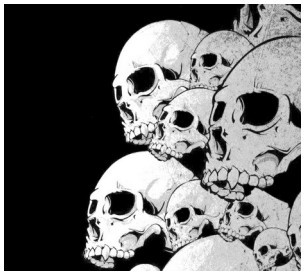
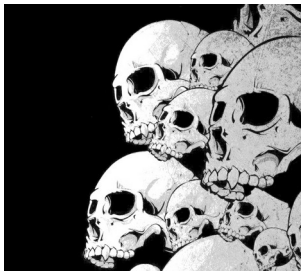


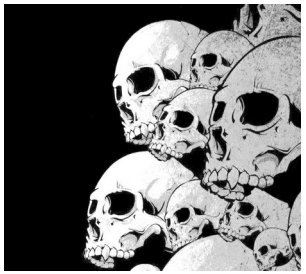
Guitarix

Y. Collette (ycollette.nospam@free.fr)
https://audinux.github.io

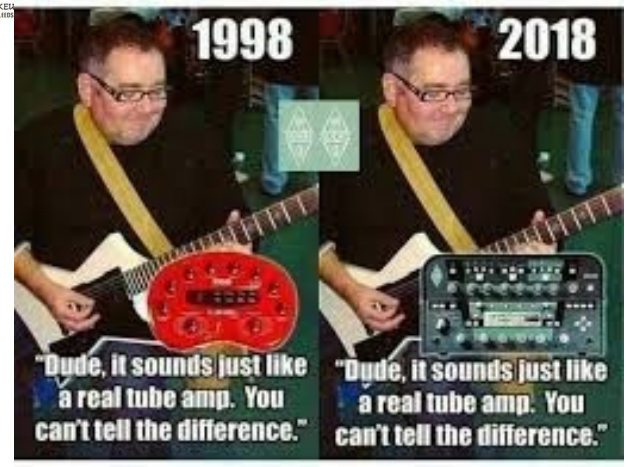


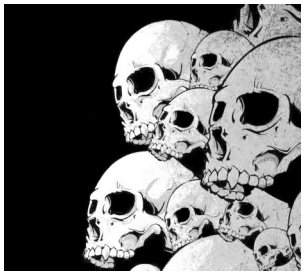
Guitarix

https://guitarix.org/



Some old ads





Some old promises

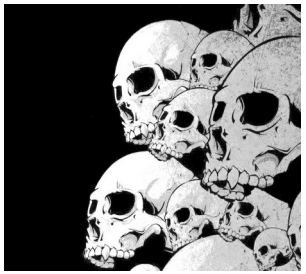
In the 1980s, the "death of tubes" was heralded with transistor distortions, the Rockman, and later the SansAmp—a promised revolution, but tube amps never disappeared.

In the 2000s and 2010s, Line 6 PODs, then Kemper/Axe-FX, were heralded as "the death of the amp and pedalboard"—again, they mostly added a new palette of tools.

Today, with NAM, AIDA-X, Neural DSP, etc., the narrative is back: "plugins/AI replace everything"... but amps, pedals, and open-source solutions like Guitarix + IR + LADSPA/LV2 under Linux continue to offer excellent sounds.

“For 40 years, every new technology has promised to kill tubes and pedals… Yet, the good sounds were already there. What’s really changing is the accessibility and diversity of tools.”

“No need to wait for the next revolution: with Guitarix, IR and free plugins under Linux, we already have the tools to sculpt a professional sound.”



Guitarix
General appearance

Activation of different windows

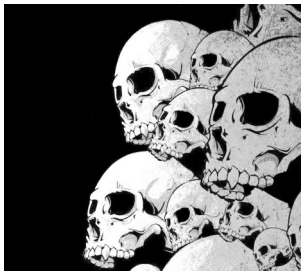
Effects available

General parameters

Tuner

Effect rack

Presets



Guitarix
The buttons bar

Pool: activates / deactivate the display of the effect list bar.

Order: Reduces the height of all effect modules and allows you to easily click and reorganize these effects. This view is practical to reorganize large amounts of effects.

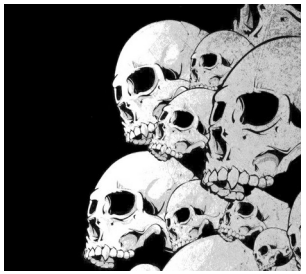
Tuner: Activates / deactivate the tacking of the tuning rack.

Rack: Activates / deactivate the Display of Effect Modules Rack.

