



User manual	2
Soundfont editor	6
> Tree	7
> Toolbar	13
> Editing pages	17
> General information	18
> Sample editor	20
> Instrument editor	25
> Preset editor	33
> Configuration summaries	35
> Tools	38
> Global tools	42
> Sample tools	44
> Instrument tools	51
> Preset tools	60
Soundfont browser	62
Settings	66
Menu	72
Annexes	75
> The different soundfont formats	76
> Command line	79
Tutorials	84
Create a soundfont from scratch	86
How to prepare a sample	90
Using custom releases in an instrument	95
Use Polyphone as a synthesizer	98
Create a synthesizer with a Raspberry Pi	102
Read a MIDI file with a soundfont	103
Technical documentation	106
Building Polyphone for Windows	108
Building Polyphone for Mac OS X	111
Building Polyphone for Fedora	113
Building Polyphone for Ubuntu	115
Using Qt Creator to build Polyphone	117
Translate Polyphone	121



USER MANUAL



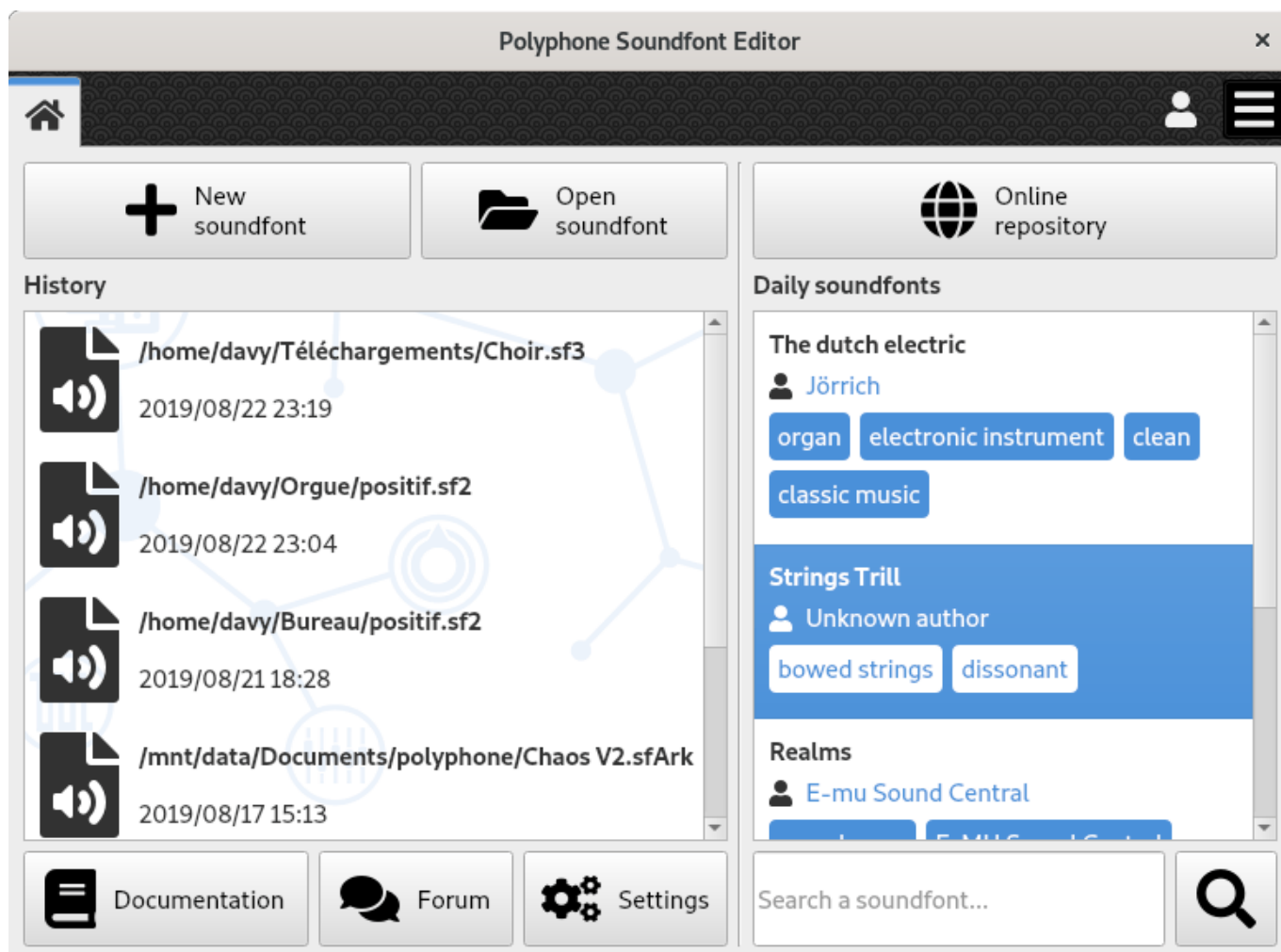
Welcome to the **User Manual** of Polyphone! To quickly discover what soundfonts are and how to edit them with Polyphone, you can start with [tutorials](#) or watch videos on [YouTube](#) (special thanks to Polyphone users for their useful videos). And if you'd like to learn all about the software's features, simply read on.

If you need more help, or if the reading of these pages puts you off, you may find assistance in the [forum](#).

HOME SCREEN

The first screen of Polyphone is basically made of 3 parts:

- a **left part**, containing a file history and shortcuts;
- a **right part**, dedicated to the soundfont repository;
- a **top area**, containing tabs and a menu.



Home screen of Polyphone



LEFT PART

Opening a soundfont



Three ways for opening a soundfont:

- a click on **Open soundfont** opens a file browser, the allowed file to select having the extension **.sf2**, **.sf3**, **.sfArk**, **.sfz** or **.organ** (the different formats are described [here](#));
- a click on **New soundfont** opens an empty soundfont, everything is to be done from scratch;
- a double-click on a soundfont from the history re-open it.

When a soundfont is open, a new tab appears with a [soundfont editor](#) inside.

Shortcuts

Three buttons are located at the bottom of the left part:

- **Documentation**: open a web browser for reading the Polyphone [documentation](#) .
- **Forum**: open a web browser with the URL of the Polyphone [forum](#) .
- **Settings**: open the [settings](#) of the software in a new tab.

RIGHT PART

Online repository

A click on the button **Online repository** opens a tab with the [soundfont browser](#).

Daily soundfonts

Every day, 5 random soundfonts are displayed here. This is made for inciting people discovering new soundfonts. Double-clicking on one of them opens a new tab with the corresponding [soundfont description](#).

Search

It is possible to write a keyword here for searching specific soundfonts and the results will be displayed in the [soundfont browser](#).

TOP AREA



Tabs

The tabs provide a quick access between:

- the **home screen** (first tab — always open),
- the **editor** of an open soundfont,
- the software **settings**,
- the **soundfont browser**,
- a **soundfont description** from the repository.

Navigation between tabs can be performed using keyboard shortcuts:

Ctrl+**Tab** to go to the next tab, excluding the home screen tab,
Ctrl+**Maj**+**Tab** to go to the previous tab, excluding the home screen tab,
Ctrl+**H** to go to the home screen.

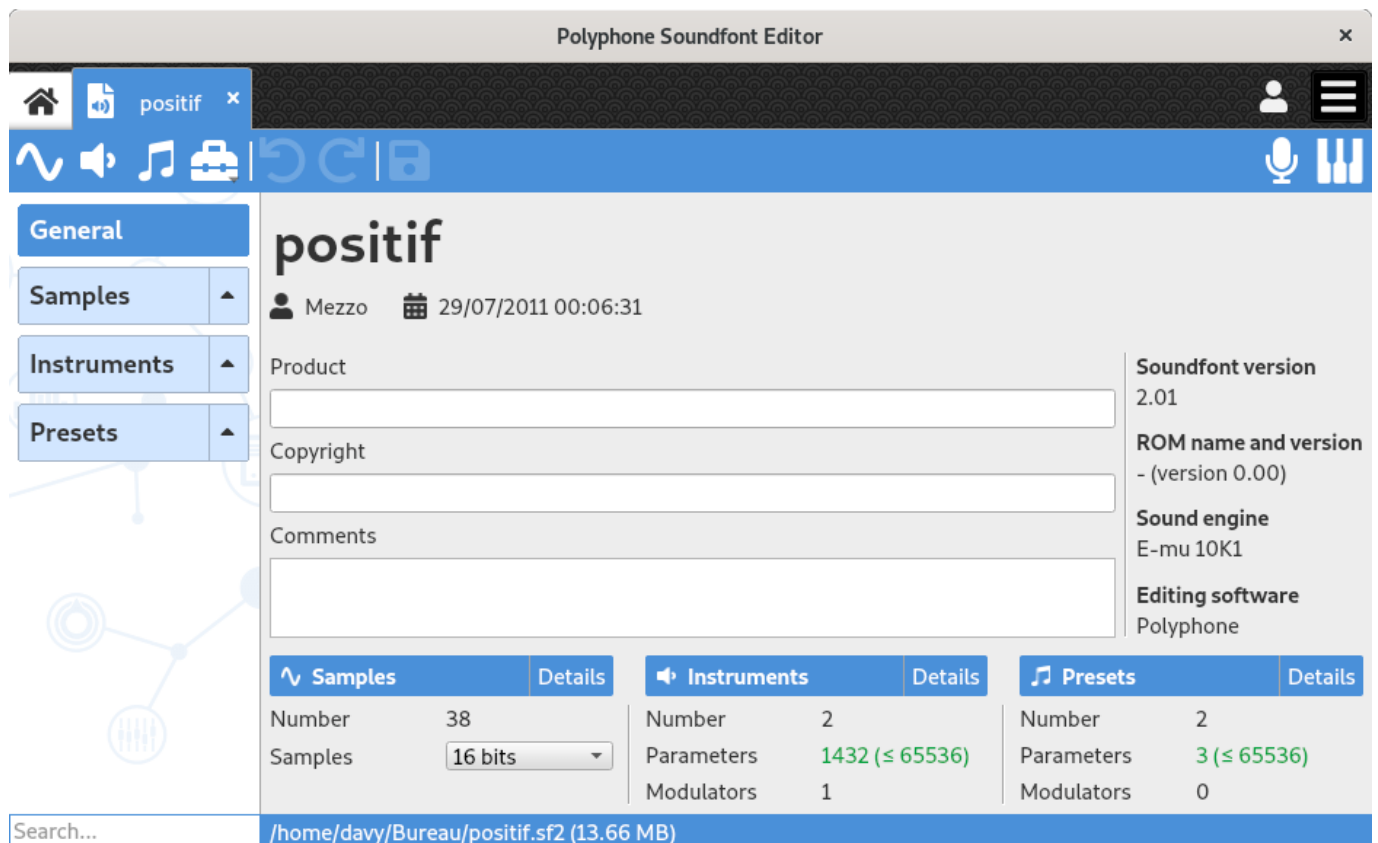
Menu

The menu, located in the top right corner, contains general functions and is described [here](#).



SOUNDFONT EDITOR

When a soundfont is created or when an existing one is opened, a new tab containing a soundfont editor appears.



Soundfont editor

A soundfont editor is made of:

- a **tree** on the left, representing the architecture of the soundfont;
- a **toolbar** on the top (in blue in the picture);
- a main area that can either contain an **editing page** or a **configuration summary**, depending on what is selected in the **tree**.



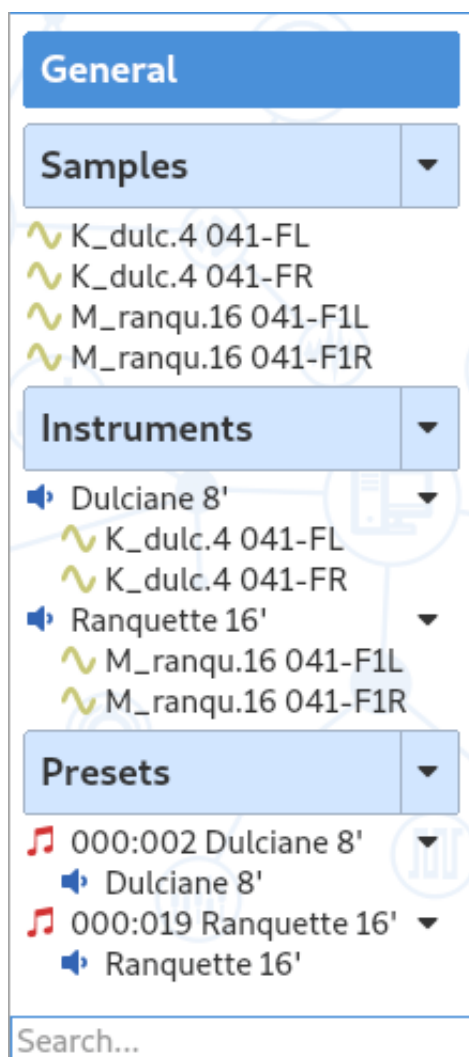
TREE

The tree is the first element in the editor you should focus on. It represents the architecture of the open soundfont and pilots the right part of the software when you select one or more elements.

The tree is always displayed with the same **structure**, provides a **contextual menu** and has **drag & drop**, **copy / paste** and **search** features.

STRUCTURE

The structure of the tree is made of 4 headers and sub-content for the headers **Samples**, **Instruments** and **Presets**.



Tree: structure



Header "General"

A click on the header **General** opens the editor of the [general information](#).

Section "Samples"

This section lists all samples contained in the soundfont, which are the raw material upon which all instruments are built. Selecting one or more samples leads to the [sample editor](#). A click on the header **Samples** displays the [configuration summary](#) of all samples.

Section "Instruments"

This section lists all instruments contained in the soundfont, an instrument being an intermediate element made of samples and used by presets. Instruments contain links to samples called "divisions", displayed as a list below each instrument. Selecting an instrument or one of its divisions leads to the [instrument editor](#). A click on the header **Instruments** displays the [configuration summary](#) of all instruments.

Section "Presets"

This section lists all presets contained in the soundfont, a preset being a final elements made of instruments and that are accessible outside the soundfont. Presets contain links to instruments called "divisions", displayed as a list below each preset. Selecting a preset or one of its divisions shows the [preset editor](#). A click on the header **Presets** displays the [configuration summary](#) of all presets.

CONTEXTUAL MENU

A right click on an element shows a menu comprising the following actions:

Bind to...

Bind the selected sample(s) to an instrument, or the selected instrument(s) to a preset.

Replace by...

Replace a sample by another sample in an instrument, or replace an instrument by another instrument in a preset. The configuration of the divisions is kept.

Copy

Copy the selected element(s), also accessible with **Ctrl+C**.

Paste

Paste the previously copied element(s), also accessible with **Ctrl+V**.

Duplicate

Duplicate the selected element(s) that can be samples, instruments, presets or divisions. Also accessible with **Ctrl+D**.

Delete

Delete the selected element(s), also accessible with **Del**.

Mute



For testing purposes, a division within an instrument or a preset can be muted. This option can also be found in the contextual menu of the **table header**.

Always play

Opposite to the “Mute” function, one or more instruments or presets can be added when playing. It is thus possible to find a good balance between different instruments or presets playing together.

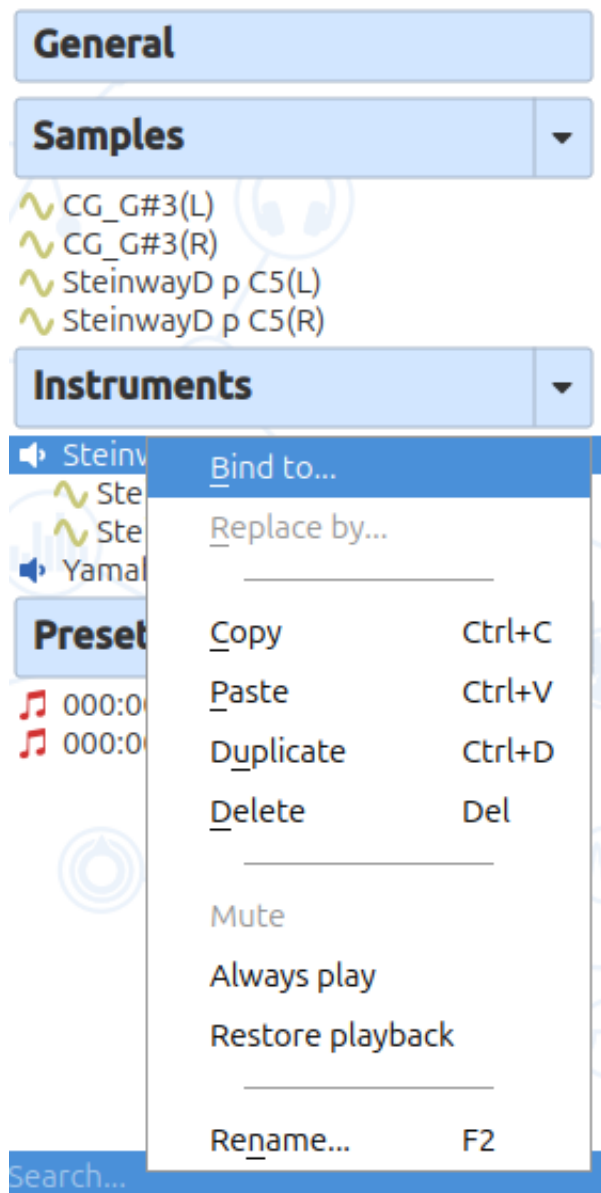
Restore playback

Cancel the effects of “Mute” and “Always play”.

Rename or bulk rename

Rename the selected element(s), also accessible with **F2**. When several elements are selected, different options are provided:

- overwrite existing name with key name as suffix (for samples only),
- overwrite existing name with numerical ascending suffix,
- replace characters,
- insert text at a specific position,
- delete character range.



Tree: context menu

DRAG & DROP

The operation “drag & drop” in the tree enables quick associations or copies. Before a drag & drop you need to select elements. This can be a single element or a multiple selection made possible with the buttons **Ctrl** and **Shift**.

Then, the result of a drag & drop depends on the kind of elements that are dragged and the kind of element that is targeted.

Sample → instrument

Association of the sample to the instrument, creation of an instrument division.

Sample → header Instruments

Creation of an instrument that contains all samples dragged or creation of one instrument per



sample dragged.

Instrument → preset

Association of the instrument to the preset, creation of a preset division.

Instrument → instrument

Copy of all divisions of the source instrument into the destination instrument (it can be the same instrument).

Instrument division → instrument or instrument division

Copy of the division of the source instrument into the destination instrument (it can be the same instrument).

Instrument → header Presets

Creation of a preset that contains all instruments dragged or creation of one preset per instrument dragged.

Preset → preset

Copy of all divisions of the source preset into the destination preset (it can be the same preset).

Preset division → preset or preset division

Copy of the division of the source preset into the destination preset (it can be the same preset).

COPY / PASTE

Everything that is possible with a drag & drop as seen previously is also possible with a copy / paste, accessible through the contextual menu or with the shortcuts **Ctrl+C** and **Ctrl+V**.

Copy / Paste is also allowed within different soundfonts. It is possible to copy one element in a first soundfont, and paste it in a second soundfont. A multiple selection is allowed.

Sample(s) selected

Copy of the samples.

Instrument(s) selected

Copy of the instruments including all samples linked.

Preset(s) selected

Copy of the presets including all instruments and samples linked.

During a copy of elements between different soundfonts, if an element having the same name already exists several options are provided:

- ignore the operation, in which case the copy is canceled for the element having the same name;
- replace the existing element, in which case the existing element is overwritten by the copied element;
- duplicate the element, in which case the element is copied next to the existing element with a different name.

SEARCH

To facilitate the search for items in the tree, a search bar is available to filter.

Tree: search bar



When a string is inserted into the search bar, the elements containing it are shown. Moreover:

- if the string is included in the name of sample, the instruments using the sample and the presets using these instruments are displayed;
- if the string is included in the name of an instrument, the samples used by the instrument and the presets using the instrument are displayed;
- if the string is included in the name of a preset, the instruments used by the preset and the samples used by these instruments are displayed.

A click on the cross to the right of the editable field cancels the filter.



TOOLBAR

The toolbar, located on the top of the editor, contains icons for:








- common **editing functions**,
- switching the **view** in editors,
- displaying the **recorder**,
- displaying the **virtual keyboard**.



Toolbar




EDITING FUNCTIONS

The elements in the left part of the toolbar are for common editing functions:



-  **New sample**
Add new **samples** to the soundfont. A dialog will be displayed for selecting audio files (**.wav**, **.flac**, **.mp3**, **.ogg**, **.aiff** and **.snd** files can be imported).
-  **New instrument**
Create a new **instrument**.
-  **New preset**
Create a new **preset**.
-  **Tools**
Clicking on this button will show a menu containing all available **tools** for the selected element(s) in the **tree**.
-  **Undo**
Undo the last modification.
-  **Redo**
Redo the last undone modification.
-  **Save file**
Save the soundfont.

SWITCH VIEW


In the right part of the toolbar, some icons may appear depending on which **editor** is displayed:

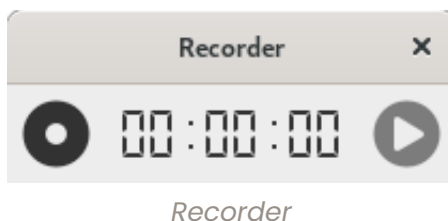
- switching between the  **table view**, the  **range editor** and the  **envelope editor** is possible when editing an instrument,



- switching between the  table view and the  range editor is possible when editing a preset,
- nothing displayed otherwise.

RECORDER

When clicking on the icon , a window opens showing the recorder that can record the output of Polyphone.



The left button can have two possible states:

Record

Start a new record if the acquisition is off. It is possible here to specify the name of the file that will contain the recorded audio signal.

Stop

Stop the current acquisition.

The right button can also have two possible states (only accessible if the acquisition is on):


Pause

Pause the current acquisition.

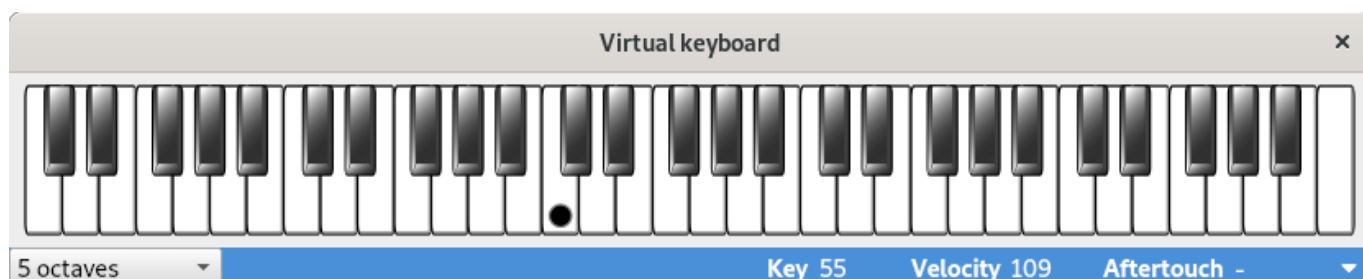
Resume

Resume the current acquisition.

VIRTUAL KEYBOARD

When clicking on the icon , a window opens showing the virtual keyboard that has been made for testing samples, instruments and presets. This window triggers MIDI signals:

- **note on / note off / aftertouch** with the piano keyboard,
- other MIDI signals if the **MIDI controller part** is expanded.





Virtual keyboard

Piano keyboard

The piano keyboard triggers signals that contains key and velocity numbers. It can be controlled in three ways:

with the mouse

The desired key may be clicked by the mouse. The velocity is defined according to the vertical position of the click. After a click, following moves within the same key will trigger aftertouch signals.

using the computer keyboard

The keys used are defined in the [settings](#).

using an external MIDI keyboard

The MIDI input is defined in the [settings](#). If the piano keyboard is visible, it will show what you are playing.

Markers and colors

- When an [instrument](#) or a [preset](#) is selected, the area not covered by the divisions is **grayed**. It is thus possible to visualize the full range of an instrument or a preset.
- When a division of an instrument or a preset is selected, its range is displayed the same way than above. Moreover, its root key is identified by an **orange marker**.
- When a key is selected on the keyboard, it appears in **blue** (or the selection color selected in the software [settings](#)). Moreover, the range of its corresponding division appears in **light blue**. This applies only at the instrument level.
- Middle C (key 60) is identified by a **black marker**.

Useful shortcuts

Ctrl+K

This shortcut can put at any time the focus on the virtual keyboard. This allows, while editing in a table for instance, to use the computer to play instead of edit a cell. A glow effect on the virtual keyboard confirms that the focus is on it.

Ctrl+1, Ctrl+2, ... Ctrl+8

These shortcuts change the octaves played with the computer keyboard. After the use of **Ctrl+1**, the lowest octave can be played. The highest octave is reached using **Ctrl+8**.

→ / ←

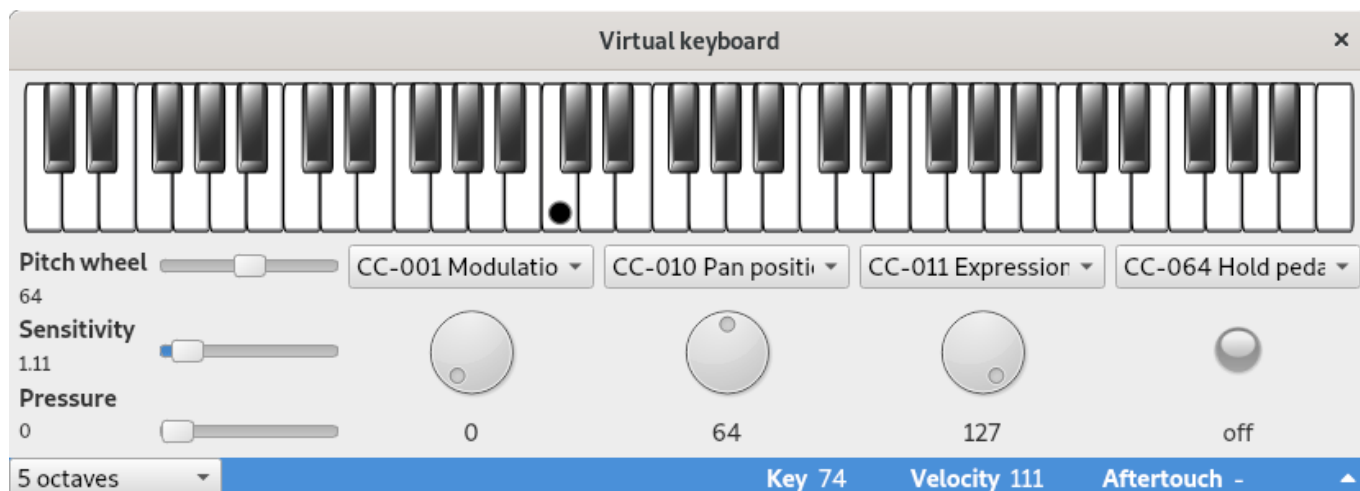
The right or left arrow will play the pitch just next or below the last key that has been played, with the same velocity. The association of the key **Shift** skips keys.

↑ / ↓

The up or down arrow will play the same pitch than the last key played, but with a stronger or weaker velocity. The association of the key **Shift** skips velocities.

MIDI controllers

When clicking on the bottom right arrow, the MIDI controller area can be displayed or hidden. If you are playing with modulators, it is advised that you use these features to test your soundfont.



Virtual keyboard with MIDI controllers

This area is made of different buttons:

Pitch wheel

The pitch wheel increases or decreases the pitch. It goes back automatically to the original position 0.

Sensitivity

The sensitivity defines the modification range in semi-tones of the pitch wheel. By default this is 2 (one full tone).

Pressure

The pressure is the amount of pressure on the keyboard. It can be compared to the aftertouch but while the aftertouch is key-specific, the pressure is a global signal for all keys.

Knobs

3 knobs are here so that MIDI controller values can be changed. For each knob you can specify a controller number and then its value.

On / off led

This led is to set a MIDI controller value to 0 or 127 (useful for the sustain pedal for instance). It is like a knob but with only two possible values.

All these elements can be synchronized with a MIDI keyboard connected to the computer if the MIDI input is correctly set in the [settings](#). Relative MIDI controller values are supported for driving the knobs.



EDITING PAGES

When a soundfont is opened, the **tree** is filled with the content of the soundfont. The editors will appear to the right of the tree when elements are selected:

- selecting the header **General** will show the editor of the **general information**,
- selecting one or more **samples** will show the **sample editor**,
- selecting **instruments** or **instrument divisions** will show the **instrument editor**,
- selecting **presets** or **preset divisions** will show the **preset editor**.

