




Become a rockstar using FOSS!

(Or at least use FOSS to write and share music for fun!)

Lorenzo Miniero

 @lminiero@fosstodon.org

FOSDEM 2023, Open Media devroom

4th February 2023, Brussels 



Clickbait title alert: not a rockstar!



Lorenzo Miniero

@lminiero@fosstodon.org

I know one of them was me, so who was the other one?! #spotify
#november



LORENZO MINIERO HAD

2 listeners



But that doesn't mean you can just have fun 😊





A few words about me



Lorenzo Miniero

- WebRTC developer (Janus)
- Hobbyist musician
- Love metal & orchestral!

Let's keep in touch!

- lminiero@gmail.com
- <https://fosstodon.org/@lminiero>
- <https://lminiero.bandcamp.com>
- <https://soundcloud.com/lminiero>



A barebones TOC



- We'll talk about (not necessarily in this order...)
 - The FOSS music ecosystem
 - How you can write some music (composition and orchestration)
 - Different ways of rendering virtual instruments
 - Recording and/or processing real instruments
 - Putting it all together in a DAW (mixing and mastering)
 - How you can distribute the result (on demand vs. live)
- Please notice we will only scratch the surface, though
 - There's so much to talk about and so little time!
 - Besides, I'm no expert: there's good chances I'll say something wrong... 😊



A barebones TOC



- We'll talk about (not necessarily in this order...)
 - The FOSS music ecosystem
 - How you can write some music (composition and orchestration)
 - Different ways of rendering virtual instruments
 - Recording and/or processing real instruments
 - Putting it all together in a DAW (mixing and mastering)
 - How you can distribute the result (on demand vs. live)
- Please notice we will only scratch the surface, though
 - There's so much to talk about and so little time!
 - Besides, I'm no expert: there's good chances I'll say something wrong... 😊



FOSS music ecosystem is *amazing!*



- Audio backends getting easier to use at low latencies
 - e.g., ALSA, JACK, Pipewire, etc.
- Port-based approach helps avoiding monolithic applications
 - Different applications to deal with different requirements
 - They can then all connect to each other
- Many existing alternatives for different requirements
 - Easy to find what works best for your needs
 - Different genres may require a different approach



FOSS music ecosystem is *amazing!*



- Audio backends getting easier to use at low latencies
 - e.g., ALSA, JACK, Pipewire, etc.
- Port-based approach helps avoiding monolithic applications
 - Different applications to deal with different requirements
 - They can then all connect to each other
- Many existing alternatives for different requirements
 - Easy to find what works best for your needs
 - Different genres may require a different approach



FOSS music ecosystem is *amazing!*



- Audio backends getting easier to use at low latencies
 - e.g., ALSA, JACK, Pipewire, etc.
- Port-based approach helps avoiding monolithic applications
 - Different applications to deal with different requirements
 - They can then all connect to each other
- Many existing alternatives for different requirements
 - Easy to find what works best for your needs
 - Different genres may require a different approach



An important distinction before we start



- Applications will work with audio and/or MIDI signals
 - Results may be similar, but they're not the same "thing"!
- Audio signals
 - Basically a waveform, an actual sound being captured/reproduced
 - What may come from a microphone, or what's sent to your speakers
- MIDI signals
 - Information about specific notes being played (e.g., on a keyboard)
 - Will often be translated to an audio signal itself (e.g., with virtual instruments)
- Some applications will only work with one, others with both



An important distinction before we start



- Applications will work with audio and/or MIDI signals
 - Results may be similar, but they're not the same "thing"!
- Audio signals
 - Basically a waveform, an actual sound being captured/reproduced
 - What may come from a microphone, or what's sent to your speakers
- MIDI signals
 - Information about specific notes being played (e.g., on a keyboard)
 - Will often be translated to an audio signal itself (e.g., with virtual instruments)
- Some applications will only work with one, others with both



An important distinction before we start



- Applications will work with audio and/or MIDI signals
 - Results may be similar, but they're not the same "thing"!
- Audio signals
 - Basically a waveform, an actual sound being captured/reproduced
 - What may come from a microphone, or what's sent to your speakers
- MIDI signals
 - Information about specific notes being played (e.g., on a keyboard)
 - Will often be translated to an audio signal itself (e.g., with virtual instruments)
- Some applications will only work with one, others with both



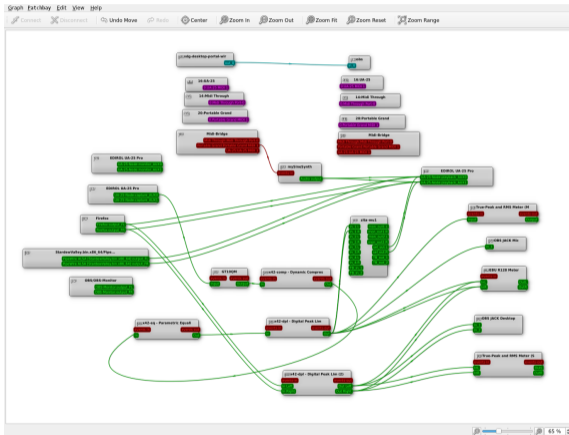
An important distinction before we start



- Applications will work with audio and/or MIDI signals
 - Results may be similar, but they're not the same "thing"!
- Audio signals
 - Basically a waveform, an actual sound being captured/reproduced
 - What may come from a microphone, or what's sent to your speakers
- MIDI signals
 - Information about specific notes being played (e.g., on a keyboard)
 - Will often be translated to an audio signal itself (e.g., with virtual instruments)
- Some applications will only work with one, others with both



One Pipewire to rule them all!



