



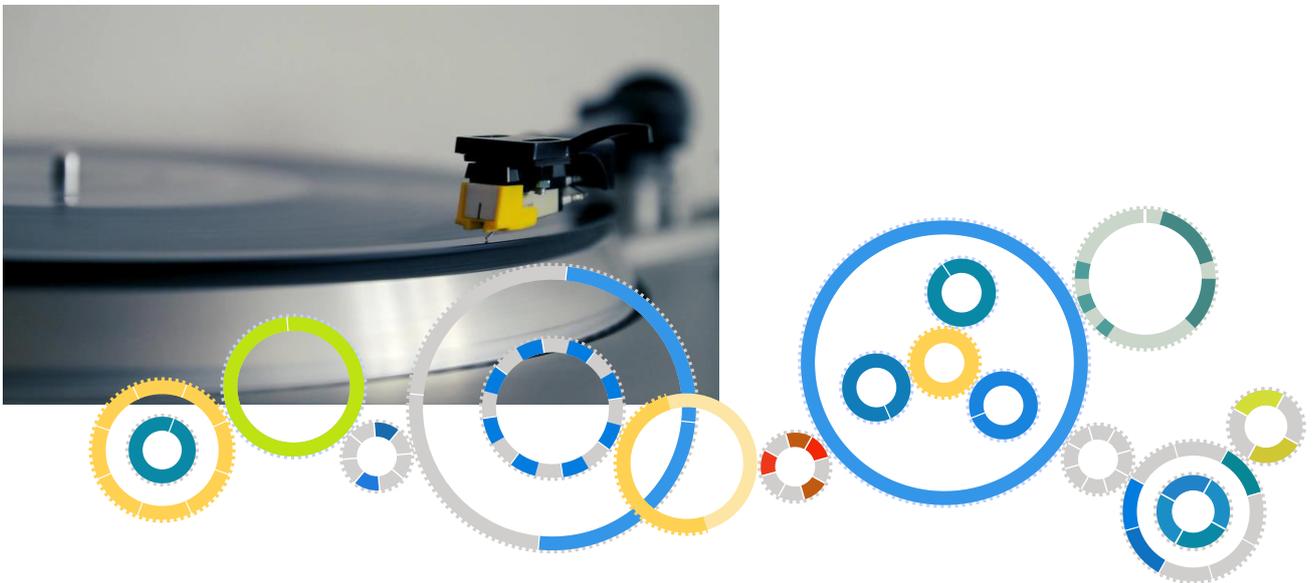
# foreverloops

play a round music

Unleash your ingenuity with foreverloops – a hybrid sandbox game and productivity software for creative minds and discerning gamers.

Virtual gears let you interface with a new world of media compositions and performances to produce killer beats, rhythmic mashups, audio visual collages, and so much more.

A unique and rewarding experience awaits!



[www.pegi.info](http://www.pegi.info)

last update: 12/20/2018, foreverloops 1.3

## contents

health warning.....	1
advice / games & health (PEGI) .....	1
quick start.....	2
create gears.....	3
connect / stack gears .....	3
program gears .....	4
gear context.....	5
user interface.....	6
canvas .....	6
menu bar.....	7
sample browser .....	9
video panel .....	10
virtual MIDI keyboard.....	10
context menu.....	11
playing.....	12
gear mechanics .....	12
media types.....	15
gear types.....	17
options .....	32
general settings.....	32
display settings.....	34
sound settings .....	34
interaction settings.....	35
export settings.....	36
key bindings .....	36
tips & tricks & troubleshooting.....	37
video decoding .....	37
audio decoding.....	37
hardware requirements.....	37
loading .....	37
system requirements.....	38
support / contact .....	39
about foreverloops .....	39
credits / special thanks / promoters.....	40
concept & idea / project lead / software engineering .....	40

interaction design / art direction / qa .....	40
application design.....	40
sound samples.....	40
SoundFont.....	40
video samples .....	40
demo composition .....	40
translations.....	40
special thanks .....	41
promoters .....	41
end-user license agreement .....	42

### advice / games & health (PEGI)

Gaming does not pose specific health risks. It is a leisure activity, just more interactive than reading books or watching television. As it involves the use of your eyesight, we generally advise players to follow these guidelines:

#### always play in a well-lit room

Take regular breaks.

Don't play a game for hours on end, no matter how compelling it may be.

Keep a minimum distance from the screen.

#### is it possible for my child to become addicted to games?

Playing video games is not a physical addiction. Children play games because they enjoy them; and some enjoy them more than others. While playing a video game they may be blinkered to everything else that goes on around them (the same goes for watching movies or television).

#### can gaming cause epilepsy?

No. Exhaustive research has found that epilepsy cannot be caused by playing videogames. However, an extremely small number of people, who already have Photosensitive Epilepsy, may actually realize that they have this condition when playing videogames. However, experiences such as watching television, going to a discotheque, or even experiencing light flickering through trees can bring about.

for more information:

[www.pegi.info/en/index/id/11](http://www.pegi.info/en/index/id/11)

[www.pubmedcentral.nih.gov](http://www.pubmedcentral.nih.gov)

[www.isfe.eu](http://www.isfe.eu)

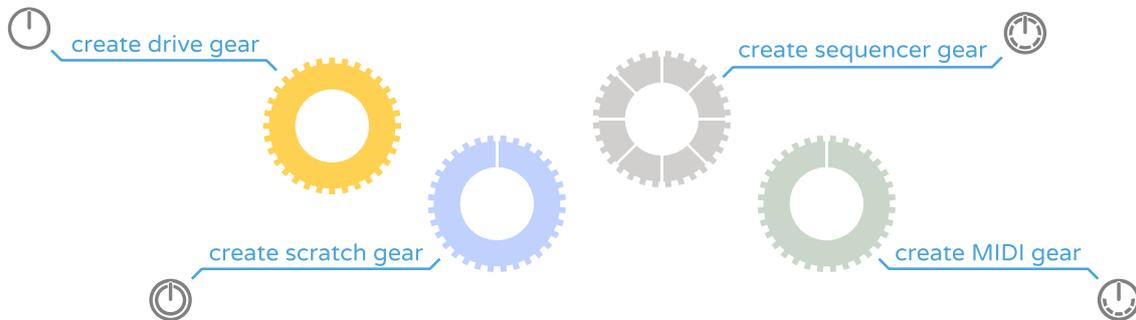
In foreverloops, gears combine mechanical principles (i.e., transmission, stacking, driving) with the virtuality of a software simulation to afford many opportunities for audio-visual creations.



Create and program your own gear trains and evolve them from simple experiments to awe-inspiring complex media machineries that can be tweaked and customized in real-time.

## create gears

To create gears, simply drag a gear icon from the menu bar onto the canvas. There is no limit to the number of constructible gears you can create.



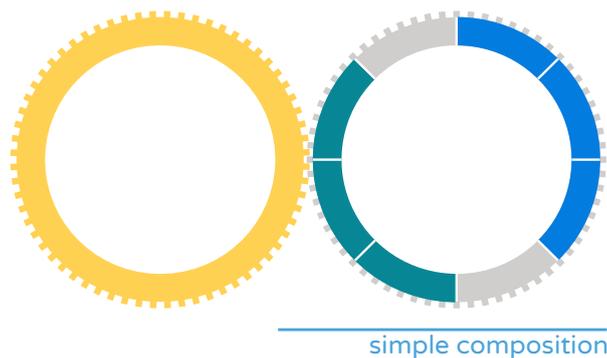
Currently, foreverloops supports four different gear types.

Drive gears activate gear trains and are responsible for the tempo.

Sequencer gears and scratch gears can be setup with media samples (audio, video, image), which play back as the gears rotate.

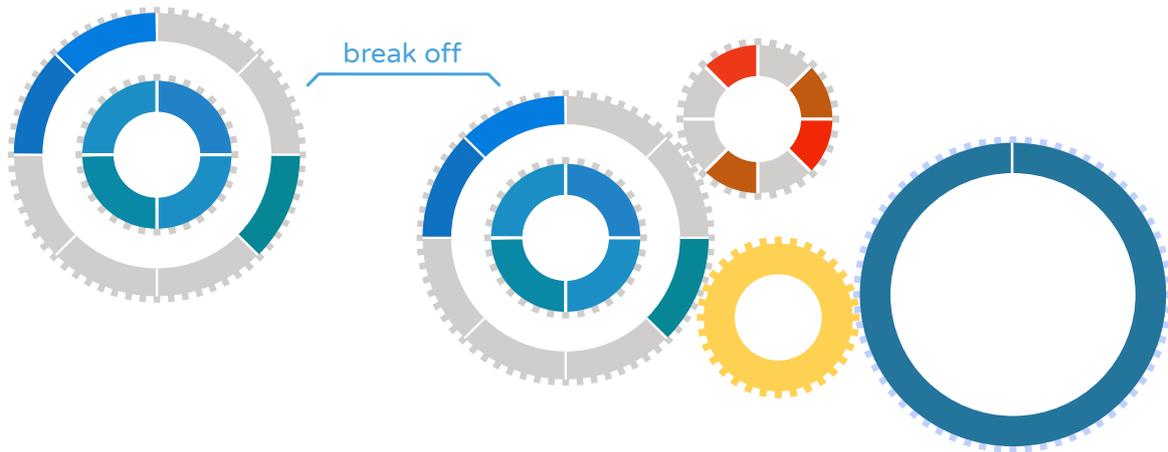
Midi gears can be programmed so called MIDI notes – they are audible instrument notes and can be recorded from an external MIDI instrument, from the virtual on-screen keyboard (using SoundFonts), or directly set via edit-view.

For a simple composition, you need a drive gear that is connected with at least one media gear filled with samples.



## connect / stack gears

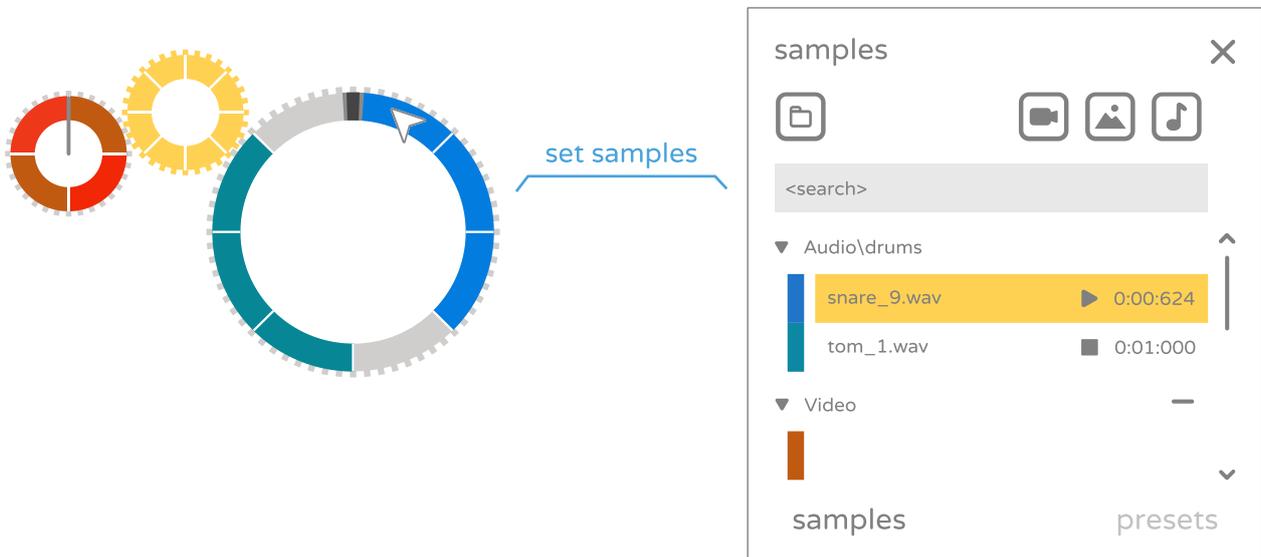
It is also possible to directly connect or stack newly created gears. This way, you can construct complex gear trains quickly.



Of course, gear connections can be established or broken at any time by dragging a gear to where you want it. You can also move groups of gears or even whole trains at once by moving the lowermost gear of a stack.

### program gears

foreverloops gears can be interfaced with all kinds of audio–visual materials. To select a sample (either audio or image), just click on it in the sample browser window. To see or hear a quick preview, you can double-click on it or hit the play button to the right to start playback immediately.