Polyphone Soundfont Editor

positif
Mezzo 29/07/2011 00:06:31

Product
Copyright
Comments

Soundfont version
2.01

ROM name and version
- (version 0.00)

Sound engine
E-mu 10K1

Editing software
Polyphone

Samples		Instruments		Presets	
Number	4	Number	2	Number	2
Samples	16 bits	Parameters	46 (≤ 65536)	Parameters	3 (≤ 65536)
		Modulators	2	Modulators	0

Search... /home/davy/Bureau/positif.sf2 (13.66 MB)

Editing pages



GENERAL INFORMATION

The page for editing the general information is displayed when the header **General** is selected in the . This page is made of three kinds of elements:

- **editable fields**,
- **general information** that are not editable,
- **statistics** about the soundfont.

The screenshot shows the Polyphone Soundfont Editor window. The title bar reads 'Polyphone Soundfont Editor'. The interface has a dark theme. On the left, there's a sidebar with a 'General' tab selected, and other tabs like 'Samples', 'Instruments', and 'Presets'. The main area displays the 'positif' soundfont. It includes fields for 'Product', 'Copyright', and 'Comments'. On the right, there's a summary of the soundfont's properties: 'Soundfont version 2.01', 'ROM name and version - (version 0.00)', 'Sound engine E-mu 10K1', and 'Editing software Polyphone'. At the bottom, there are three tables showing statistics for Samples, Instruments, and Presets. The status bar at the very bottom shows the file path '/home/davy/Bureau/positif.sf2 (13.66 MB)'.

Samples		Instruments		Presets	
Number	38	Number	2	Number	2
Samples	16 bits	Parameters	1432 (≤ 65536)	Parameters	3 (≤ 65536)
		Modulators	1	Modulators	0

Global information page

EDITABLE FIELDS

The editing page of the general information provides fields for changing the following attributes:

Title

This is the internal name of the soundfont, which does not necessarily have to do with the file name.

Author

Relates to the author name of the soundfont.

Date



Creation date of the file.

Product

Product for which is designed the soundfont.

Copyright

Mention to fill if the file is subject to copyright.

Comments

Information not included in any of the previous categories.

Kind of samples

Indicates whether the samples are stored with a 16-bit or 24-bit resolution. Please note that 24-bit samples may not be supported in some synthesizers.

GENERAL INFORMATION

The editing page of the general information comprises the following non-editable information:

Soundfont version

Version of the **sf2 format** used by the file.

ROM name and version

Name and version of a sample ROM if used.

Sound engine

Name of a sound engine.

Editing software

Editing software used for the soundfont creation.

File name

Address and name of the file.

File size

Size of the soundfont after it is saved.

STATISTICS

A count is made on:

- the number of **samples** and unused samples,
- the number of **instruments** and unused instruments,
- the number of **presets**,
- the number of parameters and modulators edited for the instruments,
- the number of parameters and modulators edited for the presets.

Unused samples and instruments can be easily removed with the tool “ **Remove unused elements**”.

Note: The number of 65536 parameters in the instruments and the presets should not be exceeded. Some synthesizers may not be adapted to read them all.

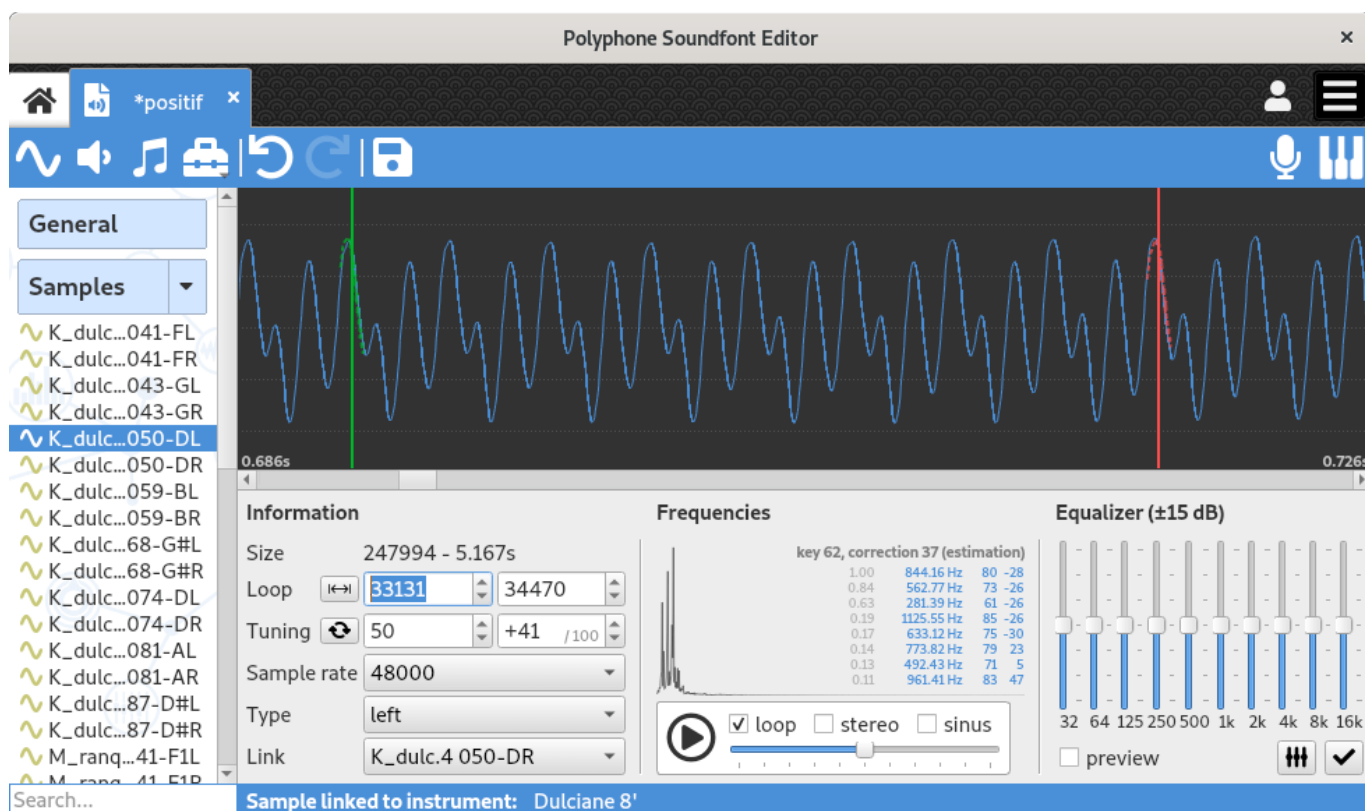


SAMPLE EDITOR

When clicking on one or more sample(s) in the **tree**, the editing page of the samples appears.

This page is divided into several parts:

- a **graph** representing the sample,
- an **information** area,
- an analyze of **frequencies**,
- an **equalizer**,
- a sample **player**.



Sample page

THE GRAPH

Zoom & drag

The graph allows the visualization of the sound wave. It is possible to zoom vertically or horizontally by holding a right mouse click and dragging:



- dragging right expands the horizontal axis,
- dragging left contracts the horizontal axis,
- dragging up expands the vertical axis,
- dragging down contracts the vertical axis.

Holding the left mouse button allows a horizontal move of the graph, if the horizontal axis has been expanded.

Loop and playing positions

A green vertical bar indicates the start position of the loop (changeable by left click), a red bar indicates the position of the end of the loop (changeable by right click). A dotted line representing a wave portion is associated with each vertical bar, useful for visualizing the superimposition of the beginning and the end of the loop.

During playback, a white vertical bar runs across the graph and indicates the playback position in real time.

Cutting the sample

When holding the **Alt** keyboard button, it is possible to define an area to cut:

- a left mouse button press defines the start of the area,
- the corresponding left mouse button release defines the end of the area.

Once an area is defined, a dialog appears to ask for a confirmation before cutting the sample.



Cutting sample



Colors of the graph

Graph colors are editable in the software [preferences](#).

SECTION "INFORMATION"

The section "Information" allows the editing of the following information:

- start and end of the loop,
- root key of the sample and its correction to be applied for the sound to be tuned in the equal temperament,
- sound sample rate (a change will cause re-sampling),
- sound type (not linked: mono, otherwise left, right or linked),
- the linked sample if the sample is stereo.

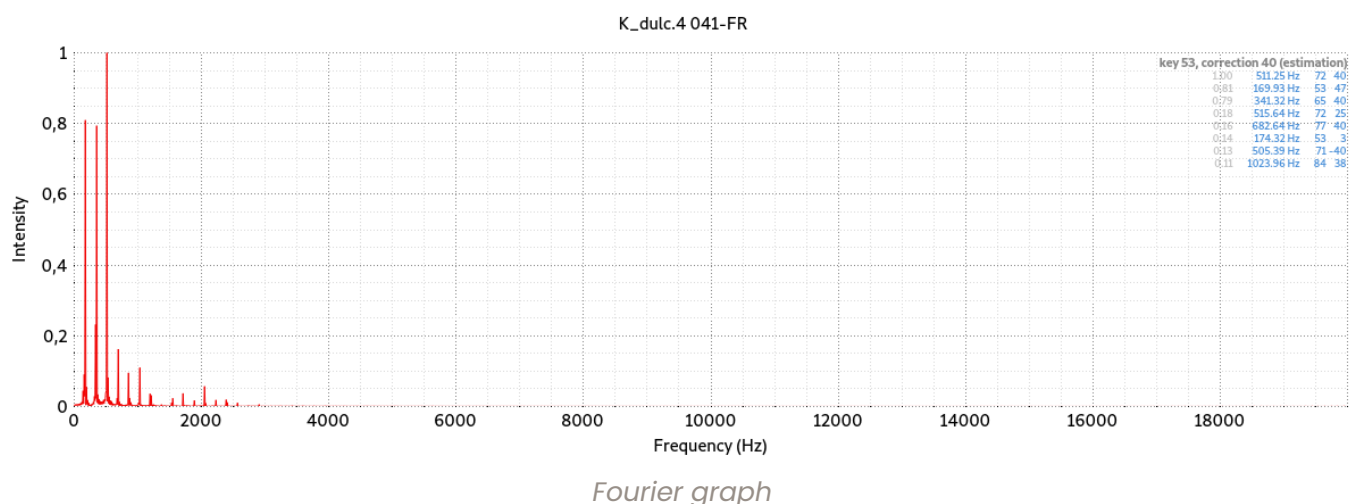
In addition, the sample size in samples (number of values) and seconds is shown.

A button edits the start and end of the loop so that the loop is the entire sample. Another button edits the root key and the correction so that they match with the evaluation of the [frequency](#).

Any change on a stereo sample may be applied on the linked sample if the option **Stereo editing** is ticked in the [preferences](#). Via a multiple selection in the [tree](#), several samples may be edited simultaneously.

FREQUENCY ANALYSIS

The frequency analysis includes a Fourier transform (graph showing the intensity of frequencies contained in the signal).



On the right is shown the list of dominant peaks with the following information:

- ratio between the peak intensity and the intensity of the maximum peak,



- frequency,
- closest corresponding key,
- correction (distance to the closest corresponding key).

From these data an estimate of the pitch of the sound is made. This estimate can be used for tuning.

A right click on the graph opens a contextual menu that allows you to:

- export the graph into a .png image,
- show the list of all peaks and a button for exporting this list into a .csv file.

Show peak frequencies...					
Sample	Peak	Intensity	Frequency	Key	Correction
F_bourdon rp 8 054L	1	1	184.384 Hz	54	6
	2	0.26137	554.497 Hz	73	0
	3	0.177638	180.346 Hz	54	44
	4	0.155903	188.421 Hz	54	-32
	5	0.0834036	176.308 Hz	53	-17
	6	0.0822446	192.459 Hz	55	32
	7	0.054803	172.271 Hz	53	23
	8	0.0544476	196.496 Hz	55	-4
F_bourdon rp 8 060L	1	1	261.098 Hz	60	3
	2	0.0864793	783.294 Hz	79	2
	3	0.070302	1567.93 Hz	91	0
	4	0.070047	269.173 Hz	60	-49
	5	0.0581711	248.985 Hz	59	-14
<div><div>Export</div><div>Close</div></div>					

Show peak frequencies

EQUALIZER

The equalizer makes it possible to increase or decrease some frequencies, from -15 to 15 dB. It is possible to listen to the effect with the option **preview** enabled and then definitely apply the effect by pressing the **Apply** button. Pressing **reset** resets all the buttons to 0 dB (but does not remove the effect on the sound: for that use the **undo** button in the **toolbar**).

Any change on a stereo sample may be automatically applied on the linked sample if the option is checked in the **preferences**.

PLAYER



The player plays the sound. During playback, a white cursor runs across the graph to indicate the playback position. Playback options are:

Loop

Plays the sound loop if the beginning and end of the loop are completed.

Stereo

Plays the sample in stereo, with its linked sample. If looping is also required, the two respective loops are used (they can be different).

Sine

A sine calibration signal is added to the sound, allowing a sample tuning to the nearest pitch hundredth. At first the root key must be filled. At this level, beats can be heard if the frequency of the sine differs slightly from the fundamental frequency of the sound. It remains to finely adjust the correction to make the beats as slow as possible, so that the sample is tuned at best. See the tutorial " [How to prepare a sample](#)" for more information.

If a playback is in progress when switching from one sample to another through the [tree](#), it automatically restarts. Also, pressing the key `space` enables or disables playback if you are navigating in the tree.



INSTRUMENT EDITOR

When clicking on one or more instrument(s) in the **tree**, the editing page of instruments appears.

This page comprises several parts:

- the **table**,
- the **range editor** when the button **Edit ranges** from the **toolbar** is clicked,
- the **envelope editor** when the button **Edit envelope** from the **toolbar** is clicked,
- the **modulator section**.

	Global	K_dulc.4 0 41-FL	K_dulc.4 0 41-FR	K_dulc.4 0 43-GL	K_dulc.4 0 43-GR	K_dulc.4 0 50-DL	K_dulc.4 0 50-DR	K_dulc.4 0 50-DL	K_dulc.4 0 50-DR
Key range		36-45	36-45	46-55	46-55	56	56	57	57
Velocity range									
Attenuation (dB)		39.3	39.3	41.2	41.2	34.7	34.7	34.7	34.7
Pan [-50;50]		50.0	-50.0	50.0	-50.0	50.0	-50.0	50.0	-50.0
Loop playback									
Sample root key		41	41	43	43	50	50	50	50
Root key		53	53	55	55	62	62	62	62
Tuning (semi-tones)									
Tuning (cents)									
Scale tuning									
Filter, cutoff (Hz)									
Filter, resonance (dB)									
Vol env delay (s)									
Vol env attack (s)									
Vol env hold (s)									

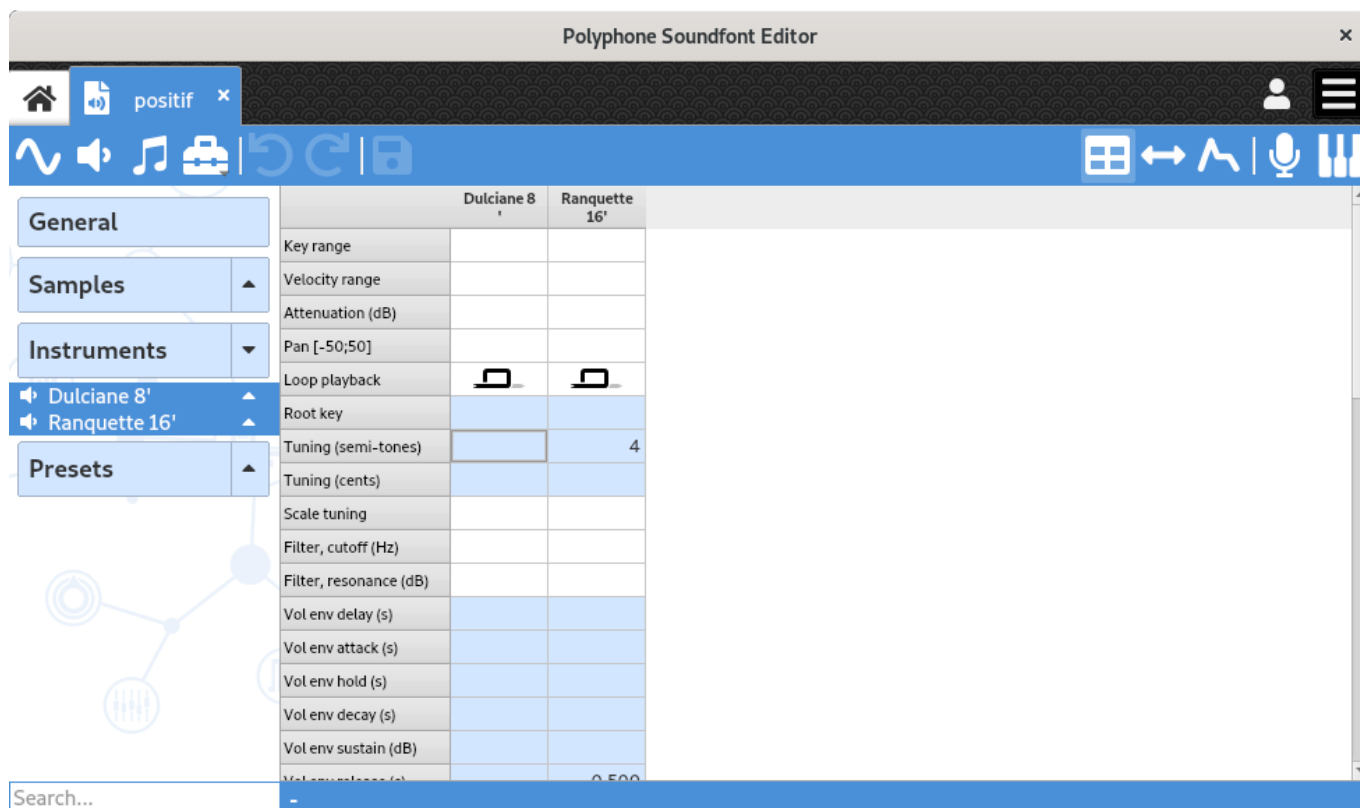
▲ 1 modulator: Tuning (semi-tones)

Search... Instrument linked to preset: Dulciane 8'

Instrument page

If only one instrument is selected, the page allows the editing of the global parameters of the instrument and the parameters of all its divisions.

In the case where several instruments are selected, the global parameters of each of these instruments can be edited.



Instrument page with multiple instruments selected

TABLE

Presentation

The table allows a global view and the editing of an instrument configuration. A change of a parameter in the **Global** division applies to all divisions, except for divisions whose same parameter is already specified (local override).

The editable parameters are:

Key range

Key range for which the sample will be heard.

Velocity range

Velocity range for which the sample will be heard. It is possible to set different sounds for the same note, with different velocity ranges, to distinguish different nuances of a piano key for example.

Attenuation (dB)

Attenuation in decibel applied to the sample.

Note: the sf2 format doesn't allow this value to be negative, no amplification is thus possible.



Pan

Balance applied to the sound, to hear it to the left or to the right in stereo.

Sample playback

→ indicates a sound reproduced with no loop. ↻ indicates a sound which loops



continuously.  is for triggering a sample playback when a key is released. The volume envelop is still used for defining the volume and a short attack (0.2 ms) is added for smoothing the start of the sample on the release.  indicates a sound which loops for the duration of key depression and then proceeds to play the remainder of the sample. Note: the playback mode "release" is NOT documented by the sf2 format, so other players such as FluidSynth may decide not to implement it. The other way to achieve this behavior is to double each sample with a sample containing a short silent loop followed by the sound to emit on the release. The added sample will be read with the playback parameter "loop on + release".

Root key

Allows you to change the normal root key of a sample to a different note from that usually used.

Tuning

Change the tuning in semitones and cents of semitones. Changing "scale" plays on the difference between two consecutive notes.

Filter

Apply a low-pass filter, by specifying the cutoff frequency and the resonance.

Volume envelope

Change the volume envelope, and possibly how it changes with the note.

Modulation

Set a modulation (envelope, delay, frequency, changes with the note) and its effect on the tone, filter and volume.

Vibrato

Set a vibrato (delay, frequency) and its effect on the tone.

Exclusive class

This parameter can define one or more exclusive classes, by assigning to a set of divisions within a class the same parameter value other than 0. When an exclusive class is defined, any note triggered from one of the divisions of the exclusive class ends all the other sounds of the same class. The scope of an exclusive class is the preset in which the class is defined (a sound triggered in an instrument will end a sound of another instrument if the two instruments are in the same preset). This effect can be used for percussion instruments like a cymbal. Basically playing one note automatically terminates another in the same group.

Chorus and reverb

Indicates the intensity of chorus and reverb effects. The effects are configurable in the software [settings](#).

Fixed key and velocity

Force the value of the key and of the velocity on the full extent of the division.

Offsets

Changes the positions of the start and end of a sound during playback, as well as the start and end of the loop. The unit of an offset is samples (number of values).

Editing facilities in the table

To facilitate and speed up editing in the table, the following features have been implemented:

- the mouse wheel may be used to change a value (the cell has to be in edit mode first),
- an external keyboard can be used to set the rootkey, key range and velocity range (the cell has to be in edit mode first),
- a multiple selection, followed by the editing of the last cell, will edit all cells simultaneously,
- the keys `Del` and `Backspace` clear the content of all selected cells,
- stereo divisions may be edited simultaneously if the option is enabled in the software [preferences](#),



- key names may be numeric or as a text (having C3, C4 or C5 as middle C), depending on the option chosen in the [preferences](#),
- copy / paste / cut of a set of cells (not necessarily adjacent) may be done via **Ctrl+C** / **Ctrl+V** / **Ctrl+X**.

Do not forget the [tools](#) available! In particular the tool allowing a [key-based configuration](#).

Possibility to mute divisions

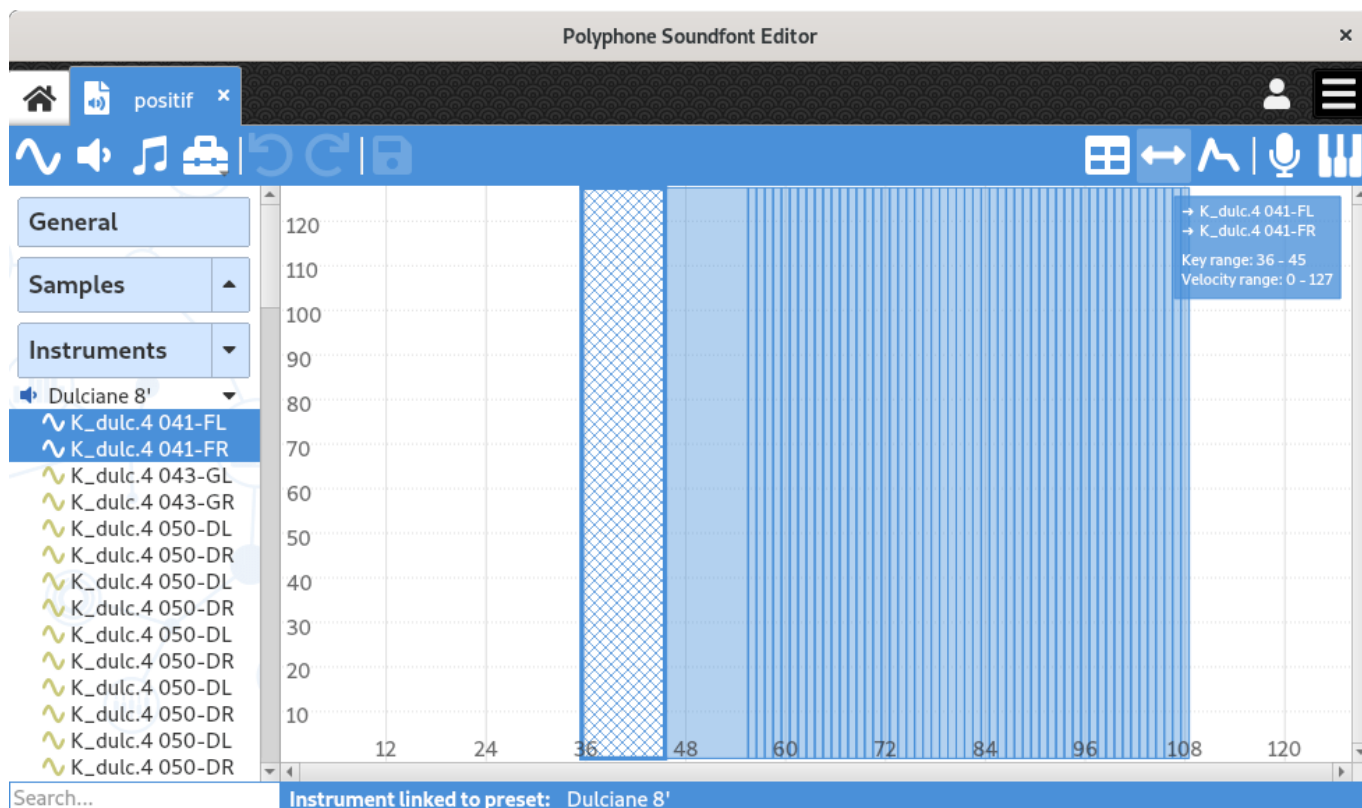
For testing purposes, it is possible to mute a division or mute all divisions but one in the tables. The options are accessible with a right click on the header of a division, and also in the contextual menu of the [tree](#).

Global	Koto A2	Koto E3	Koto G3	Koto G5
	0-61			08
	16.0			.7
	0.4	0.4	0.4	0.4
	60	60	60	60
	68	73	77	101

Mute divisions in a table

RANGE EDITOR

The range editor allows you to quickly and graphically dispose samples according to their key and velocity ranges. This graph further highlights configuration problems, such as zones free of samples or unwanted superpositions.



Range editor

Each sample is represented by a semitransparent rectangle, so that the superposition of several elements is identifiable.

Rectangle selection

A simple left click on a rectangle will select it and removes all other selections. If several rectangles are superimposed, consecutive clicks change the selection under the mouse cursor.

A multiple selection is possible:

- by maintaining button **Ctrl** and then clicking on the different rectangles to select one by one,
- by using button **Shift**: all rectangles between the first rectangle being clicked and the next one will be selected.

Updating the rectangles

After having selected one or more rectangle(s), it is possible with the left button to:

- change the size, when you grab one of the edges of a rectangle;
- change the position, by pointing to the center of a rectangle.

If several rectangles are selected, all changes are made synchronously.



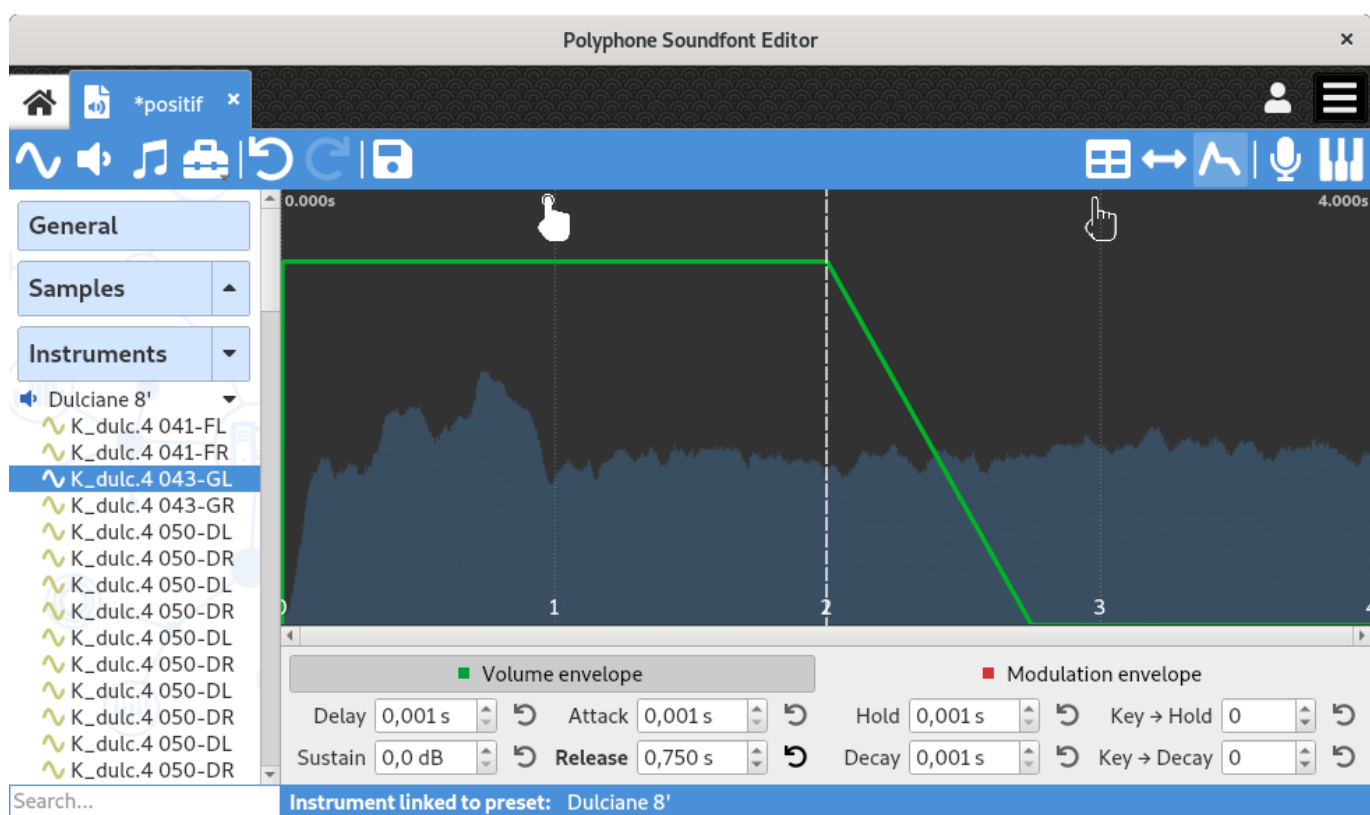
Drag & zoom

Maintaining the right click allows you to change the zoom around the selected point.