<https://pipewire.org/>



Capturing an instrument to make some noise



- You have your guitar/bass and your laptop, what do you do now?
 - You can't just plug the jack in the mic slot!
- You'll need an audio interface of some sort
 - e.g., USB audio interface with XLR/3.5mm inputs
 - Interface ports then available as a system "capture"
- What to capture depends on what you need
 - e.g., straight from the cable vs. already processed for electric guitar
 - Acoustic guitars sometimes recorded from multiple sources at the same time



Capturing an instrument to make some noise



- You have your guitar/bass and your laptop, what do you do now?
 - You can't just plug the jack in the mic slot!
- You'll need an audio interface of some sort
 - e.g., USB audio interface with XLR/3.5mm inputs
 - Interface ports then available as a system "capture"
- What to capture depends on what you need
 - e.g., straight from the cable vs. already processed for electric guitar
 - Acoustic guitars sometimes recorded from multiple sources at the same time



Capturing an instrument to make some noise



- You have your guitar/bass and your laptop, what do you do now?
 - You can't just plug the jack in the mic slot!
- You'll need an audio interface of some sort
 - e.g., USB audio interface with XLR/3.5mm inputs
 - Interface ports then available as a system "capture"
- What to capture depends on what you need
 - e.g., straight from the cable vs. already processed for electric guitar
 - Acoustic guitars sometimes recorded from multiple sources at the same time

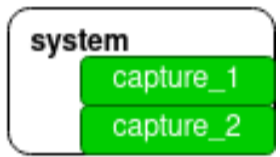
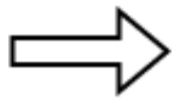


The one I personally use (note: cat not included)