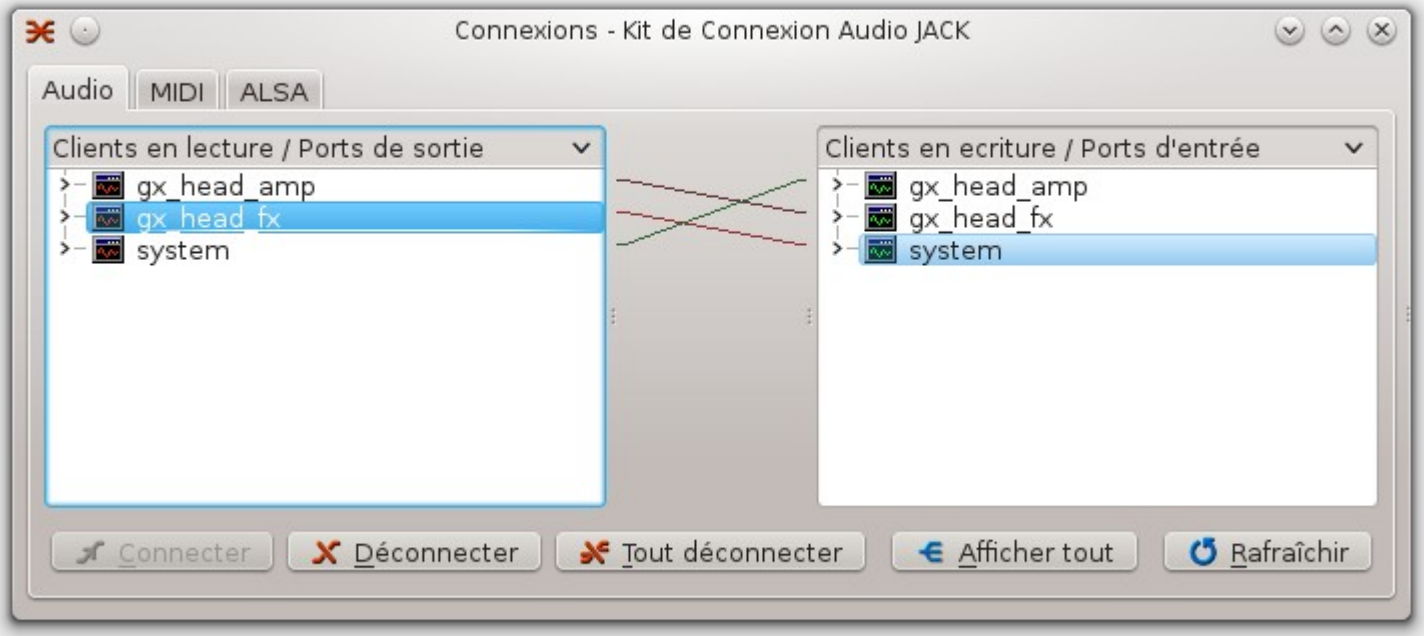
Split: displays the modules rack in two columns, mono grafts on the left column and stereo grafts on the right column.

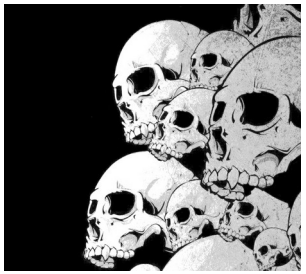
Fold / Show: the button less reduces the height of all effect modules. The button plus dilates.

Live: Displays a minimum full screen poster with preset information. Practical for live performance.