Maintaining the left click allows you to move the graphic if the initial point is not on a rectangle.

ENVELOPE EDITOR

The envelope editor provides a graphical overview of the 8 parameters defining the volume envelope and the 8 parameters defining the modulation envelope. This is thus a tool for better understanding the impact of each of these parameters.



Envelope editor

The buttons above the graph select alternatively the volume envelope (displayed in green) or the modulation envelope (displayed in red) for the editing. During the editing of the modulation envelope, the volume envelope is displayed with a thin line.

The left part of the graph represents the evolution of the envelope when a key is **triggered**. 7 out of the 8 parameters are involved in this representation:

Delay

Duration during which the envelope remains at 0 just after a key is triggered. In the majority of cases we want to hear audio immediately after pressing the keys and then this parameter is left at zero. However **delay** comes in handy for example when we want to create richer pad



sounds constructed with several layers. Suppose we want to introduce a second sound layer 0.5 seconds after the key is pressed, then we can set it by entering "0.5" in this -section. A delay does not hold the playback of the sound - it mutes the audio. If we have set the delay to 0.5 seconds, we will start hearing audio after 0.5 seconds of the sample that has already been played.

Attack

Duration after **delay** during which the envelope grows from 0 to the maximum value. For example a bowed string sound like a violin has an attack which is quite long, while a plucked string sound such as a guitar is very short. Even when we want to use an attack as short as possible, an advise is to at least enter a value of 0.001 otherwise a "click" might be heard after pressing a key.

Hold

Duration after **attack** during which the envelope keeps its maximum value.

Decay

Duration after **hold** during which the envelope progressively goes from the maximum value to the value specified by **sustain**. When the **sustain** value is set to its maximum or is not set, the volume during the **decay** will stay at its maximum no matter which value is specified.

Sustain

Value after **decay** sustained as long as the triggered key is not released. Just as **attenuation** the sustain value refers to the amount of decibel-reduction. In the case where we want to reduce the volume by its half after the decay-time, a parameter of 6 in the sustain-section must be set. Number "144" can be specified if we want to completely mute the sound.

Key → Hold

Change of the **hold** duration depending on the key (see below).

Key → Decay

Change of the **decay** duration depending on the key (see below).

A positive value in **Key → XXX** gradually decreases the duration of XXX toward the trebles, whereas a negative value does the opposite. For instance:

- value 100 halve the duration of XXX per octave above middle C (60) and double the duration of XXX per octave below middle C,
- value -100 double the duration of XXX per octave above middle C (60) and halve the duration of XXX per octave above middle C.

These parameters can be useful when designing a guitar or a piano with the sustain-pedal on: the lower the tone the longer it lasts until we hear no sound at all.

The right part represents the evolution of the envelope when a key is **released**. Only the **release** duration is involved in this representation. The envelope gradually returns to 0 during this duration. Sounds of violins or synth-pads have a relatively long release time, while that of other sounds such as organs are very short. In practice for short release times we still have to set its duration to at least 0.2 seconds, because a release time which is too short doesn't sound pleasant or natural (however it also depends what we have in mind about the construction of the instrument we want to create).

When only one division of an instrument is selected in the tree, the corresponding sound **sample** is displayed on the background. It is thus possible to better visualize the evolution of the envelope relative to the sample being played.

MODULATORS

The lower section of the editing page of instruments is intended for the creation of modulators, globally for an instrument or for a particular division. When you play an instrument, a modulator



listens to one or two MIDI signal(s) and modify one of the parameters in the table according to the mathematical rule configured. The output of a modulator can also be used as input to another modulator (this function appeared in version 2.04 of the [sf2 format](#) and may not be supported by some synthesizers).

	Global	K_dulc.4 0 41-FL	K_dulc.4 0 41-FR	K_dulc.4 0 43-GL	K_dulc.4 0 43-GR	K_dulc.4 0 50-DL	K_dulc.4 0 50-DR	K_dulc.4 0 50-DL	K_dulc.4 0 50-DR
Key range		36-45	36-45	46-55	46-55	56	56	57	57
Velocity range									
Attenuation (dB)		39.3	39.3	41.2	41.2	34.7	34.7	34.7	34.7
Pan [-50;50]		50.0	-50.0	50.0	-50.0	50.0	-50.0	50.0	-50.0
Loop playback									
Sample root key		41	41	43	43	50	50	50	50
Root key		53	53	55	55	62	62	62	62
Tuning (semi-tones)									

+

Modulator #1

Poly pressure

CC-016 Gener

0 → 1

×

× 5

-5 ↔ 5

no operation

Tuning (semi-)

Add from: -5 semi-tones

To: 5 semi-tones

Modulator #2

CC-017 Gener

CC-012 Effect

0 → 1

×

× 1200

-1200 ↔ 1200

no operation

Vol env releas

Multiply from: 0.5

To: 2

Modulator editor

Buttons on the left are made for:

- expanding / collapsing the modulator area,
- adding / removing a modulator,
- copying / pasting / duplicating one or more modulators.

When a modulator is added, the border of the corresponding cell in the table is thicker (see the tuning in the **Global** column above).

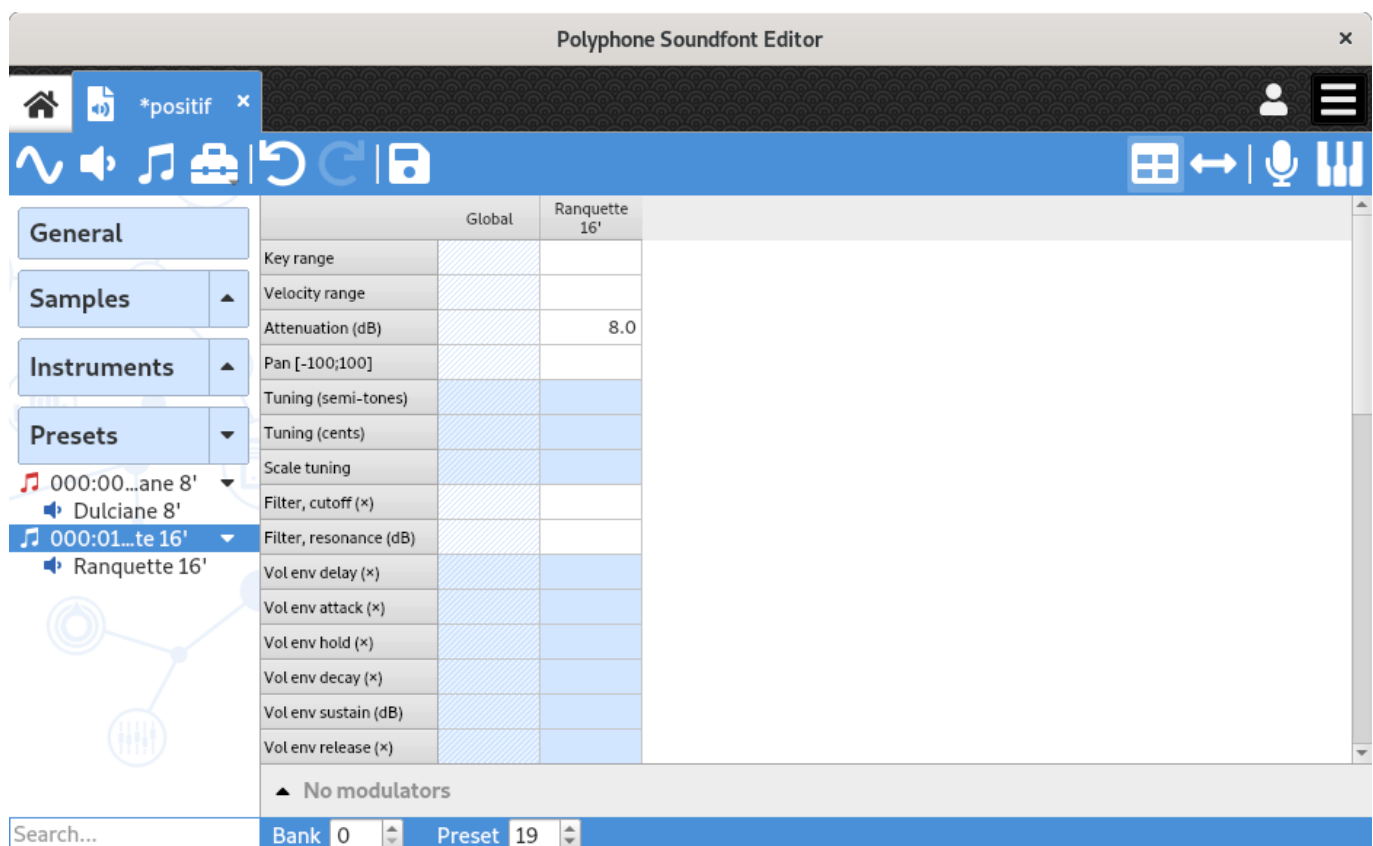


PRESET EDITOR

When clicking on one or more preset(s) in the [tree](#), the editing page of presets appears. A preset is a combination of one or more instruments and is the element visible from outside a [soundfont](#).

Similarly to the [instrument editor](#), several parts are present:

- the table,
- the range editor when the button **Edit ranges** of the [toolbar](#) is clicked,
- the modulator section.



Preset page

If only one preset is selected, the page allows the editing of the global parameters of the preset and the parameters of all its divisions. In the case where several presets are selected, the global parameters of each of these presets can be edited.

DIFFERENCES WITH THE INSTRUMENTS

The parameters edited here are said relative unlike parameters of an instrument which, except in



special cases, are said absolute. Values of a preset parameters are added or multiplied to the instrument parameters that compose it. Some parameters are not editable at the preset level:

- loop playback,
- root key,
- exclusive class,
- fixed key,
- fixed velocity,
- the four offsets.

A preset is identified by a preset number and bank number, this combination being unique and adjustable below the table. Each bank comprises 127 presets.

The last bank (128) is usually used for percussion kits.



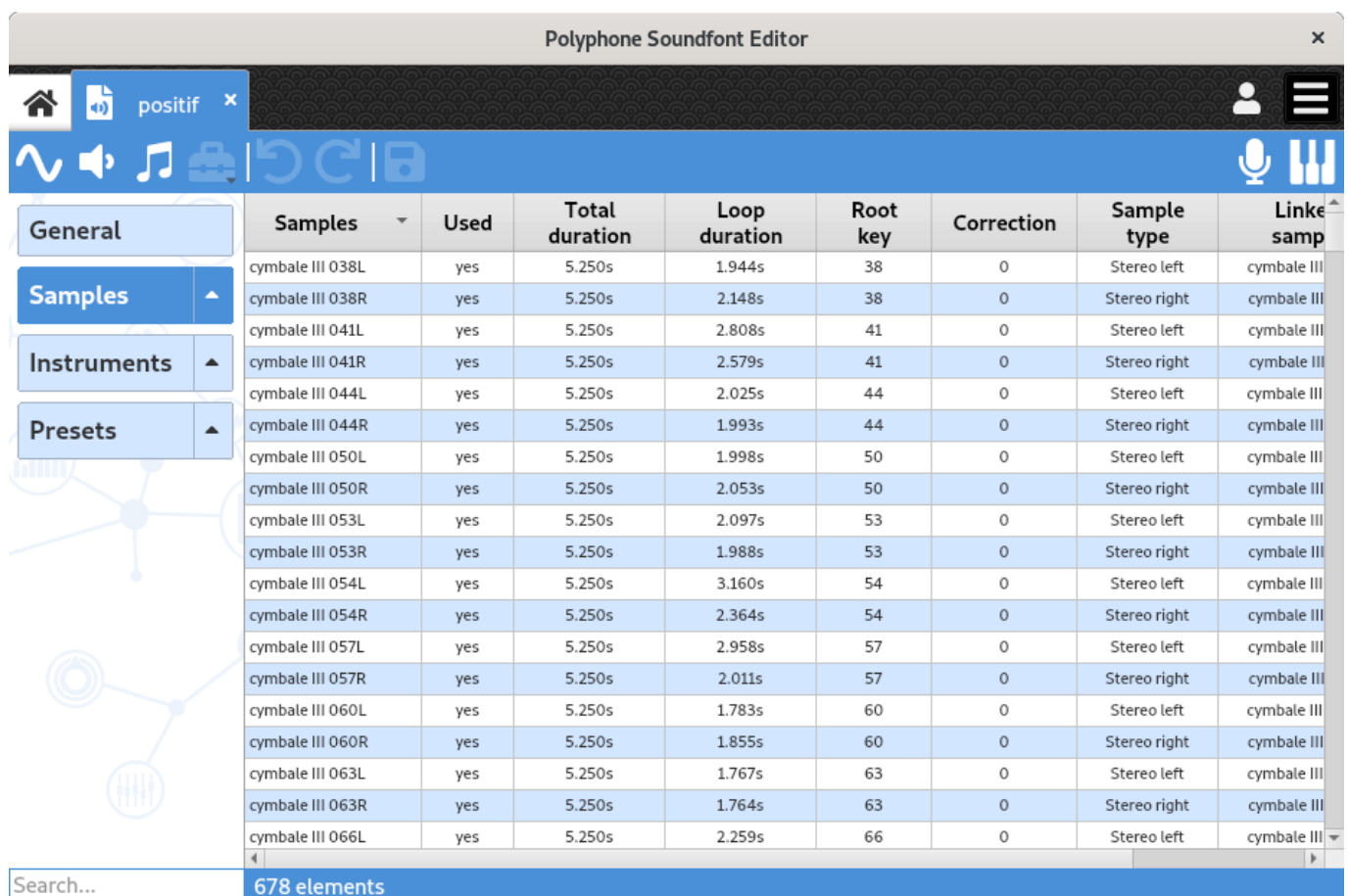
CONFIGURATION SUMMARIES

The configuration summaries provide an overview of all elements contained in a soundfont. They are accessible via the headers **Samples**, **Instruments** and **Presets** in the **tree**, and allow the visualization of their corresponding elements.

Information gathered in these pages can be used to support inspections within large soundfonts.

SAMPLES

When clicking on the **Samples** header of a soundfont in the **tree**, the sample configuration summary appears.



The screenshot shows the Polyphone Soundfont Editor window. The 'Samples' header is selected in the left sidebar. The main area displays a table with the following columns: Samples, Used, Total duration, Loop duration, Root key, Correction, Sample type, and Linke samp. The table lists 20 cymbale III samples, each with a duration of 5.250s and a loop duration of 1.944s to 2.259s. The 'Used' column is set to 'yes' for all samples. The 'Sample type' column is set to 'Stereo left' or 'Stereo right'. The 'Linke samp' column is set to 'cymbale III'. A search bar at the bottom left shows '678 elements'.

Samples	Used	Total duration	Loop duration	Root key	Correction	Sample type	Linke samp
cymbale III 038L	yes	5.250s	1.944s	38	0	Stereo left	cymbale III
cymbale III 038R	yes	5.250s	2.148s	38	0	Stereo right	cymbale III
cymbale III 041L	yes	5.250s	2.808s	41	0	Stereo left	cymbale III
cymbale III 041R	yes	5.250s	2.579s	41	0	Stereo right	cymbale III
cymbale III 044L	yes	5.250s	2.025s	44	0	Stereo left	cymbale III
cymbale III 044R	yes	5.250s	1.993s	44	0	Stereo right	cymbale III
cymbale III 050L	yes	5.250s	1.998s	50	0	Stereo left	cymbale III
cymbale III 050R	yes	5.250s	2.053s	50	0	Stereo right	cymbale III
cymbale III 053L	yes	5.250s	2.097s	53	0	Stereo left	cymbale III
cymbale III 053R	yes	5.250s	1.988s	53	0	Stereo right	cymbale III
cymbale III 054L	yes	5.250s	3.160s	54	0	Stereo left	cymbale III
cymbale III 054R	yes	5.250s	2.364s	54	0	Stereo right	cymbale III
cymbale III 057L	yes	5.250s	2.958s	57	0	Stereo left	cymbale III
cymbale III 057R	yes	5.250s	2.011s	57	0	Stereo right	cymbale III
cymbale III 060L	yes	5.250s	1.783s	60	0	Stereo left	cymbale III
cymbale III 060R	yes	5.250s	1.855s	60	0	Stereo right	cymbale III
cymbale III 063L	yes	5.250s	1.767s	63	0	Stereo left	cymbale III
cymbale III 063R	yes	5.250s	1.764s	63	0	Stereo right	cymbale III
cymbale III 066L	yes	5.250s	2.259s	66	0	Stereo left	cymbale III

Sample configuration summary

For each sample, this page allows for example to perform the following checks:

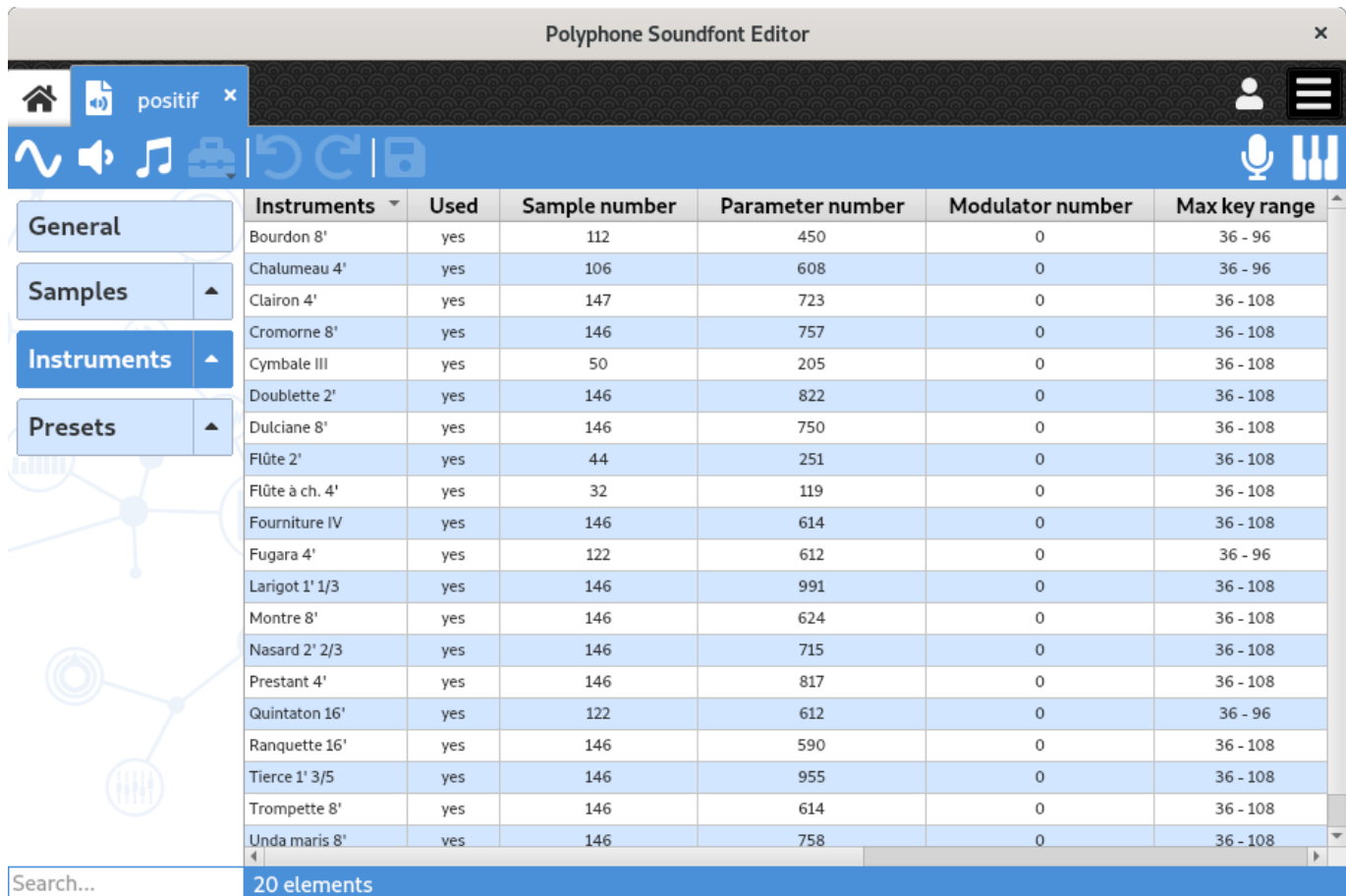
- Is it used?



- Is its total duration (and possibly its loop duration) long enough?
- Has the tuning been done?
- Is it properly linked in stereo?
- Is the sampling rate high enough for a good audio quality?

INSTRUMENTS

When clicking on the **Instruments** header of a soundfont in the **tree**, the instrument configuration summary appears.



The screenshot shows the Polyphone Soundfont Editor window. On the left is a sidebar with a 'tree' view containing 'General', 'Samples', 'Instruments' (selected), and 'Presets'. The main area displays a table of instrument configurations. The table has columns: Instruments, Used, Sample number, Parameter number, Modulator number, and Max key range. There are 20 elements listed in the table. At the bottom left is a search bar, and at the bottom right, it says '20 elements'.

Instruments	Used	Sample number	Parameter number	Modulator number	Max key range
Bourdon 8'	yes	112	450	0	36 - 96
Chalumeau 4'	yes	106	608	0	36 - 96
Clairon 4'	yes	147	723	0	36 - 108
Cromorne 8'	yes	146	757	0	36 - 108
Cymbale III	yes	50	205	0	36 - 108
Doublette 2'	yes	146	822	0	36 - 108
Dulciane 8'	yes	146	750	0	36 - 108
Flûte 2'	yes	44	251	0	36 - 108
Flûte à ch. 4'	yes	32	119	0	36 - 108
Fourniture IV	yes	146	614	0	36 - 108
Fugara 4'	yes	122	612	0	36 - 96
Larigot 1' 1/3	yes	146	991	0	36 - 108
Montre 8'	yes	146	624	0	36 - 108
Nasard 2' 2/3	yes	146	715	0	36 - 108
Prestant 4'	yes	146	817	0	36 - 108
Quintaton 16'	yes	122	612	0	36 - 96
Ranquette 16'	yes	146	590	0	36 - 108
Tierce 1' 3/5	yes	146	955	0	36 - 108
Trompette 8'	yes	146	614	0	36 - 108
Unda maris 8'	yes	146	758	0	36 - 108

Instrument configuration summary

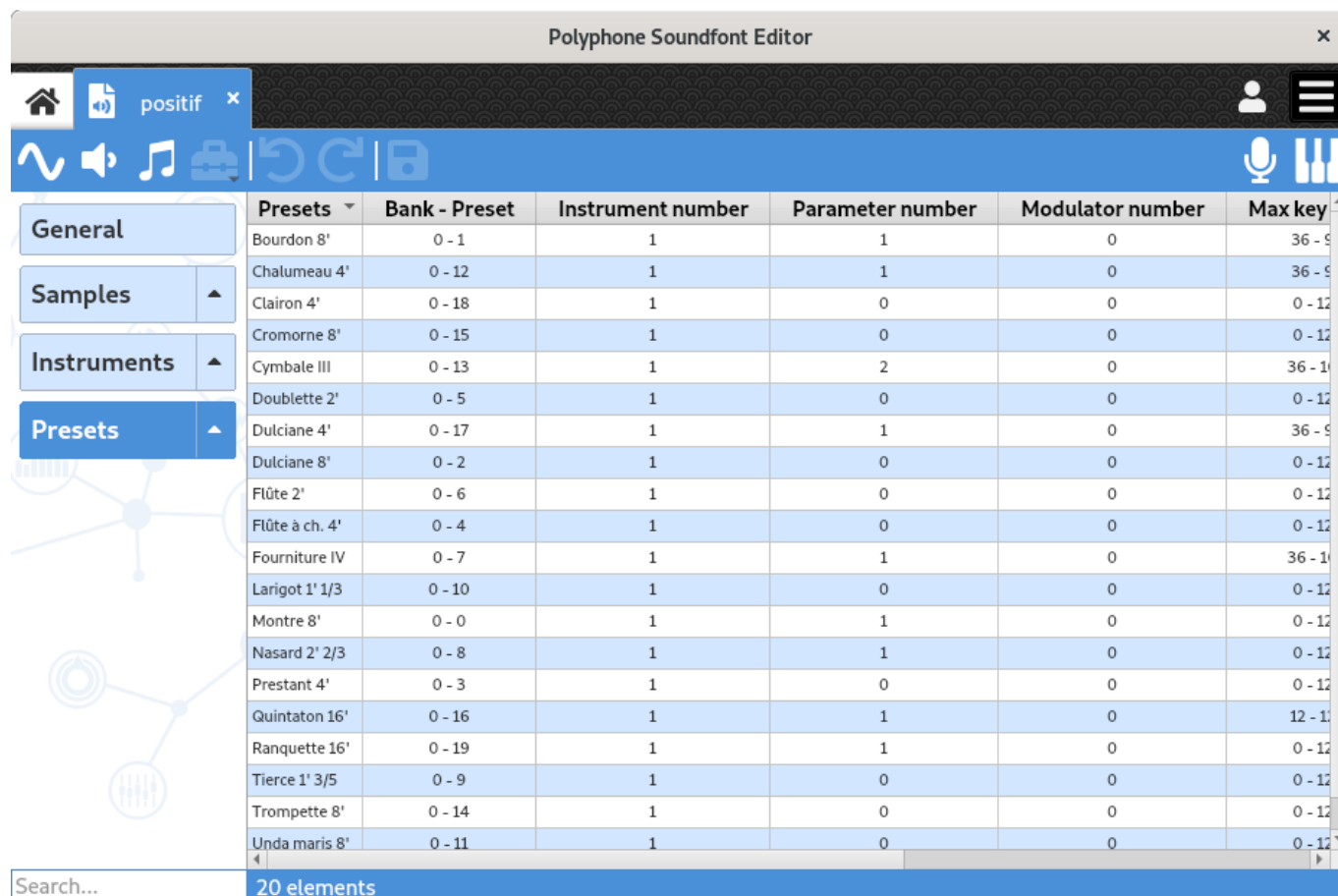
For each instrument, this page allows for example to perform the following checks:

- Is it used?
- Are the numbers of linked samples, parameters and modulators consistent?
- Are the key and velocity ranges correct?
- Have the samples been properly attenuated?
- Is the loop playback on?
- Have the chorus and reverb settings not been forgotten?



PRESETS

When clicking on the **Presets** header of a soundfont in the **tree**, the preset configuration summary appears.



The screenshot shows the 'Polyphone Soundfont Editor' window. On the left is a sidebar with a 'tree' view containing categories: General, Samples, Instruments, and Presets (which is selected). The main area displays a table titled 'Preset configuration summary' with the following data:

Presets	Bank - Preset	Instrument number	Parameter number	Modulator number	Max key
Bourdon 8'	0 - 1	1	1	0	36 - 9
Chalumeau 4'	0 - 12	1	1	0	36 - 9
Clairon 4'	0 - 18	1	0	0	0 - 12
Cromorne 8'	0 - 15	1	0	0	0 - 12
Cymbale III	0 - 13	1	2	0	36 - 10
Doublette 2'	0 - 5	1	0	0	0 - 12
Dulciane 4'	0 - 17	1	1	0	36 - 9
Dulciane 8'	0 - 2	1	0	0	0 - 12
Flûte 2'	0 - 6	1	0	0	0 - 12
Flûte à ch. 4'	0 - 4	1	0	0	0 - 12
Fourniture IV	0 - 7	1	1	0	36 - 10
Larigot 1' 1/3	0 - 10	1	0	0	0 - 12
Montre 8'	0 - 0	1	1	0	0 - 12
Nasard 2' 2/3	0 - 8	1	1	0	0 - 12
Prestant 4'	0 - 3	1	0	0	0 - 12
Quintaton 16'	0 - 16	1	1	0	12 - 11
Ranquette 16'	0 - 19	1	1	0	0 - 12
Tierce 1' 3/5	0 - 9	1	0	0	0 - 12
Trompette 8'	0 - 14	1	0	0	0 - 12
Unda maris 8'	0 - 11	1	0	0	0 - 12

At the bottom of the table, there is a search bar with the text 'Search...' and a summary bar indicating '20 elements'.