Audio interface + instrument + JACK



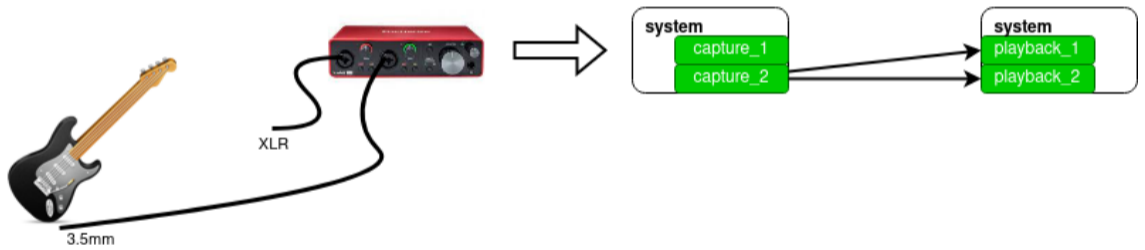


Audio interface + instrument + JACK



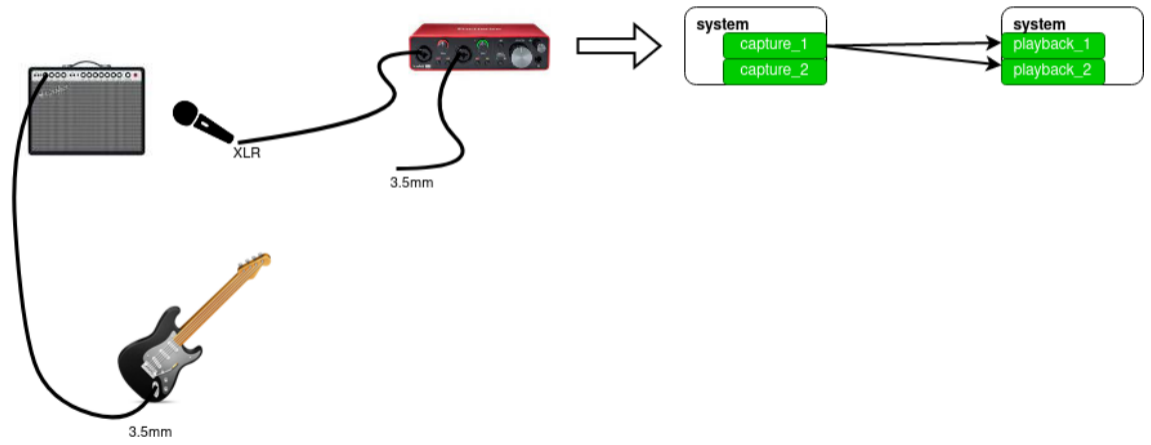


Audio interface + instrument + JACK



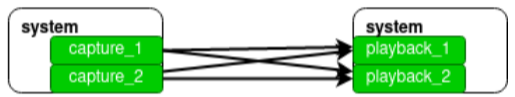
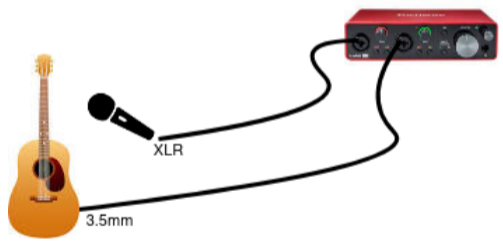


Audio interface + instrument + JACK





Audio interface + instrument + JACK





Guitar/bass processing: Guitarix



<https://guitarix.org/>



Guitar/bass processing: Guitarix





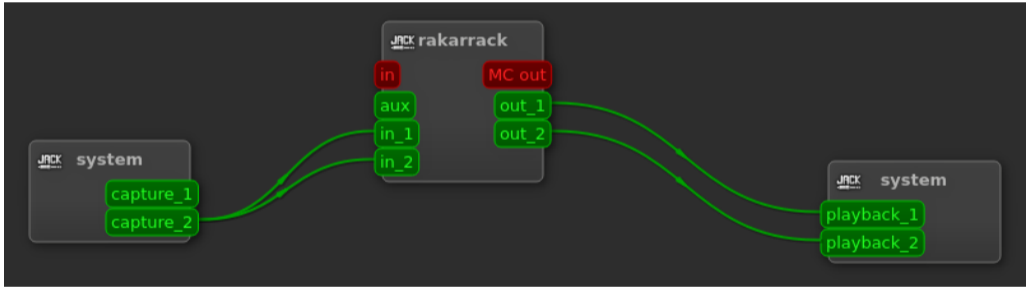
Guitar/bass processing: Rakarrack



<https://rakarrack.sourceforge.net/>

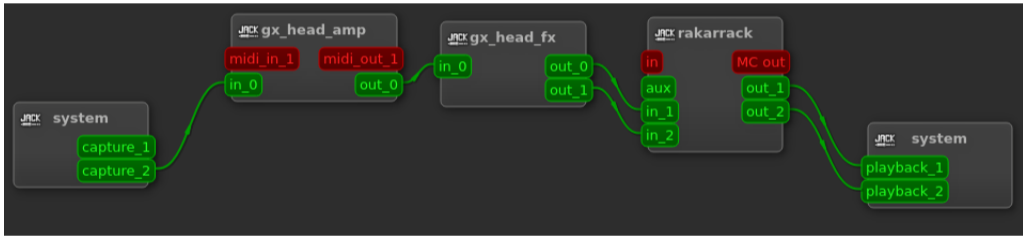


Guitar/bass processing: Rakarrack





Why not both at the same time? 😊





What if we want to record it, now?



- Any recording application that can act as a JACK sink will do
 - e.g., Audacity, GStreamer, etc.
- Good chances are you want to work within a “project”, though
 - Multiple tracks for different instruments
 - Ability to add filters to work on tracks dynamically
- That’s where a DAW can help!
 - Digital Audio Workstation
 - Conceived to record, edit and produce audio files of different kinds
 - Modular nature adds support for filters, EQ, compression, etc.



What if we want to record it, now?



- Any recording application that can act as a JACK sink will do
 - e.g., Audacity, GStreamer, etc.
- Good chances are you want to work within a “project”, though
 - Multiple tracks for different instruments
 - Ability to add filters to work on tracks dynamically
- That's where a DAW can help!
 - Digital Audio Workstation
 - Conceived to record, edit and produce audio files of different kinds
 - Modular nature adds support for filters, EQ, compression, etc.



What if we want to record it, now?



- Any recording application that can act as a JACK sink will do
 - e.g., Audacity, GStreamer, etc.
- Good chances are you want to work within a “project”, though
 - Multiple tracks for different instruments
 - Ability to add filters to work on tracks dynamically
- That’s where a DAW can help!
 - Digital Audio Workstation
 - Conceived to record, edit and produce audio files of different kinds
 - Modular nature adds support for filters, EQ, compression, etc.