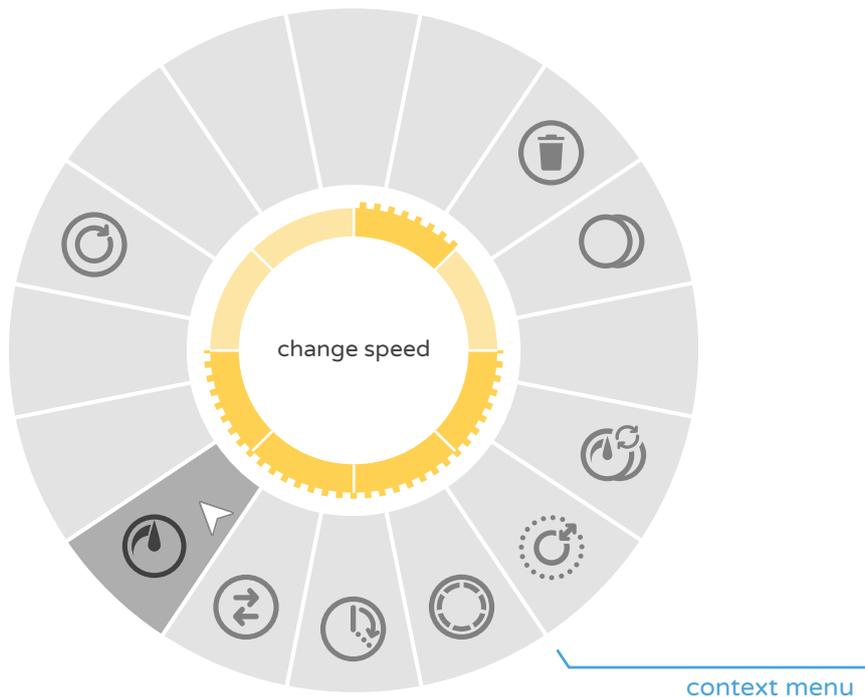
As soon as you select a sample, you can feed it into the gears (sequencer and scratch gears). To do so, simply click on a sequencer or scratch gear – this sets up a sample that will be triggered as soon as it passes the sampler (shown as line). Obviously, to get an output, the relevant gear must be active or, in other words, rotated by a connected or stacked drive gear.

## gear context

In foreverloops, virtual gears have different contexts that depend on the gear type and its current topology: You can alter drive gears in different ways than you do sequencer gears.

For instance, the speed of a drive gear can be altered as follows:

- hovering the gear and pressing the shortcut key for either doubling or halving the speed (default: '+' / '-')
- right-clicking the drive gear to open the context menu, and then
- clicking reverse speed, or change speed

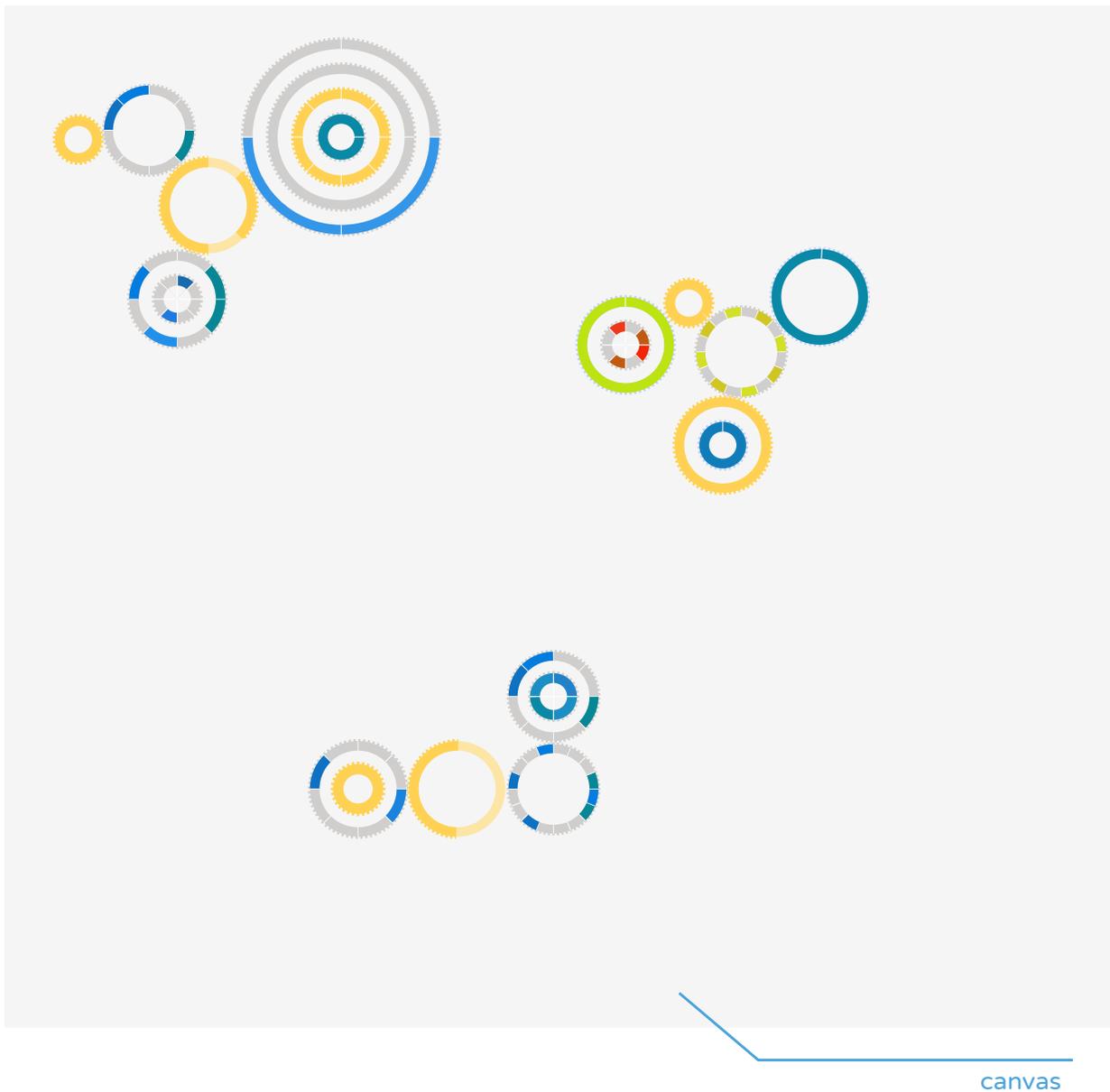


When you right-click a sequencer gear, a different context menu will open. With it, you can setup sample volumes, among other things. However, it is not possible to alter its speed, as this is entirely dependent on the connected drives.

This section covers the central user interface elements of foreverloops: canvas, menu bar, sample (and directory) browser, video panel, virtual MIDI keyboard, as well as the different gear context menus.

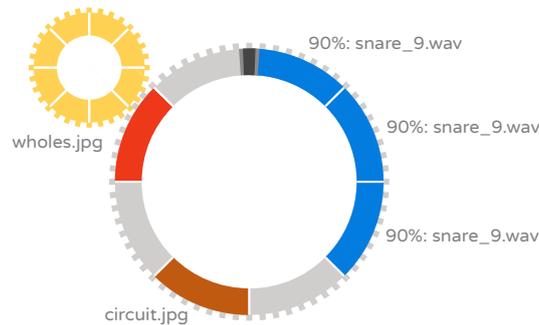
### canvas

The canvas is the area where gears can be created, moved around, and setup with samples. It offers plenty of space for multiple gear trains and individual gears. It also provides an overview of ongoing occurrences.



With the canvas, you can enlarge and focus on particular sections by zooming (using the mouse wheel) and panning (dragging).

In addition, pushing the right mouse on an empty canvas area labels the media samples and the used MIDI notes temporarily.



## menu bar

The menu bar allows you to create gears, receive feedback, acquire information, engage or disable particular modes, and enable or hide UI elements.

The menu is organized into several sections:



## gear creation section



This section contains icons for the different gear types: drive, sequencer, scratch and MIDI gear. By dragging an icon onto the canvas, you can create and connect or even stack gears. Clicking on the icon and then clicking on the canvas has the same effect.

## information

The information section displays context information and gives general feedback.

## undo / redo



Undo will revert to the last action, whereas redo will repeat them. Also, the information section will display more details concerning the relevant operation when you hover over the undo or redo button.

## modes

The mode section enables you to alter global behaviors:



Pause / play stops or re-activates all the drives on the canvas.



Alter the gear synchronization behavior. If enabled (default), gears you connect or stack with other gears will be aligned towards the target. This function is very useful for achieving appealing compositions quickly; however, sometimes you will prefer to manually synchronize.



Visual representation of the gear rotation. You can select either the gear-rotation mode (default); or the rotate-sampler mode, which is more common in radial sequencers.



Toggle between the gear move and scratch modes. With the move mode, you can move, connect, and stack gears. The scratch mode on the other hand lets you drive and activate gears as well as gear trains by manually rotating (scratching) relevant gears.

## capture



The capture function lets you record your composition / performance. You can either record just the audio output, which generates an audio file in the recordings folder, or a combined audio/video output. For the latter, a video file is created. In both cases, pushing either button will start the capture, which will finish when you press the button again or hit the escape key.

## UI



The UI (user interface) functions allow you to show / hide virtual keyboard, the video panel, and the sample browser.

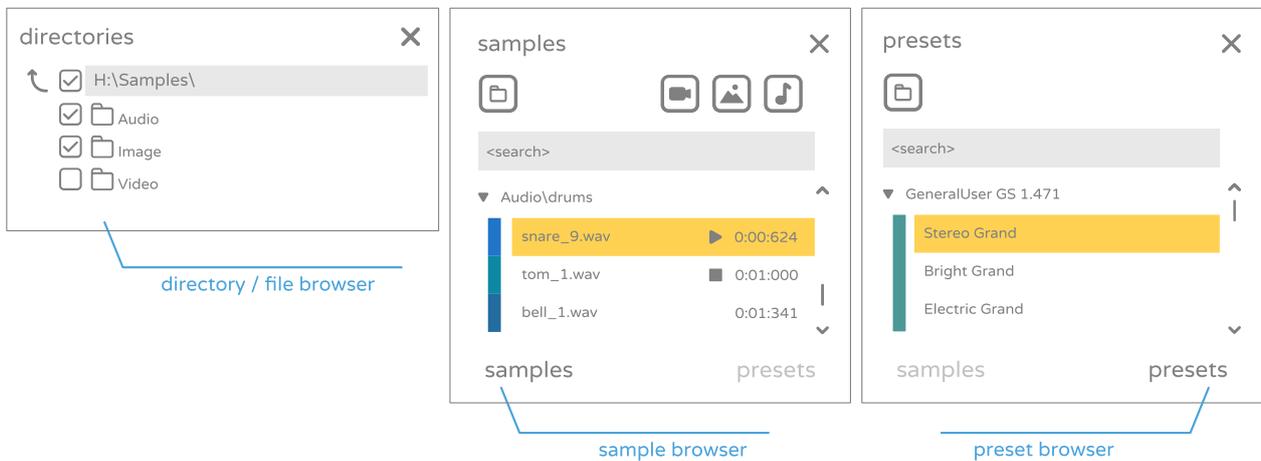
## exit to menu



Lastly, pushing the menu button will bring you back to the menu scene. From there, you can save and load scenes, alter the settings, or quit foreverloops.

## sample browser

With the sample browser, you can flip through media material, initiate sample playback, and import your own loops, pics, and flicks. Most importantly, the sample browser allows you to select the sample to be used for gear programming. Also, by changing into preset mode, you can browse and select the available instruments of your MIDI SoundFonts.



## directory browser

The directory browser, once folded out, lets you add additional media directories. For instance, you can add “My pictures” this way and use all the image files from there in foreverloops. Likewise, you can also remove sample directories if the sample browser becomes packed too full. By holding down the shift-key while adding or removing a directory, sub-directories, if any, are added or removed as well.

In preset mode, you can add individual SoundFont packs - files that contain multiple instrument presets to be used in conjunction with MIDI gears. Currently, sf2 and sfz files can be loaded.

## sample browser

The sample browser lets you check over your media material, preview it (double clicking a sample makes it play once), and select it for later use in gears. It upholds a folder structure that is analogue to an underlying file system, and displays the sample type (either audio, image, or video), the name of the sample along its file extension, and its duration. On the very bottom, you will find a button for folding out or hiding the directory browser. On the top, you can enable or disable filters, and search for specific samples by name.