Preset configuration summary

For each preset, this page allows for example to perform the following checks:

- Have the bank and preset numbers been properly filled?
- Are the numbers of linked instruments, parameters and modulators consistent?
- Are the key and velocity ranges correct?
- Have the instruments been accurately attenuated?
- Have the chorus and reverb settings (as offset) not been forgotten?



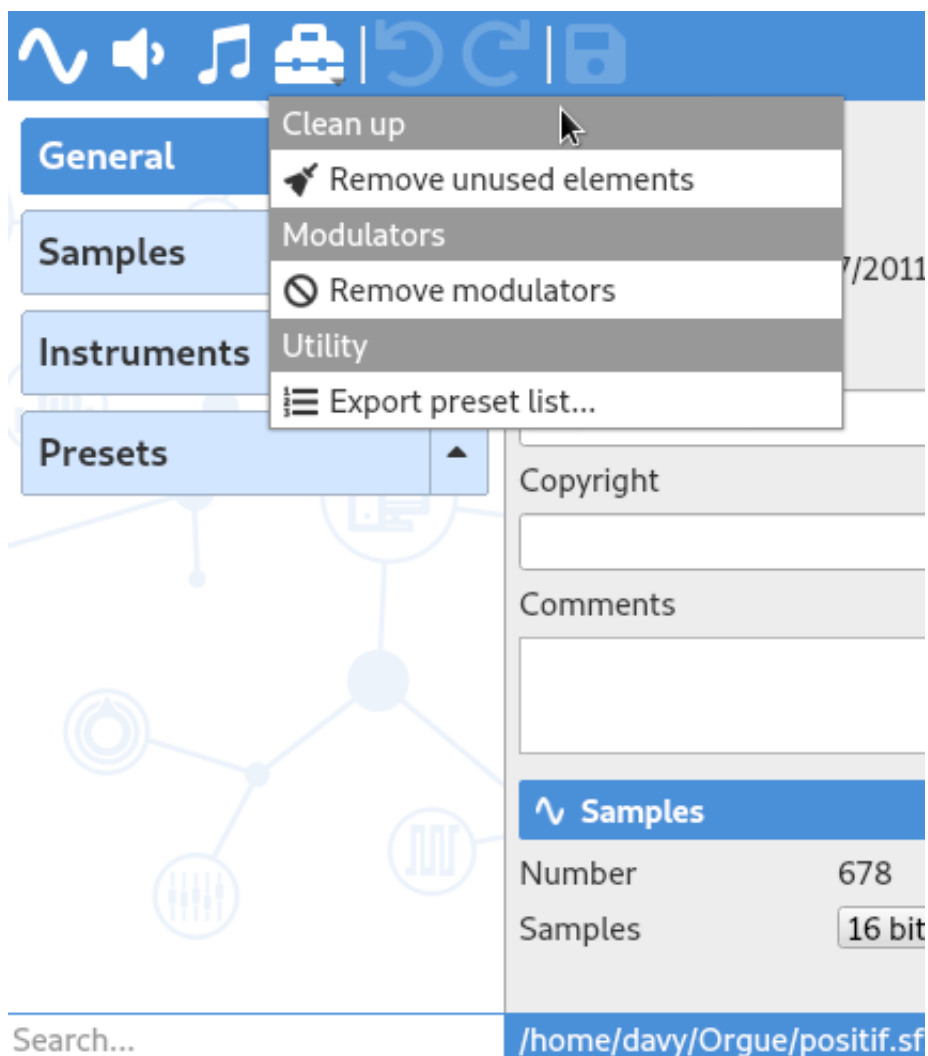
Tools

Once you understand how soundfonts work (through the different [editors](#)), tools are probably what you should care of. They are all designed to fasten / automate the editing and to avoid common errors or imprecision that a human usually does.

Tools are all available in a menu that is displayed when clicking on the toolbox icon in the [toolbar](#). The content of this menu depends on the selection in the [tree](#): first on the kind of selection (as described below) but also on the number of selected elements. A few tools cannot process multiple elements so they will be hidden with a multiple selection.

GLOBAL TOOLS

When the header **General** is selected in the [tree](#), the tool menu is filled with [global tools](#).

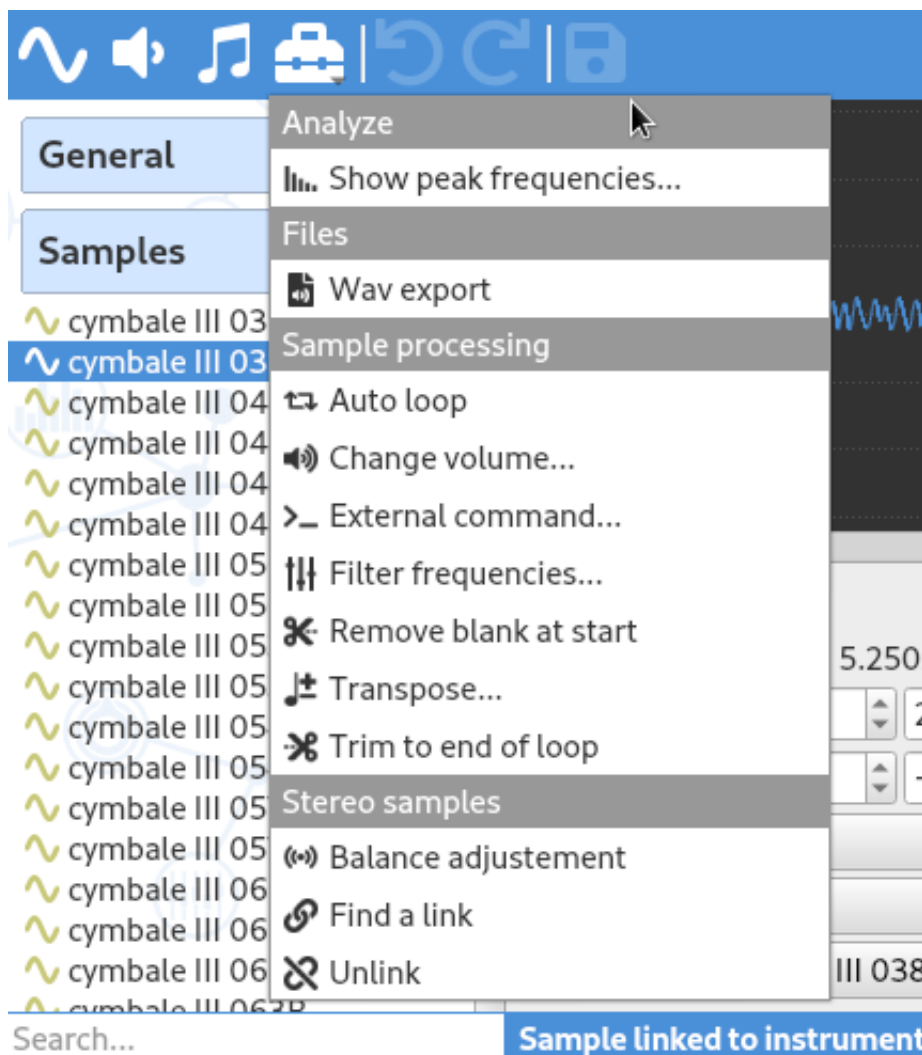




Tool menu showing global tools

SAMPLE TOOLS

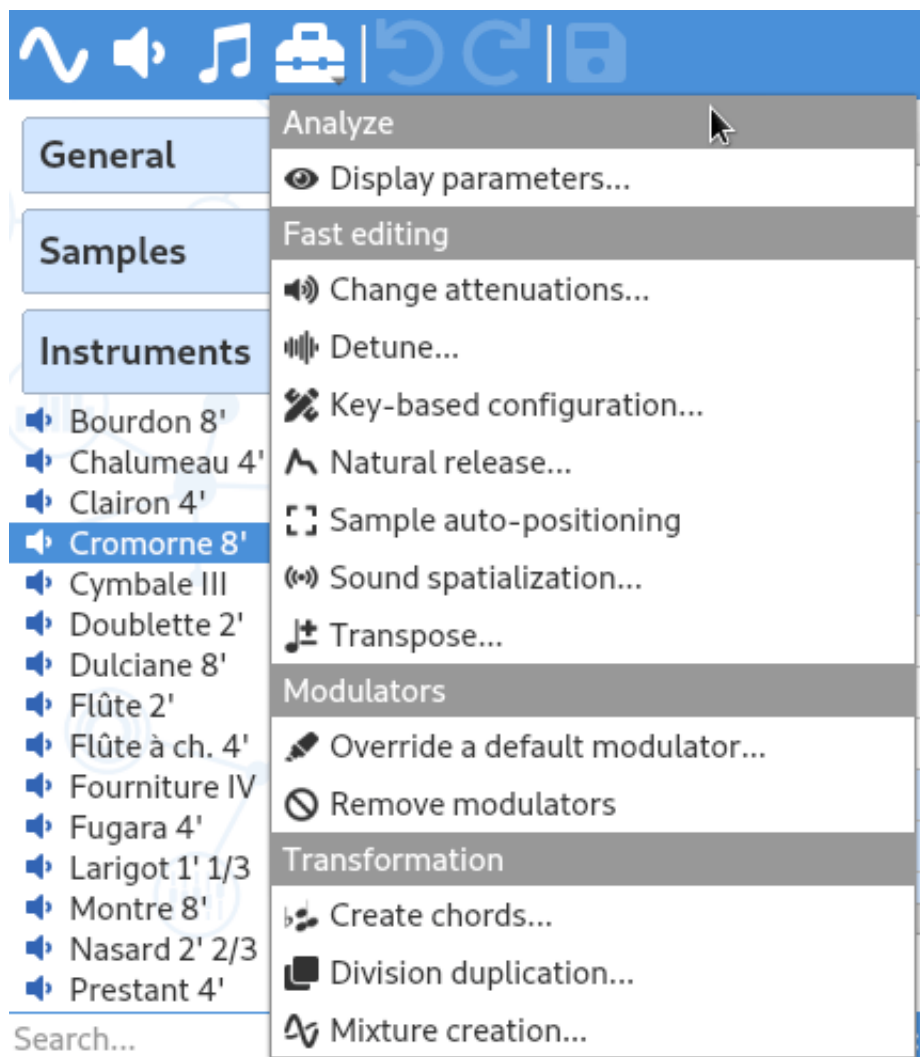
When one or more samples are selected in the **tree**, the tool menu is filled with **sample tools**.



Tool menu showing sample tools

INSTRUMENT TOOLS

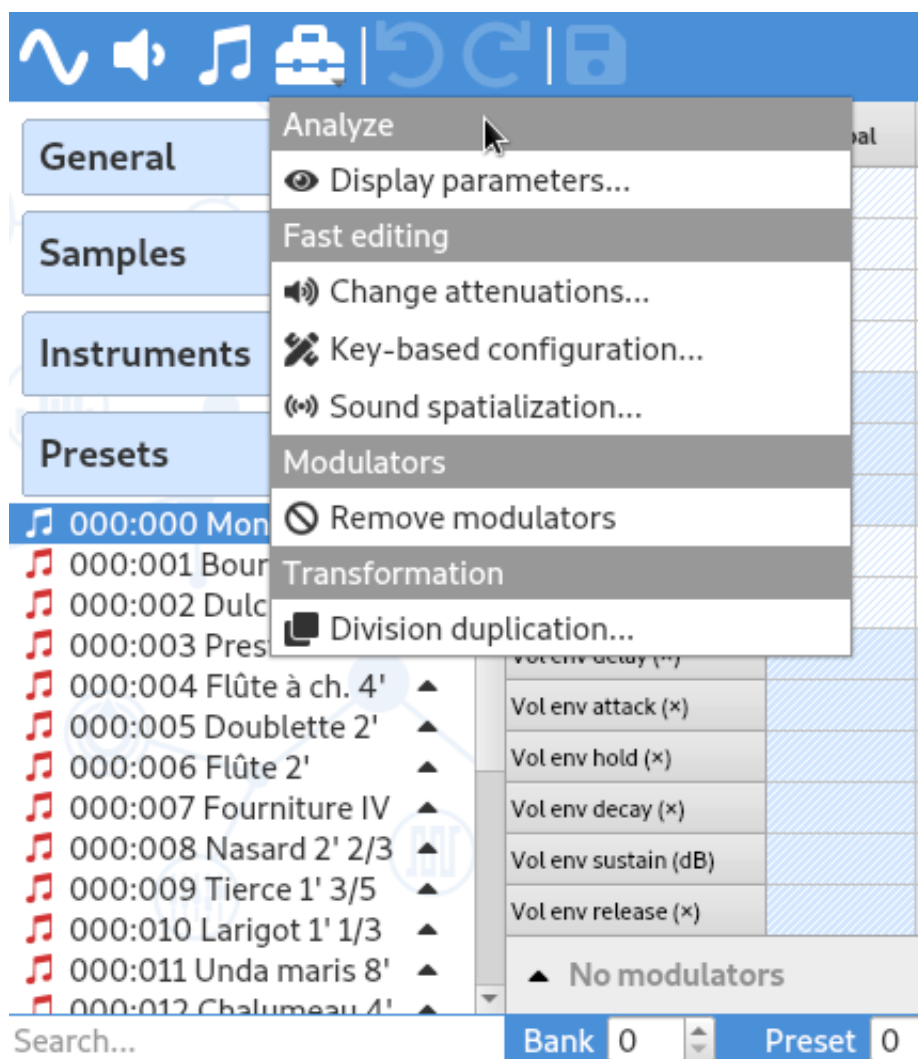
When one or more instruments (including the divisions) are selected in the **tree**, the tool menu is filled with **instrument tools**.



Tool menu showing instrument tools

PRESET TOOLS

When one or more presets (including the divisions) are selected in the **tree**, the tool menu is filled with **preset tools**.



Tool menu showing preset tools



GLOBAL TOOLS

When the header **General** is selected in the **tree**, the **tool menu** provides tools sorted in 3 categories:

- **clean up**,
- **modulators**,
- **utility**.

CLEAN UP

Remove unused elements

All samples and instruments being used by no instruments or presets are automatically deleted.

MODULATORS

Remove all modulators

All **modulators** present in all **instruments** and **presets** will be removed.

UTILITY

Export preset list

With this tool, the list of all **presets** will be written in a text. This text can be formatted to be integrated in a CSV file or in HTML to be integrated in a website.



Export preset list

☒ csv format

☐ html table

0	0	Montre 16'
0	1	Violon 16'
0	2	Principal 8'
0	3	Bourdon 16'
0	4	Salicional 8'
0	5	Octave 4'
0	6	Octave 2'
0	7	Flûte à ch. 8'
0	8	Flûte conique 4'
0	9	Clairon 4'
0	10	Grand plein jeu V
0	11	Grosse mixture IV

Copy

Close

Preset list



SAMPLE TOOLS

When one or more samples are selected in the **tree**, the **tool menu** contains several tools sorted in 4 categories:

- **audio processing**,
- **fast editing**,
- **files**,
- **loop / playback**,
- **stereo samples**.

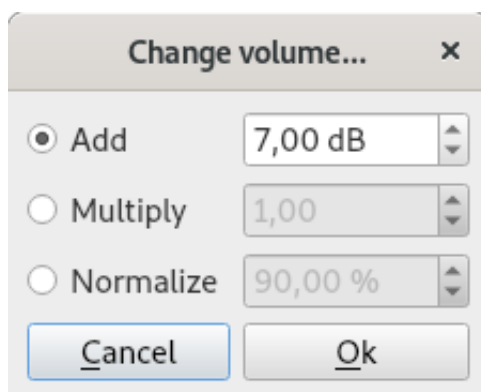
Using the tools simultaneously on several samples is possible (with a multi selection).

AUDIO PROCESSING

Change volume

You can change the sample volume in three different ways:

- in adding a certain amount of dB (positive or negative amount),
- in multiplying the signal by a number,
- in normalizing the signal so that the maximum of this signal represents some percentage of the maximum intensity.



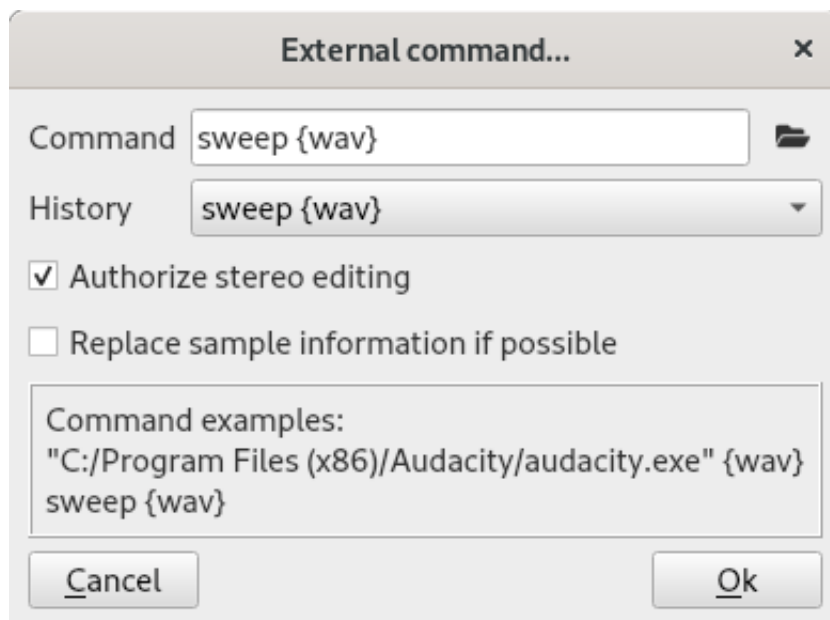
Change volume tool

External command

This tool allows you to use an external command or program to edit a sample. You don't need to export and re-import a sample into Polyphone anymore, everything is done automatically:



1. download an editing software for .wav files such as **Audacity** or **Sweep** (Linux),
2. use the tool **External command** for one or more samples,
3. fill the command line in the right field to call the external software (you may need to read the documentation of this software),
4. edit the sample(s) with the external tool,
5. save the modified version of the wav file(s) with the external tool and close it,
6. the sample(s) will be automatically loaded into the soundfont.

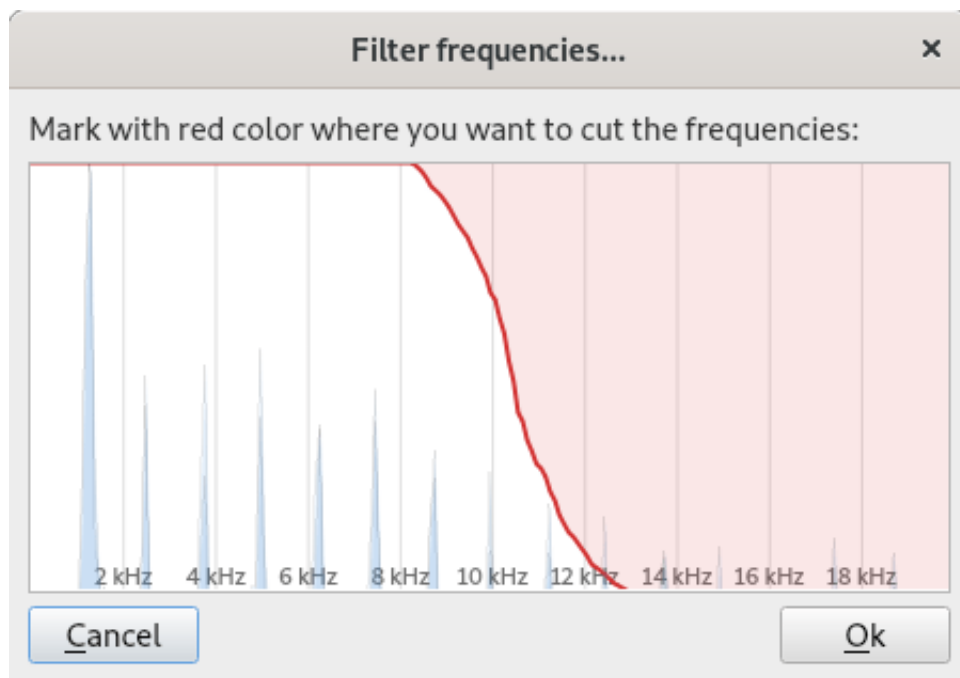


External command tool

Note: during the process, Polyphone exports a .wav file and will reimport this same file after the external command has been run. If the external command creates other files or saves the result as another file, those files will be ignored and Polyphone will reimport the file initially exported that will probably be not modified.

Filter frequencies

When you use this tool, a window will be displayed showing the frequencies contained in the sample(s) you selected. You can then use the mouse to draw a red area. Everything that is in this area will be removed: this means that you can manually decrease or remove some frequencies.

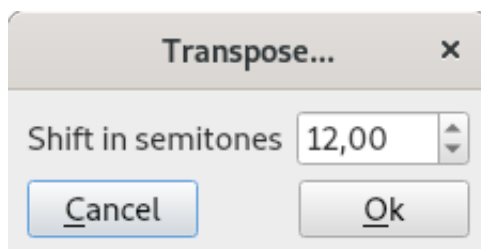


Filter frequency tool

Note: This filter is not common since it cuts frequencies in a hard way. This tool can be useful to design a brick-wall filter but in most cases you will probably want to use the [equalizer](#).

Transpose

The tool asks for a shift in semitones and then re-samples the sound to change the pitch.



Transpose tool

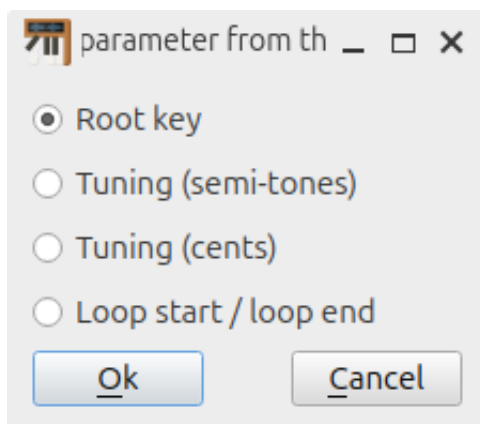
FAST EDITING

Load a parameter from the instruments

Parameters at the instrument level can override or offset the parameters defined at the sample level. With this tool, it is possible to fix or complete the sample parameters with what is defined in the instruments using these samples:



- modifying the sample root key with the override defined at the instrument level,
- shifting the sample root key with the "Tuning (semitones)" parameter,
- shifting the sample pitch correcting with the "Tuning (cents)" parameter,
- defining the sample loop start and end with the loop offset parameters.

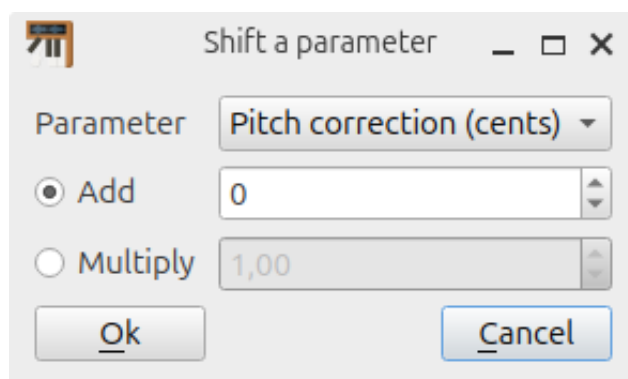


Tool for loading a parameter from the instruments

Shift a parameter

This tool allows the simultaneous editing, by adding or multiplying a value, of one of these parameters for all selected samples:

- root key,
- correction,
- loop start,
- loop end.



Fast edit tool

FILES

Wav export



After having selected a destination directory, all selected samples will be exported as .wav files.





LOOP / PLAYBACK

Auto loop

The software seeks two positions in the sample, distant and suitable for creating a loop. If needed, a short sound segment is copied with a cross-fade so that the transition at the join location is as smooth as possible.

If it fails, a warning message appears indicating that no loops can be found.

Warning: This tool modifies the original sample. Modifications can be undone via the “undo” function, but if the file is saved and Polyphone is closed there is no way to step back.

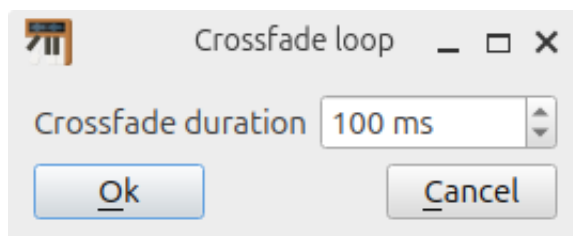
Note: By default, the looping is disabled in an instrument. To hear it, you must select  in the **Loop playback** parameter row of either the **Global** or individual notes columns in the **table**.  turns it On,  turns it Off.  turns the looping On and the sample will play to the end after a key is released.

Cross-fade loop

The cross-fade tool can be seen as the manual version of the **auto loop tool**. First, you need to position the start and end of the loop in the graphic (WAV display area) around a stable area. Then, this tool creates a loop from these positions by applying a cross-fade whose duration is configurable. Everything is therefore configurable but care must be taken when searching for the start and end of the loop:

- at both positions, the harmonics of the sound must match so as not to hear a scratched vinyl record effect during the transition,
- the phases of the different harmonics must also match so as not to hear a frequency cancellation during the cross-fade, which strongly alters the timbre of an instrument,
- as much as possible, the end of the loop must be far from the start to avoid a sound that is too synthetic.

A tip is to focus on positions where the wave crosses the x-axis (value 0).

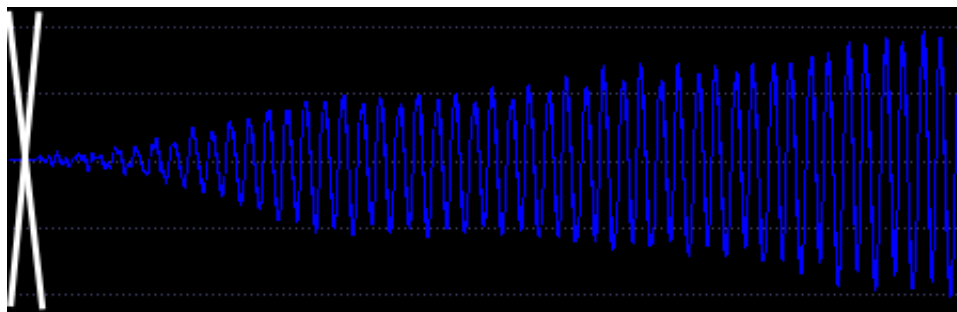


Cross-fade loop tool



Remove blank at start

Any blank area before the sample begins is eliminated, reducing the time before the sound is heard.

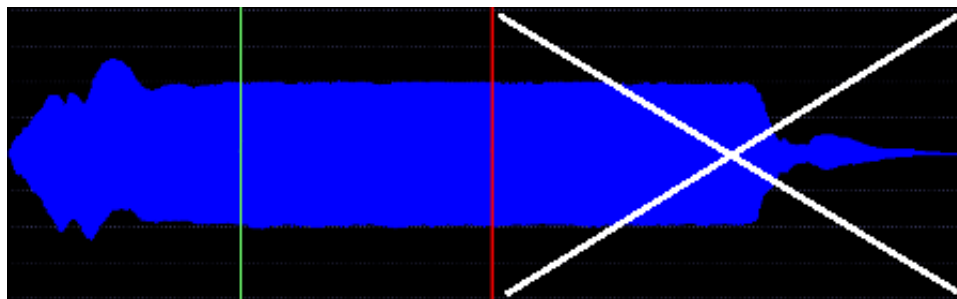


Remove blank at start

Note: this tool may be applied automatically each time a sample is imported, the option being in the software [settings](#).

Trim to end of loop

If a loop is defined for the selected sample, the software will cut the sound after the end of the loop (leaving a small margin at the end).



Trim to loop

Note: this tool may be applied automatically each time a sample is imported, according to the software [settings](#).

STEREO SAMPLES

Balance adjustment

If the sample is linked to another one (stereo), the software can equalize the mean intensity of the two samples to adjust the balance. This will then be easier to accurately pan a sample in an [instrument](#).