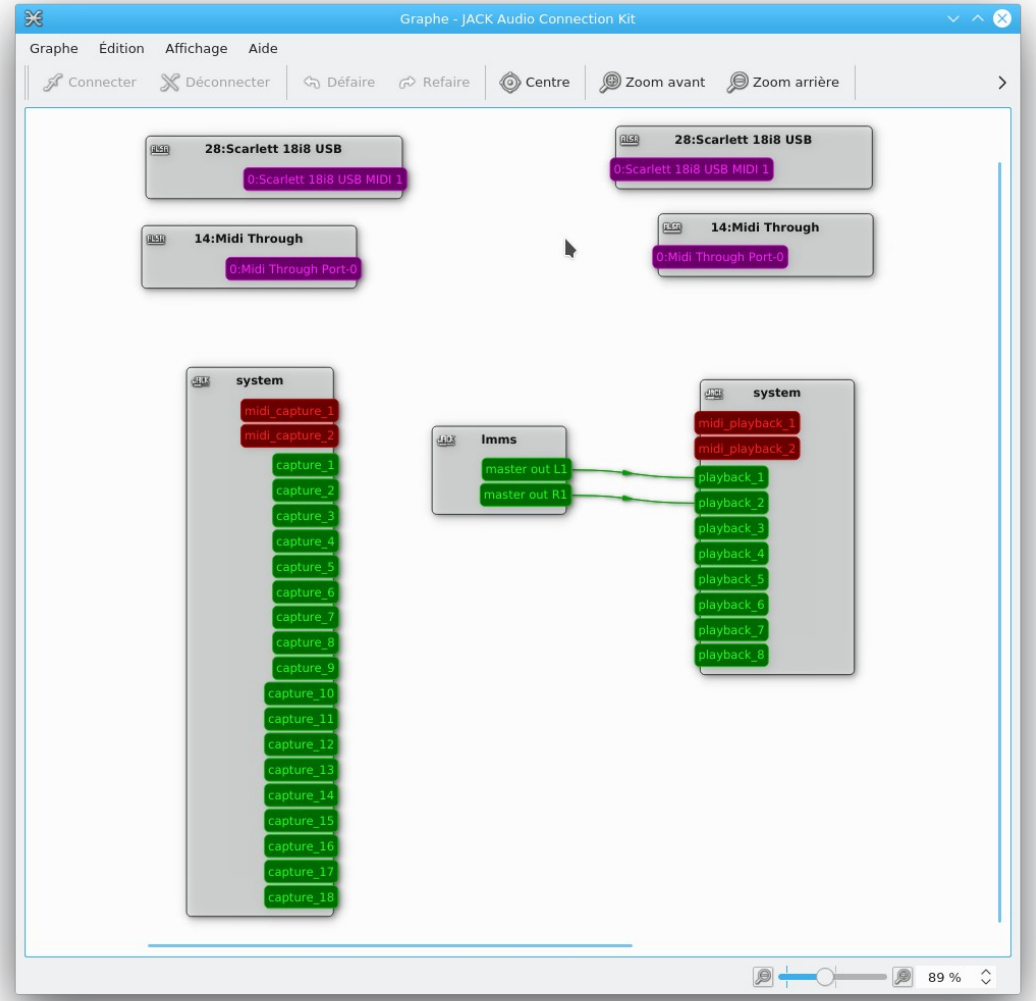


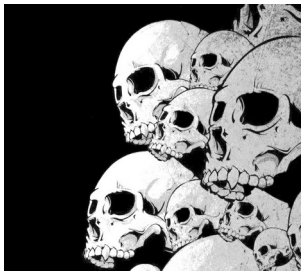
Guitarix
Jack side



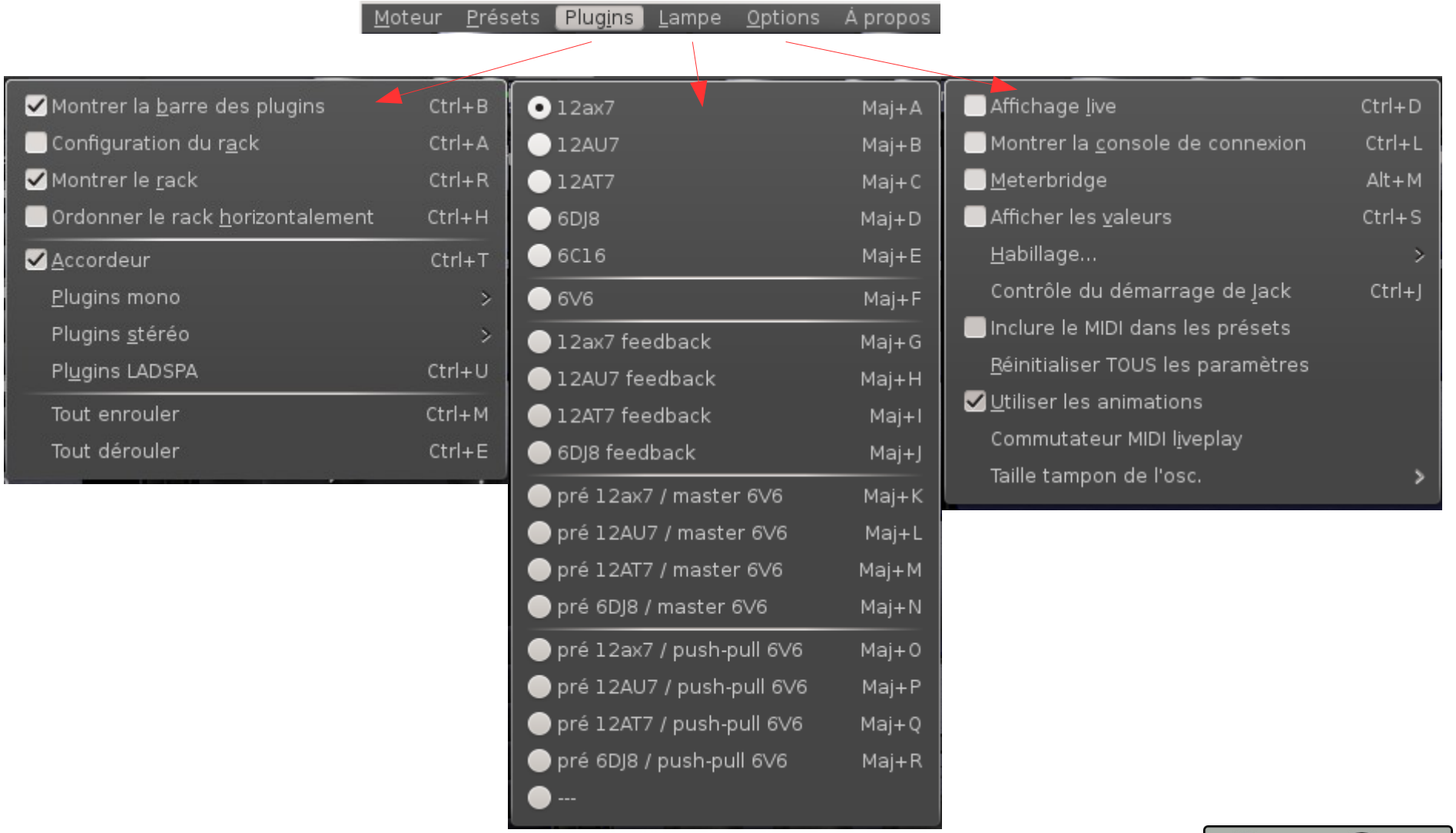


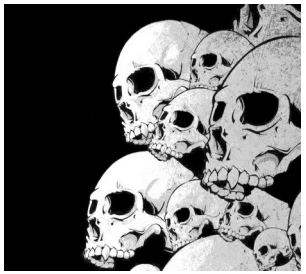
Guitarix
Jack side - New connection window





Guitarix
The Menus



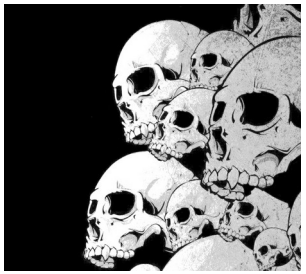


Guitarix
The structure of a rack

Pre-amplifier effects

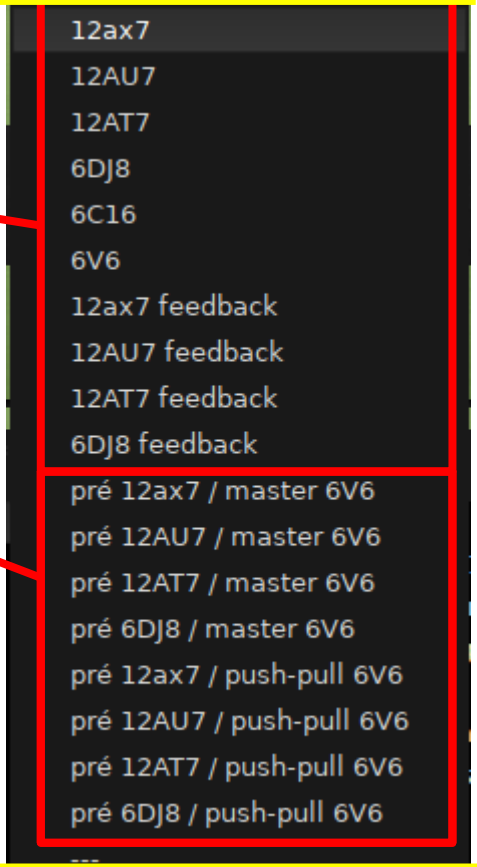
Amplifier

Post-amplifier effects



The tube amp emulator