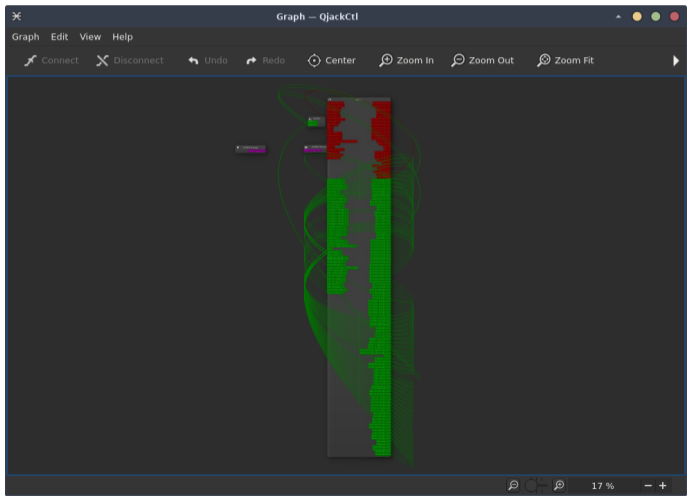
The one I personally use: Ardour



<https://ardour.org/>



That's a lot of connections!





Adding drums (without a drummer!)



- Let's assume we have bass and guitars, now: what about drums?
 - Unless you have a “real” one, that is!
- First good example of “virtual instrument”
 - We'll need to “write” the drum parts
 - Something will then need to “sequence” them (drum sounds)
- While you can write MIDI manually, patterns are easier to work with
 - Rhythmic parts can be repeated, with some variations
 - Percussive nature of instrument helps too



Adding drums (without a drummer!)



- Let's assume we have bass and guitars, now: what about drums?
 - Unless you have a “real” one, that is!
- First good example of “virtual instrument”
 - We'll need to “write” the drum parts
 - Something will then need to “sequence” them (drum sounds)
- While you can write MIDI manually, patterns are easier to work with
 - Rhythmic parts can be repeated, with some variations
 - Percussive nature of instrument helps too



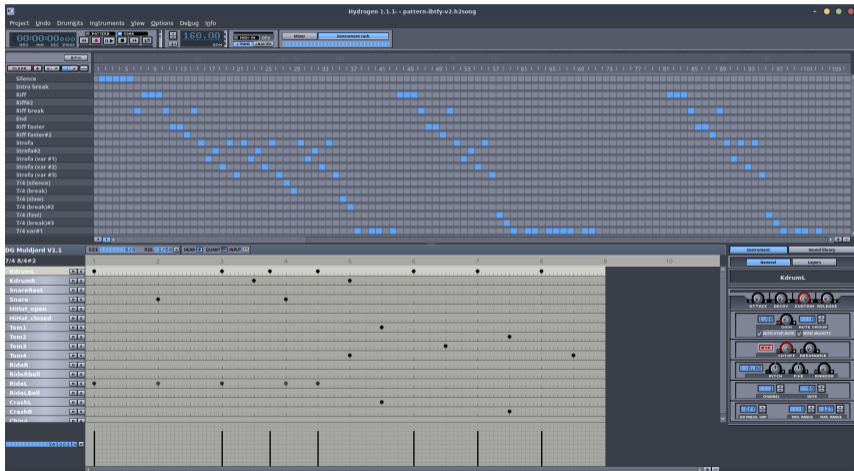
Adding drums (without a drummer!)



- Let's assume we have bass and guitars, now: what about drums?
 - Unless you have a “real” one, that is!
- First good example of “virtual instrument”
 - We'll need to “write” the drum parts
 - Something will then need to “sequence” them (drum sounds)
- While you can write MIDI manually, patterns are easier to work with
 - Rhythmic parts can be repeated, with some variations
 - Percussive nature of instrument helps too



An excellent pattern based drum machine: Hydrogen



<http://hydrogen-music.org/>




Multichannel drums composition: DrumGizmo



DrumGizmo Wiki

⌘ ⚙ Search Search



- [About DrumGizmo](#)
- [The Team](#)

DrumGizmo

- [Getting DrumGizmo](#)
- [Download drumkits](#)
- [Drumgizmo Roadmap](#)

DGEdit

- [DGEdit](#)
- [DGEdit Roadmap](#)


General

- [Documentation](#)
- [Runs/Features](#)


Drumkits

This is a list of DrumGizmo drumkits. Over time we plan on releasing more kits, some in collaboration with the community.

[CrocellKit](#)



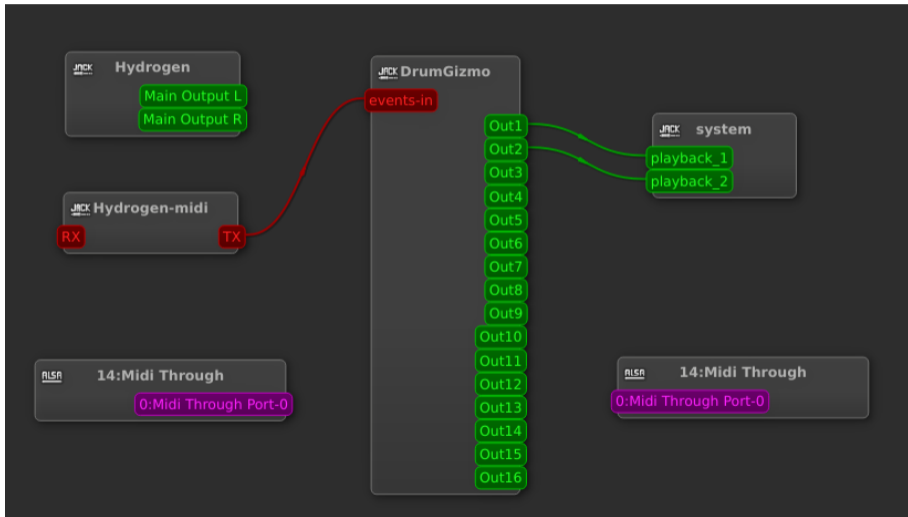
[DRSKit](#)