Find a link

Among the selected samples, samples whose names only differ by the last letter (L / R) are automatically linked. For example, if in a soundfont two samples are called `bassoon45L` and `bassoon45R`, `bassoon45L` will be recognized as the left part of `bassoon45R` and `bassoon45R` will be recognized as the right part of `bassoon45L`.

With this information the right panning will be used when these samples are added in an instrument. You can also enjoy the simultaneous editing of left and right parts in the instrument [table](#) if the stereo editing is activated in the [settings](#).

Warning: This tool can break existing links if the names don't match the rule above.

Unlink

All selected samples that are linked (stereo) are unlinked.

Thus, all samples will be processed as mono sounds and they will all have the same panning if they are added in an instrument. The stereo editing, if activated in the [settings](#), will not be applicable anymore.



INSTRUMENT TOOLS

When an instrument or one of its divisions is selected in the **tree**, the **tool menu** contains several tools sorted in 4 categories:

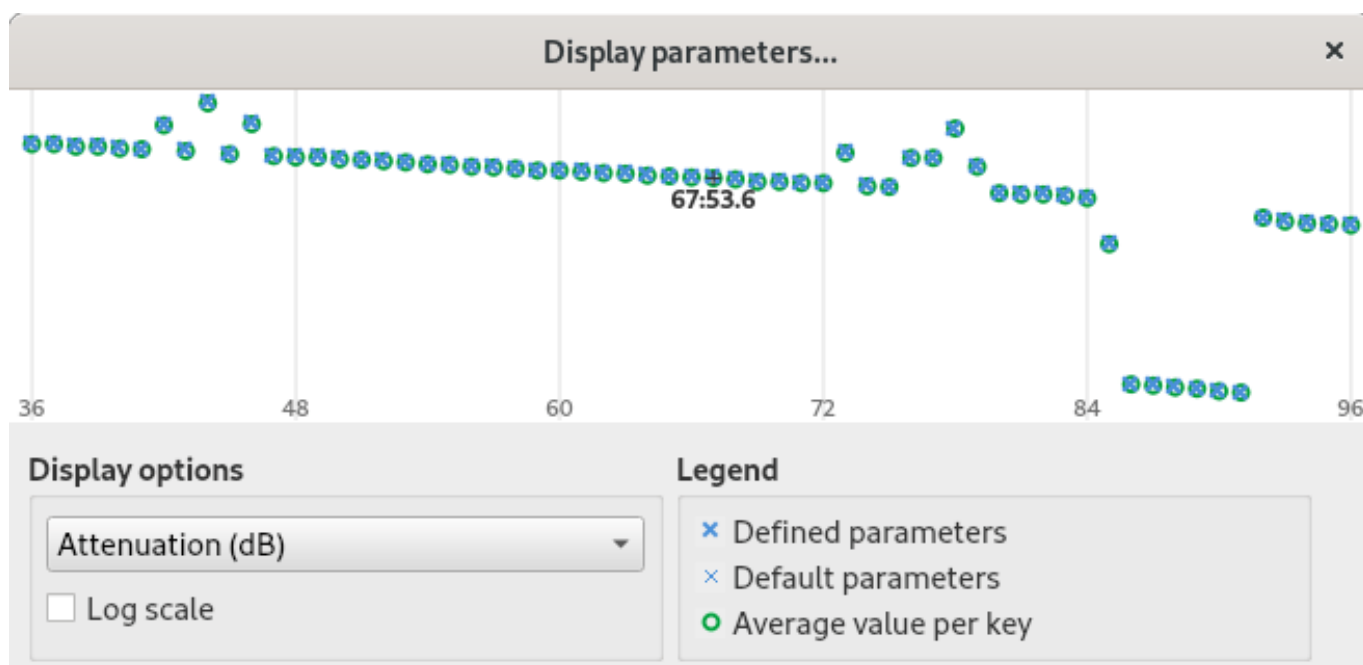
- **analyze**,
- **fast editing**,
- **modulators**,
- **transformation**.

Using the tools simultaneously on several instruments is possible (with a multi selection), except for creating chords, mixtures and for displaying parameters.

ANALYZE

Display parameters

This tool displays the evolution of a parameter according to the key. A logarithmic scale is available (take care of values equal to or less than 0!).



Display parameter tool



FAST EDITING

Change attenuations

The tool computes the minimum and the maximum attenuations in all selected instruments, and gives you the possibility to add an offset to all attenuation values without exceeding the limits.

For example, if the lowest attenuation of the divisions within the selected instruments is 16 dB and if the desired minimal attenuation is 5 dB (to increase the general volume), the attenuation of all divisions of all instruments can be decreased by 11 dB.

A dialog box titled "Change attenuations..." with a close button (X). It contains three rows of labels and values: "Attenuation range" with the value "[+19.5, +65.5]", "Possible offset" with the value "[-19.5, +78.5]", and "Desired value" with a text input field containing "0,0 dB" and a vertical spinner. At the bottom are two buttons: "Cancel" and "Ok".

Change attenuation tool

Detune

This feature slightly detunes the instrument notes, so as to create an undulating effect in conjunction with a tuned instrument. The intensity of the effect is expressed in number of beats per second and is adjustable:

- number of beats per second at key 60, which is "C" in the middle of the keyboard;
- evolution of the number of beats according to the note, dividing or multiplying the number of beats by a constant factor from one octave to another (value "1" means that the number of beats will be constant all over the keyboard).

A dialog box titled "Detune..." with a close button (X). It contains three rows of labels and values: "Number of beats per second (key 60)" with a text input field containing "4,0" and a vertical spinner; "The algebraic sign determines the tuning direction."; and "Division to the next octave" with a text input field containing "1,00" and a vertical spinner. At the bottom are two buttons: "Cancel" and "Ok".

Detuning tool



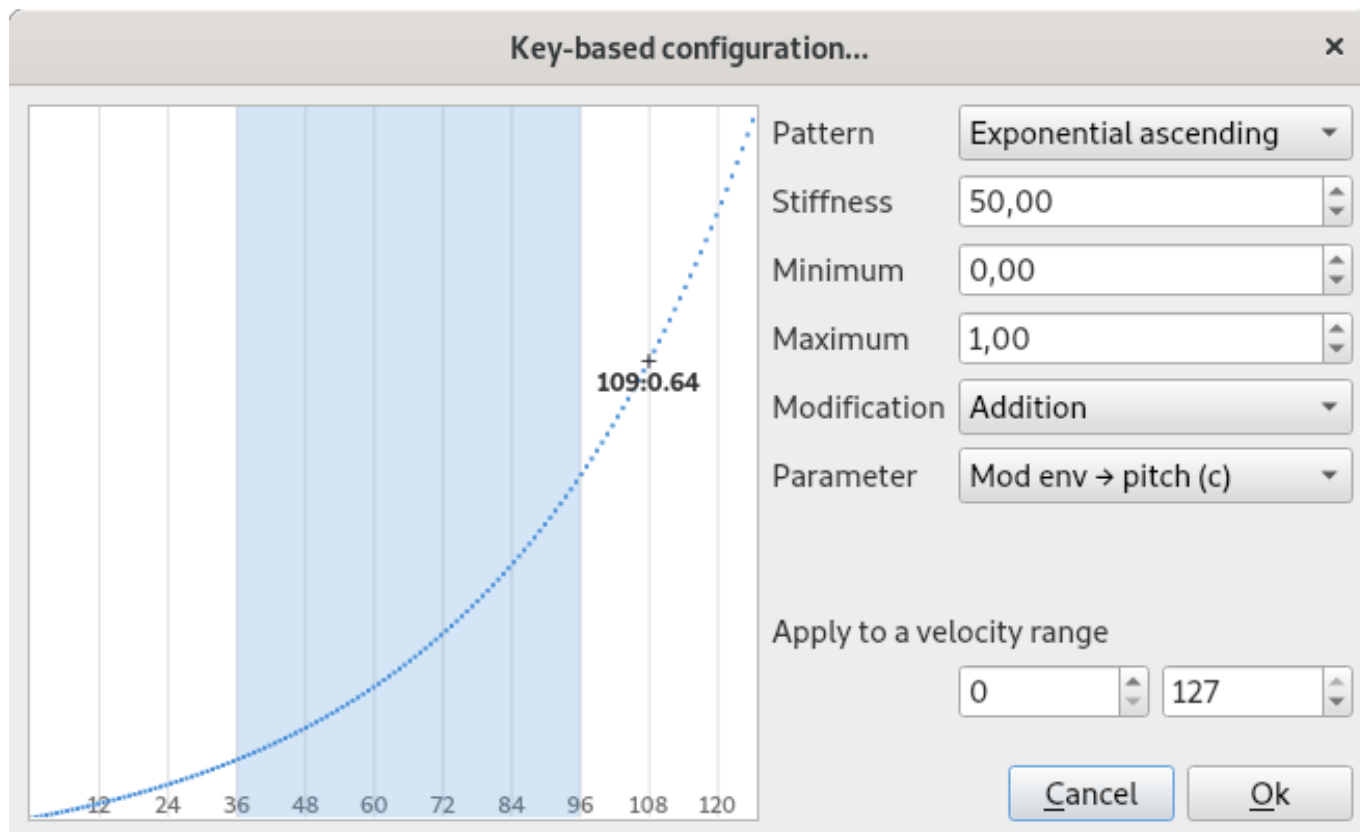
Here is an example of an instrument that is [tuned](#):

And this instrument has been combined with itself in a version that is [detuned](#):

The corresponding .sf2 file can be downloaded [here](#).

Key-based configuration

This tool allows the editing of a parameter for all divisions of an instrument simultaneously.



Key-based configuration tool

The graph represents the modification intensity, the first value on the left corresponding to the change applied to the division comprising the key 0, and the last value on the right corresponding to the change applied to the division comprising the key 127. By changing the pattern, it is possible to edit the curve:

- manually using the mouse,
- by using a linear law,
- by using an exponential law,
- by using a random generation.

In the case where a linear or exponential law is used, a left click in the graph defines the beginning of the linear or exponential area, a right click defines the end.

The parameter to be modified is chosen in **Parameter**, and the modification type is chosen in



Modification:

Addition

The modifier is added to the value of the selected parameter.

Multiplication

The modifier is multiplied with the value of the selected parameter.

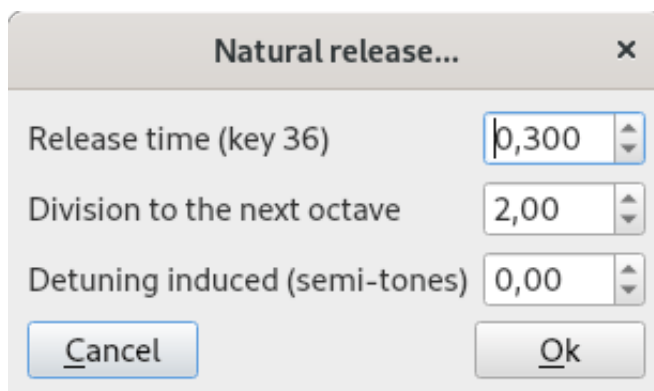
Replacement

The modifier replaces the value of the selected parameter.

It is possible to apply the changes only on divisions whose velocity range is comprised in a specific range, useful for example in the case where a piano has several velocity layers.

Natural release

This tool creates a release for each division of the instrument. Release times are computed taking into account the first two parameters and a pitch modification may be added by using the third one.



The image shows a dialog box titled "Natural release...". It contains three input fields with numeric values and up/down arrows: "Release time (key 36)" with value 0,300, "Division to the next octave" with value 2,00, and "Detuning induced (semi-tones)" with value 0,00. At the bottom are "Cancel" and "Ok" buttons.

Parameter	Value
Release time (key 36)	0,300
Division to the next octave	2,00
Detuning induced (semi-tones)	0,00

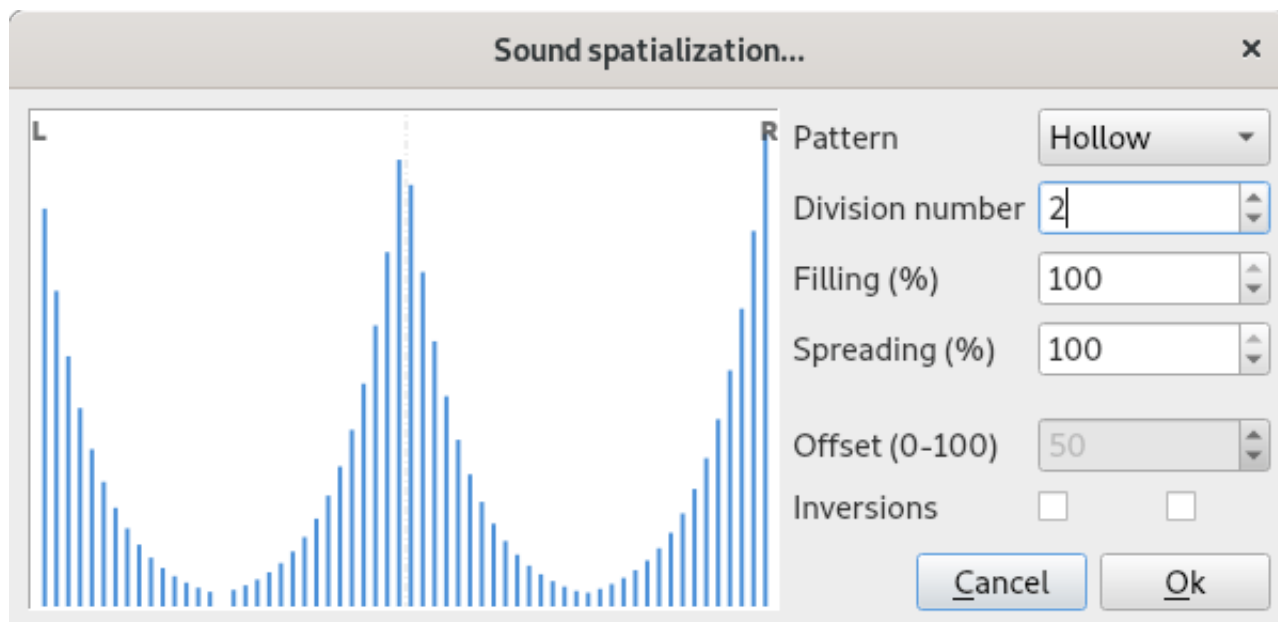
Natural release tool

Sample auto-positioning

The software determines the key ranges of all divisions of an instrument, in order to optimize their use all over the keyboard.

Sound spatialization

This feature spatializes sounds according to their pitch, which can be useful for a piano for example (bass and treble left to right).



Sound spatialization tool

The bars in the graph represent each a string or pipe, having a length in relation to the pitch. The shortest bar corresponds to the highest note and the longest bar corresponds to the lowest note.

Each note is positioned in the space, the far left corresponding to a sound coming from the left and the extreme right corresponding to a sound coming from the right. Several patterns can be selected:

Ascending

Trebles are in the left side, basses are in the right side.

Descending

Basses are in the left side, trebles are in the right side.

Hollow

Basses are at both extremities, trebles are centered.

Spike

Trebles are at both extremities, basses are centered.

Random

Trebles and basses are positioned randomly.

Other options are available:

- number of pattern repetitions (**Divisions number**),
- intensity of the effect (**Spreading**),
- filling of a division (**Filling**),
- offset left — right if the spreading is not 100% (**Offset**),
- **Inversions**.

The graph is updated every time a setting changes for a better visualization. On mouse over, the correspondence note number / panning is displayed.

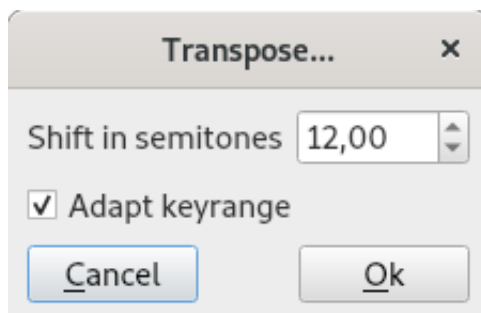
Transpose



The transposition tool allows the transposition of an entire instrument. The sounds of the instrument are thus the same, but at a different position over the keyboard.

As input, a shift in terms of semitones is required. For the pitch to be higher, the shift must be positive. Conversely, for the pitch to be lower the shift must be negative. The operations performed by the tool, for each division of the instrument, are then as follows:

- the root key is adjusted,
- modification of the fine tuning if the transposition is performed by a number of semitones including decimals,
- shift of the key range according to the number of semitones if **adapt key range** is checked.

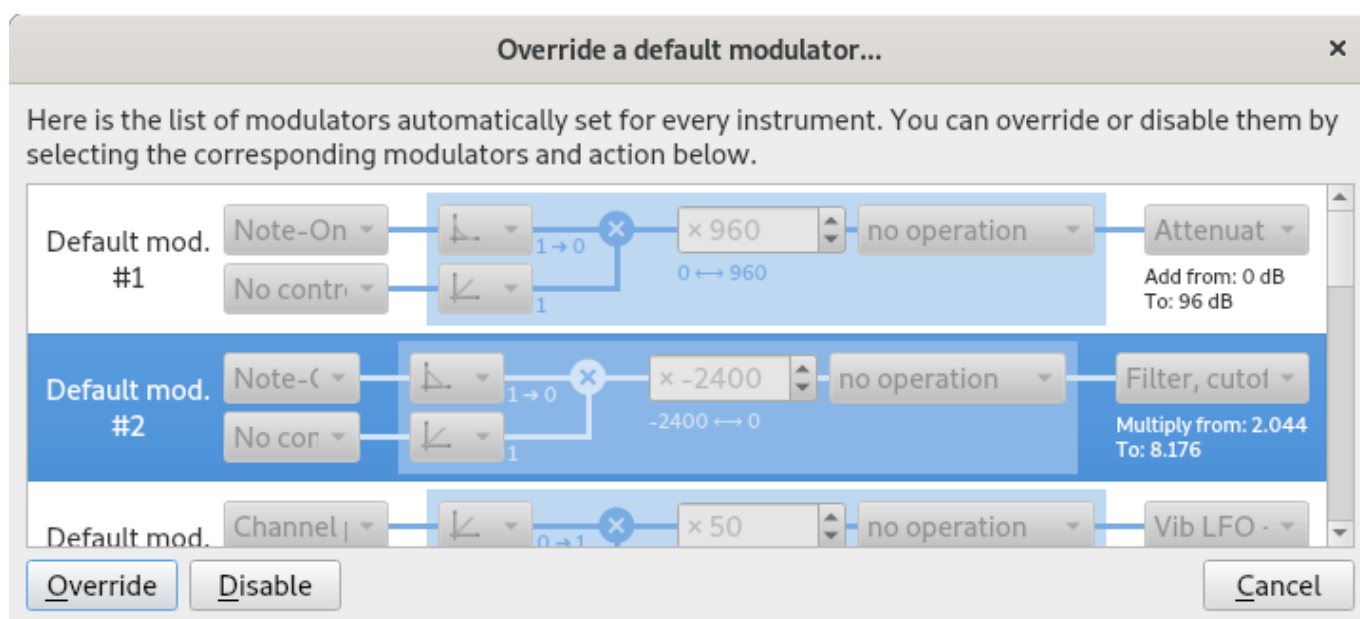


Transposition tool

MODULATORS

Override a default modulator

Default modulators are automatically assigned to every instrument. This tool can create modulators that will override or disable them.



Overriding tool



Remove modulators

All modulators of all selected instruments will be removed.

Note: Default modulators will still apply. The only way to disable a default modulator is to override it with the tool " [Override a default modulator](#)".

TRANSFORMATION

Create chords

Based on an instrument, this tool allows you to create all possible chords. This could be interesting for example for the creation of a choir or pads.

Create chords...

Instrument name: ☒ Stereo

Sample creation: ☒ Auto loop

Chord

Root key:

Third:

Fifth:

Seventh:

Ninth:

Inversion number:

Octave:
 -3 -2 -1 0 +1 +2 +3

Tool for creating chords



Here is an example, [before](#)

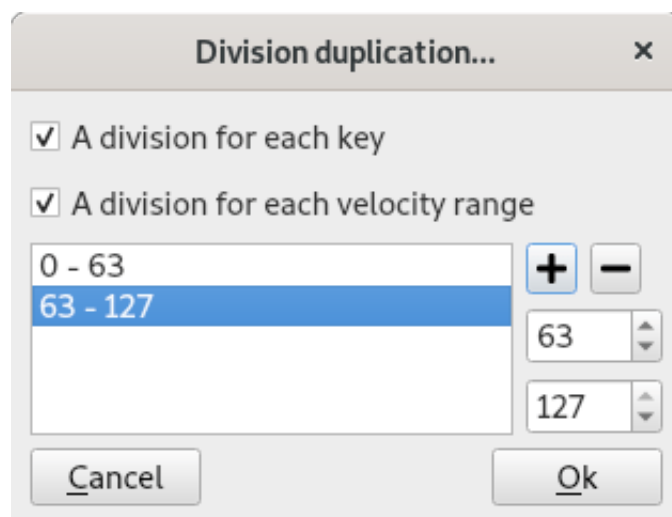
And [after](#)

An example .sf2 file, based on a flute, can be downloaded [here](#) .

Division duplication

This action has for effect the duplication of all divisions of an instrument. For example, if a sample linked to an instrument has a key range of 36-38, the software duplicates this division so as to have this linked sample for the key ranges 36-36, 37-37 and 38-38. It makes then possible to thoroughly edit each sample parameter for each different note, instead of editing whole divisions with the same parameters.

It is also possible to duplicate the divisions according to velocity ranges.



Division duplication tool

Mixture creation

Specific to the organ, this tool allows the creation of a mixture from an existing instrument. An interface appears for specifying the different mixture divisions, and for each the list of the ranks. Types of possible ranks are very diverse, ranging from the octave to the 27th harmonic.



Mixture creation...

Name of the mixture: ☒ Stereo

Sample creation: ☒ Auto loop

Divisions

36 - 53	+ -
54 - 77	
78 - 101	
102 - 108	

Key range

Ranks

2'	+ -
1' 1/3	
1'	
2/3'	

Type

Mixture creation tool

To work, this tool relies on all samples associated to the selected instrument, the range defined for each sample and the corresponding attenuation. Samples have to be tuned regardless the definition of the instrument (via the [sample editor](#)), and the root key specified must not take into account membership in a stop (no transposition if the sound is a principal 4' for example).

An option allows the automatic loop of created samples, another sets the creation interval of samples. Finally, it is possible to choose the type of the created sounds: mono or stereo.

Here is an example, [before](#)

And [after](#)

An example .sf2 file, based on an organ stop, can be downloaded [here](#) .



PRESET TOOLS

When a preset or one of its division is selected in the **tree**, the **tool menu** contains several tools sorted in 4 categories:

- **analyze**,
- **fast editing**,
- **modulators**,
- **transformation**.

Using the tools simultaneously on several presets is possible (with a multi selection), except for the visualizer. All tools are also available for the **instruments**.

ANALYZE

Display parameters

Same tool than " **Display parameters**" for instruments.

FAST EDITING

Change attenuations

Same tool than " **Change attenuations**" for instruments.

Key-based configuration

Same tool than " **Key-based configuration**" for instruments.

Sound spatialization

Same tool than " **Sound spatialization**" for instruments.

MODULATORS



Remove modulators

Same tool than “ [Remove modulators](#)” for instruments.

TRANSFORMATION

Division duplication

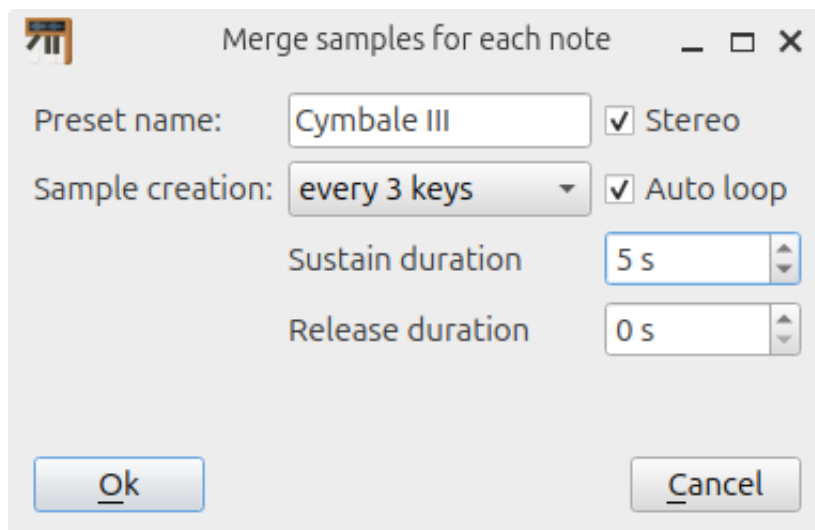
Same tool than “ [Division duplication](#)” for instruments.

Merge samples for each note

In the case where a preset calls several instruments and samples simultaneously when a key is struck, this tool will merge, for each note, all the samples played. The result is a new preset that plays only one sample per key, thus reducing the CPU load required to run the preset.

The options defined in the image below will make the tool simulate every 3 notes a 5-second key press. The audio recording will be in stereo and the release will not be recorded. The samples thus created will then be looped using the [auto-loop tool](#) and the instrument containing all these samples will be named “Cymbale III”.

Note: using the auto loop option will remove the release part and slightly decrease the sustained duration of each sample.



Merge samples for each note

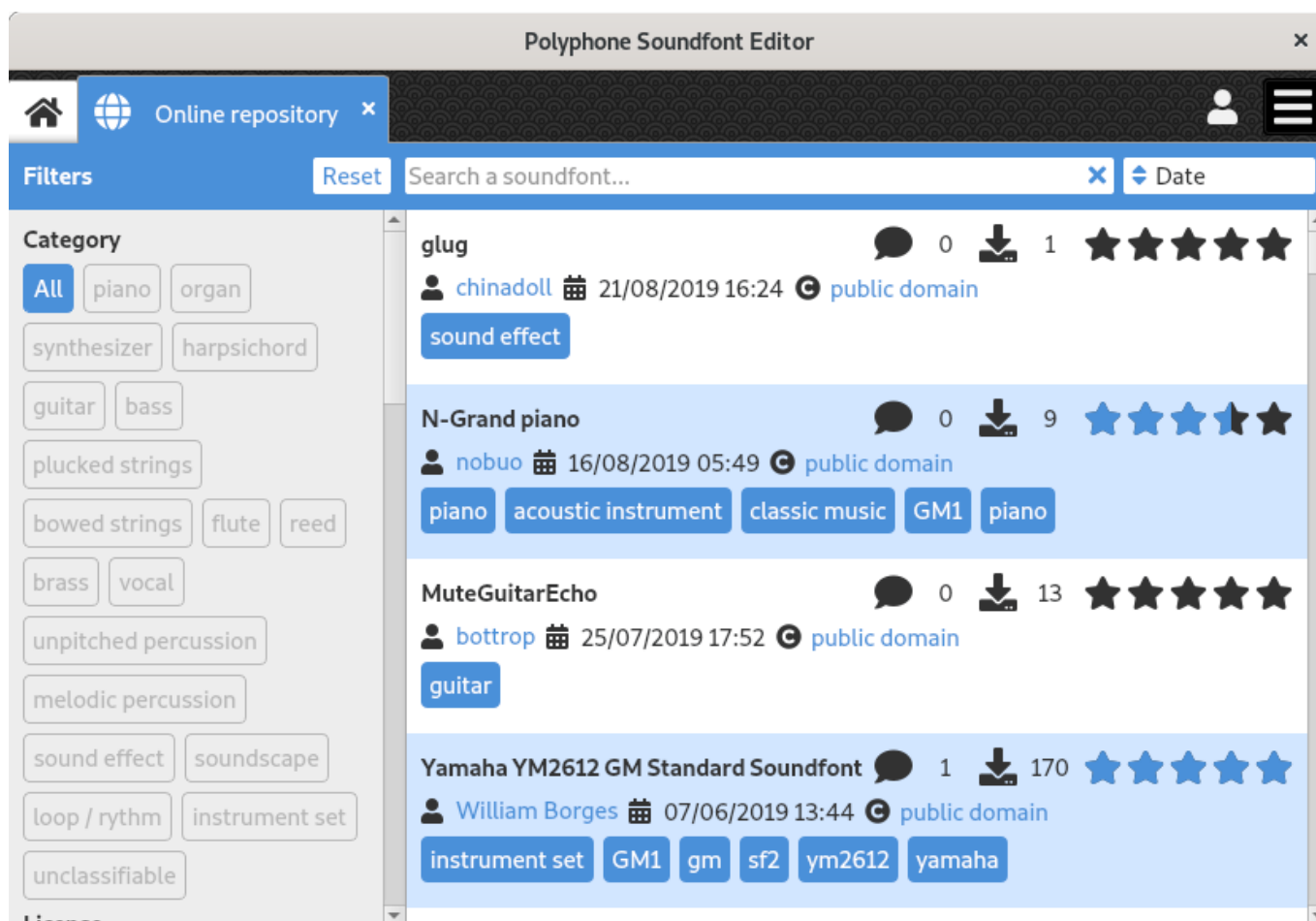


SOUNDFONT BROWSER

The soundfont browser is a link between Polyphone and the soundfont repository of the Polyphone website. It allows you, directly in Polyphone, to download, test and possibly re-use soundfonts.