Setting with lamp choice just for the amplifier



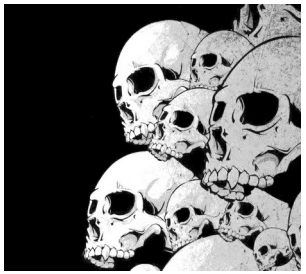
General volume

Sound characteristics settings

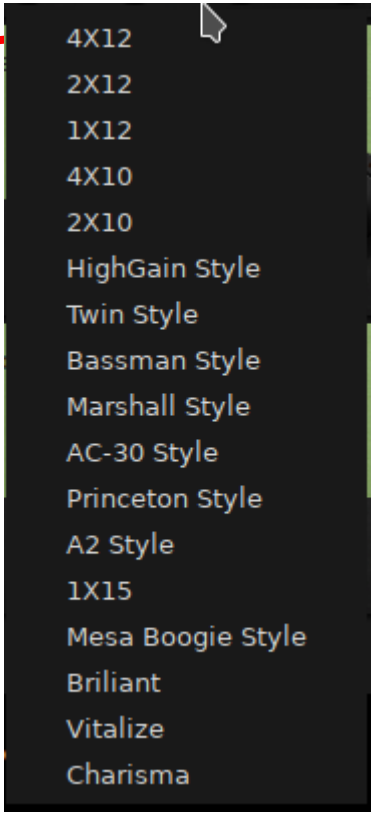
Mixture between signal with and without distortion

Signal gain

Setting with choice of lamp for pre-amplifier / amplifier



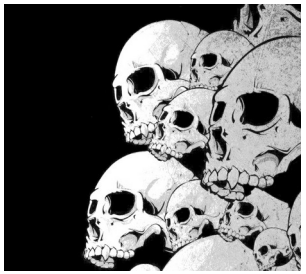
Cabinet



A cabinet is a baffle simulator.
It allows you to simulate different types of amplifier or a speaker combination.