## preset browser

The preset browser, lets you select instruments for MIDI gears and instruments. To preview the preset, a MIDI note has to be triggered (either by using the virtual keyboard, hitting a key / pad of an external MIDI instrument, or by playing an armed MIDI gear).

## video panel

The video panel represents the primary output channel for visual sample material, e.g., pictures and videos. So, when you double-click on an image sample, it is displayed in the video panel.

The video panel can be hidden, resized, and put into the foreground or background (using the mouse wheel).



gears over fullscreen video

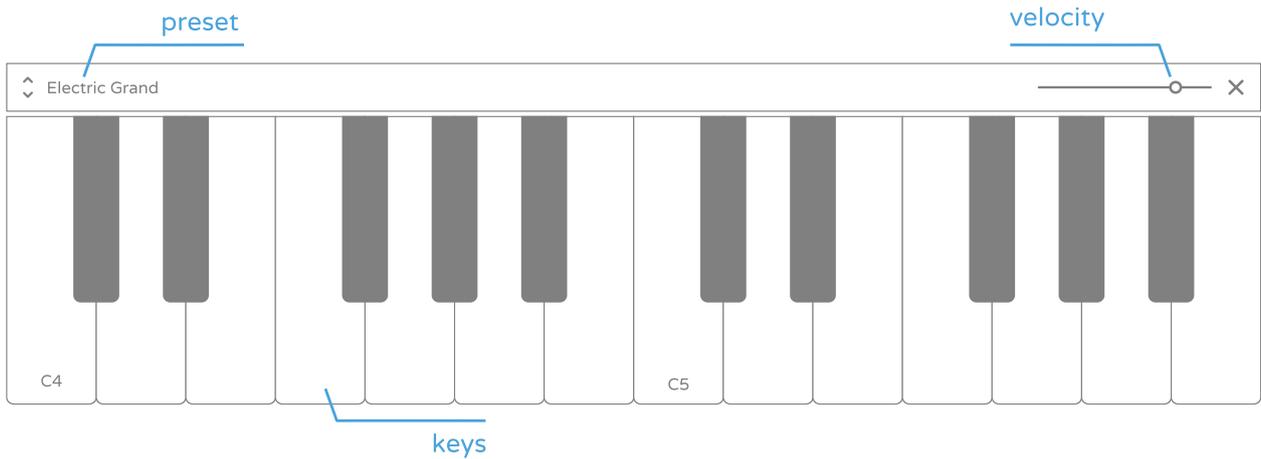


gears behind video panel

## virtual MIDI keyboard

The on-screen keyboard can be used to play MIDI notes in real time. In conjunction with MIDI gears, it can be used to record note sequences. You can also set the instrument (preset) and the default velocity (i.e. the level of the played notes). The number of shown keys can be changed by resizing the panel top left or top right or by scrolling the mouse wheel.

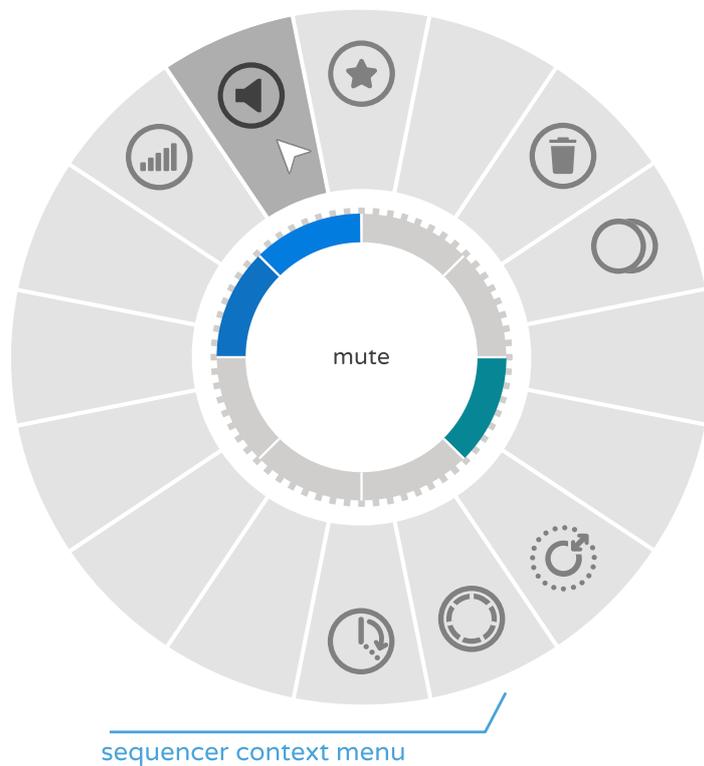
To choose which key / octaves are shown, resize the panel on the right or left hand side or drag along the keys with the right mouse button pressed.



### context menu

With the context menu, you can alter various aspects of individual gears. For example, you can alter sample volumes, de-activate drives, alter the number of segments, or delete gears.

To open this menu, right-click a gear on the canvas. The context menu is then displayed above the gear.



The available functions of the different context menus are explained in more detail in the gear type section.

The underlying physics simulation inspires a playful and adventuresome attitude. Playful activities and free-form experimentation stimulate engaging compositions, arrangements, and performances. Also, by simply playing with gears you learn how to use them.

This section explains relevant mechanical principles in detail.

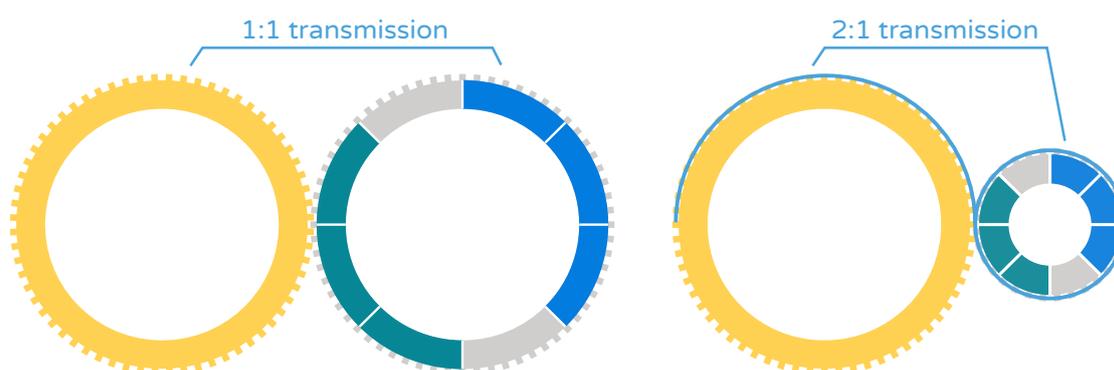
## gear mechanics

Gears, gear meshes, and gear trains provide unprecedented possibilities for music-making and performances. foreverloops lets you create virtual gear trains - mechanical systems comprising connected gears whose teeth are engaged.

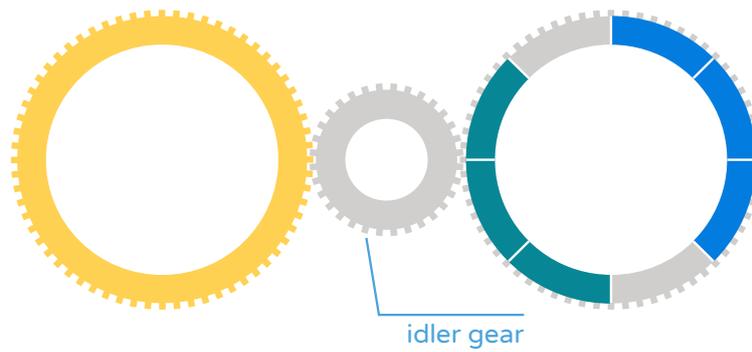
In the following, we introduce the uses and effects of translation / transmission, driven gear trains, and Leibniz gears. Note that foreverloops ignores gear slippage for simplicity and usability. Also, the number of cogs is assumed to be proportional to the circumference of the gears.

## translation / stacking / transmission

In foreverloops, the angular speed ratio of two gears is inversely proportional to the ratio of the circumference. The speed ratios are determined by the circumference of the interconnected gears, as shown below:

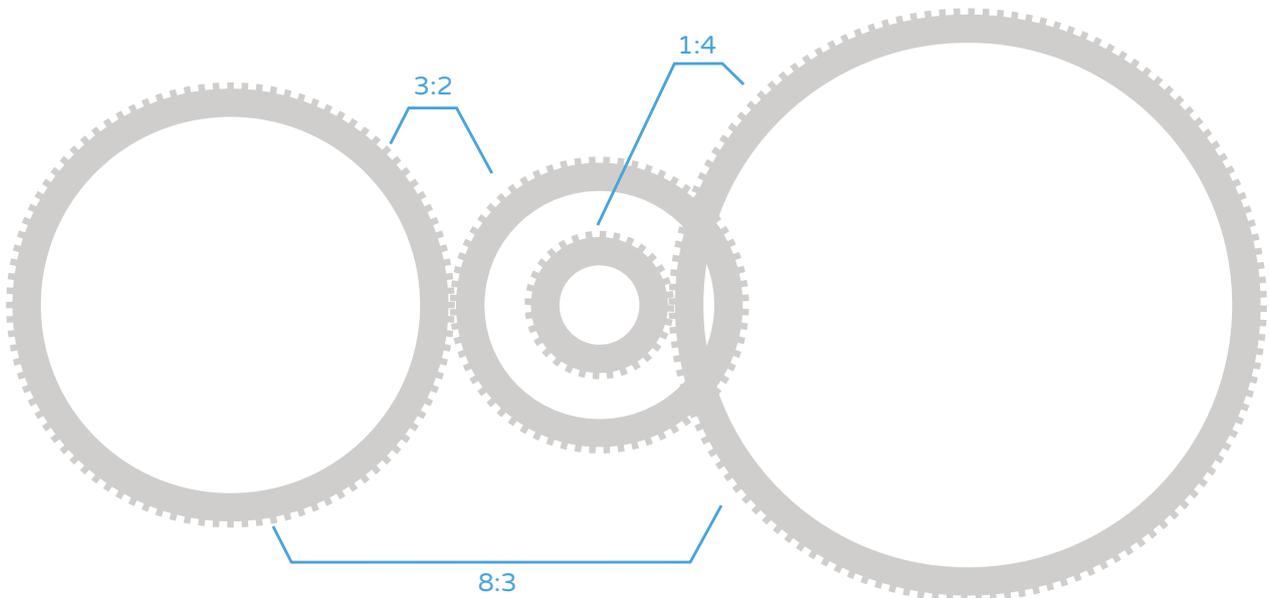


Following the laws of mechanics, the direction of rotation is reversed for interleaved gears. So-called in-betweeners, or idler-gears, can be inserted to cancel this direction change, as shown below:

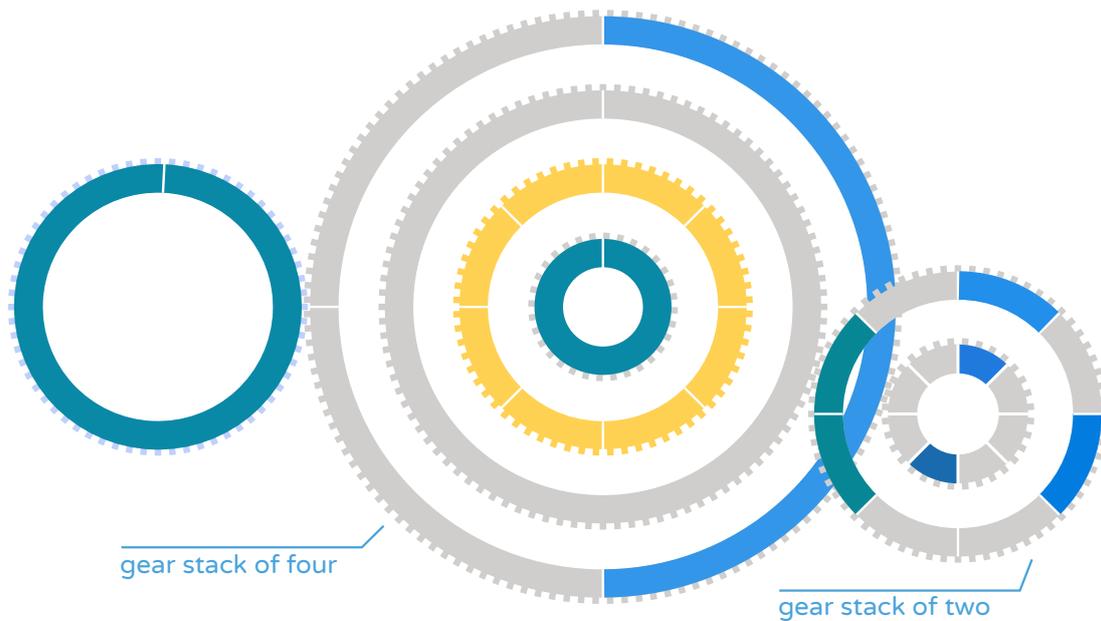


Generally, if an output gear has more cogs than the input gear, the gear train is *amplified* (input torque), resulting in higher beats per minute (BPM). Likewise, if the output gear has fewer cogs than the input gear, the gear train is *reduced* (input torque), thereby lowering the BPM.

foreverloops supports several different gear sizes to cover a wide range of well-known rhythm patterns.



