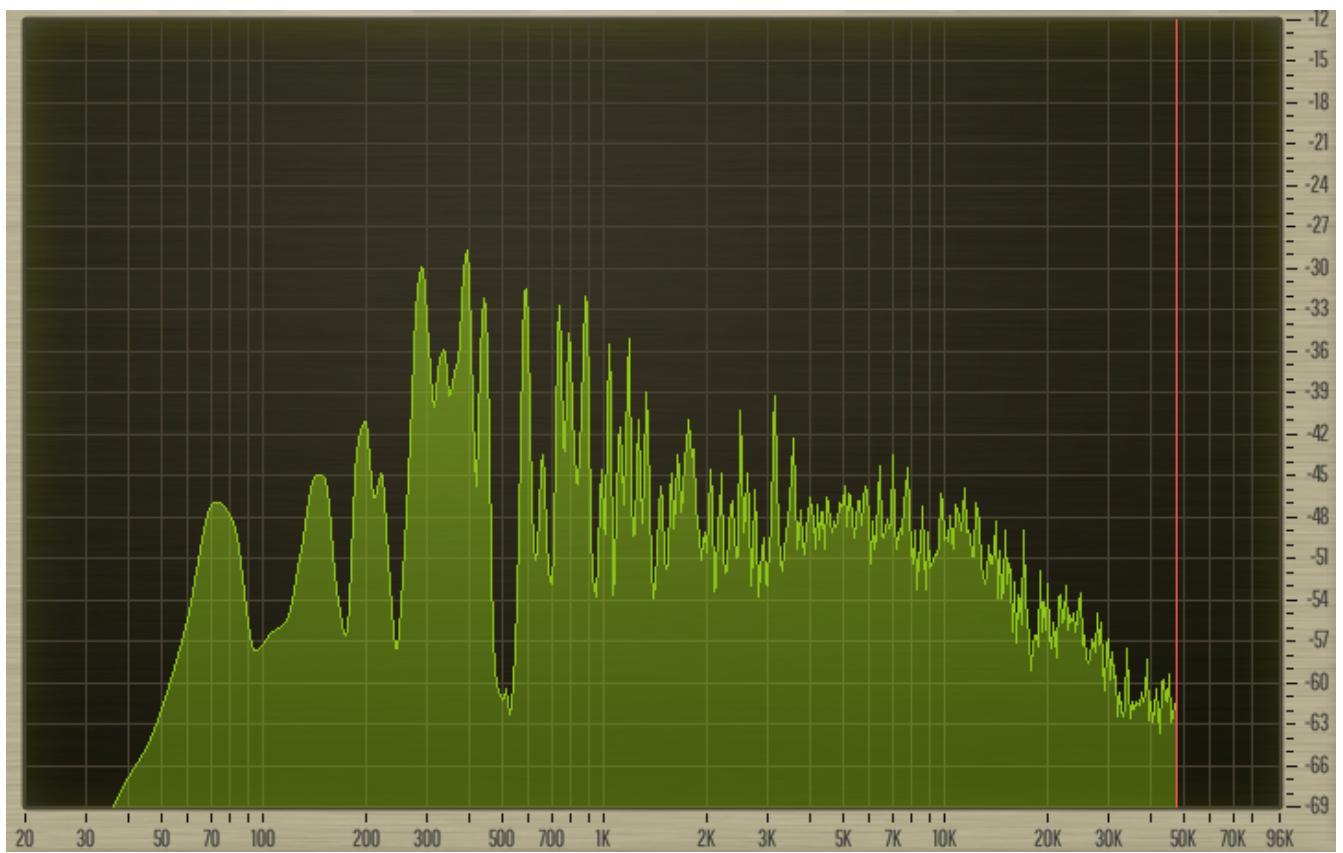




Then two images of the same ‘test-sound 01’, converted from 44.1 kHz to 96 kHz but this time **WITH** IRIS, where we can see there is now sound (Both tonal and transients) above ~22 kHz.

We can also see the hard compression has been masked out fairly well by IRIS, but not completely (So, again, it is always better to use IRIS as early in the music-production-phase as possible):





Some of IRIS' key-points:

- No theoretical upper sample-rate limit (But only tested up to 192 kHz)
- For Windows 7, 8, 10.
- Zip with .dll (No .exe installation. Good for portable use)
- No DRM (Such as PhoneHome, iLok, etc)
- Output-level generally matches input-level.
- Works on mono and stereo signals.

Technical description:

IRIS operates on the principle of ‘sympathetic resonance’, mixed with ‘transient response’. This means it can handle both tonal and percussive sounds.

IRIS is inspired by AI-based image-resolution enhancers, but is not AI-trained (Although Python-based AI-software for audio sample-rate up-conversion was studied during the creation of IRIS, those approaches still created too many artifacts to be really useful, or to be even stable with different audio-material. At some point in the future such methods may surpass the method used in IRIS of course.)

This means 2 things:

1: IRIS will NOT produce the same result every time you play the same sound-sample, but ‘think up’ an artificial suitable high-end construction to mix into your audio (This gives the sound a more ‘organic’ quality, and is the part we want from the AI-method)

2: IRIS will NOT create the known ‘random’ AI-based artifacts, which AI-systems create when they ‘dream up’ a ‘wrong’ dream about the content they work on (This leads IRIS to give stable predictive results that are always musically fit, but obviously slightly less organic/random than when an AI-system ‘gets it right’.)

The idea for IRIS came after the incident where the Golden Gate bridge was ‘singing’ in the summer of 2020. If you have not yet seen the videos of this incident, search for them online and treat yourself to a most remarkable sonic experience. An immensely beautiful phenomenon risen from the mix between nature and human creation.

END OF TEXT

Website: https://jelstudio.dk/VSTplugin_AutoLoudness/

Email*: jelstudio@hotmail.com

Twitter: <https://twitter.com/JELSTUDIO>

Facebook* (personal to Jacob Larsen): <https://www.facebook.com/jacoblarsen.248>

General DSP programming: Jacob Larsen

Loudness-mode algorithm-programming: Jacob Larsen

GUI programming: Jacob Larsen

VST compiler: Derek John Evans

BETA-tester and sound-quality inspector: Sébastien Wittebolle.

*If you receive 'weird' nonsense/spam email from this account, it is NOT sent by JELSTUDIO!

Feel free to use Twitter/Facebook instead.

(When using Facebook do NOT accept game-invitations or other 'crap' like that!)