Cabinet 4x12

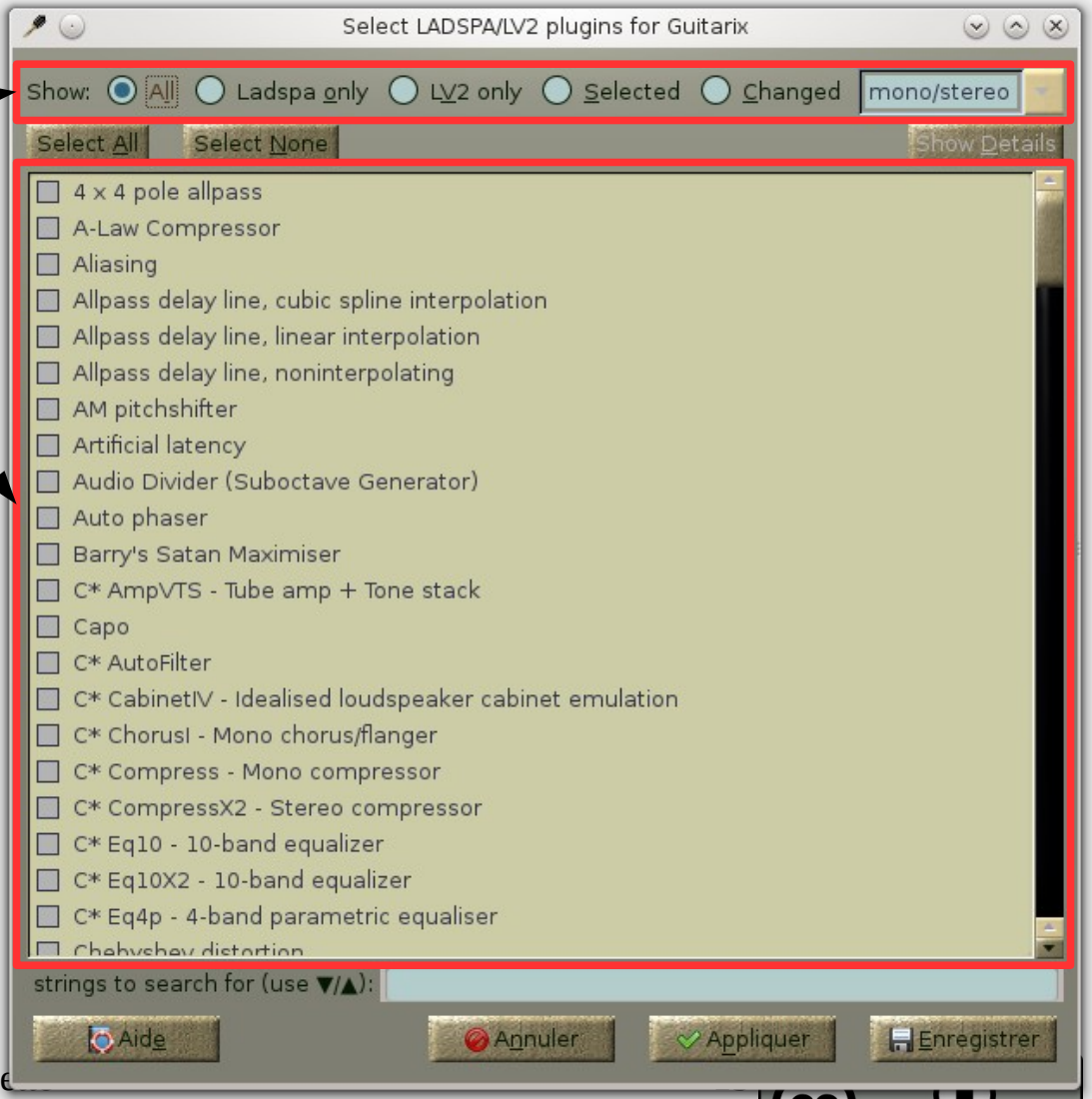
Cabinet Princeton style

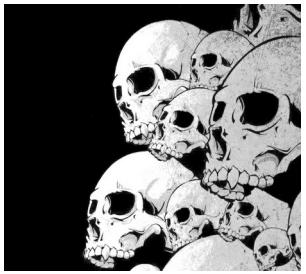


Plugin management LADSPA / LV2

Plugin type selection area (LADSPA / LV2)

List and selection of plugins to add to rack





Plugins for guitar

Interesting plugins for the guitar:

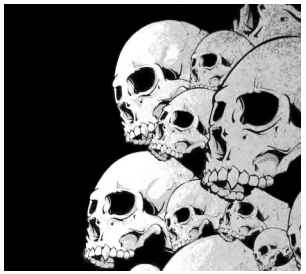
https://portalmod.com/

A distortion in LV2 format from Portalmod:

$ git clone https://github.com/portalmod/mod-distorition.git
$ CD MOD-DISTORTION
$ make # compilation
$ ./mod_build # Installation in the user directory
$ dnf installer mod-discontion

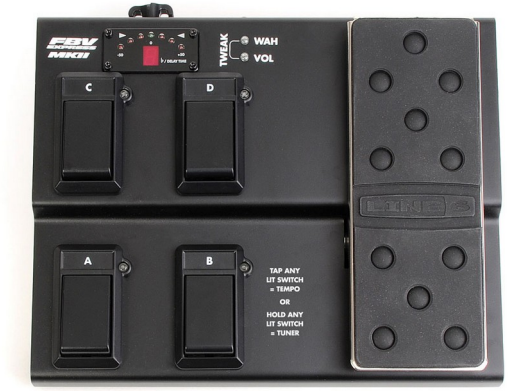
Tonality pedals in LV2 format from Portalmode:

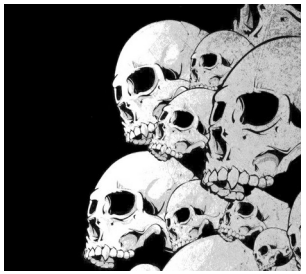
$ git clone https://github.com/portalmod/mod-pitchshifter.git
$ CD MOD-PITCHSHIFTER
$ ./make.sh # compilation
$ ./mod_build # Installation in the user directory
$ dnf installer MOD-GXPITCHSHIFTER



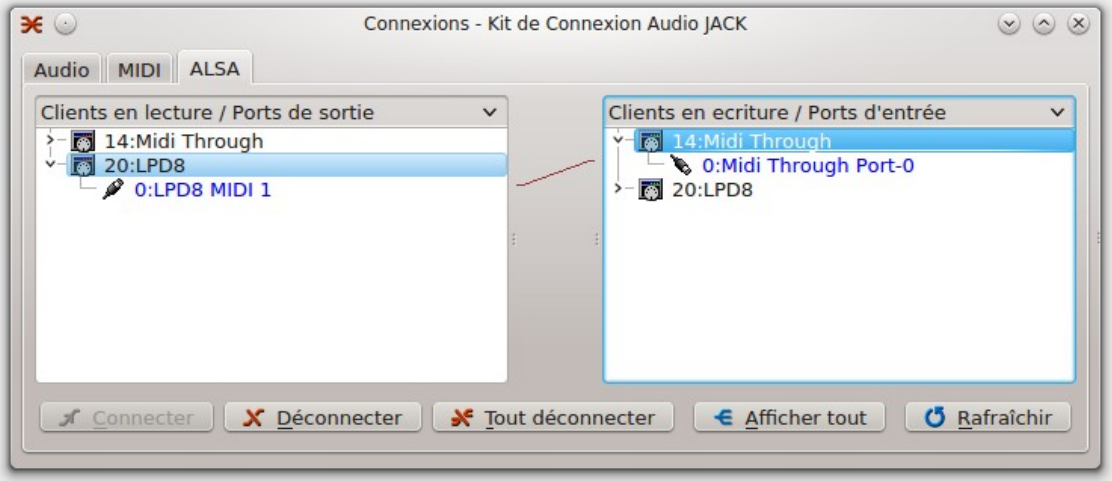
Check via MIDI

Objective: control an effect button via a MIDI instrument



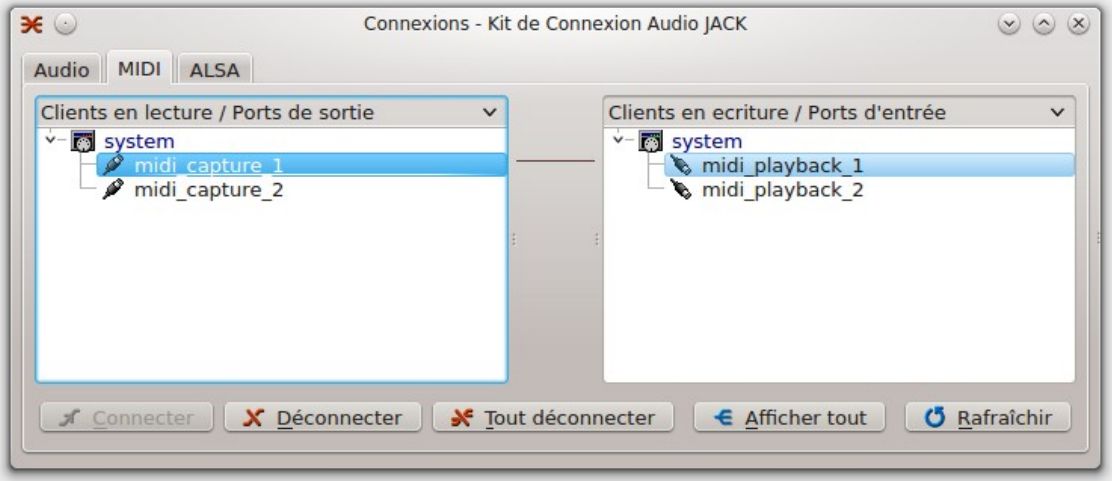


Control via Midi Jack connections



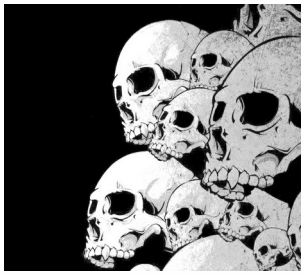
First step:
Alsa tab:

We connect the MIDI output of the LDP8 to the System system input: MIDI_PLAYBACK_1



Second step:
Midi tab:

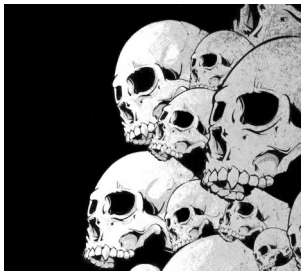
We connect the system system output: MIDI_CAPTE_1 to the system system output: MIDI_PLAYBACK_1



Control via Midi
MIDI / Guitarix association

A click on the guitarix button to control with the middle of the mouse button or on the two buttons for a two buttons mouse reveals this dialog window.
Then you have to move the MIDI button you want to use. The references of this MIDI button are then recovered by guitarix.
You have to click on "Validate" to record the Association Midi / Guitarix button.

The engine menu → MIDI controller reveals this dialogue.
By clicking on "Including MIDI controllers in the Presets" allows you to save MIDI / Guitarix associations when you save the presets.



The Dubber

Reading sense

Overdub

Mix between signal and dubber

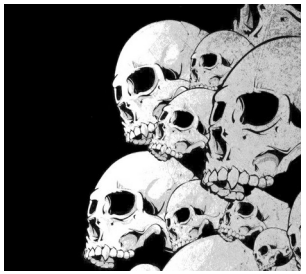
Registration

Start and end adjustment

Delete

Dubber Level

The Dubber allows you to record a sound loop and to replay it

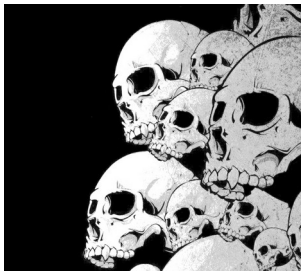


The metronome

Metronome pulsation

Metronome level

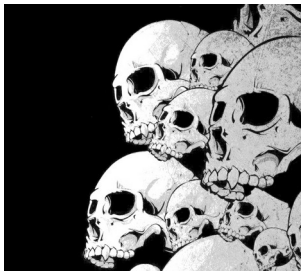
The metronome diffuses a beat. The sound level of the beat is proportional to the level of the sound at the input.



Guitarix and web server 1/2

Guitarix can be launched without graphical interface.
It is then controllable via a web server.
This web server can then be checked via a graphics tablet for example.

Guitarix compilation in web server mode:
 # We recover the source code of guitarix (Git must be installed)
$ git clone git: //git.code.sf.net/p/guitarix/git guitarix
 # We initialize guitarix dependencies
$ git submodule update -in
$ CD Trunk/Webui/Bootplate/
$ git submodule update -in
$ cd ../ ..
 # We are now compiling guitarix (GCC must be installed)
$ ./waf Configure –Prefix =/Opt/Guitarix-Devel
$ ./waf
 # We install everything
$ su -c ./waf install

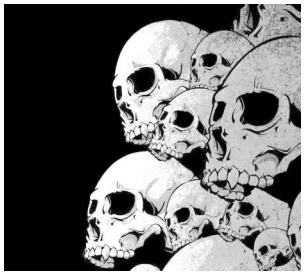


Guitarix and web server 2/2

1. Starting guitarix:
   $ /opt/guitarix-devel/bin/guitarix -N
2. start the websockify program from the current directory (webui):
   $ python -m websockify.websocketproxy --web=. '*':8000 localhost:7000
3. open the URL http://localhost:8000 in the desktop browser

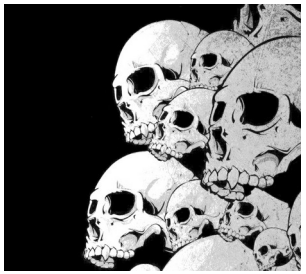
You should be redirected to debug.html. index.html expects preprocessed files which can be generated with the deploy script:
  $ ./deploy.sh
This script creates the build/ directory which is used in index.html.
deploy.sh needs the node program from node.js. For Debian/Ubuntu:
  $ su -c dnf install nodejs

Next step :
- open the port 8000 on the tablet and on the server ;
- connect to the guitarix web server via the tablet ;
- install this web server on the Raspberry Pi.