Additional transmission ratios can be achieved by combining transmission with stacking, i.e. the axial connection of gears.

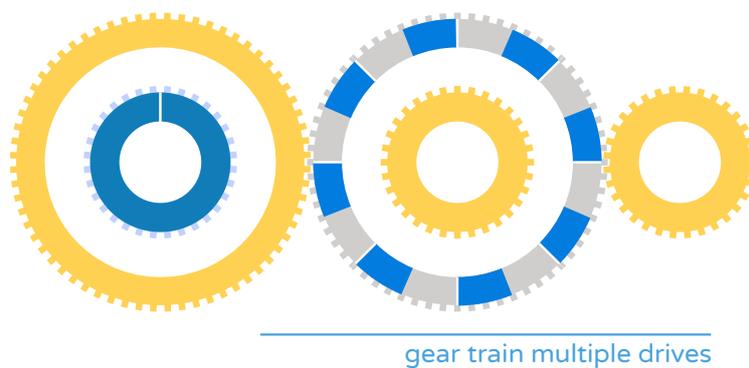


This option can be useful for merging different gear types or configurations into a stack. A stack is created when several gears share an axial connection. To initiate axial connections, smaller gears must be put onto larger gears.

### driving gear trains

Without explicit user input, only drive and scratch gears can activate gear trains, whereas only sequencer and scratch gears can produce visual and / or audio output.

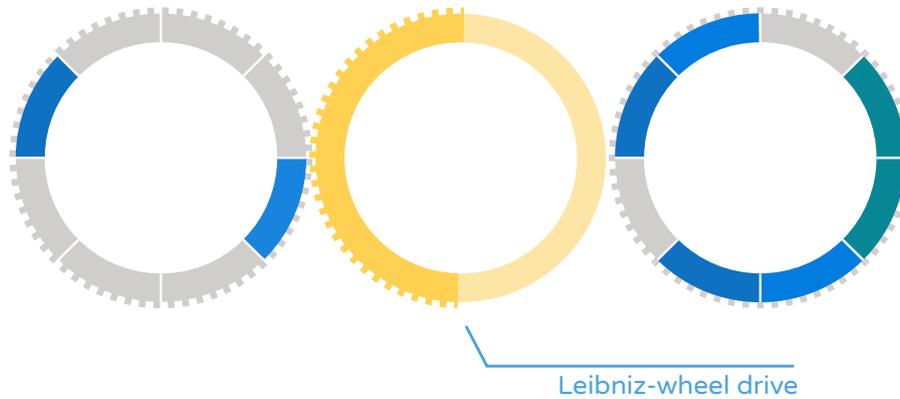
It is possible to activate gear trains with multiple drives; drives can be added and removed from a train at any time. Adding to or reducing the overall speed of a train in this way can also be a performative element.



The total speed applied to a gear train equals the sum of all the applied rotational speeds: trailed and stacked gears are considered recursively.

## logics

Drive gears can be Leibniz-wheels, or in plain English, gears with truncated cogs:



A drive gear with truncated cogs does not contribute to the overall gear train speed for a certain time period.

This mechanism facilitates basic programmable logic.

## media types

With audio-visual samples, foreverloops evolves from a basic real-time gear mechanics simulator to a multi-media composition and performance software toy.

The supported media types include sound and music files, images, and videos. Audio material is decoded in real time and played back concurrently, whereas videos and images use interruption semantics and they are played back exclusively.

## trigger / scratch

For audio and video samples, we differentiate between sequencing and scratching.

In sequencer gears, samples, once triggered, are played in a forward direction at normal speed. This is useful when compositions are formed by many short samples, such as rhythm patterns, stop motion clips, or melodies.

In contrast, scratching matches the sample playback direction and speed with the current angular speed of the scratch gear. It is primarily intended for use with longer, more ambient samples, when tempo changes are desired in either the audio or video, or when the source material needs to be a specific length.

## image

Image samples are decoded on load time, and then played back exclusively. They are active as long as their segment is active. In other words, for images, the playback durations depend on both the segment widths and the playback speeds.

## video

Video samples are decoded and played back in real-time. Only one video or image sample can be active at any time, and the most recently triggered video or image sample is given preference.

Video samples can be scratched and triggered, and the audio track of the video, if available, is played back accordingly.

## MIDI notes

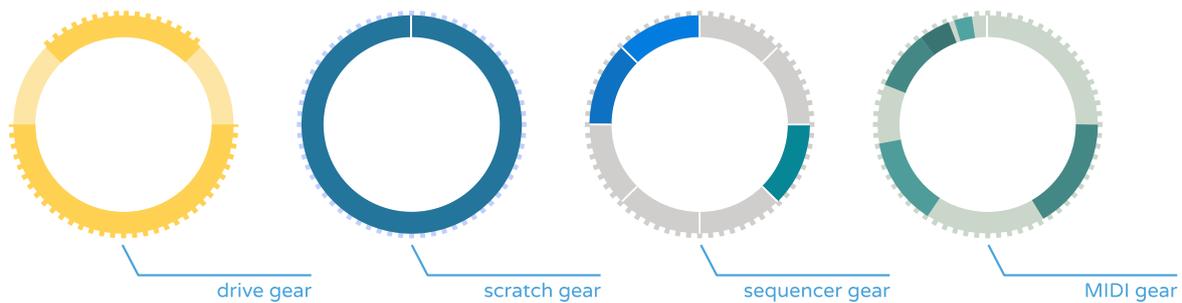
MIDI notes can be used in conjunction with MIDI gears – they can be set manually (editing) or recorded by using the virtual keyboard or external MIDI input devices. MIDI notes have multiple properties:

- the note value (key)
- a velocity (level)
- a SoundFont / instrument preset
- starting point / duration
- optionally, a quantization value

## gear types

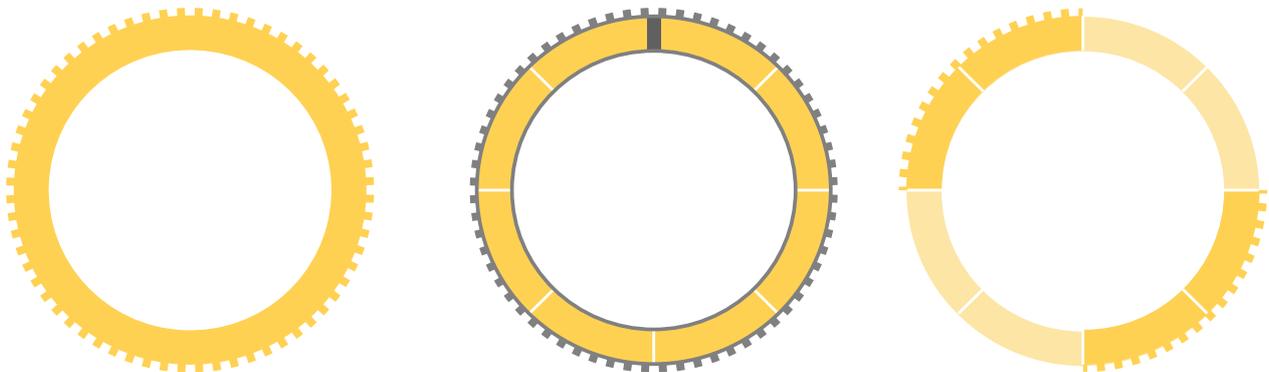
There are two core gear types: the drive gear and the media gear. Drive gears are used to manipulate rotation speeds, whereas media gears embed and play audio, image, and video samples.

The scratch gear and the MIDI gear count towards both categories: they can actively drive gear trains, but can also initiate media playback on their own.



In a nutshell, drive gears activate gear trains, sequencer gears implement sample triggering, scratch gears combine drives with single-sample scratch playback, and MIDI gears can play musical instrument notes and also drive.

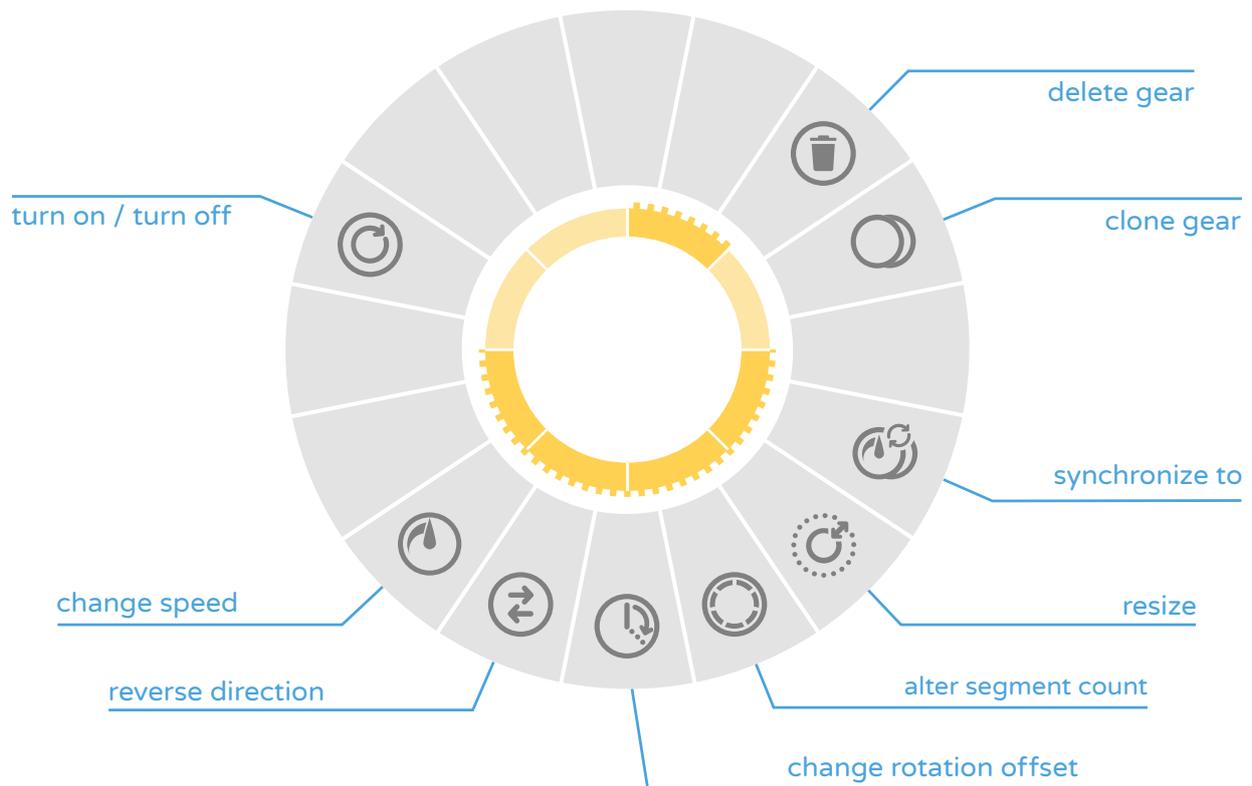
## drive gear



### A drive gear

- cannot contain media samples
- can activate gear trains
- is programmable (to become a Leibniz wheel)
- can have different sizes, which affects the speed ratios of attached gears
- can be combined with other drives to achieve additional speed / translation ratios
- when combined with other drives, causes the speeds to add up

With the drive context menu, you can:



- delete the drive gear
- clone the drive gear
- synchronize it to another drive:

When you synchronize a drive to another drive, MIDI or scratch gear, the synchronized drive will have the same speed as the other individual drive, as well as the same current rotation. Also, any connected or stacked gears will be affected by a rotational synchronization. With this feature, you can straighten up multiple asynchronous trains. Drives can also take over the speed of the scratch or MIDI gears, which enables the combined use of scratch or MIDI gears and Leibniz wheels.