<https://drumgizmo.org/>



Hydrogen + DrumGizmo





What about other virtual instruments?



- We added guitars, bass, even drums, but we may want more
 - Maybe a keyboard background...
 - ... or a piano/synth solo ...
 - ... or even a full orchestra!
- Just as with drums, this is where MIDI helps
 - We'll have information on which notes to play, and how
 - Something will then turn that to "sounds" (e.g., a trumpet)
- Notes can come from different places
 - Hardware keyboards often have USB MIDI support (a source in JACK)
 - Or we might simply want to write the parts ourselves instead



What about other virtual instruments?



- We added guitars, bass, even drums, but we may want more
 - Maybe a keyboard background...
 - ... or a piano/synth solo ...
 - ... or even a full orchestra!
- Just as with drums, this is where MIDI helps
 - We'll have information on which notes to play, and how
 - Something will then turn that to “sounds” (e.g., a trumpet)
- Notes can come from different places
 - Hardware keyboards often have USB MIDI support (a source in JACK)
 - Or we might simply want to write the parts ourselves instead



What about other virtual instruments?



- We added guitars, bass, even drums, but we may want more
 - Maybe a keyboard background...
 - ... or a piano/synth solo ...
 - ... or even a full orchestra!
- Just as with drums, this is where MIDI helps
 - We'll have information on which notes to play, and how
 - Something will then turn that to “sounds” (e.g., a trumpet)
- Notes can come from different places
 - Hardware keyboards often have USB MIDI support (a source in JACK)
 - Or we might simply want to write the parts ourselves instead

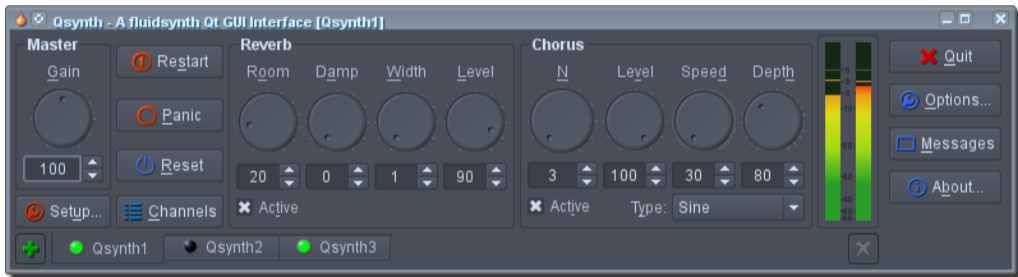


You don't need that many keys to have fun!





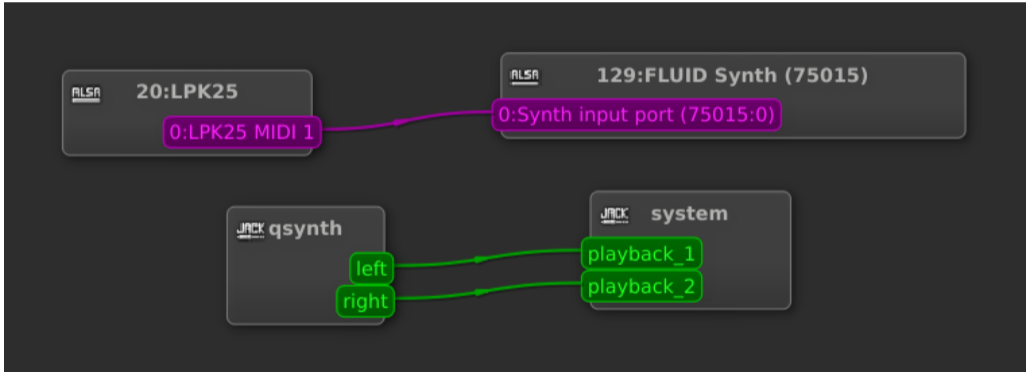
Using soundfonts (SF2): Fluidsynth/Qsynth



<https://qsynth.sourceforge.io/>



Using soundfonts (SF2): Fluidsynth/Qsynth





Using a more complex synth: Yoshimi



The screenshot displays the Yoshimi software interface, divided into two main windows.