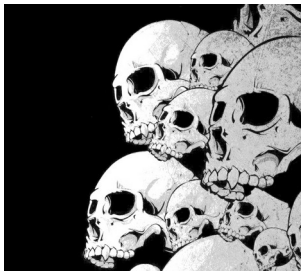
Guitarix Scope

The Scope allows you to visualize the signal to a given place in the rack.
It is practical because it makes it possible to visualize the influence of the settings of an effect.
In this way, we can dose the quantity of distortion.
Too much gain will generate an almost square signal.
Not enough gain, the signal will remain sinusoidal.
It remains to find the right dose.



Guitarix / VST3

It’s now possible to use Guitarix as a VST3 plugin in Carla for example

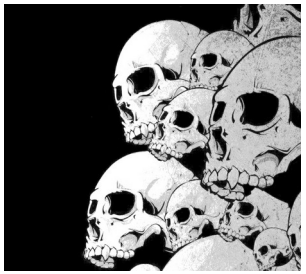


Webography

You can access to a huge set of presets for Guitarix on Musical Artifacts :

https://musical-artifacts.com/artifacts?apps=guitarix

All these artifacts can be easily downloaded into Guitarix because Musical Artifacts is now integrated into Guitarix.

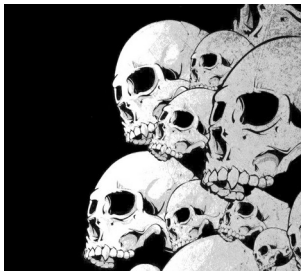


Documentary

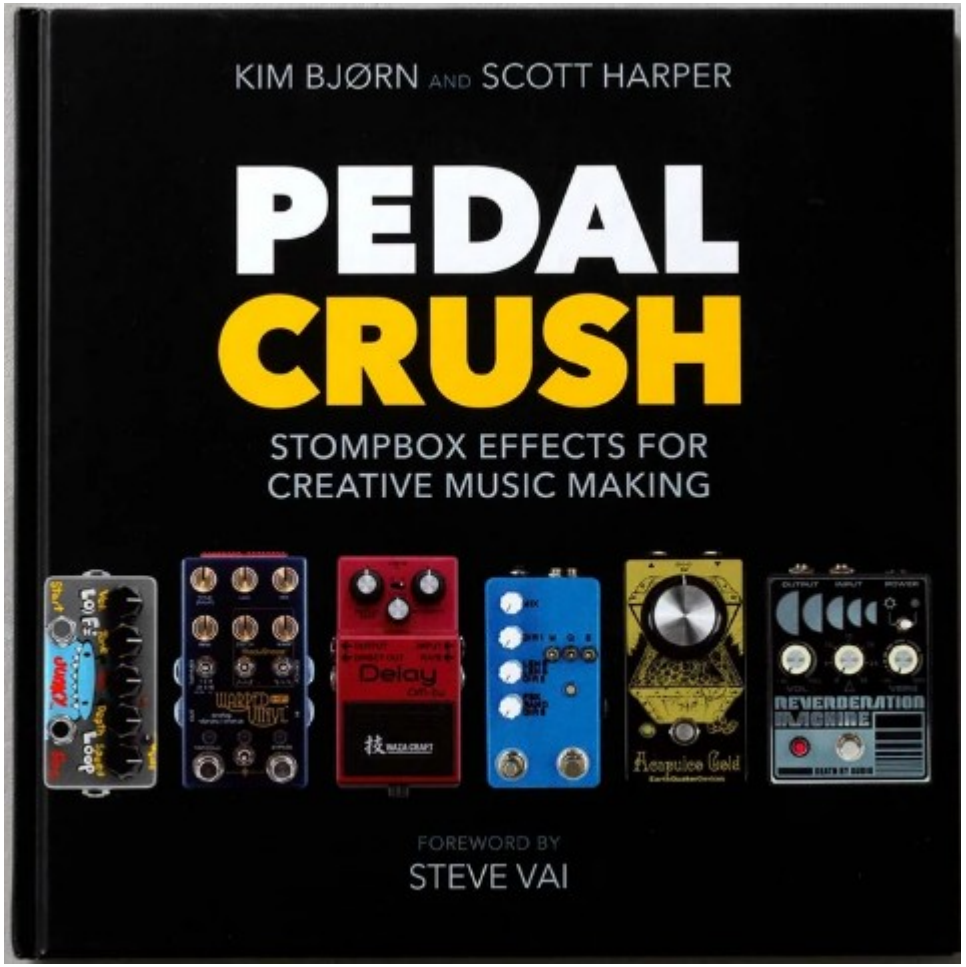
A good documentary about pedals, who made them …

The PEDAL Movie :

https://www.imdb.com/fr/title/tt11811310/



Book



A good book about pedals :

PEDAL CRUSH

https://bjooks.com/en-fr/products/pedal-crush-stompbox-effects-for-creative-music-making