- resize the drive gear:

When you resize a drive gear, its angular velocity is not changed - it still needs the same time to complete one rotation. However, connected gears are affected by a different transmission ratio, and therefore rotate either faster or slower.

- edit segment count:

Altering the number of segments of a drive gear can be useful for several reasons. It affects the Leibniz semantics, i.e. you can create a counter from 1 to 10 by specifying (and later programming) 10 segments. Also, when you specify just one segment, you can then activate or deactivate gear trains by clicking once on the drive.

- change offset:

Altering the gear offset only affects Leibniz wheels. With it, you can create a temporal skew.

- change speed

Change speed allows you to set the drive speed directly, using a circular BPM-oriented interface.

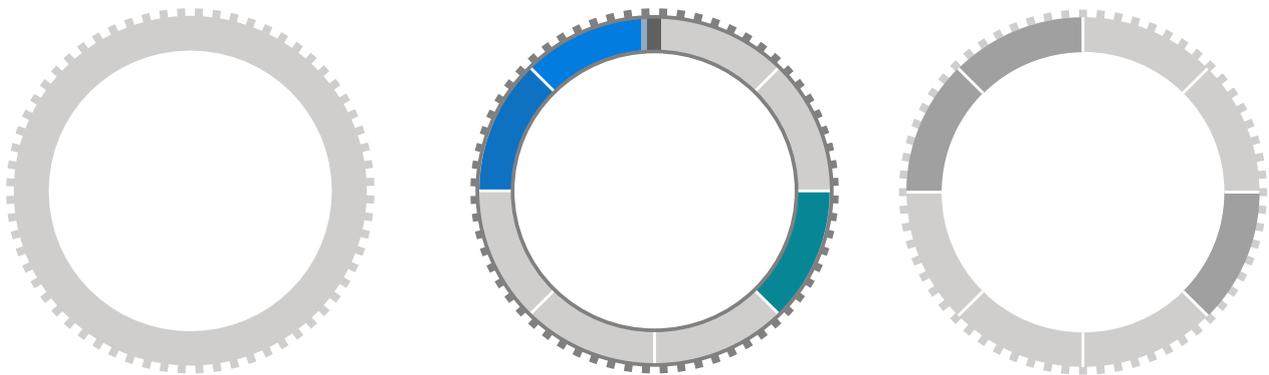
- reverse direction

With reverse direction, you can alter the driving direction of the gear, and let the entire train play backwards. In conjunction with Leibniz drives, this feature enabled interesting effects.

- turn off / turn on:

When a drive gear is turned off, it no longer activates gear trains or contributes to the overall speed of its train. However, it still may rotate when it is driven by other gears. With a turned-off drive, you can add Leibniz semantics to a scratch gear by simply stacking gears onto it.

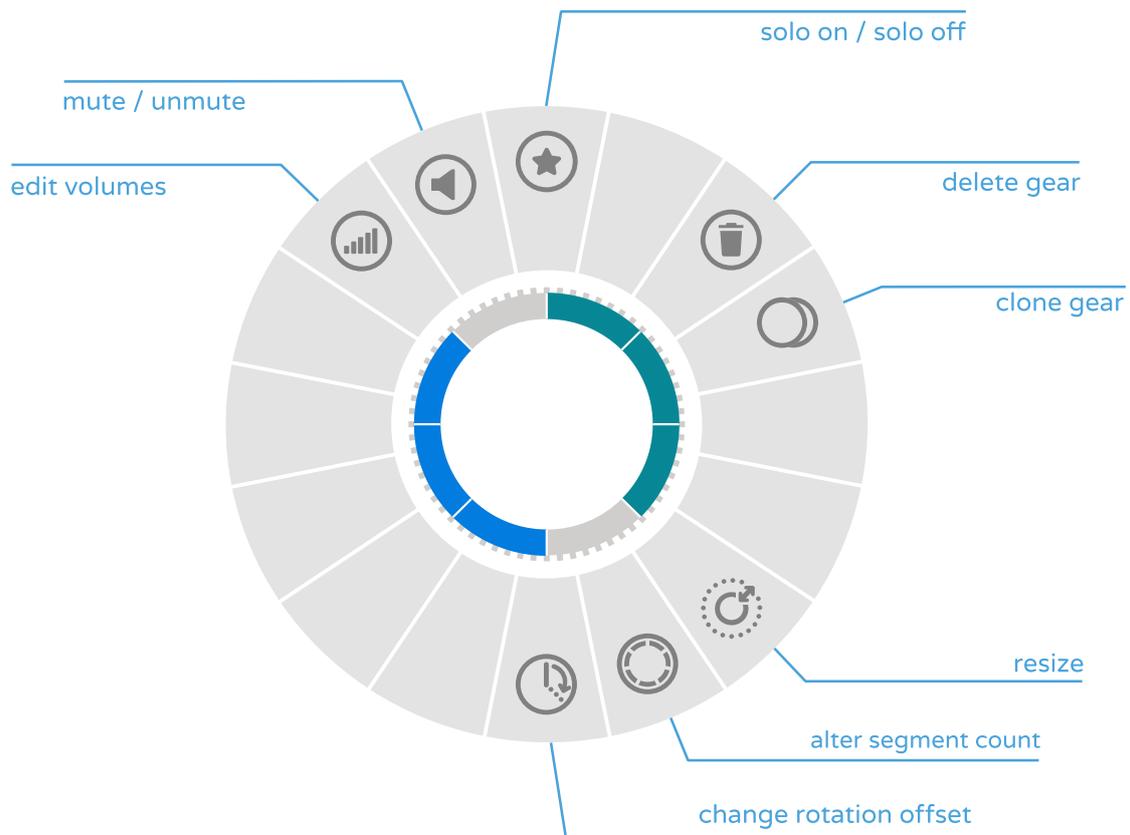
## sequencer gear



### A sequencer gear

- cannot activate gear trains by itself
- can be programmed with media samples
  - that, once triggered, are played back in normal speed
- can contain mixed and arbitrary media samples and types
- can be stacked with other sequencer gears to achieve interesting bar measures

With the sequencer context menu, you can:



- delete the sequencer gear
- copy the sequencer gear
- resize the sequencer gear
- edit the segments of the sequencer gear:

When you alter the number of sequencer gear segments, foreverloops will try to preserve any set samples and apply them to the new segment count. It is possible to specify an arbitrary segment count, up to 64. In combination with stacking, different segment counts allow for particularly interesting bar measures and rhythm patterns.

- change offset:

Altering the rotational offset allows you to introduce temporal skews. For example, you can shift the sequencer gear by half a stroke to easily vary your rhythm. Alternatively, you can copy a sequencer gear, change its rotational offset, and play it alongside the original for an interesting effect.

- change volume:

This function lets you specify the volumes of the gear media samples. Note that samples volumes are also applied when new samples are programmed. Also, foreverloops will try to manually preserve set sample volumes when the segment count is altered.

- mute:

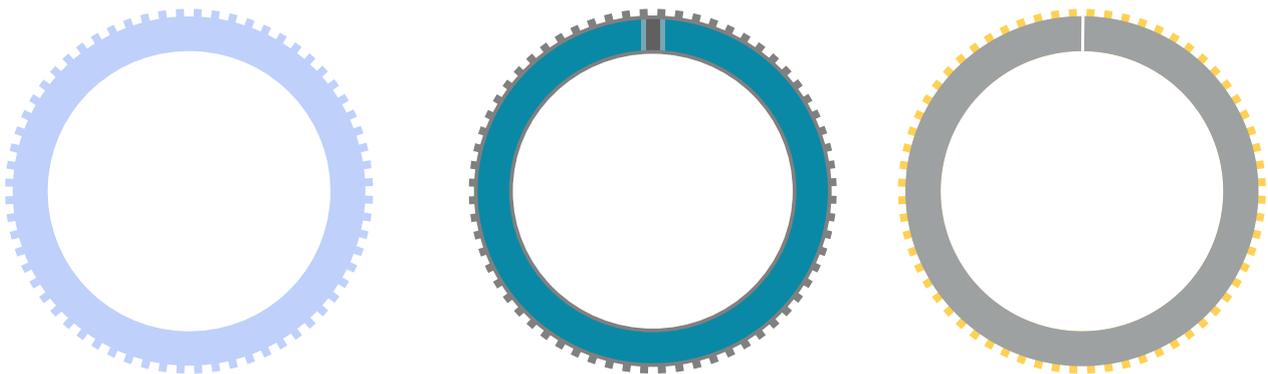
This function mutes / unmutes the entire gear, and thus enables or disables sample triggering. Note that image and image samples are not displayed when their gears are muted.

- set solo mode:

The solo mode plays back gears exclusively. Multiple gears can have the solo mode enabled and foreverloops provides shortcuts to quickly enable or disable the solo mode for one gear (default: L), a gear stack (default: CTRL + L) or all connected gears (default: SHIFT + L).

Gears in solo mode are visualized by showing an inner border in grey.

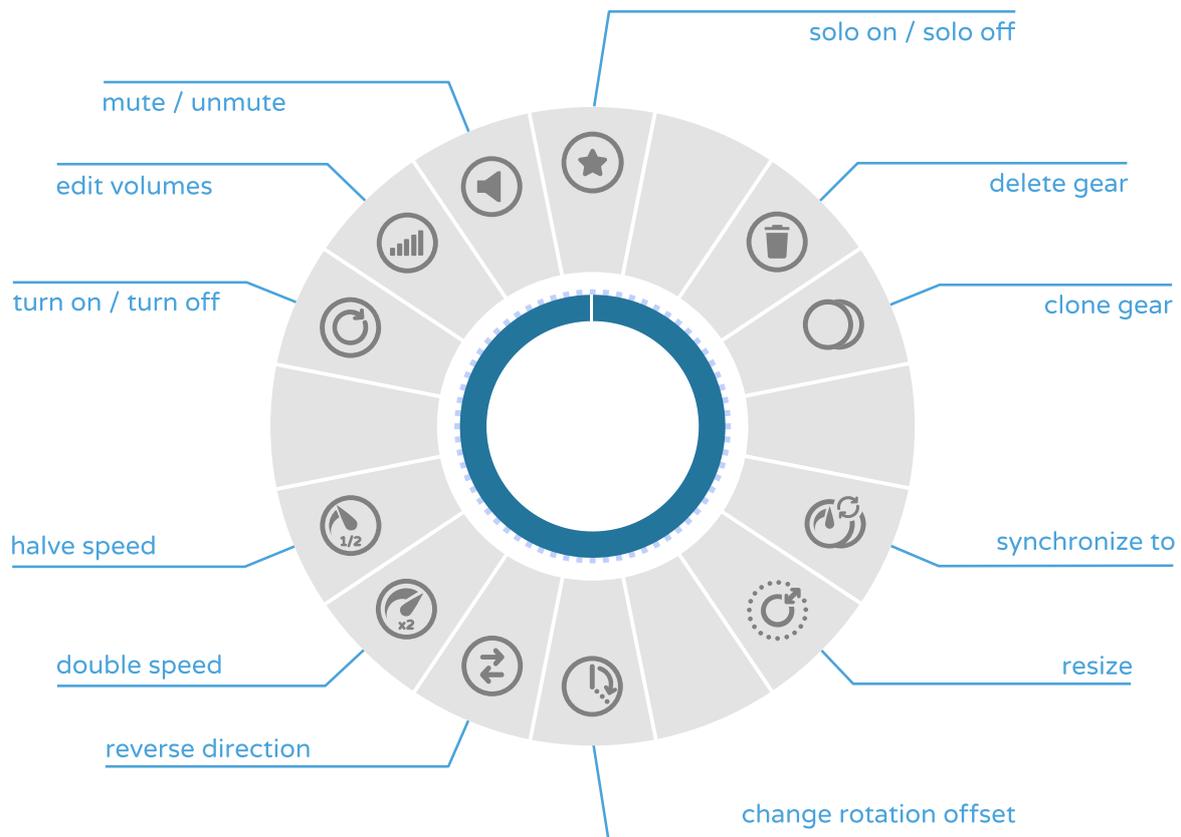
### scratch gear



### A scratch gear

- can activate gear trains AND
- initiate sample playback
- implements scratch semantics, i.e., the actual speed affects the media playback speed
- does not start by itself, i.e., it must be turned on once a sample is set to actively drive
- has a default speed that corresponds to the sample duration, i.e., will play its sample with normal speed by default
- can contain ONE sample of an arbitrary type
- cannot become a Leibniz wheel by itself
  - but can easily be combined with a Leibniz wheel (stacking / attaching)

With the scratch gear context menu, you can:



- delete the scratch gear
- copy the scratch gear
- synchronize the scratch gear to another drive:

When you synchronize to another drive, MIDI or scratch gear, the scratch gear will have the same speed as the other drive, and the same current rotation. Also, any connected or stacked gears will be affected by a rotational synchronization.