BROWSING SOUNDFONTS

To open the soundfont browser, click on **Online repository** or write a keyword on the [home screen](#). Once the soundfont browser is open in a new tab, it is possible to search, sort and filter the results.



Browsing soundfonts

Searching a soundfont

For searching a soundfont, write one or more keywords in the top bar and then press **Enter**. It is possible to write `Author:someone` to search soundfonts of a specific author.



Sorting the results

Next to the search bar, a drop-down menu provides 4 ways to sort soundfont results:

Date

Newest soundfonts are first.

Downloads

More downloaded soundfonts are first.

Rating score

Highest rated soundfonts are first.

Title (A→Z)

Soundfonts results are sorted according to the alphabetical order.

Filtering the results

Results can be filtered in different ways:

Category

Kind of instrument.

License

It's possible to list soundfonts having a license allowing a commercial and / or a sharing after the soundfont is edited.

Sample source

Origin of the samples.

Timbre

Description of the timbre.

Articulation

Description of the articulation.

Genre

Genre targeted by the soundfont.

MIDI standard

Listing only soundfonts compliant with a standard is possible.

Tags

Show soundfonts having a specific tag.

SOUNDFONT DESCRIPTION

Opening a soundfont description can be made with a double-click on a cell in the **Daily soundfonts** of the [home screen](#) or in the [soundfont browser](#). A Premium account is required for soundfonts that are not in the **Daily soundfonts**. Connecting Polyphone to your account is possible in the [settings](#) of the software.



The screenshot shows the 'Polyphone Soundfont Editor' window. The top bar has tabs for 'Online repository', 'Timbres O...3.4 Final', and 'Yamaha Y...oundfont'. The main content area displays the 'Yamaha YM2612 GM Standard Soundfont' page. On the left, there's a sidebar with user information (William Borges, 07/06/2019 13:44, public domain, willbsantos.weebly.com), category (instrument set), MIDI standard (GM1), and tags (gm, sf2, ym2612, yamaha). The main area has sections for 'Description' (A soundfont with GM Standard built by Yamaha YM2612 Dumped Voices, Sample Rate: 44.1KHz, 128 Instruments, 1 Percussion (Key 27-87)), 'Downloads' (YM2612.part1.rar (20.00 MB, 07/06/2019), YM2612.part2.rar (4.12 MB, 07/06/2019)), and 'Comments' (FC, 01/07/2019 14:50, These are very high quality samples and the soundfont will give your music a nice Yamaha FM / Megadrive (Genesis) feel. Excellent work!).

Description page of a soundfont

The description of a soundfont is made of:

- a **header**, in which you can find the title, the number of comments, the number of downloads and the rating,
- a **left part**, comprising the author name, a date, the license type (please take care of this), the category and other attributes that characterize the soundfont,
- a **central part**, displaying a textual description, the list of files to download and possible comments.

DOWNLOAD

For downloading a file, click on the icon **download** next to its file name in the section **Downloads**. You will see the download progress in a menu next to the **main menu**, at the top right corner of the software. Once a file is downloaded and if the file is a soundfont, it is possible to open it in a **soundfont editor** by clicking on the icon **open**.

It is possible from the menu to clear the completed downloads. If everything is cleared, the download menu disappears.



ory x

'Decresc...mp Organ x

Gothic organ x

Decrescendo Pump Organ.sf2 100%

Gothorgn.sf2 100%

Clear completed downloads

Description

Bank	Preset	Name
0	0	Gothic Organ I
0	1	Gothic Organ II

Downloads

Gothorgn.sf2 (781.59 kB, 04/03/2017)

Downloading soundfonts



SETTINGS

The software preferences are accessible either from the [home screen](#) or from the [menu](#). They are divided into five sections:

- [General](#)
- [Interface](#)
- [Sound](#)
- [Virtual keyboard](#)
- [Online repository](#)

GENERAL

The **General** section allows the modification of the following parameters:

Audio backend

Specifies the audio backend used to play samples. JACK and ASIO drivers (the latter for Windows only) may be chosen.

Buffer size

Specifies the buffer size used for JACK and ASIO drivers. A small buffer results in a small latency, but the sound may become scratchy.

MIDI input

Specifies the MIDI input controlling the virtual keyboard.

Moreover, it is possible to check the following options:

Stereo editing: change linked sample

When editing one side of a stereo sample, the other side is automatically edited the same. The same applies when editing a stereo instrument division: the division corresponding to the other channel (with the same note and velocity ranges) is edited the same.

Sample import: trim to loop

When importing a sample, data located after the end of loop are automatically removed (same effect as the tool " [Trim to end of loop](#)").

Sample import: remove blank at start

When importing a sample, any blank area located before the beginning of the attack is automatically removed (same effect as the tool " [Remove blank at start](#)").



General

Input / Output

Audio backend	ALSA - Default ALSA Device
Buffer size	128
MIDI input	-

Sample management

Stereo editing	<input checked="" type="checkbox"/> change linked sample
Sample import	<input type="checkbox"/> trim to loop
	<input type="checkbox"/> remove blank at start

Settings, general

INTERFACE

This section shows options related to the interface of the software:

Language

The language may be changed here (a restart of the software is required).

Key names

Keys may be named depending on their number (according to MIDI specification), or depending on the name of middle C (C3, C4 or C5). Key names may comprise sharps or flats.

Sort divisions

The division order within an instrument or a preset can be changed here.

Unique instance

Open all files in a single window.

Theme

Different styles and predefined color sets are available for customizing the appearance of Polyphone. A manual selection of colors is also possible.

Window borders

Display or hide the window borders, for saving space on the screen.

Decorations

Background decorations in some lists can be deactivated for readability.



Interface

Text

Language

American English

Key names

Middle C → 60, then 61

Behaviour

Sort divisions

by key range

Unique instance

☒

Theme

Style

Fusion

Colors

Gnome

	Window	Button	Selection	List and table
Text	<div></div>	<div></div>	<div></div>	<div></div>
Background	<div></div>	<div></div>	<div></div>	<div></div> <div></div>

Window borders

☐

Decorations

☒

Settings, interface

SOUND

The **Sound** section makes it possible to modify the chorus and reverb of the synth, along with its global volume. The global volume may be changed with the volume MIDI controller if an external keyboard is connected.


Since the default modulator modifying the filter frequency has been changed between versions 2.01 and 2.04 of the sf2 specifications, it is possible to define here which version to take:

- 2.01: use the rule "lower the filter frequency taking into account the velocity IF the velocity is less than 64" (we can hear a clear and problematic gap between velocities 63 and 64);
- 2.04: use the rule "lower the filter frequency taking into account the velocity" (no more gap);
- disabled: this default modulator is not set.



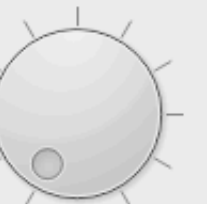
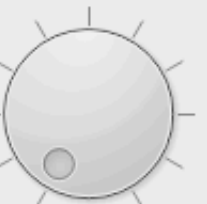
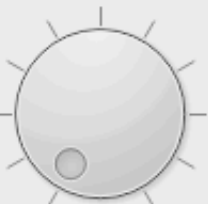
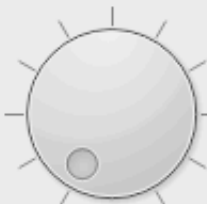
Sound

Options

General volume  +0 dB

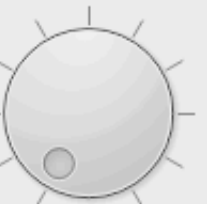
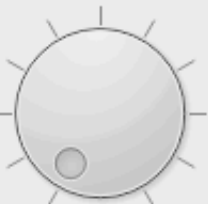
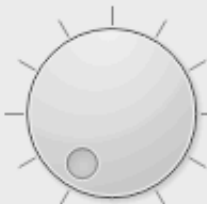
Default modulator "velocity → filter cutoff" soundfont 2.04

Reverb



Level Room Width Damp

Chorus



Level Depth Frequency

Settings, sound

VIRTUAL KEYBOARD

The keys of the computer keyboard used to control the **virtual keyboard** are defined in the **Virtual keyboard** section. The pitch may be increased or decreased octave by octave.

For more advanced musicians, the tuning and temperament can be changed here.



Keyboard

Mapping

	C	C#	D	D#	E	F	F#	G
Octave #1	Z	S	X	D	C	V	G	E
Octave #2	Q	2	W	3	E	R	5	T
Octave #3	Shift+Z	Shift+S	Shift+X	Shift+D	Shift+C	Shift+V	Shift+G	Shift+E
Octave #4	Shift+Q	Shift+@	Shift+W	Shift+#	Shift+E	Shift+R	Shift+%	Shift+T

First C

Tuning

Tuning fork

Temperament

Settings, keyboard

ONLINE REPOSITORY

Identifiers can be entered here to link Polyphone with an account, a Premium account unlocking the .

The download directory used by the [soundfont browser](#) can be changed here.



Online repository

Account

Email address

Password

Connection



Create an account

Downloads

Directory

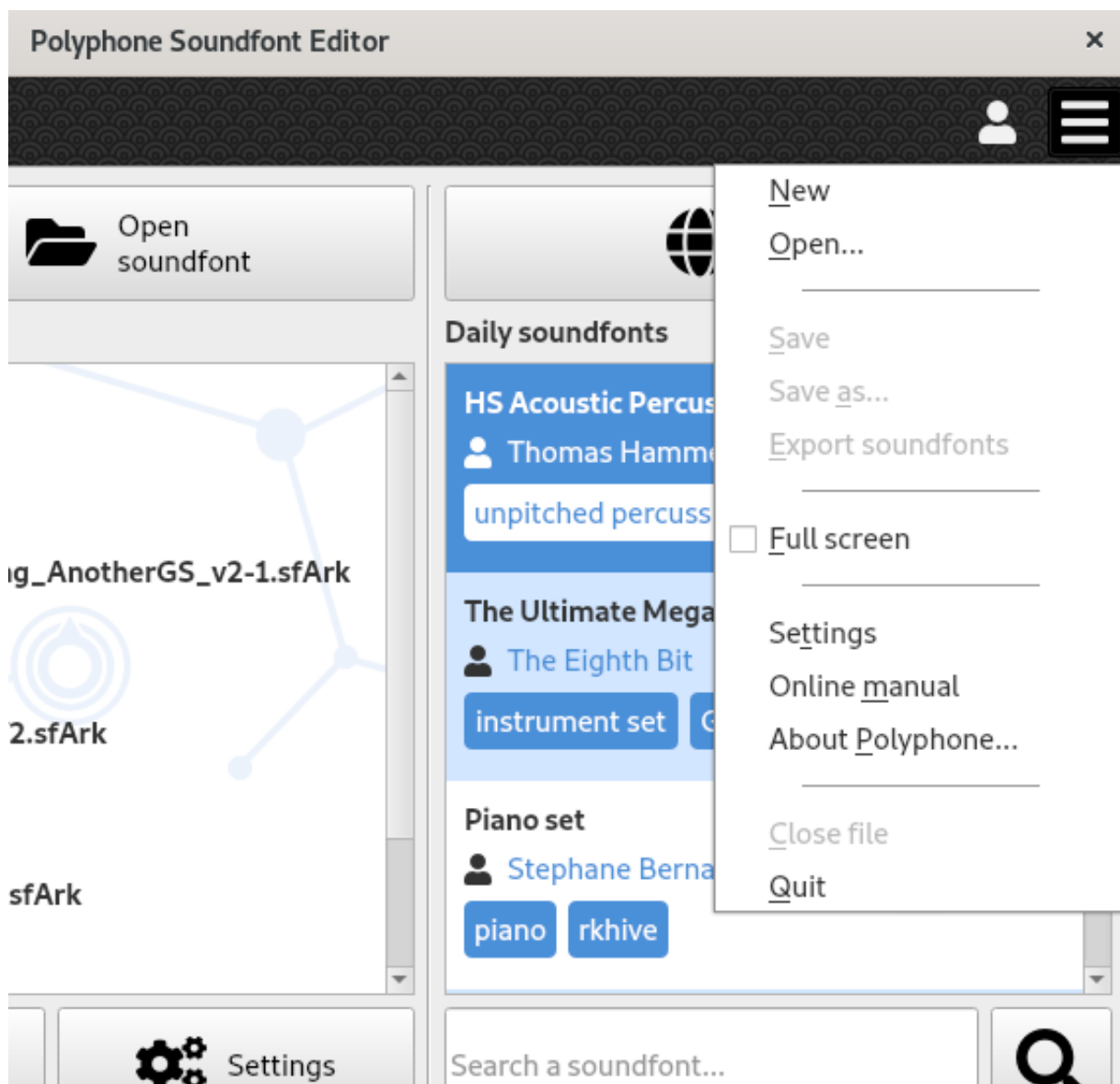


Settings, repository



MENU

The menu, located at the top right corner of the software, contains several items allowing you to access different features.



Menu

OPENING A SOUNDFONT EDITOR

New



Create a new .sf2 file from scratch. Also accessible with the shortcut **Ctrl+N**.

Open...

Open an .sf2, .sf3, .sfz, .sfArk or .organ file. Also accessible with the shortcut **Ctrl+O**.

SAVING A SOUNDFONT

Save

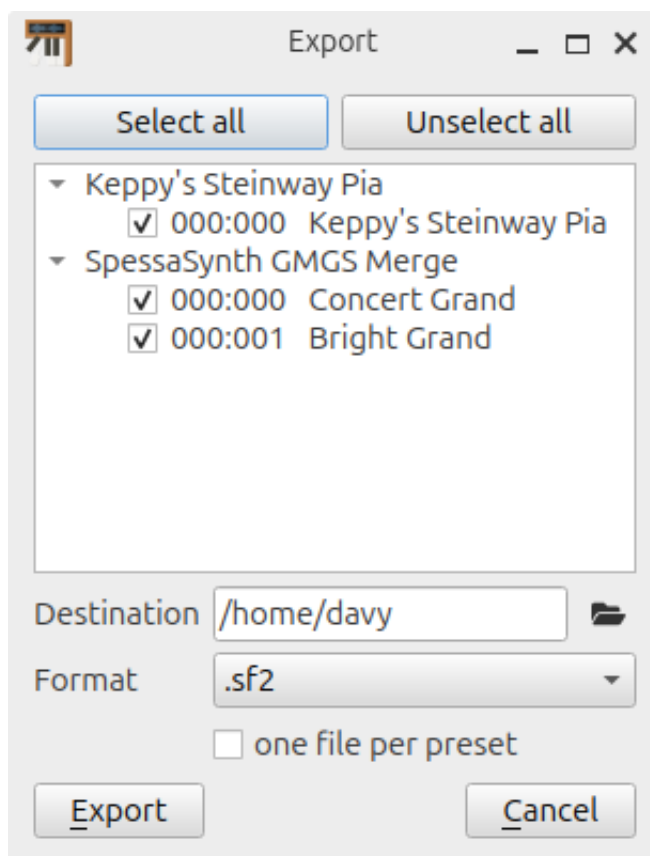
Save the current file. Also accessible with the shortcut **Ctrl+S**.

Save as...

Save the current file in another file. Also accessible with the shortcut **Ctrl+Shift+S**.

Export soundfonts...

Merge and export one or several presets from the open soundfonts in the .sf2, .sf3 or .sfz format. Also accessible with the shortcut **Ctrl+E**.



Export window

VIEW OPTIONS

Full screen

Toggle the full screen mode. Also accessible with the shortcut **F11**.



SHORTCUTS

Settings

Open the software [settings](#).

Online manual

Show this user manual in a web browser. Also accessible with the shortcut **F1**.

About Polyphone...

Show the software version and the list of people who contributed to the project.

CLOSING

Close file

Close the current file. Also accessible with the shortcut **Ctrl+W**.

Quit

Quit the software. Also accessible with the shortcut **Ctrl+Q**.



ANNEXES

Here is the list of pages giving extra information:

- [The different soundfont formats](#)
- [Command line](#)



THE DIFFERENT SOUNDFONT FORMATS

Polyphone supports several soundfont formats:

- [sf2](#),
- [sf3](#),
- [sfz](#),
- [sfArk](#),
- [organ](#).

SF2 SOUND FONTS

Sf2 soundfonts are the main files edited by Polyphone. Each sf2 file comprises one or more musical virtual instruments, made of audio samples and a lot of parameters. Parameters define how the samples should be played throughout the keyboard, possibly modulated by predefined signals ([modulators](#)). Sf2 files are built according to a 3-level structure:

- [samples](#), coming from a trumpet or a piano for instance,
- [instruments](#), made of samples,
- [presets](#), made of instruments.

Further to the definition and setting of these elements, a sf2 file contains also [general information](#) (the author, copyright or edit time for instance).

Soundfonts are used by software synthesizers, such as [Fluidsynth](#), driven by MIDI signals. It is also possible to use sf2 soundfonts to listen MIDI files with [TiMidity](#), [WildMIDI](#) or QuickTime and listen scores with [MuseScore](#). Hardware such as samplers, soundcards or even synthesizers may also support the .sf2 format.

Further information on soundfonts may be found for example on [Wikipedia](#). The sf2 specifications are available [here](#) for version 2.01 and [there](#) for version 2.04.

SF3 SOUND FONTS

The sf3 format, developed by [MuseScore](#), is similar in all respects to the sf2 format except that the are stored in the OGG format (like the MP3 format but open source) instead of being stored as raw data. The consequence is that the sf3 format is about 10 times lighter than the sf2 format for a comparable quality.

A soundfont exported in this format is not intended to be edited later because successive compressions would result in a lower sound quality. This format should be seen as a final product and is very interesting for all end users of soundfonts in that:

- downloads are made easier on internet,
- space is saved in computers,



- the combination MIDI + sf3 provides a lightweight and powerful solution to play soundtracks (video games, mobile applications, ...).

Unlike the [sf2pack](#) format which answers the same issues, the sf3 format is entirely open-source. The source code is available, thus ensuring its sustainability (let's avoid the mistakes done with sfArk or sfPack!).

Sfz SOUNDFONTS

The sfz format has the same goal than the sf2 format: create a musical instrument by disposing and configuring samples over the keyboard. The main difference is that while the sf2 format is only 1 file that contains everything, the sfz format is a text file delivered with a set of .wav samples. Since it was meant to be editable by a human, the main advantage was to edit the file without the need of a complex editor. But an editor is still highly recommended for big instruments, the quantity of parameters can indeed be quickly discouraging. The sfz format is also not defined as strictly as the sf2 format: differences may appear in the way to edit and in the way to play an sfz instrument. But at the same time, the capabilities of the sfz instruments have been increased compared to the sf2 standard.

Limitations on the sfz import

- Only opcodes having a counterpart in [tables](#) are imported.
- The envelope applied to pitch and filter being the same in sf2 files (Mod env), importing different envelopes for the pitch and filter is not possible (fileg_* and pitcheg_* opcodes).
- The LFO applied to volume and filter being the same in sf2 files (Mod LFO), importing different LFOs for the volume and filter is not possible (amplfo_* and fillfo_* opcodes).
- The only filter that can be imported is the second order low pass filter (fil_type=lpf_2p).
- To be imported correctly, group and off_by opcodes must contain the same value within the same group.

Warning: width and position opcodes may not be interpreted correctly.

Limitations on the sfz export

- [Modulators](#) are currently not exported.
- The sfz format (version 1) allowing only one pitch modulation, the two parameters **Vib LFO → pitch (c)** and **Mod LFO → pitch (c)** cannot be exported simultaneously.
- A loop 0-1 can be exported but is ignored by most of the sfz players.
- 24-bit samples may not be supported in some synthesizers, please refer to their specification manual.

In general

The matching of sound levels (in dB), during the sfz / sf2 conversion process, has been tested with Sforzando software. There may however remain some differences.

Moreover, the way parameters change according to the key (via **key → Vol env hold / decay** and **key**



→ **Mod env hold / decay**) have no exact match. The sf2 format uses an exponential law to define the decay and hold durations according to the key, while the sfz format uses a linear law.

SfArk ARCHIVES

SfArk archives, like the sf3 format, is a compressed format. This means that the quality of the sounds is a bit lowered (but often imperceptible). This format tends to be obsolete, that's why Polyphone only allows importing sfArk archives (not the export).

GRANDORGUE DISPOSITIONS

GrandOrgue is a program whose main purpose is to emulate instruments like pipe organs. An organ is defined by a series of wav files, each containing the sound of a pipe, and a file with the extension ".organ" which contains all the parameters needed to create an organ based on all wav files. Basically, all samples are associated with specific keys and are grouped by stops, which can be activated or deactivated when playing the organ. Polyphone is able to create soundfonts based on this mapping, but do not support the conversion of soundfonts to the GrandOrgue format.

Import limitation

Transpositions are currently not supported.



COMMAND LINE

Polyphone can be run with a command line to convert soundfonts in the format **sf2**, **sf3** or **sfz**. Supported file formats for the conversion are **sf2**, **sf3**, **sfz**, **sfArk** and **organ**.

An option can also be called for opening Polyphone in **synth mode**.

CONVERSION TO SF2

Command line

```
polyphone -l -i <file/to/convert> -d <output/directory> -o <output_file_name>
```

Note: with Windows you need to write the full path of Polyphone instead of just `polyphone`, which is for example `C:/Program files/Polyphone/polyphone.exe`.

Arguments

- l** Mode "l" is "convert to sf2".
- d** Output directory in which the input file will be converted. This argument is optional, by default this is the same directory than the input file.
- o** Output name of the converted file without the extension (the extension ".sf2" will be automatically added). This argument is optional, by default this is the same name than the input file.

Example

```
polyphone -l -i /path/to/file.sfArk
```

CONVERSION TO SF3

Command line



```
polyphone -2 -i <file/to/convert> -d <output/directory> -o <output_file_name> -c <configuration>
```

Note: with Windows you need to write the full path of Polyphone instead of just `polyphone`, which is for example `C:/Program files/Polyphone/polyphone.exe`.

Arguments

- 2**
Mode "2" is "convert to sf3".
- d**
Output directory in which the input file will be converted. This argument is optional, by default this is the same directory than the input file.
- o**
Output name of the converted file without the extension (the extension ".sf3" will be automatically added). This argument is optional, by default this is the same name than the input file.
- c**
Conversion configuration. It is possible to specify the compression quality: "0" is low, "1" is medium, "2" is high. This argument is optional, by default this is "1" (medium quality).

Example

```
polyphone -2 -i /path/to/file.sf2 -c 2
```

CONVERSION TO SFZ

Command line

```
polyphone -3 -i <file/to/convert> -d <output/directory> -o <output_file_name> -c <configuration>
```

Note: with Windows you need to write the full path of Polyphone instead of just `polyphone`, which is for example `C:/Program files/Polyphone/polyphone.exe`.

Arguments

-3



Mode "3" is "convert to sfz".

- d**
Output directory in which the input file will be converted. This argument is optional, by default this is the same directory than the input file.
- o**
Output name of the converted file without the extension (the extension ".sfz" will be automatically added). This argument is optional, by default this is the same name than the input file.
- c**

Conversion configuration made of 3 values separated by the "|" character. This argument is optional, by default this is "0|0|0".

- The first value is "1" if each preset must be prefixed by its preset number, "0" otherwise.
- The second value is "1" if a directory per bank must be created, "0" otherwise.
- The third value is "1" if the General MIDI classification must be used to sort presets, "0" otherwise.

Example

```
polyphone -3 -i /path/to/file.sf3 -c '0|1|1'
```

SYNTH MODE

Command line

```
polyphone -s -i <file/to/read> -c <configuration>
```

Note: with Windows you need to write the full path of Polyphone instead of just `polyphone`, which is for example `C:/Program files/Polyphone/polyphone.exe`.



Polyphone opened with the synth mode option

Arguments

- s Mode "s" is "open Polyphone in synth mode".
- c

Synth configuration made of 3 values separated by the "|" character. This argument is optional, by default this is "0|off|off". These parameters can then be changed via the graphical interface.

- The first value is the MIDI channel number (between 1 and 16) to be listened to for playing. A value of 0 or "all" indicates that all channels will be used and processed as one channel. A value of -1 or "each" listens to each channel separately, which is a prerequisite for [playing MIDI files](#) for instance.
- The second value is "off" or "on", enabling or disabling the selection of multiple presets for playing.
- The third value is "off", "on", or "toggle", enabling or disabling the selection of presets by the keys corresponding to the preset number (key 0 activates preset 0, for example).



Example

```
polyphone -s -i /path/to/file.sf2 -c '1|on|toggle'
```



TUTORIALS



Create a soundfont from scratch

Learn the basics to create a new soundfont.

How to prepare a sample

An instrument built in a soundfont is made of samples, coming for example from the recordings of a real instrument. This tutorial describes how to add and prepare a sample in a soundfont.

Custom releases

The purpose of this tutorial is to create expressive instruments using looped samples whose releases have been recorded. You will learn how to fully use the capabilities of such a sample, by using the attack and sustain of the sample but also its release.

Use Polyphone as a synthesizer

Polyphone isn't just a soundfont editor, it's also a full-featured synthesizer that lets you play along with a MIDI keyboard. This tutorial shows you how to launch Polyphone in synth mode.

Create a synthesizer with a Raspberry Pi

Coming soon!

Read a MIDI file with a soundfont

MIDI files contain MIDI messages which, when put together, form a melody. They can be combined with a synthesizer using a soundfont, and this tutorial shows how to use a MIDI file to test the rendering of a soundfont.



CREATE A SOUNDFONT FROM SCRATCH

To create a **soundfont**, click on **Create a soundfont** in the **home screen**. Soundfonts are made from a 3-level structure, so the creation of a new soundfont involves three fundamental steps:

- the **sample preparation**,
- the **instrument creation**, comprising samples,
- the **preset creation**, comprising instruments.

SAMPLE PREPARATION

Sample loading

The sample preparation begins by **loading** .wav files. To do this first select the **Samples** category in the **tree** and select **Import samples** from the **toolbar**. Samples may be obtained:




- through Internet,
- by recording a real musical instrument,
- by synthesis using a specialized software.

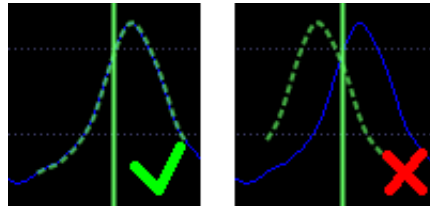
Sample looping

Next, a **loop** may be required in the sample, if you would like it to sound longer than its normal length. An example of this would be if you only have a 1 second flute sample, but would like it to sound indefinitely. This can be done by manually or automatically assigning loop points (start + end) within the sample boundaries, so that the loop area can be repeated and thus make the sample play longer. While you can assign loop points manually, using the " **Auto loop**" function to assign them automatically usually produces better results and is a lot faster.

To assign loop points manually:

1. click in the **tree** on the sample to be looped,
2. in the **sample editor**, arbitrarily position the start and end of the loop in the **graphic** (WAV display area) unless a loop is already defined. Use left-click for start and right-click for end. The right loop point should be entered first (since the right point is by default at the position 0 and that it's not possible to assign a left point after the right point).
3. press the **Play** button after having selected the loop function,
4. adjust the start and / or end of the loop, while the sample is playing, until the transition between the two positions is as smooth as possible.

Note: to hear looping in an instrument, you must select  in the **Loop playback** parameter row of either the Global or individual notes columns in the **Parameters Table**.  turns it On,  or a blank turns it Off.



Loop illustration

Sample tuning

Finally, a **tuning** has to be done. To do this, a calibration tool (sine) is available in the editing page of . For each sample the method is as follows:

1. begin the playback (press the **play** button), if possible with loop function selected,
2. select the sine function,
3. adjust the volume slider to hear the two sounds as clearly as possible,
4. change the **Root key** until the two sounds most closely match,
5. adjust the **Correction (cents)** to tune the sample to the Rootkey. To do this, pay attention to any beats that occur and make sure they are as slow as possible.

The **frequency** analysis can be a good indicator to start tuning.

To go further

Several tools are available to edit samples like:

- removing blank at start,
- equalizing the sound,
- transposing,
- normalizing the volume,
- adjusting the balance for stereo samples.