Left Window: Bank Selection

Yoshimi - Root 10, Bank 105 - /usr/share/yoshimi/banks/The_Mysterious_Bank

This column uses MIDI Extended Program Change

105. The Mysterious Bank	Search	Roots	Banks	
1. percussive lead	33. organ	65. bell pad	97. powerful bass	129. falling stars
2. heavy funky	34. organ 2	66. sines	98. doublebass	130. glitch
3. funky lead	35. organ 3	67. res strings pad	99. small bass	131. radio
4. funky lead 2	36. organ 4	68. sine pad 1	100. synth bass 1	132. starting machine
5. delay lead	37. organ 5	69. sine pad 2	101. synth bass 2	133. strange world
6. old lead	38. space organ	70. sine pad 3	102. synth bass 3	134. ufo invasion
7. 8bit evolved	39. space organ 2	71. sine pad 4	103. synth bass 4	
8. pulse bell		72. sine pad 5	104. synth bass 5	
9. vangelis bell	41. 8bit love	73. sine pad 6	105. synth bass 6	
10. simple bell	42. 80s pad	74. violent pad	106. synth bass 7	
11. dreaming bells	43. analog strings	75. violent pad 2	107. synth bass 8	
12. fantastic bell	44. smooth strings	76. compad	108. synth bass 9	
13. outerspace bell	45. synth strings	77. hoover	109. synth bass 10	
14. fantasy bell	46. synth strings 2	78. old pad	110. synth bass 11	
15. mystic bell	47. strings pad 1	79. supersaw	111. synth bass 12	
16. pulse bell 2	48. strings pad 2	80. supersaw 2	112. vintage bass	
17. sine bell	49. strings pad 3	81. supersaw 3		
18. synth bell	50. soft pad	82. supersaw 4	114. 80s rhodes	
	51. trance pad	83. good pad	115. rhodes	
20. popcorn	52. another planet	84. synth pad	116. rhodes 2	
21. popcorn 2	53. another galaxy		117. new arpeggio 3	
22. china synth	54. another space		118. musicbox	
23. weird	55. will in space	87. bass	119. jazz guitar	
	56. in space	88. bass 2	120. heavy metal qui	
25. church organ	57. near saturnus	89. bass pad	121. percussive	
26. 60s organ	58. echoes	90. bass pad fat	122. pseudo steeldrur	
27. metronomy organ	59. far far away	91. depeche mode	123. synth reed	
28. hard organ	60. not so far	92. dubstep bass	124. brass	
29. cool organ	61. lost	93. fat bass		
30. cool organ 2	62. somewhere	94. fingered bass	126. 8bit car	
31. cool organ 3	63. chip pad	95. fm bass	127. computer	
32. rock organ	64. gauss pad	96. hard bass	128. drunk world	

Buttons: SELECT, RENAME, SAVE, DELETE, SWAP, Engines Used (Add, Sub, Pad), Close

Right Window: Instrument Control Panel

Yoshimi Instrument PatchSet Path State Scale

Buttons: Stop!, Mixer Panel, Vectors, Reset, Virtual Keyboard, Undo, Stereo, Midi Learn, Redo

Detune Volume Key Shift 0 F. BPM 120

System Effects Insertion Effects

No Effect On Send to

No Effects Applied

Part 1 of 16 Simple Sound Edit

On Poly Mode

Midi 1 Portamento Vel.Sense Vel.Offs Panning Volume

Controllers Midi CCs Pan Law Default: +3dB, -3dB

Min Note 0 Key Shift 0

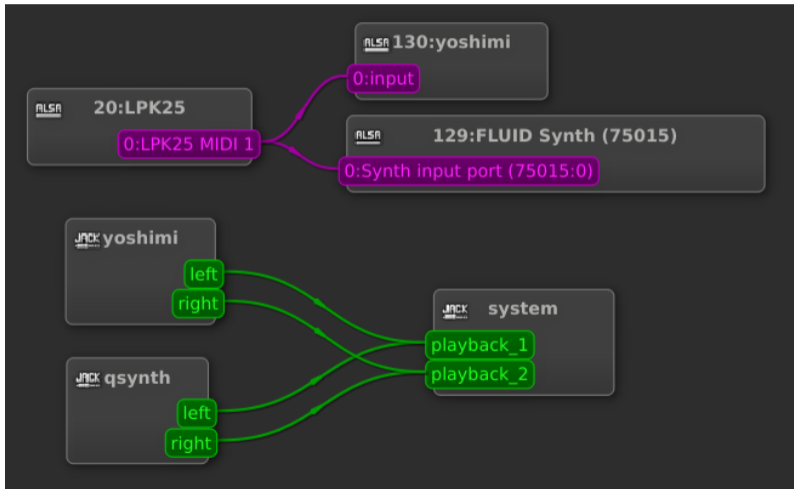
Max Note 127 Key Limit 20

Send to System Effect

<https://yoshimi.sourceforge.io/>



Using a more complex synth: Yoshimi





Using Sforzando format (SFZ): sfizz



The screenshot displays the sfizz software interface. At the top left, there is a 'File' menu and a 'Presets' button. The main area is divided into several sections:

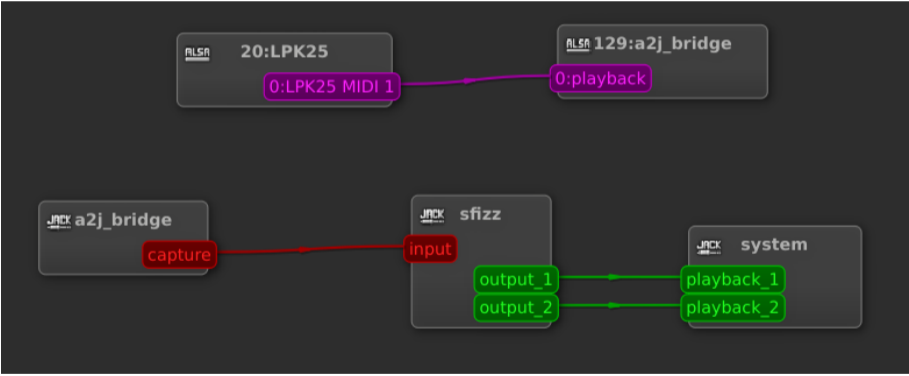
- sfizz logo:** Includes a home icon, a switch icon, and a gear icon.
- DefaultInstrument:** A dropdown menu with left and right arrow icons and a checkmark.
- Key switch:** A text input field.
- Voices:** 0
- Max:** 64
- Memory:** (no value shown)
- Volume:** A green circular meter showing 0.0 dB.
- Engine:** A panel with three settings: Polyphony (64), Oversampling (1x), and Preload size (32 kB).
- Tuning:** A panel with four settings: Scala file (DefaultScale), Root key (C), Frequency (4), and Stretch (440.0 Hz).

At the bottom, there is a piano keyboard with numbered labels from -1 to 8.

<https://sfz.tools/sfizz/>

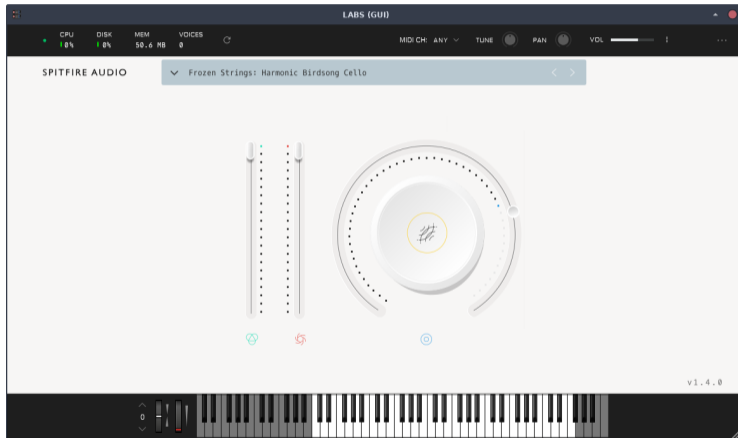


Using Sforzando format (SFZ): sfizz





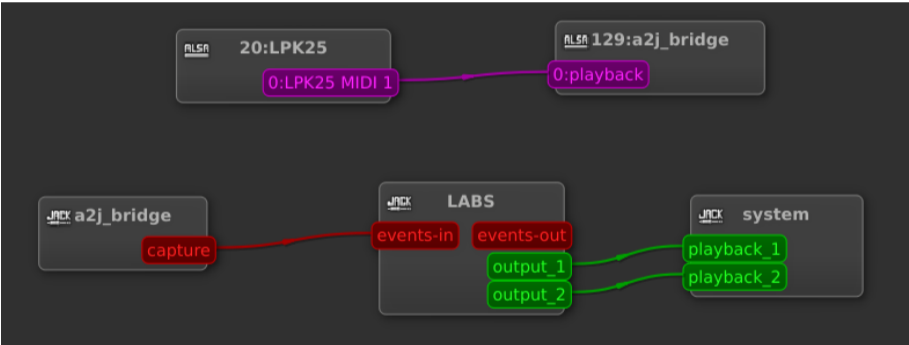
Using Windows VSTs: LinVST (e.g., via Carla)



<https://github.com/osxmidi/LinVst>



Using Windows VSTs: LinVST (e.g., via Carla)





Writing music as if you were programming: Lilypond



The screenshot shows the Lilypond software interface. On the left, there is a sidebar with various toolbars for articulation, dynamics, and other musical notations. The main window is divided into two panes. The top pane shows the source code for a musical score, including notes, rests, and dynamic markings like `mf` and `ff`. The bottom pane shows the rendered musical score with staves for different instruments and vocal parts. The score includes lyrics in Dutch: "Lau-da - te De - ho - vam, om nea gen - tes! om nea gen tes! Lau - da - te De - ho - vam om nea gen tes! om nea gen tes!". The interface also shows a menu bar at the top and a status bar at the bottom.