- resize the scratch gear:

Resizing a scratch gear has the same implications as resizing a drive gear: When you resize it, its speed is not changed, it still requires the same time to complete one rotation. However, connected gears are affected by the different translation ratio, and therefore rotate faster or slower.

- change offset:

Altering the rotational offset allows you to introduce temporal skews.

- reverse gear rotation:

When you reverse the rotation of a scratch gear, it drives attached gears into the opposite direction. Its sample is played backwards as well.

- double speed

When you double the rotation speed of a scratch gear, it also drives other gears with double speed. Its sample is played twice as fast.

- halve speed

When you halve the rotation speed of a scratch gear, it also drives other gears with half speed. Its sample is played half as fast.

- turn off / turn on:

When a scratch gear is turned off, it no longer activates gear trains or contributes to the overall speed of its train. However, it still may rotate when driven by other gears. In this case, its set sample is played back accordingly.

- change volume:

This function lets you specify the sample volume.

- mute:

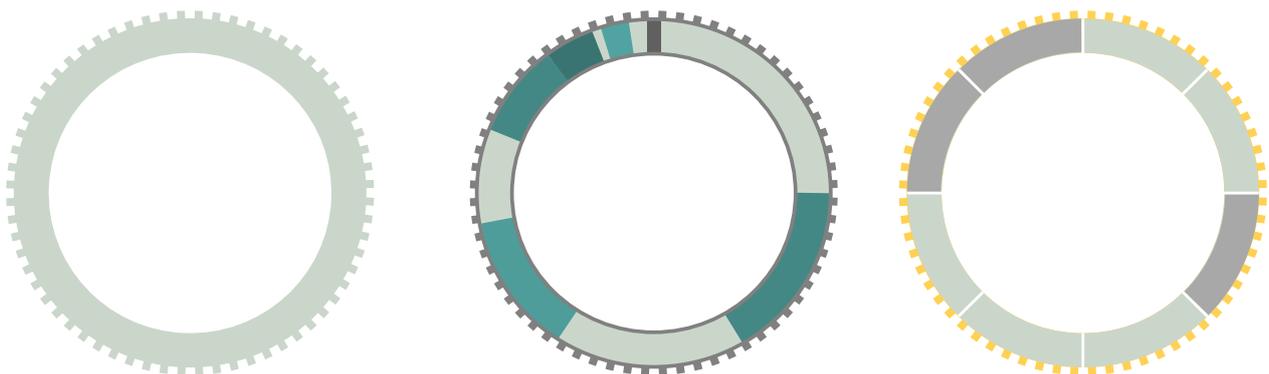
This function mutes / unmutes the entire gear, and in thus enables or disables the sample playback. Note that image samples are not shown when the gear is muted.

- set solo mode:

The solo mode plays back gears exclusively. Multiple gears can have the solo mode enabled and foreverloops provides shortcuts to quickly enable or disable the solo mode for one gear (default: L), a gear stack (default: CTRL + L) or all connected gears (default: SHIFT + L).

Gears in solo mode are visualized by showing an inner border in grey.

## MIDI gear

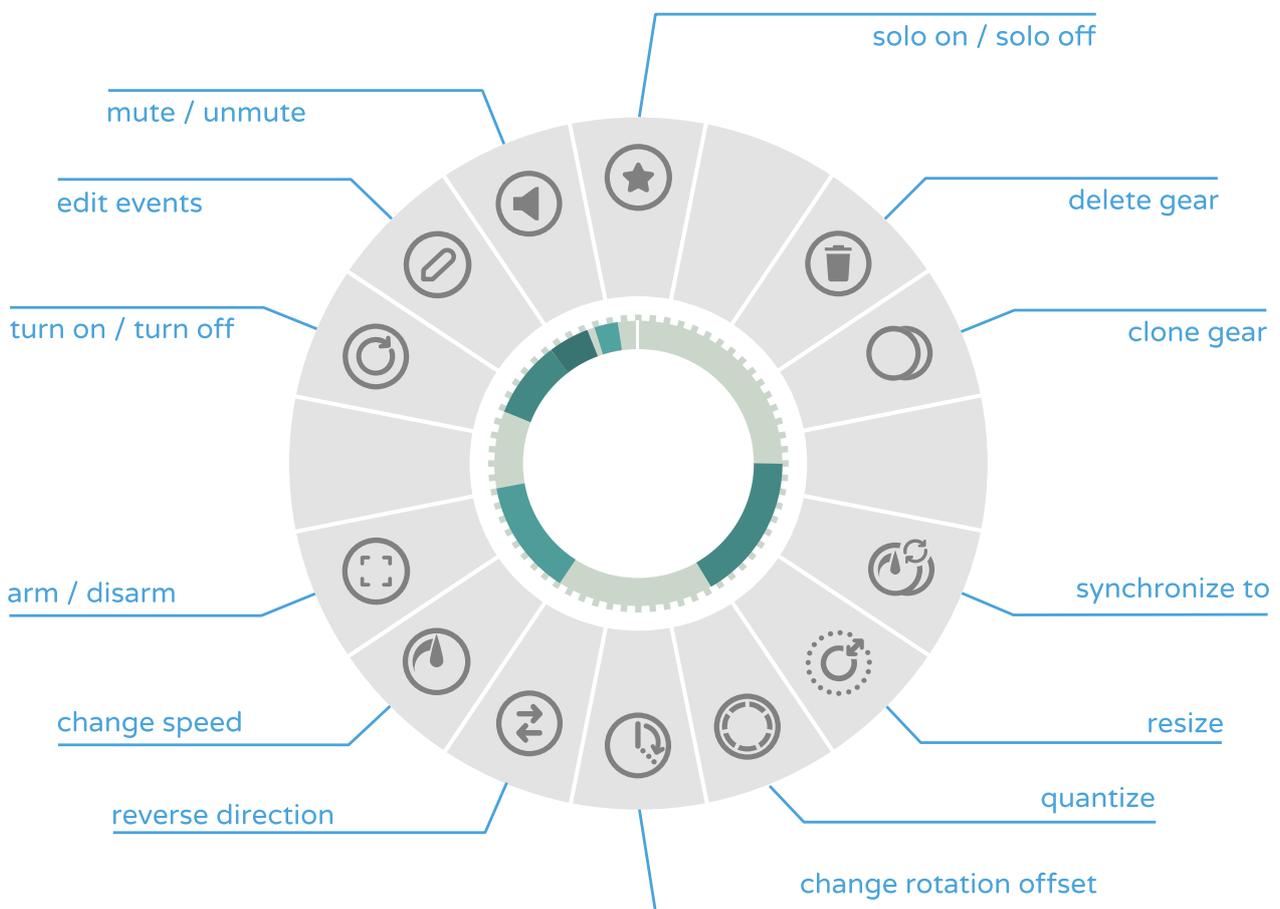


### A MIDI gear

- can activate gear trains AND
- initiate MIDI note playback
- does not start by itself, it must be turned on in order to activate its train and / or to listen to the MIDI note sequence
- has a preset that corresponds to a MIDI instrument

- can be programmed in multiple ways
  - by editing the notes in a piano-roll like view
  - by directly recording MIDI notes (from an external MIDI input device or the virtual midi keyboard). In order to receive MIDI notes, the gear must be armed.
- cannot become a Leibniz wheel by itself
  - but can easily be combined with a Leibniz wheel (stacking / attaching), also for recording purposes
- has a quantization property that can be set via context menu. Quantization affects the starting points and durations of all gear notes. A quantization value of zero means that no quantization occurs.

With the MIDI gear context menu, you can:



- delete the MIDI gear
- copy the MIDI gear
- synchronize the MIDI gear to another drive:

When you synchronize to another drive, MIDI or scratch gear, the MIDI gear will have the same speed as the other drive, and the same current rotation. Also, any connected or stacked gears will be affected by a rotational synchronization.

- resize the MIDI gear

Resizing a MIDI gear has the same implications as resizing a drive gear: When you resize it, its speed is not changed, it still requires the same time to complete one rotation. However, connected gears are affected by the different transmission ratio, and therefore rotate faster or slower.

- setup note quantization

When using the quantization mechanism, MIDI notes are aligned towards virtual segment boundaries, concerning both starting points and durations. A value of zero means that no quantization is conducted, in this case the initial record timings are used. By quantizing MIDI notes, you can easily achieve specific bar measures, and use them in conjunction with other bar measures. Note that quantization can also be used for recording – by setting a quantization value prior to the recording of notes, you can pretty much ensure that your bars are complied with.

- alter offset

Altering the rotational offset allows you to introduce temporal skews. For example, you can shift the MIDI gear by half a stroke to easily vary your rhythm or melodies. Alternatively, you can copy a MIDI gear, change its rotational offset, and play it alongside the original for interesting effects.

- change speed

Change speed allows you to set the drive speed directly, using a circular BPM-oriented interface.

- arm / disarm

Only armed MIDI gears can be used as recording targets. As soon as the MIDI gear is armed, it will listen to and receive MIDI input notes, either from the virtual MIDI keyboard, or from external devices. For the time being, only MIDI notes are accepted. Note that only one MIDI gear can be armed at the same time.

- turn off / turn on:

When a MIDI gear is turned off, it no longer activates gear trains or contributes to the overall speed of its train. However, it still may rotate when driven by other gears. This also has an effect on the recording behavior – recording into a turned off (and empty) MIDI gear enables an auto-scale behavior, i.e. the speed and note lengths are automatically adjusted to the recording duration. Recording into a turned on or otherwise active MIDI gear avoids auto-scaling. This is useful for recording multiple tracks into a single gear.

- edit MIDI notes

Shows a circular-style piano-roll interface that lets you alter, create, and delete MIDI notes. More details are covered in the subsequent section Editing MIDI notes.

- mute / unmute

This function mutes / unmutes the entire gear, and thus enables or disables MIDI playback.

- set solo mode:

The solo mode plays back gears exclusively. Multiple gears can have the solo mode enabled and foreverloops provides shortcuts to quickly enable or disable the solo mode for one gear (default: L), a gear stack (default: CTRL + L) or all connected gears (default: SHIFT + L).

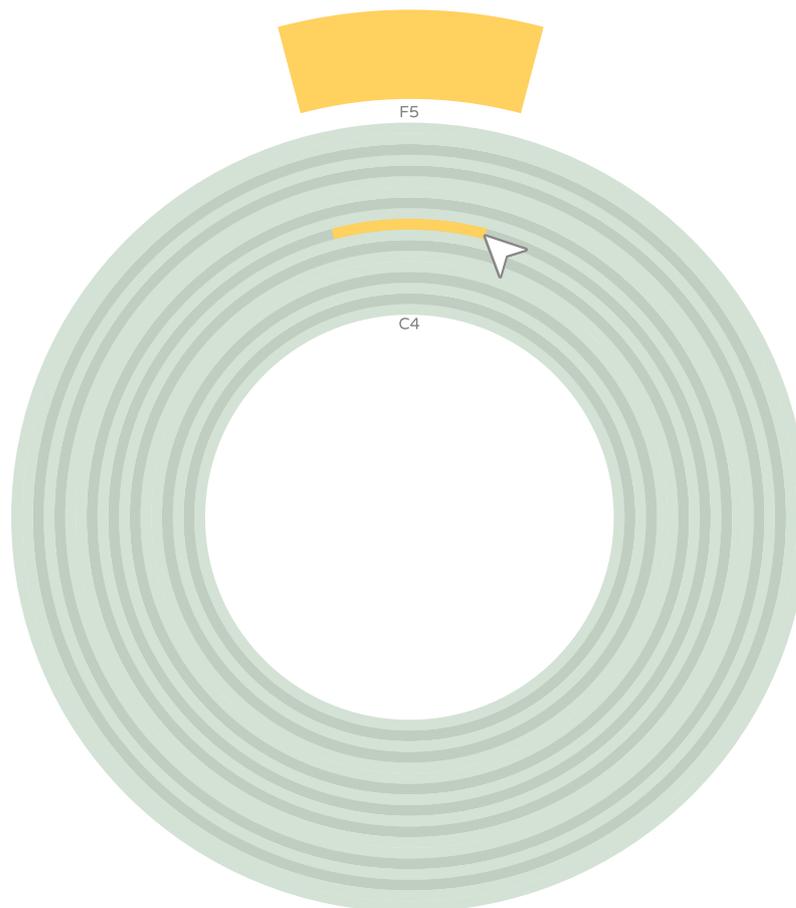
Gears in solo mode are visualized by showing an inner border in grey.