INSTRUMENT CREATION

For this step, it is assumed that one or more samples are available for the creation of an instrument.

Creating an instrument and adding samples

First, click on **New instrument** in the **toolbar**. A name must now be entered.

Then add samples to the instrument using a “ **drag & drop**”. In the tree you will notice that divisions appear in the instrument.

Note: When divisions are created the samples are not copied but linked. There are no constraints on the number of divisions.







Distribution of divisions

When samples are added to instruments, they appear as divisions (columns) in the **instrument editor table**. Each division must then be positioned on the keyboard by changing the **Key range** in the table. Generally, it is a good practice to have the range of a division include the root key of its represented sample. The entire surface of the keyboard must be covered (eg, key 36 to key 96 for a classic synthesizer keyboard).

The tool " **Sample auto-positioning**" automatically distributes the samples over the keyboard.

Setting the divisions

To enable the loop playing mode in an instrument containing looped samples, select  in the **Loop playback** parameter row of either the **Global** or individual notes columns in the parameter table.

-  turns it On,
-  or a blank turns it Off.
-  turns the looping On and the rest of the sample is read after the key is released.

The global division, as its name suggests, allows you to enter a parameter for the entire instrument. A global parameter applies only to the divisions whose same parameter is not specified. This means that the parameters of the standard divisions take precedence over the settings of the global division.

In the context of this simple tutorial, no other parameters have to be edited for the instrument to work. It should be already possible to play with the **virtual keyboard**.

However, to improve the instrument it is advisable to enter a value in the **Vol env release (s)** row, which prevents the sound from stopping abruptly, as soon as a key is released. The other parameters are described in the description of **tables**.

PRESET CREATION

For this step, it is assumed that one or more instruments are available for the creation of a preset.

What is a preset?

A preset is the equivalent of a sound patch name found in hardware synths. It is visible from outside a soundfont and is identified by a bank number and a preset number. A soundfont can have one or many presets. Each preset contains one or more instruments, just as instruments contain one or more samples.

Creating a preset and adding instruments



First, click on **New preset** in the **toolbar**. A name must now be entered.

Then add instruments to the preset using a “ **drag & drop**”. In the tree you will notice that one or more divisions appear in the preset. Often there may only be one, as in the case of a preset containing a single instrument.

It is possible to create as many or more presets as instruments, and each preset can contain one or more instruments.

Setting up a preset

Changing the preset parameters may not be required since:

- when creating a preset, the bank number and the preset number are automatically assigned,
- when adding an instrument to a preset the key range is automatically calculated according to the instrument.

For the scope of this tutorial, the creation of a soundfont is finished!



HOW TO PREPARE A SAMPLE

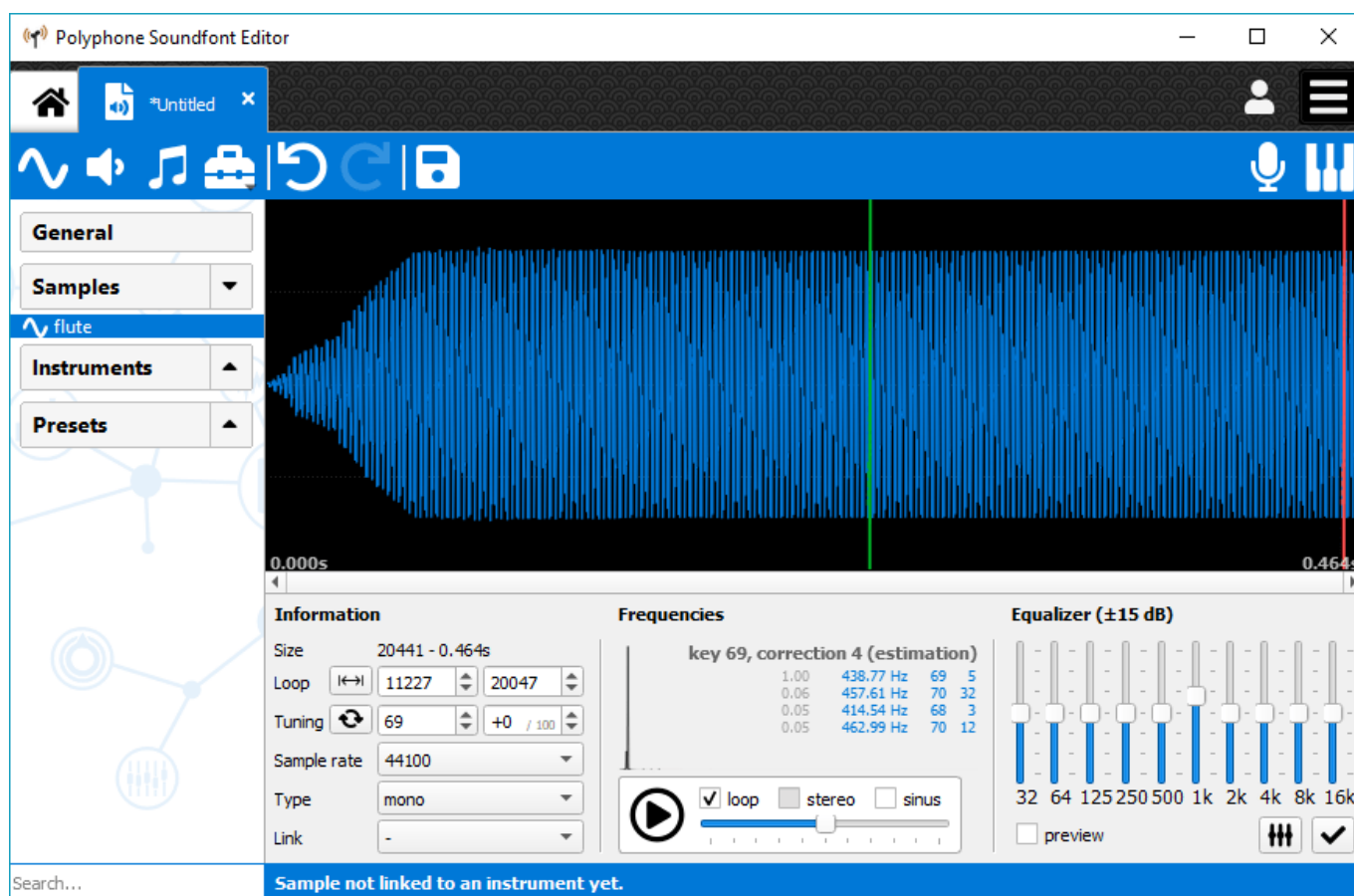
An instrument built in a soundfont is made of samples, coming for example from the recordings of a real instrument. This tutorial describes how to add and prepare a sample in a soundfont.

IMPORT

You must first have samples on your computer. A lot of free samples can be found on internet, being either recorded from real instruments (from a flute, guitar, cello, ...) or created with different kinds of audio synthesis.

For this tutorial this flute sample is provided: [flute.wav](#).

Once you have it, create a new soundfont with Polyphone (**Menu** ► **New**) and import the sample (**Add a sample**).



Import sample



Loop

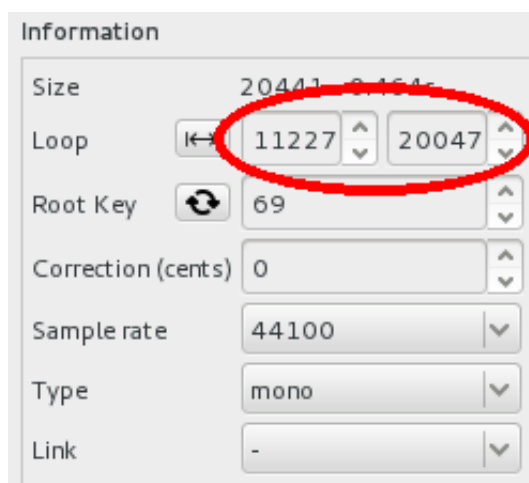
Introduction

A loop is an area within a sample that can be repeated to extend the duration of the sample. Let's say a sample lasts only 0.5 second, with a nice loop the sound can be extended up to 10 seconds or more. This is useful only for samples of non-percussive instruments such as flutes, trumpets, organs, violins. To the opposite, drums require no loops. This might also be the case for a piano or a guitar, depending on how these sounds are synthesized.

Thus, this section is necessary if and only if you feel the need of extending the original length of a sample. If not, go directly to the next section "Tuning".

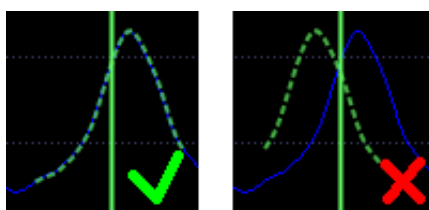
Configuring a loop

In the tree, click on the sample you want to loop and focus on the loop parameters.



Loop sample: loop parameters

If you are lucky, these values were packaged in the .wav sample and these fields are already filled. If not, or if it's been badly configured (this can happen...), arbitrarily click left and right on the graphic (WAV display area) to define the beginning and the end of the loop. This is generally done on the steady phase of the sound, not within the attack or the release. Then, zoom around the beginning of the loop and try to superimpose the green (or red) and blue curves.

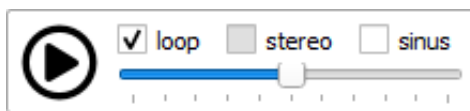


Loop sample: wave alignment



Listening to the result

When the loop is configured, press **play** on the player with the **loop** enabled to check the result.



Loop sample: check the result

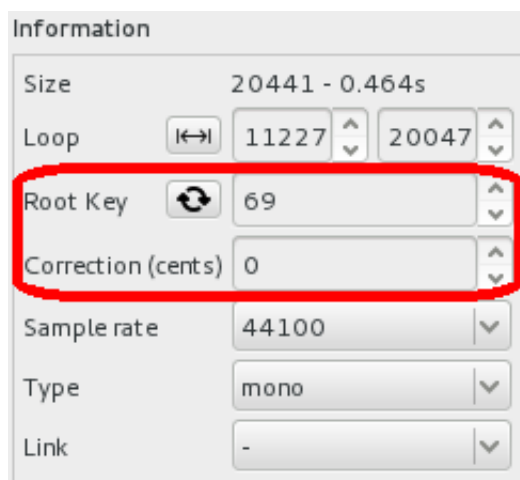
If you don't hear the transition once the end of the loop is read, this means that the loop is good. Otherwise, keep the player on and slightly move the end and / or the beginning of the loop until you find points with a transition as smooth as possible. Try the next records to see the difference between a bad and a good loop:

- [bad loop](#)
- [good loop](#)

Note: A tool exists to “cheat” if you cannot find a loop easily: the **auto-loop** tool. It will automatically create a loop by cutting the sample and superimposing two sections, so that the end of the section will exactly match its beginning. See the related documentation for more information.

TUNING

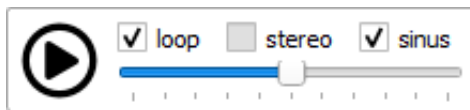
All samples have to be tuned if you don't want an out-of-tune instrument. By indicating in the soundfont the exact pitch of all imported samples, the soundfont will be able to adapt them to keys. Some samples may already have these information, in which case you shouldn't have to change the values in **Root Key** and **Correction (cents)**. But as a precaution it is always good to check.



Tune sample: pitch information



Enable the loop if the sample has one and press **play** on the player. Then add the calibration sine and adjust its volume so that you can hear both sounds as clearly as possible.

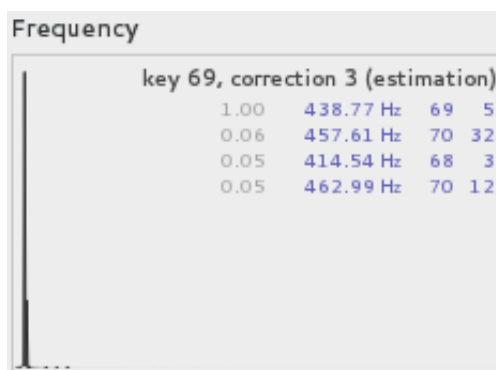


Tune sample: play sample along with calibration signal

Start the tuning by changing the root key until the two sounds most closely match. Continue by adjusting the correction. For this step, pay attention to any beats that occur and make sure they are as slow as possible. Here are examples containing the sound of a flute and a calibration sine:

- **very bad tune** [🔗](#) (the root key doesn't match)
- **bad tune** [🔗](#) (the correction is not correct and you can hear beats)
- **good tune** [🔗](#) (both values are good)

Note: The **frequency analysis** can be a good indicator to start tuning. But keep in mind that no algorithm is perfect for this task and only your ears will find the right tune.



Tune sample: frequency analysis

TO GO FURTHER










Several tools are available in the menu **Tools**. You can edit samples by:

- **removing blank** at start,
- **equalizing** the sound,
- **transposing**,
- **normalizing** the volume,
- **adjusting the balance** for stereo samples.

NOTE ON THE STEREO SAMPLES

Stereo samples are made of two mono samples. The link is stored in the information of a sample.



Information	
Size	346101 - 7.848s
Loop	 20209  34609 
Root Key	 48 
Correction (cents)	-3 
Sample rate	44100 
Type	left 
Link	F_bomb.16 036-CL 

Stereo sample

Managing two mono samples may add difficulty to loop them because a loop can sound fine for one side and not for the other. In that case, don't hesitate to cheat by configuring two different loops on the right and left side. The quality of the sound will not be impaired.



USING CUSTOM RELEASES IN AN INSTRUMENT

The purpose of this tutorial is to create expressive instruments using looped samples whose releases have been recorded. You will learn how to fully use the capabilities of such a sample, by using the attack and sustain of the sample but also its release.

TRADITIONAL RELEASE VS CUSTOM RELEASE

When you trigger a note you will hear first the beginning of the sample (the attack) and then, as long as you don't release the key, the loop within the sample will be played again and again (the sustained phase). When you release the key, the loop will still be used and the sound will progressively be attenuated until you cannot hear it anymore. This is the traditional way to do a release with soundfonts: easy to set but ignoring transient harmonics that could occur during the real release of an acoustic instrument. This type of release can be viewed as a simple extension of the sustained phase.

A more realistic release would be to use the real release provided by the recorded sample. The attack and sustained phase will be done the same way but when the key is released the sustained phase will suddenly drop, leaving room for the recorded release that will begin to sound. A custom release is used instead of extending the sound of the sustained phase, adding details to the sound.

Creating such an instrument is made by splitting the problem in two:

- first defining the attack and sustained phase,
- then triggering the release.

The example that will be taken here is the sound of a trumpet in an organ, for its high harmonic content and audible progression.

SAMPLE PREPARATION

Two cases may appear: you have a file comprising the full sample from the attack to the release, or you have an additional file containing only the release that has been already extracted.

Full sample with release

Load the sample twice in Polyphone and then, in the second instance of the sample, remove everything in the sample except the release. The removal can be done by using the key **Alt** and then defining an area on the **sample graph**. To find the point where the release begins, play the sample several times and try to define the point where you think no more power was inducing the sound. This last part will be the release. Then follow the instructions below.



Separate sample and release

Samples for the example: [sample 1](#), [sample 2](#).

If you have 2 samples, the first one containing the beginning and the sustained phase of the sound and the second one containing the release, you only need to make sure that the release begins with a very short blank (a couple of points) being close to 0. A short inaudible loop will thus be made with this portion.

Tune and loop the first sample as explained in the tutorial “ [how to prepare a sample](#)”.

Loop the second sample on a very short part at the beginning, for example from the point 1 to the point 2 or 3. Specify the same root key and correction than the first sample.

FIRST INSTRUMENT: ATTACK AND SUSTAINED PHASE

Create a first instrument with the samples comprising the attack and the sustained phase of the sound. Configure the global division this way:

- choose the loop option in loop playback (2nd choice),
- specify 0.01s in the volume envelop release (very short release).

Now you have an instrument comprising no releases.

SECOND INSTRUMENT: RELEASE

Create a second instrument for the release comprising the release samples with a very short blank loop at the beginning. Configure it this way:

- choose the option **loop + release** in **loop playback** (3rd choice),
- select a high number such as 100 seconds in the **Vol env release (s)** parameter so that the release is not attenuated,
- determine approximately the duration of the attack in the first sample and use this value in the volume envelop attack.

Now you have an instrument remaining silent as long as you don't release a key, and then you can hear the release phase. You could also notice that a very fast shot on a key will trigger the release in an attenuated way. This is barely noticeable here with the example since the attack of a trumpet is very fast. The duration of the attack has been slightly augmented in the example.

MERGING BOTH INSTRUMENTS



Create a preset using these two instruments. The first instrument is heard when a key is triggered and the second instrument is triggered when the key is released. You can then listen to the result and try to make a smoother transition between the sustained phase and the release:

- by attenuating the first or second instrument,
- by increasing the release duration of the first instrument.



The result with the example files is [here](#) .

You can listen here the differences between:

- a [classical release](#) 
- and a [custom release](#) 




USE POLYPHONE AS A SYNTHESIZER

Polyphone isn't just a soundfont editor, it's also a full-featured synthesizer that lets you play along with a MIDI keyboard. This tutorial shows you how to launch Polyphone in synth mode.

PREREQUISITES

First you need a **master keyboard** or other MIDI controller, which you connect to the computer via a **MIDI-to-USB adapter**.

Download the latest version of Polyphone [here](#) , then install and open the program. In the settings, you should find your MIDI device among the **MIDI inputs**. If this is not the case, it's likely that the MIDI adapter driver is not installed on the computer (resolve this before continuing).

Still in the settings, check that the **audio output** is correct.

OPEN POLYPHONE IN SYNTH MODE

Windows

After installing Polyphone, click on the **Polyphone (synth mode)** shortcut in the Windows start menu.

Linux

Open a terminal and write the following command line:

```
polyphone -s
```

Mac

Open a terminal and write the following command line:

```
/Applications/Polyphone.app/Contents/MacOS/polyphone -s
```

Note: if you have moved the Polyphone application to a folder other than **Applications**, you will need to adapt the above path.



PLAYING AN INSTRUMENT

To play an instrument, simply open a soundfont that you can download from the [repository](#), for example, and select an instrument contained in that soundfont. Instruments are arranged by bank: select a bank to display the instrument list below.



Polyphone opened with the synth mode option

Once you have selected an instrument, its key range is displayed on the virtual keyboard, and you can play it using the corresponding keys on your MIDI controller.

OPTIONS

Playing several instruments at the same time

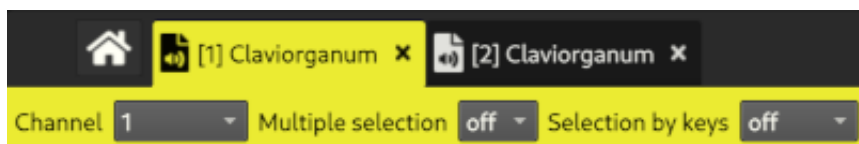
To play several instruments when a key is pressed on the keyboard, simply set the **Multiple selection** option to **on**. Multiple instruments can now be selected in the instrument list.



Playing several instruments separately

If you have several MIDI controllers connected to the computer, you can open several soundfonts and choose a specific channel for each of the opened soundfonts. In this way, for example, a first keyboard will be used to play one instrument, and a second keyboard will be used to play another one. The same soundfont can be opened several times to play several of its instruments simultaneously.

Channel assignment is done by modifying the value of **channel**. For example, the first tab opened will listen to channel 1, while the second will listen to channel 2.



2-channel synth configuration

Note: some keyboards can be split into two parts, so that a single keyboard can be used to send keys to 2 different channels.

Switching instruments with a key

Switching the instrument is generally made via a MIDI message of type "Program change" and Polyphone is able to read these.

However, another mechanism can be used: the use of keys programmed to send "Note ON" / "Note OFF" messages, with the note number corresponding to the instrument number. This mechanism can be useful when the soundfont features only a few instruments, and when keys can be programmed to send specific notes (e.g. master keyboard pads).

If the **Selection by keys** option is set to **on**:

- an instrument is activated when a "Note ON" MIDI message is received,
- an instrument is deactivated when a "Note OFF" MIDI message is received.

This option can be used when the pad on a master keyboard is configured as a toggle (one press activates the note, another press deactivates it).

If **toggle** is selected:

- an instrument is activated when a "Note ON" MIDI message is received,
- an instrument is deactivated when another "Note ON" MIDI message is received.

This time, the switchover takes place at Polyphone level.

POLYPHONE LAUNCH OPTIONS

All the options seen above can be called up when opening Polyphone by adding parameters to the



command line. It is also possible to specify the soundfont to be opened.

For **Linux** or **macOS**, simply complete the command line. For **Windows**, create a shortcut by right-clicking on the executable file **polyphone.exe**, then select **Send to desktop (create shortcut)** from the **Other options** of the context menu. Then, by displaying the properties of this shortcut, it is possible to add parameters to the target, which could be:

```
"C:\Program Files\Polyphone\polyphone.exe" -s -c "all|on|toggle"
```

The various options are described on this [page](#).

SYNTH USES

Now that you know how to turn Polyphone into a synthesizer, why not [use a Raspberry Pi to create a musical instrument?](#)

Polyphone's synth mode can also be used to [read MIDI files](#).



CREATE A SYNTHESIZER WITH A RASPBERRY PI

Coming soon!



READ A MIDI FILE WITH A SOUNDFONT

MIDI files contain MIDI messages which, when put together, form a melody. They can be combined with a synthesizer using a soundfont, and this tutorial shows how to use a MIDI file to test the rendering of a soundfont.

A BIT OF BACKGROUND

When playing on a synthesizer keyboard, [MIDI](#) signals are emitted, containing information such as the notes played and their velocity (a term indicating the force with which the key is struck). A MIDI file, in simple terms, contains a chronological record of the signals emitted by such a keyboard. Until then, no sound has yet been created.

For a sound to be emitted by the loudspeaker, a conversion step is necessary, and this is where the synthesizer comes in: it triggers sounds from the MIDI signals received. Several types of synthesizer exist, and here we will be talking about sample based synthesizers, and more specifically synthesizers using sound banks in the SoundFont format.

A MIDI file can control up to 16 different channels, allowing 16 different instruments to be used in parallel. As well as indicating which notes to play, the MIDI file contains instructions telling the synthesizer to use instrument X from bank Y for the upcoming notes on channel Z. To ensure a correct instrument selection, the sound bank used by the synthesizer must comply with the [General MIDI](#) standard, which assigns a specific instrument to each X and Y number. For example, instrument **001** must be mapped to a piano in the soundfont.

To recap, we are going to talk here about the playback of a MIDI file driving a synthesizer that triggers sounds contained in a soundfont that complies with the General MIDI standard. What follows is simpler!

PREREQUISITES

To play a MIDI file using the method described below, you will need:

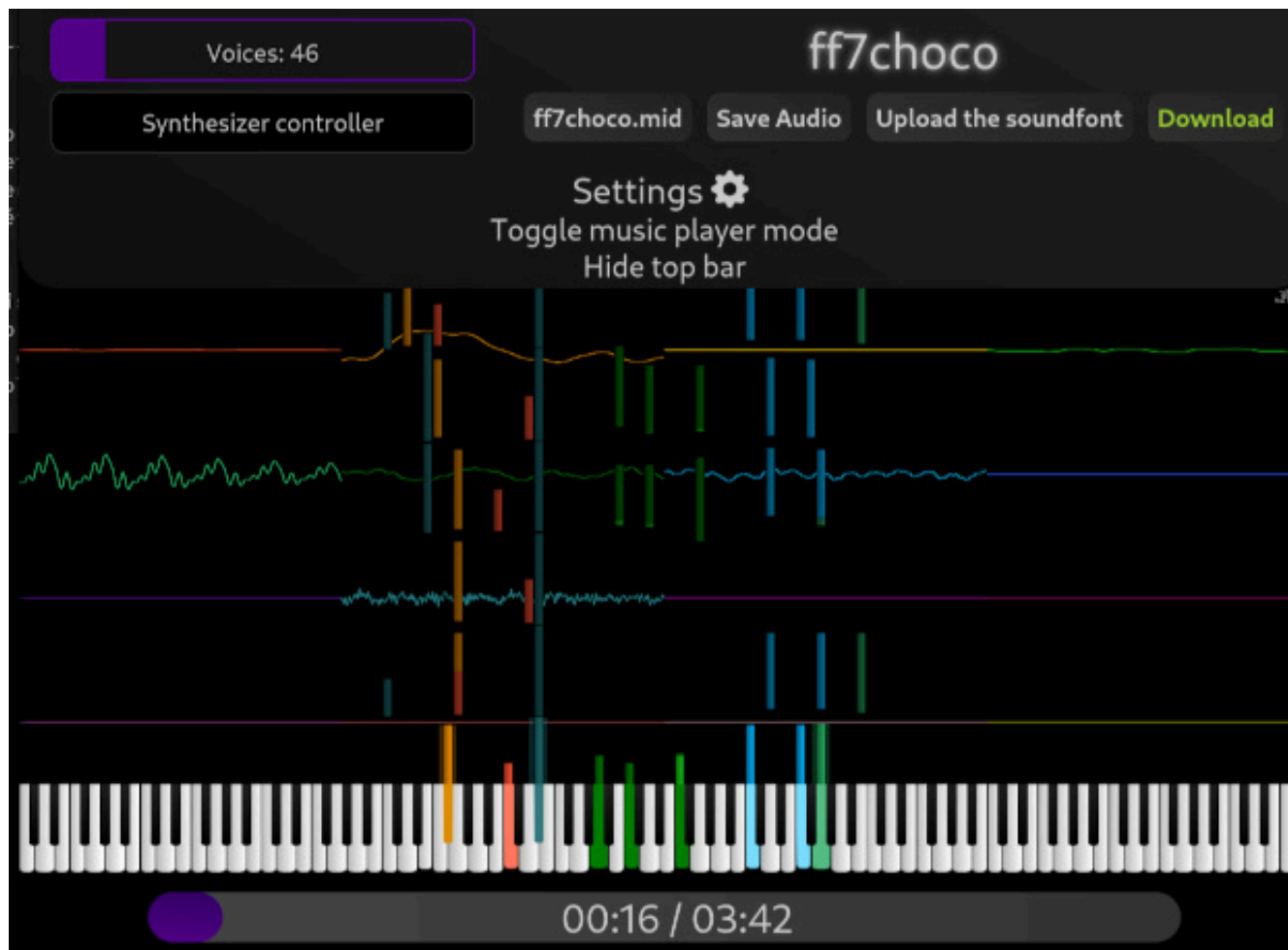
- an internet access,
- a web browser,
- a MIDI file, such as [this one](#),
- optionally, a soundfont that you want to test and that complies with the [General MIDI](#) standard (a filter exists in the [repository](#)),
- optionally, the latest version of Polyphone available on this [page](#).



READING A MIDI FILE

A simple method is to use the [SpessaSynth](#) synthesizer, which is both a MIDI file reader and a soundfont synthesizer. It has the advantage of being available online and requires no special installation.

Click on the **Load MIDI file** button, select your MIDI file, and you are ready to go.



The SpessaSynth MIDI player and synthesizer

SELECTING A SOUNDFONT

To use a soundfont of your choice, click on the **Upload soundfont** button and select your file.

CONNECTING THE MIDI OUTPUT TO POLYPHONE

As mentioned above, **SpessaSynth** is both a MIDI file reader and a synthesizer. It is possible to use only the MIDI file reader feature and send the signals to Polyphone, which will take over for sound synthesis. This can be useful if the soundfont is too large to be sent over the Internet.



To connect SpessaSynth to Polyphone (method tested with Linux):

- launch the Polyphone software in **synth mode** and open the configurations,
- select "MIDI through" as **MIDI input** (remember the name),
- open SpessaSynth in your web browser and click on the **Configurations** button,
- select the MIDI output "MIDI through" (identical name to Polyphone's MIDI input).

As a MIDI file can send messages to 16 channels, you will need to open the soundfont 16 times and specify a different channel from 1 to 16 for each tab. For convenience, when opening Polyphone, you can specify the soundfont to be opened and the option "each" to open all 16 tabs. The command would be, for example:

```
polyphone -s -i "/home/davy/Downloads/gm.sf2" -c "each/off/off"
```

See this [page](#) for more information about command lines.



TECHNICAL DOCUMENTATION



All sources of the software are hosted by GitHub and developers can go [here](#) if they want to help in the project.

Some instructions can be found below. If something is not up to date, don't hesitate to report it in the [forum](#).

Building Polyphone for Windows

Instructions to build Polyphone for Windows.

Building Polyphone for Mac OS X

Instructions to build Polyphone for Mac OS X.

Building Polyphone for Fedora

Instructions to build Polyphone on Fedora.

Building Polyphone for Ubuntu

Instructions to build Polyphone with Ubuntu.