<https://lilypond.org/>



A simpler WYSIWYG approach: MuseScore



The screenshot displays the MuseScore application window. The main workspace shows a piano score for a piece titled "Reunion-36". The score is in 4/4 time and consists of three systems of music. The first system is marked "Andante con moto" with a tempo of $\text{♩} = 120$ and a dynamic of *mp*. The second system is marked "più mosso" and includes a *rit.* (ritardando) marking. The third system is marked "tempo primo" and includes a *rit.* marking and a dynamic of *mf*. The Inspector panel on the right is open, showing settings for the selected element, including visibility, automatic placement, minimum distance, offset, stacking order, leading space, and barline style. The status bar at the bottom indicates "Barline: Single barline; Measure: 1; Beat: 1; Staff: 1 (Piano)" and "Normal mode 1:01:000".

<https://musescore.org/>



Don't know music notation? Piano rolls to the rescue!



A screenshot of a digital audio workstation (DAW) piano roll window. The window title is "Piano-Roll - juno_pad01.ogg". The interface includes a toolbar with various editing tools like play, stop, solo, mute, and eraser. A timeline at the top shows measures 1 through 17. The piano roll itself is a grid with a vertical axis for pitch (labeled C4, C5, C6) and a horizontal axis for time. Green rectangular notes are plotted on the grid. At the bottom, there is a "Note Volume" section showing vertical stems with orange dots representing the volume of each note. The interface is dark-themed with a blue header bar.



Your song's ready: what now?



- Many places where you could upload it to
 - e.g., Soundcloud, Jamendo, Youtube/Peertube, etc.
 - Bandcamp good option to try and monetize your music
 - Spotify, Apple Music, etc. need intermediary (e.g., Distrokid)
 - ... but they're only worth it if you want to "be there"
 - Royalties per play are beyond ridiculous
- Make sure you engage the community for feedback and learning!
 - <https://linuxmusicians.com/viewforum.php?f=9>
 - <https://discourse.ardour.org/c/community/made-with-ardour/13>



Your song's ready: what now?



- Many places where you could upload it to
 - e.g., Soundcloud, Jamendo, Youtube/Peertube, etc.
 - Bandcamp good option to try and monetize your music
 - Spotify, Apple Music, etc. need intermediary (e.g., Distrokid)
 - ... but they're only worth it if you want to "be there"
 - Royalties per play are beyond ridiculous
- Make sure you engage the community for feedback and learning!
 - <https://linuxmusicians.com/viewforum.php?f=9>
 - <https://discourse.ardour.org/c/community/made-with-ardour/13>



Your song's ready: what now?



- Many places where you could upload it to
 - e.g., Soundcloud, Jamendo, Youtube/Peertube, etc.
 - Bandcamp good option to try and monetize your music
 - Spotify, Apple Music, etc. need intermediary (e.g., Distrokid)
 - ... but they're only worth it if you want to "be there"
 - Royalties per play are beyond ridiculous
- Make sure you engage the community for feedback and learning!
 - <https://linuxmusicians.com/viewforum.php?f=9>
 - <https://discourse.ardour.org/c/community/made-with-ardour/13>



“All I want for Christmas is Odin”



<https://www.youtube.com/watch?v=hxjwb8mzDQQ>
<https://peertube.tv/w/jGRAueDQFvhhNC2y1CYdKE>



“Can WebRTC help musicians?”

The background of the slide is a white surface with a prominent vertical crack running down the right side and several smaller horizontal and diagonal cracks, giving it a weathered or broken paper appearance.

Can WebRTC help musicians?

Going beyond traditional and boring use cases to support the arts

Lorenzo Miniero

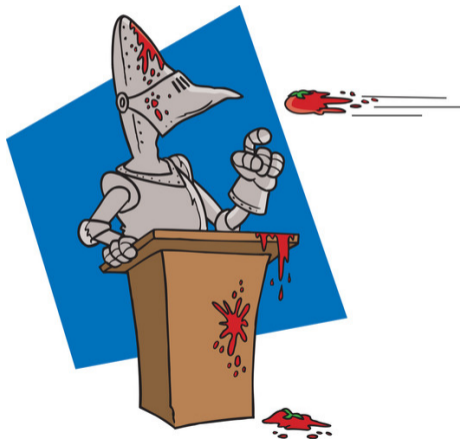
D.rtc
2021-02-06

The logo for FOSDEM 2021, featuring the year '2021' in large black letters with a globe and purple virus-like icons integrated into the '0' and '2's, and the word 'FOSDEM' in purple below it.

https://fosdem.org/2021/schedule/event/webrtc_musicians/



Thanks! Questions? Comments?



Contacts

-  <https://lminiero.bandcamp.com>
-  <https://soundcloud.com/lminiero>
-  <https://fosstodon.org/@lminiero>
-  <https://twitter.com/elminiero>