## Editing MIDI notes

The MIDI gear edit mode can be accessed via context menu. It resembles a piano-roll, but is arranged in a circular layout. With this view, you can set new MIDI notes, delete and move notes, and alter durations and velocities.

- create new MIDI notes:

To create a new MIDI note, click in an empty space of the note area. The bigger the distance from the center, the higher the note. If the gear quantization is enabled, it is also applied to the newly created note. The velocity value depends on the current value of the velocity slider on the virtual MIDI keyboard.



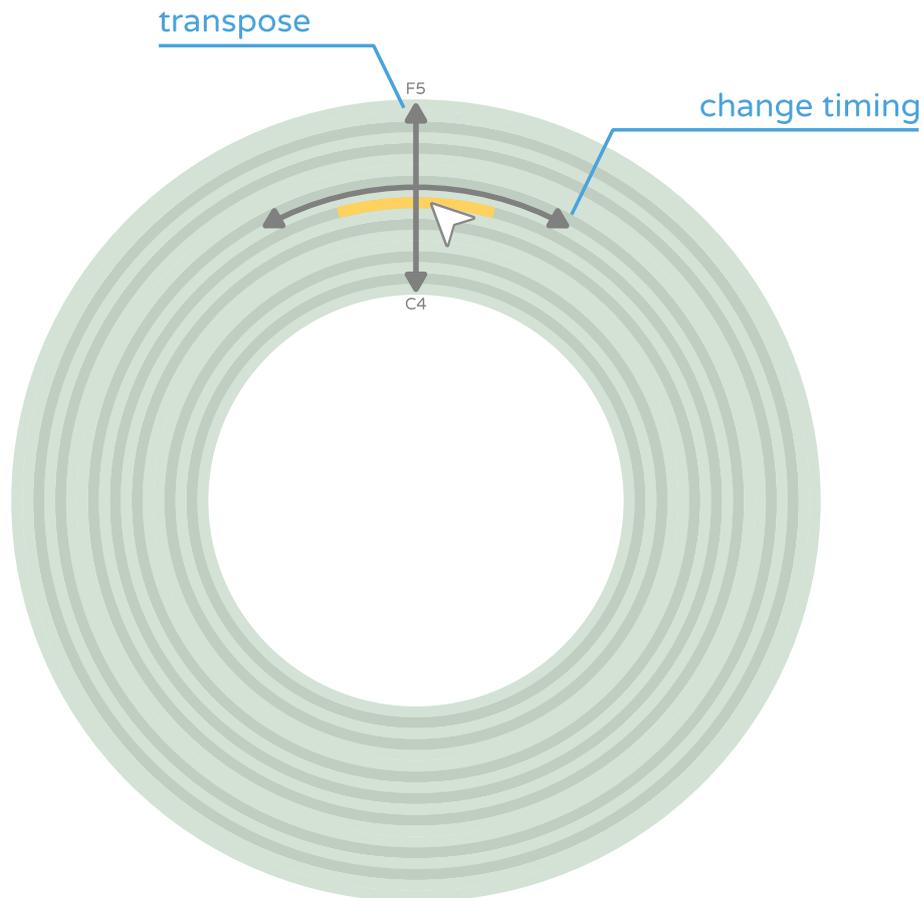
- delete MIDI notes:

Drag the selected note outside the piano roll, or hit the delete key to delete notes.

- transpose MIDI notes / change timing of MIDI notes:

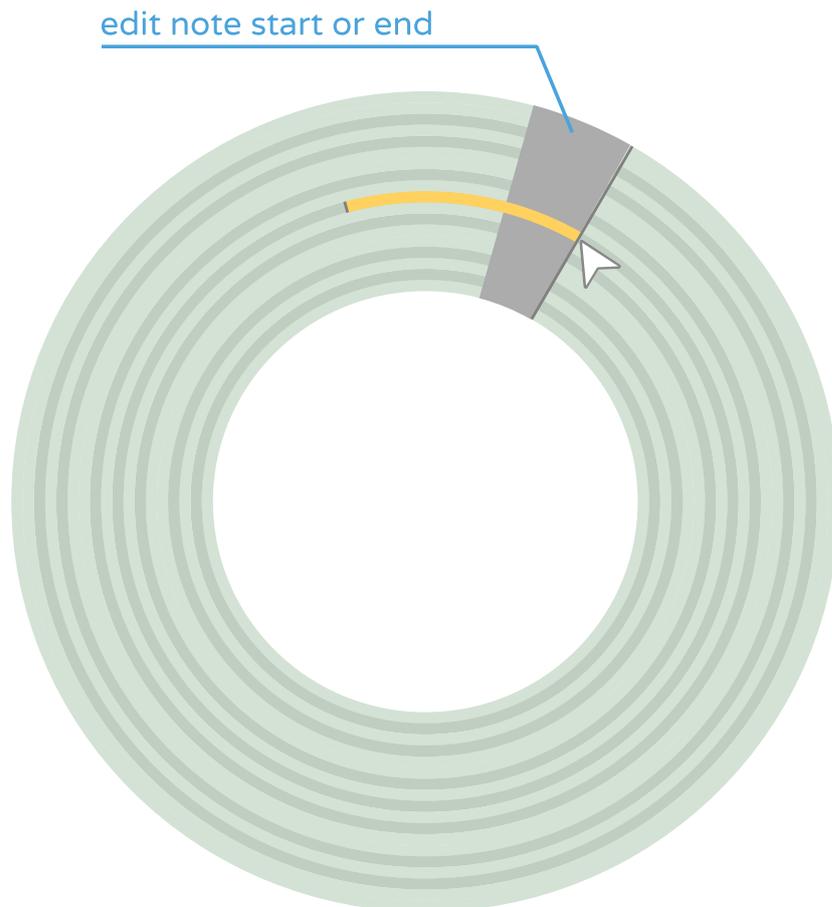
MIDI notes can be moved by dragging. Dragging affects all currently selected notes, so multiple notes can be re-positioned at the same time. By moving a note, you can alter its starting point (angle), as well as its note height (distance from center / transpose).

Transpose all notes by selecting all notes (default: CTRL+A) and dragging them along the radius. By locking the dragging direction (default: SHIFT+Dragging), you can either transpose or change the timings.



- alter MIDI note durations:

MIDI note durations can be changed in two ways: first, it is possible to relocate the starting point without changing the end point by using the handle that becomes apparent when hovering over the starting section of the note. Altering the starting point this way changes the note duration as well. Second, it is possible to alter the note's end point by dragging its handle to change the duration. Be aware that because notes are arranged in a circular manner, starting and end points may be interchanged, depending on playback direction.

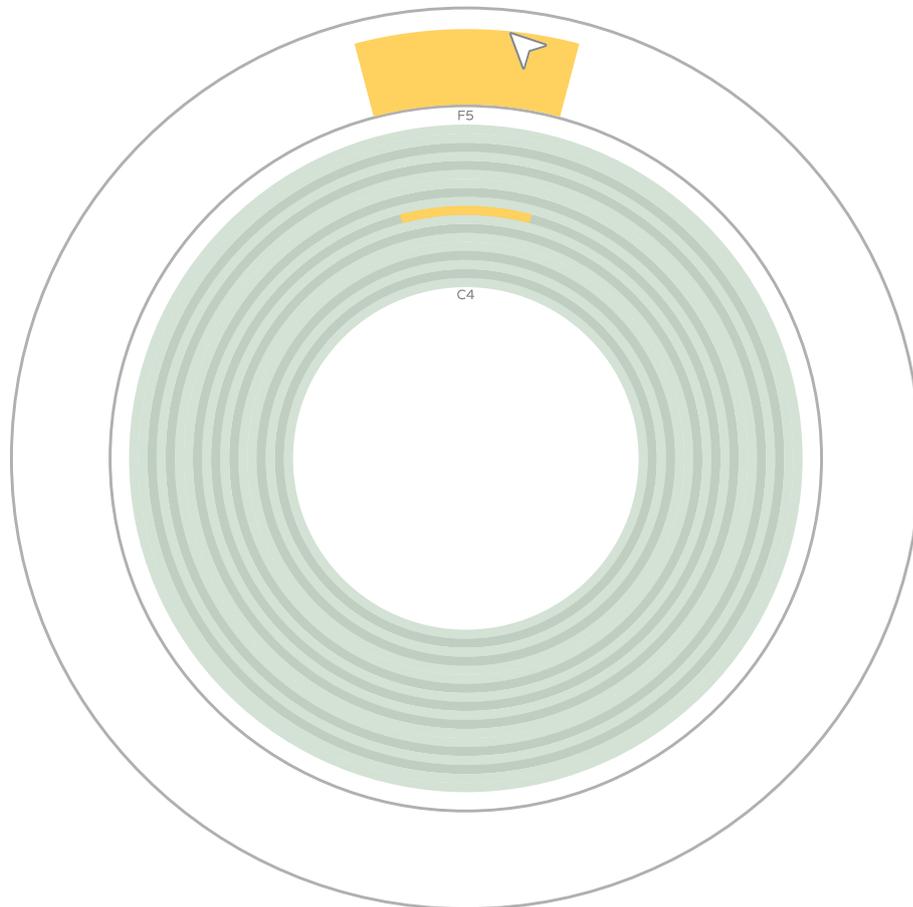


- alter MIDI note velocities:

MIDI note velocities roughly correspond to the levels notes are played back with. All MIDI notes have a velocity value that can be altered using the edit mode. Velocities are displayed in the outer ring area, they can be modified by clicking or dragging. The longer the distance from the center, the higher the velocity value. It is also possible to alter the velocity values of all gear notes at the same time, with CTRL pressed. Holding SHIFT while dragging changes all selected notes.

With ALT pressed while dragging, all hit velocities are changed, similar to the edit volume behavior on sequencer gears.

To change a velocity of a note overlapping with another, scroll the mouse wheel to pick the desired velocity.



## Recording MIDI notes

While recording notes into MIDI gears, implicit modes that depend on gear activity and current synchronization mode apply:

- loose recording
- loop recording

### Loose recording mode

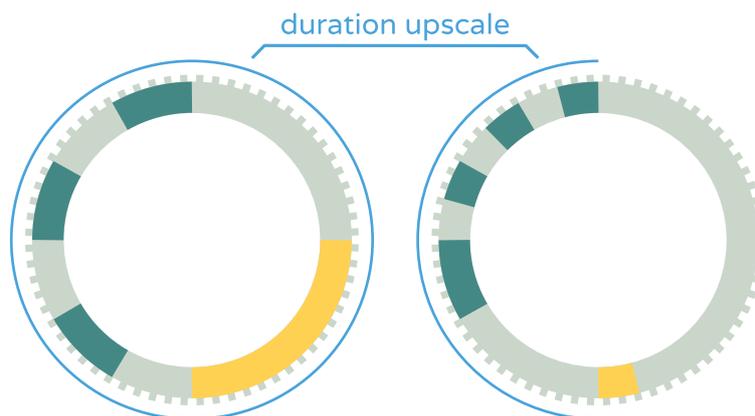
Free recording for longer or melodic tracks.

Create an unpowered MIDI gear, and play your melody: if the recording duration exceeds the gear duration (one turn if it was actively powered), the duration adapts to the recording (by upscaling).

- with sync enabled: the recording start position is synced to the virtual clock and the duration is prolonged by doubling the current duration to sync the recording end.
- with sync disabled: the recording start matches the 12 o'clock position and the duration is set to the current recording length.