Using Qt Creator to build Polyphone

Instructions to build Polyphone with Qt Creator.

Translate Polyphone

For all people willing to see Polyphone in their own language.



BUILDING POLYPHONE FOR WINDOWS

Use these instructions for building Polyphone for Windows with the latest version of the sources on [GitHub](#). You will have the software at its most advanced level of development, including the latest features, but perhaps also some instabilities.

PREREQUISITE

Qt installation

Polyphone is written in C++ with the Qt development environment. To install Qt, go to this [page](#) and download the installer. You will need to create an account first.

Then launch the installer and select the following components :

▶ Qt Design Studio	1.0.0-0	2024-06-20	
▶ Extensions	1.0.0-0-202406111003	2024-06-11	
▼ Qt	1.0.19	1.0.19	2024-05-23 2.76 Go
▶ <input type="checkbox"/> Qt 6.8.0-beta2	6.8.0-0-202407101146	2024-07-10	
▼ <input checked="" type="checkbox"/> Qt 6.7.2	6.7.2-0-2024061...	6.7.2-0-202406110335	1.28 Go
<input checked="" type="checkbox"/> MinGW 11.2.0 64-bit	6.7.2-0-2024061...	6.7.2-0-202406110335	1.03 Go
▼ <input checked="" type="checkbox"/> Developer and Designer Tools	1.2.2-0-2024052...	1.2.2-0-202405230632	2024-05-23 1.48 Go
<input type="checkbox"/> LLVM-MinGW 17.0.6 64-bit	17.0.6-202401290621	2024-01-29	
<input checked="" type="checkbox"/> Qt Creator 14.0.1	14.0.0-0-202407...	14.0.1-0-202408060...	2024-08-06 560.23 Mo
<input checked="" type="checkbox"/> Qt Creator 14.0.1 CDB Debugger Support	14.0.0-0-202407...	14.0.1-0-202408060...	2024-08-06 103.02 Mo
<input checked="" type="checkbox"/> Debugging Tools for Windows	2024-02-26-0-20...	2024-02-26-0-20240...	2024-08-06 68.47 Mo
<input type="checkbox"/> Qt Creator 14.0.1 Debug Symbols	14.0.1-0-202408060...	2024-08-06	
<input type="checkbox"/> Qt Creator 14.0.1 Plugin Development	14.0.1-0-202408060...	2024-08-06	
<input type="checkbox"/> MinGW 13.1.0 64-bit	13.1.0-202401290621	2024-01-29	
<input checked="" type="checkbox"/> MinGW 11.2.0 64-bit	9.0.0-1-2022032...	9.0.0-1-202203221220	2022-03-22 571.68 Mo
<input type="checkbox"/> MinGW 8.1.0 32-bit	8.1.0-1-202004170606	2020-04-17	
<input type="checkbox"/> MinGW 8.1.0 64-bit	8.1.0-1-202004170606	2020-04-17	
<input type="checkbox"/> MinGW 7.3.0 32-bit	7.3.0-1-202004170606	2020-04-17	
<input type="checkbox"/> MinGW 7.3.0 64-bit	7.3.0-1-202004170606	2020-04-17	
<input type="checkbox"/> MinGW 5.3.0 32-bit	5.3.0-2	2017-04-27	
<input type="checkbox"/> MinGW 4.9.2 32-bit	4.9.2-1	2016-05-31	
<input type="checkbox"/> MinGW 4.9.1 32-bit	4.9.1-3	2016-05-31	
<input type="checkbox"/> MinGW 4.8.2 32-bit	4.8.2	2014-05-08	
<input type="checkbox"/> MinGW 4.8 32-bit	4.8.0-1-1	2013-07-01	
<input type="checkbox"/> MinGW 4.7 32-bit	4.7.2-1-1	2013-07-01	
<input checked="" type="checkbox"/> CMake 3.29.3	3.29.3-20240515...	3.29.3-202405151154	2024-05-15 112.55 Mo
<input checked="" type="checkbox"/> Ninja 1.12.0	1.12.0-20240515...	1.12.0-202405151040	2024-05-15 555.54 Ko
▶ <input type="checkbox"/> OpenSSL 3.0.12 Toolkit	3.0.12-1	2023-12-04	
Qt Maintenance Tool	4.8.0-202405141...	4.8.0-202405141156	2024-05-14 47.44 Mo



Components to select

Take the latest version available if these choices are not exactly the same.

Compiler installation

Qt needs a compiler to compile Polyphone sources. It is necessary to use MinGW for the following reasons:

- there's above all a practical reason, since the dependencies below are compatible with MinGW
- but also to enable correct import/export of files in sfArk format.

You can download MinGW on this [page](#), taking the version corresponding to what has been selected in the previous paragraph.

Unarchive the whole in directory **C:/mingw64** for instance.

Then go to Windows system settings to add the folder **C:/mingw64/bin** to the Path :

- open the Windows menu,
- type "Environment variables",
- open the system properties and click on "Environment variables",
- double-click on "Path" in the system variables,
- and add **C:/mingw64/bin** to the directories listed in variable "Path".

In doing so, mingw64 executables can be called by Qt.

Polyphone dependencies

Download the project dependencies available [here](#). These are additional elements used by Polyphone that are not supplied by Qt. If some of the dependencies are out of date, it is possible to download them [here](#).

BUILD

Sources

Start by downloading the Polyphone sources [here](#) or from [GitHub](#) and unarchive the files on your Windows desktop, for example.

Then, unarchive the additional Polyphone dependencies in directory **windows** that must be next to the **sources** folder.

Continue now with this tutorial for building Polyphone with Qt Creator: [Using Qt Creator to build](#)



Polyphone. No modification of the file **polyphone.pro** will be necessary.

Launch Polyphone without Qt Creator

Go to the directory **windows/bin**. If compilation went well, you should see the executable file **polyphone.exe**. To enable it to run independently, this folder must be completed. Run the following command in a terminal (adapt the paths beforehand):

```
C:\Qt\6.7.2\mingw_64\bin\windeployqt.exe "C:\Users\User\Desktop\polyphone\windows\bin\polyphone.exe"
```

The Qt libraries will appear (**Qt6Core** for example) and a double-click on file **polyphone.exe** will launch the software. To enable Polyphone to be launched on another Windows computer, these files from **C:\mingw64\bin** are also needed to complete the folder:

- libgcc_s_seh-1.dll
- libssp-0.dll
- libstdc++-6.dll
- libwinpthread-1.dll

TROUBLESHOOTING

This forum [category](#)  will give you some help if needed.



BUILDING POLYPHONE FOR MAC OS X

Use these instructions if you cannot install Polyphone on your Mac with the available installers in the section "[Software](#)".

PREREQUISITE

The following applications are required:

- [Qt Creator with its framework](#),
- Xcode or just the "Command Line Tools" (the version to use is specified on the Qt website - [here](#) for instance).

You also need the sources of Polyphone, available [here](#) or on [GitHub](#), along with the necessary libraries you can download [here](#).

For correctly opening a project with Qt Creator, it must first be given the permission to read all files from your computer (in "Security and privacy" => "Full disk access" => add "Qt creator.app" which is in its installation directory).

If some of the previously provided libraries are not up to date or not compatible with the version of Mac, like SSL for example, it will be necessary to look for their sources on the internet and to recompile them by enabling a static linking:

```
configure --enable-static --disable-shared --prefix=...
make
make install
```

Then copy all .h (headers) and .a (static libraries).

Another way to get the libraries already built is to use [Homebrew](#). After installing this tool, it is for example possible to get "libsndfile" along with its dependencies with this command:

```
arch -x86_64 brew install libsndfile
```

The .h and .a files will then be located in "/usr/local/Cellar".

BUILD

Unzip the libraries and place the directory **macos** just next to the directory **sources**.



Open the file **polyphone.pro** with **Qt Creator**. Build the project, a bundle **polyphone.app** should appear in the directory **macos**.

If the SDK path cannot be resolved, try modifying this variable in file **polyphone.pro** for matching your SDK version (linked to the Xcode version):

```
QMAKE_MAC_SDK = macosx10.15
```

If the compiler is now complaining that some features are not available with the specified MacOS target, you can increase this variable in the .pro file:

```
QMAKE_MACOSX_DEPLOYMENT_TARGET = 10.15
```

This value is also to be written in file "sources/contrib/mac/polyphone.plist" below "LSMinimumSystemVersion".

COMPLETE THE BUNDLE

Run the following command to include libraries and frameworks inside the bundle (adjust the path first!):

```
/Path/To/Qt/5.2.0/clang_64/bin/macdeployqt /Path/To/macos/polyphone.app
```

You can then execute the program or compress it as a .zip or .dmg file to share it.

TROUBLESHOOTING

See this [topic](#) if you need more information or for getting some help.



BUILDING POLYPHONE FOR FEDORA

Use these instructions if you cannot install Polyphone on your Fedora distribution with the available installers in the section "[Software](#)" (verified: Aug 27, 2020 on Fedora 32, 64bit).

PREREQUISITE

Qt installation

After a base installation of Fedora, install the following packages (including a compiler) as pre-requisites for Qt development:

```
sudo dnf update
sudo dnf install @development-tools
```

Then you can either install the default Qt framework with the following command:

```
sudo dnf install qt-creator qt5-qtsvg-devel qt5-qttools-devel
```

Polyphone dependencies

Install the following dependencies (if you have **Synaptic** installed you could alternatively use it for installation):

```
sudo dnf install alsa-lib-devel jack-audio-connection-kit-devel rtaudio-devel zlib-devel libogg-devel flac-devel libvorbis-devel glib2-devel openssl-devel rtmidi-devel stk-devel libsndfile-devel
```

BUILD

Sources

First, get Polyphone sources from [here](#) or from [GitHub](#).

If you want to continue the build with Qt Creator, follow now this tutorial: [build with Qt Creator](#). Otherwise, continue the reading of this document.



Getting the executable

Go into source directory (where the file **polyphone.pro** is located), open a terminal and build Polyphone with this command:

```
qmake-qt5 && make
```

If everything goes well you will have the compiled file **polyphone** in the **build-XXX** subdirectory next to **sources**.

TROUBLESHOOTING

This forum [category](#)  will give you some help if needed.



BUILDING POLYPHONE FOR UBUNTU

Use these instructions if you cannot install Polyphone on your Linux distribution with the available installers in the section "[Software](#)". This method has been tested with Ubuntu 16.04 (Xenial).

PREREQUISITE

The following libraries are required:

- qt (qt5-default + libqt5svg5-dev)
- alsa (libasound2-dev)
- jack (libjack-jackd2-dev)
- rtaudio (librtaudio-dev)
- rtmidi (librtmidi-dev)
- stk (libstk0-dev)
- vorbis (libvorbis-dev)
- ogg (libogg-dev)
- flac (libflac-dev)
- ssl (libssl-dev)

Use for example **synaptic** to install them.

You also need the sources of Polyphone, available [here](#) or on [GitHub](#).

BUILD

In the root directory of the project, open a terminal and build Polyphone by running the following command:

```
qmake polyphone.pro PREFIX=/usr && make
```

An executable file **polyphone** in the directory **RELEASE** should then appear if everything went right.

In the case where libraries such as **RtAudio**, **RtMidi**, **Stk** are missing or are not in a compatible version in your distribution, you can edit the **.pro** file to use local copies of them. To do this, uncomment the corresponding lines. For instance

```
#DEFINES += USE_LOCAL_RTMIDI
```

will become



```
DEFINES += USE_LOCAL_RTmidi
```

(with no leading "#").

Note: if you are using **Qt Creator**, the project may be opened via its **.pro** file present in the root directory.



USING QT CREATOR TO BUILD POLYPHONE

Use these instructions to know how to build Polyphone with Qt Creator. It is assumed you already have on your computer Qt Creator installed with the Qt framework, a compiler and the necessary dependencies to build Polyphone (see related pages for this).

PREREQUISITE

The following elements are required:

- a C++ compiler (VC++, Clang, GCC, ...),
- Qt Creator with its framework,
- Polyphone dependencies.

The instructions for installing them are dependent on the OS you are using, see the related documentation pages for this.

BUILD POLYPHONE

Copy the sources

Get Polyphone sources from [here](#) or from [GitHub](#).

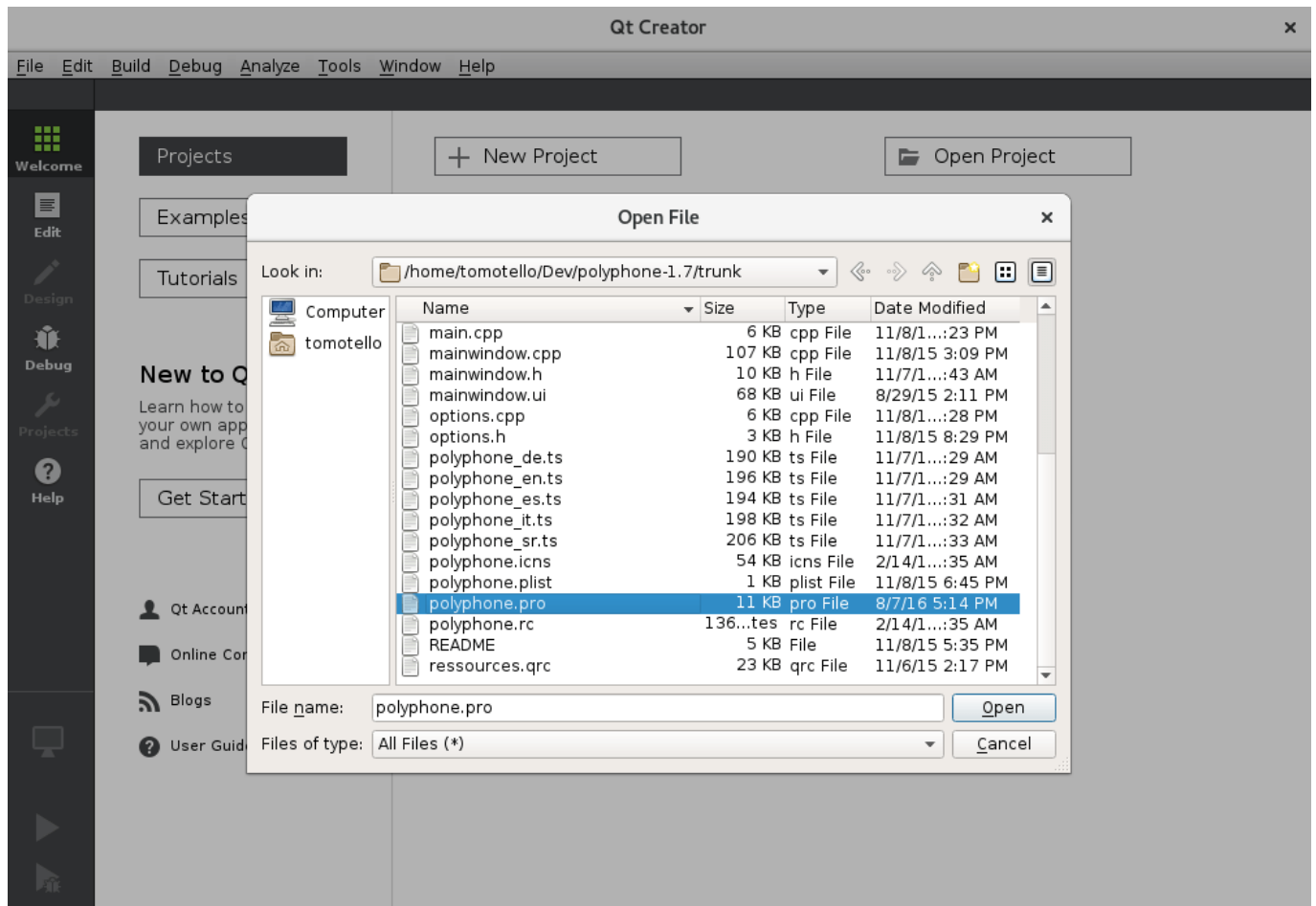
Open and configure the project

First, open the file **polyphone.pro** and edit it to possibly use the distributed versions for **rtaudio**, **rtmidi** and **stk** if your OS doesn't provide them. Uncomment the following lines by removing the leading '#':

```
DEFINES += USE_LOCAL_RTAUDIO
DEFINES += USE_LOCAL_RTMIDI
DEFINES += USE_LOCAL_STK
```

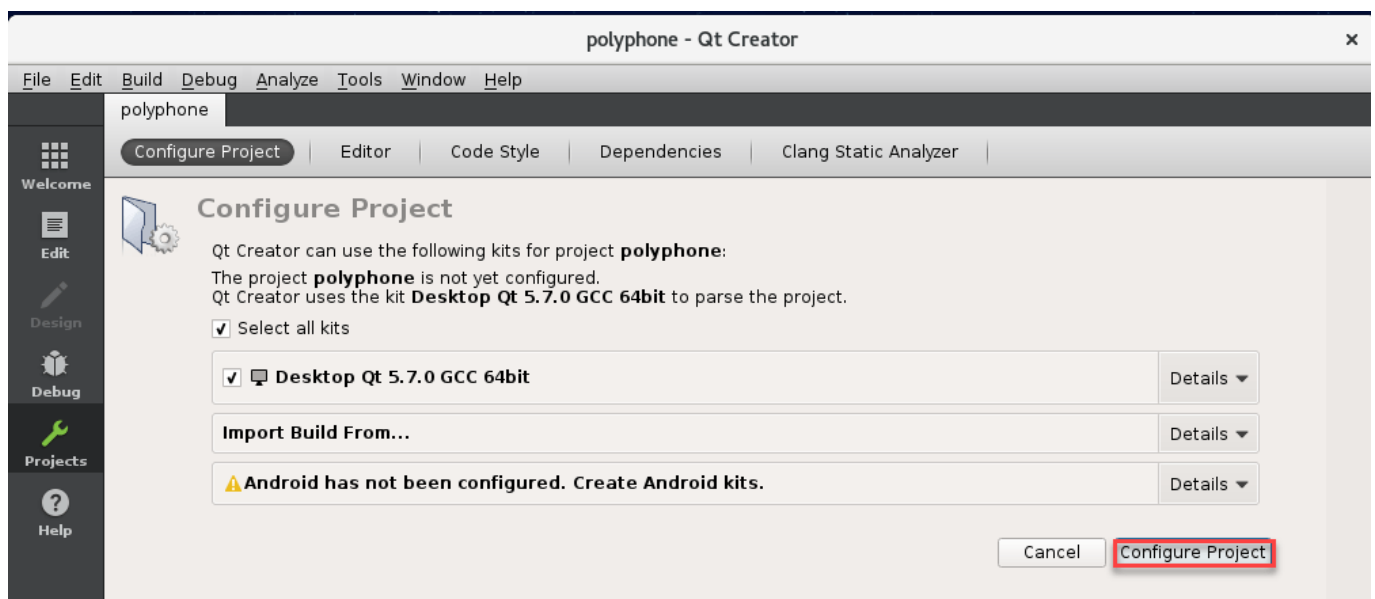
You can skip these modifications if you want to use libraries already installed in your computer (make sure you have them!).

Start Qt Creator and open the file **polyphone.pro**.



Open Qt Creator

Then configure the project.

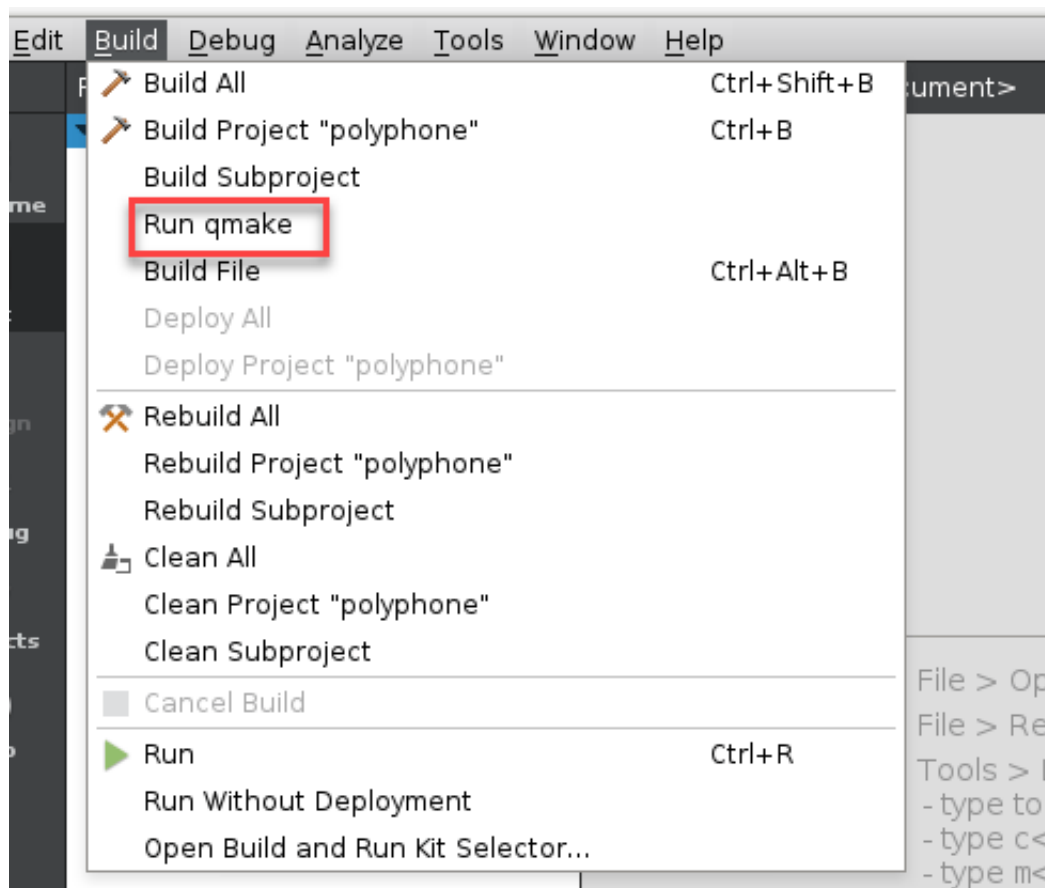


Configure project



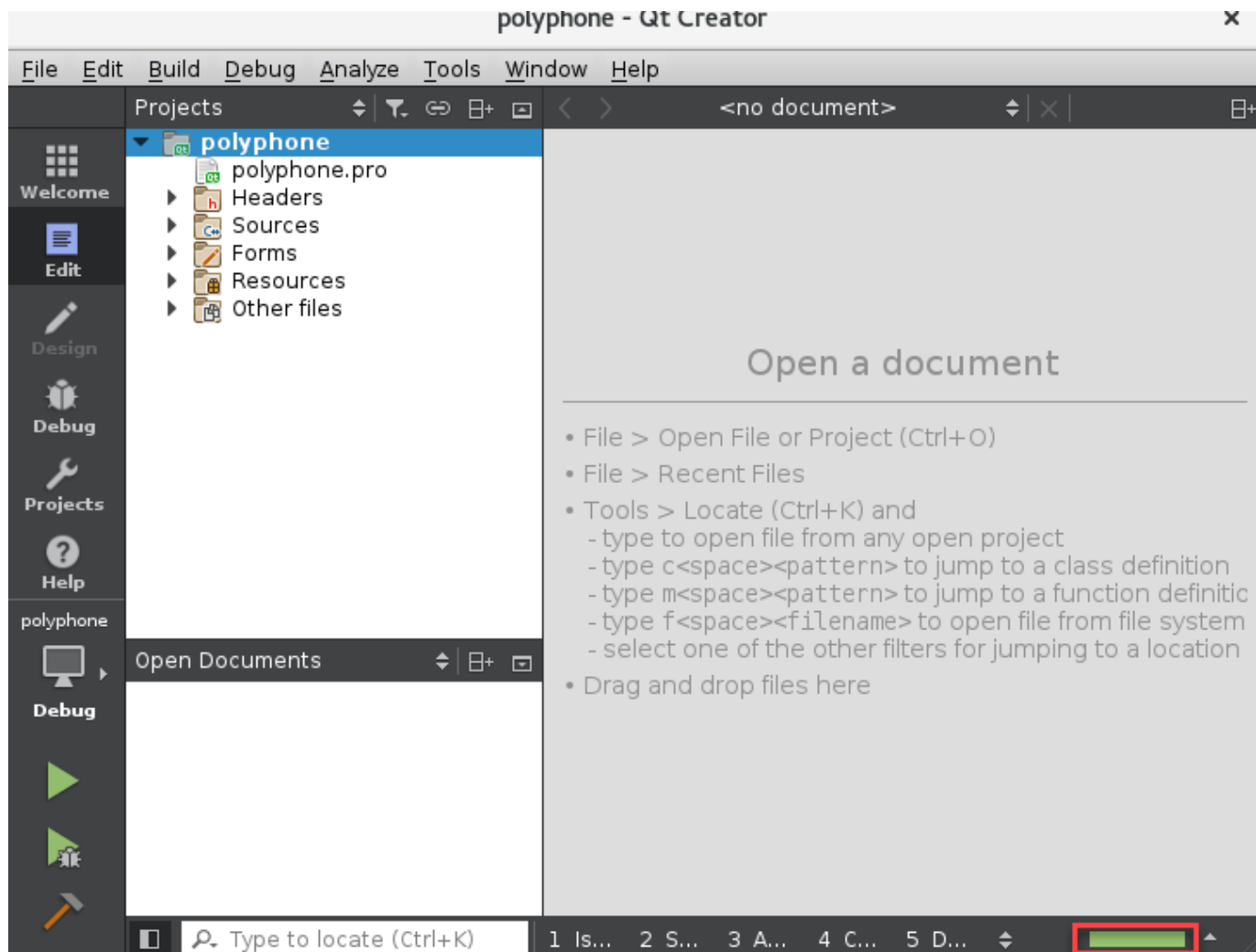
Run qmake and build the project

Go in the menu **Build** and choose the item **Run qmake**.



Run qmake

`qmake` run is finished when the proceeding bar on the bottom right corner has reached the 100% green.



qmake finished

In the section **Project** in the left column, then in the section **Build&Run**, select the build configuration (**Release** is preferred).

Finally, click on the action **Build project Polyphone** in the menu **Build**. During the project build you should see warnings but no errors.

Now you should have the executable **polyphone** either in the **RELEASE** or **DEBUG** subfolder of the source code, depending on your choice in QtCreator build configuration.

TROUBLESHOOTING

See this [topic](#) if you need more information or for getting some help.

Special thanks to [Tomotello](#) for this article.



TRANSLATE POLYPHONE

Polyphone has been designed to support translations in order to increase the accessibility. Everyone can help in this process.

PREREQUISITE

Translation file

Download the translation package corresponding to the language you want to translate from this [page](#). If the language is not in the list yet, use the [contact](#) form to ask for it (you just need to specify the new language you want). You will quickly be provided with a new translation package.

Qt Linguist

The only software you need to translate is **Qt Linguist**: a tool for generating translation files. This software can be found [here](#) for Windows users or in repositories for Linux users (via **Synaptic**, **Yum**, ...).

OPENING THE TRANSLATION FILE

Unzip the translation package you just downloaded. You will find a .ts file **polyphone_XX.ts**, **XX** being related to your language. With **Qt Linguist**, open **polyphone_XX.ts** and you will be able to start the translation.

TRANSLATING

Each item in each context needs to be translated. All you have to do is to review each element one by one and fill your translation. Three cases may appear depending on your confidence level.

- If you don't know how to translate don't hesitate to skip the element. This is not an issue: other people can complete it.
- You think you have the right translation but you are unsure: write your proposition and then use **Ctrl+J**. The item will not be validated but the translation will nevertheless be used in the translated interface. It just tells other people improving the translation that you were unsure.
- You have the right translation: write it and use **Ctrl+Enter**. This item will be validated.

A translation needs attention (in a world where everything is being made for [soliciting it](#)). Please take care of:



- the punctuation,
- the first and final spaces,
- the character case (lowercase / uppercase),
- singular and plural forms.

A final advice: **take your time!** And don't forget to save your work with **Ctrl+S**.

TESTING YOUR WORK

At anytime you can check your work. In **Qt Linguist** go to **File ► Release As...** and create a .qm file corresponding to the translation you are providing. Follow the instructions in the translation package to install it (**readme.txt** file), then open Polyphone. Go to the configurations, select your translation and restart Polyphone. You will now be able to review your translation.

Note: the operation **Release As...** may translate only text comprising no accent. If this bug happens, open a terminal and use the following command: `lrelease polyphone_XX.ts` (adapt it to your current file).

SUBMITTING THE TRANSLATION

Once you are done (or bothered!), send the .ts file (not .qm!) you were editing to contact@polyphone.io. The translation package of this [page](#) will be updated.