Recording starts as soon as the first note is played back with newly created MIDI gears (which are automatically armed). Disarm and re-arm the gear manually to start recording immediately (and create a pause before the first note plays).

Recording into quantized MIDI gears aligns note start and end towards the segmentation.



### Loop recording mode

For rhythmical compositions or recording polyphonic tracks.

Recording into a powered (either by its own power or powered by any drive connected) MIDI gear, adds notes to the previous ones after exceeding one gear turn. Very long tracks are still possible, either by reducing the speed or by respective transmission.

- with sync enabled: the recording start position is synced to the virtual clock.
- with sync disabled: the recording start position remains

Recording into quantized MIDI gears aligns note start and end towards the segmentation.

## options

foreverloops can be customized via options from the main menu. You can change the language, switch to full screen mode, alter the audio device, and define your own key bindings, to name a few. In this section, we walk you through the possible adjustments.

### general settings

The general settings section comprises more behavioral settings:

- language  
Allows you to setup the language. Once changed, the new language is applied immediately, no restart is required.
- logging  
Enables / disables real-time logging to a file. If enabled, it may help find the causes of problems or bugs. Logging is enabled by default.
- uniform rotation editing  
This setting affects how gears are displayed when you hover over them. Usually, when you hover over a gear, its visual representation stops rotating and instead the sampler circulates. Otherwise, it would be very difficult to set up media samples. When uniform rotation editing is disabled, the visible rotation stops right where it was, thereby minimizing visual disruptions. When enabled, the visible rotation is always reset to twelve o'clock. Therefore, it emphasizes the beginning of each loop, and streamlines the programming of rhythm patterns.
- performance profile  
Here, the overall foreverloops performance can be set. For the time being, three different profiles are supported:  
**max performance** – this setting allows foreverloops to make use of all available CPU resources. We recommend this setting to power users on desktop systems, but also to players with older hardware.

**high performance** – the default setting. We expect most users to stick to it, as it provides a nice tradeoff between performance and power economics. It is also suitable for laptop users with newer hardware.

**normal performance** – the setting for mobile computers. We recommend this one for older laptops or computers with loud fans.

- mute sounds in menu

If enabled, the menu scene sounds are not played back.

## display settings

In the display settings section, you can alter several graphics properties:

- full screen  
If enabled, foreverloops switches into full screen mode, otherwise it can be used as a windowed application.
- show fps  
If enabled, the current frames per second are displayed in the main scene.

## sound settings

In the sound settings sections, you can alter properties related to the audio device and sound sample playback:

- audio device  
Here, you can choose the audio device you want to use with foreverloops. Altering the audio device will re-initialize the audio system. Any changes are applied on-the-fly, with no restart necessary.
- sampling frequency  
This option lets you specify the sound sampling frequency. Common values are 44100 Hz for CD quality and 48000 Hz for DVD quality.
- auto ramp in  
If enabled, the setting will avoid clicks and other annoyances during playback, as newly triggered audio samples will be ramped. If disabled, audio triggering is slightly faster.
- scratching affects pitch  
This setting determines how audio samples should playback from a scratch gear with reduced or increased speed. If scratching affects pitch is set, both sound tempo and pitch are affected, which sounds like an LP being played at the wrong speed.
- audio threads  
Here, you can set the number of audio threads for playback processing. Generally, more threads take advantage of additional CPU cores. You can also set this parameter to zero for synchronous updating. Depending on your sound hardware, this may lead to more or even reduce audio glitches.
- buffer size  
This setting determines the playback buffer length. For slower computers, we recommend larger values to avoid stutters and breaking-ups. Otherwise, smaller values generally improve latency.

- auto arm MIDI gears

This option determines if MIDI gears are armed directly after creation, and therefore are able to receive MIDI notes without further user interaction. If disabled, MIDI gears always must be armed manually.

- remap MIDI inputs to full velocity

If this setting is enabled, any MIDI note from external devices is played back / recorded with full velocity.

- MIDI input devices

Actually, this is not a setting, but an information panel where any connected MIDI input devices are listed. The list is updated automatically if new MIDI devices are attached, or if existing devices are disconnected.

## interaction settings

The interaction settings section lets you specify zoom and pan behavior:

- pan speed

If you find the default pan speed too slow or too fast, you can alter this value.

- reverse pan direction

If enabled, panning direction will be the reverse interaction direction. In many games, this is the default behavior.

- zoom speed

If you find the default zoom speed too slow or too fast, you can alter this value.

- reverse zoom direction

If enabled, zooming direction will be the reverse interaction direction. In many games, this is the default behavior.

## export settings

In the export settings sections, you can specify the overall quality of video captures, and where captures are filed:

- video capture quality

Here you can specify the quality with which video captures are recorded. As a rule of thumb, the higher the quality, the bigger the resulting files.

- capture directory

Here you can specify the directory in which the captures are filed.

- open capture directory

If pushed, this button will open the currently set capture directory.

## key bindings

The key bindings section lets you alter the default key bindings and create your own shortcuts. Every control can have up to two key assignments, and these key assignments can combine multiple keys (e.g., the letter 'E' and the modifier key 'SHIFT'). There are specific key bindings you cannot alter (modify, exit, delete). Also, it is not possible to assign ambiguous keys. If you assign a key that is already used for something else, the old binding will be discarded, as indicated by a warning message.

Custom key bindings can really speed up your workflow. For example, you can assign the left mouse button (LMB) alongside modifier keys (ALT / SHIFT / CTRL) to the gear creation commands.

control	key binding 1	key binding 2
modify	LMB	
exit	Escape	
delete	Backspace	Delete
create drive gear	8	Left Shift + LMB
create sequencer gear	9	Left Ctrl + LMB

### video decoding

Real-time video scratching requires specific codecs that support real-time jumps between various frames. Therefore, we use the MJPEG codec to encode your footage. If you experience any troubles (like stutter or delays) in the video output, we recommend the following to help improve the situation:

- lower the resolution of your footage
- use an SSD for the video material
- capture videos to a different drive than that in which your source material is located to reduce bus clashes

Video decoding depends on both, the CPU and the GPU.

### audio decoding

Audio decoding depends on your CPU power and your audio chip. If you experience any acoustical problems (like clicks or shutter), adjust the audio buffer size in the sound settings of the options menu. It may also help to increase the performance profile.

### hardware requirements

The computational power needs increase with:

- the number of audio files played back at the same time
- the number of images used in a scene
- the resolution of the video files used
- the quality set for video captures

The following only slightly influences the hardware requirements:

- number of gears used in a scene
- number of audio or video files fed into inactive gears
- higher gear segmentation

### loading

If loading the scenes slows down, try to decrease the number of files listed in the sample browser to the required minimum.

minimum requirements

OS	Windows 7 (Service Pack 1 + convenience rollup) 32-bit or higher
CPU*	Intel Core 2 Duo Processor 2.0 GHz or equivalent
RAM*	4 GB or more
GPU*	DirectX 10 compatible graphics card OpenGL 3.3 support with newest drivers installed
	at least 1 GB video RAM
HDD	1 GB free hard disk space SSD recommended for samples
Sound card	required

\* foreverloops requires hardware with less memory or computational power than listed here, but using more media files, and playing them back at the same time, increases these requirements.

recommended system setup

OS	Windows 10 or higher
CPU	Intel i5 2.4 GHz or equivalent
RAM	8 GB or more
GPU	DirectX 10 compatible Nvidia or ATI graphics card OpenGL 3.3 support with newest drivers installed
	at least 2GB video RAM
HDD	2 GB free hard disk space SSD recommended for samples
Sound card	required

If you're having a problem with a particular foreverloops feature or you're experiencing a bug, you can email our support team and they'll get back to you quickly: [support@foreverloops.com](mailto:support@foreverloops.com)

For other inquiries, please email us at [contact@foreverloops.com](mailto:contact@foreverloops.com)

## [about foreverloops](#)

foreverloops is also the name of our newly founded (2017) Austrian start-up.

Our mission and our passion is to build software that helps you effect astonishing outputs by simply playing around.

We strongly believe in your creative potentials through (digital) play activities and we want you to utilize and advance them!

Connect with us through social media to stay up-to-date with our latest news and ongoing developments, join topical discussions and share your creations!

Subscribe our YouTube channel for up-to-the-minute videos, to watch tutorials, and to learn new tricks!

If you would like additional features, have questions about specific components, or want to critique foreverloops, our forum on Steam is a good place.

concept & idea / project lead / software engineering  
interaction design / art direction / qa

marlene brandstätter & ulrich brandstätter

application design

LIVINGROOM digital solutions GmbH

[www.livingroom.co.at](http://www.livingroom.co.at)

sound samples

reinhard gupfinger

[www.gupfinger.net](http://www.gupfinger.net)

phil gallardo

SoundFont

s.christian collins

video samples

michael wirthig & verena henetmayr

demo composition

xorGrox aka Gregor Woschitz

translations

Level Up Translations

[www.leveluptranslations.com](http://www.leveluptranslations.com)

## special thanks

Andrea Jelinek

Christa Sommerer

Daniela Wageneder-Stelzhammer

Daniel Hug

Erwin Poganitsch

Eva Schmuckermair

Helene Hochrieser

Hubert Preisinger

Jeremiah Diephuis

Jörg Schlipfing

Nicholas Wright

Katharina Kloiber

Kathrin Anzinger

Lisa Nebl

Margarete Jahrmann

Nikolaus Dürk

Peter Purgathofer

Roland Richter

Ruth Browne

Simon Schmuckermair

Volkmar Klien

## promoters

foreverloops is promoted by aws impulse XL of BMWFW processed by aws



This is a license agreement ("EULA") between the end user and the foreverloops GmbH (hereinafter abbreviated as foreverloops).

This Agreement establishes the end-user's right to use this application.

By installing the software, the end user agrees to this agreement.

### 1. Definitions

- 1.1 "Software" means the computer program licensed under this Agreement, including all updates and upgrades, except for other content
- 1.2 "Content" means any audio, video or photo files included in the application, other sounds and templates containing appropriate files
- 1.3 "Application" means Software and other content
- 1.4 "Upgrades" means Supplementary versions of a software
- 1.5 "License Seat" means the license assigned to a particular terminal of the end user
- 1.6 "Terminal" means any physical or virtual device capable of processing data using a programmable calculation rule (such as PC, laptop, mobile device, etc.) in conjunction with a processor
- 1.7 "Commercial Use" means any commercial use or use for the purpose of making profit

### 2. Scope of use

- 2.1 foreverloops grants the end user the non-exclusive, non-sublicensable, non-transferable right to use the application on one (1) license seat for personal use as well as for commercial purposes, under the conditions specified in this Agreement.
- 2.2 The contents may be used for both personal and commercial purposes. However, a transfer to third parties for commercial purposes is not permitted.
- 2.3 Contents may not be extracted from the templates and used independently of the work results created by the software.
- 2.4 The use of upgrades requires the use of the basic product. An isolated transfer of an upgrade to third parties is, therefore, not permitted. The End User agrees that the Application may automatically download and install reasonable and/or necessary updates and upgrades. There is no obligation for foreverloops to support the previous version (s).
- 2.5 If the application contains third-party content, the (license) provisions of the corresponding providers shall be observed in addition to the General Terms and Conditions and the EULA of foreverloops.
- 2.6 Intellectual property rights and other intellectual property rights are recognized by the End User and remain with foreverloops.

### 3. Usage restrictions and protection of the software

- 3.1 The End User is prohibited from passing on, selling, lending, renting, or copying the Application to third parties.
- 3.2 The End User is prohibited from copying, editing, modifying, decrypting or copying the Software. Furthermore, the decompilation, the reverse translation of the object code into the source code or the reverse engineering is prohibited. No derivative or modified forms of the Software may be created.
- 3.3 The End User is prohibited from making the application or parts of the application available through a network that allows multiple users to simultaneously use it.
- 3.4 The End User is prohibited from removing, altering or disguising product identifications, copyright notices or other legal notices of intellectual property in the application.

#### **4. Support Services**

foreverloops may provide the End User with technical support and community management services in conjunction with the Application ("Support Services").

foreverloops reserves the right to change, suspend or terminate these Support Services at any time and for any reason.

You can contact foreverloops for support via [support@foreverloops.com](mailto:support@foreverloops.com).

#### **5. Warranty and Liability**

5.1 To the extent permitted by law, the Application shall be made available to the End User as provided. foreverloops does not warrant that the Software is free of defects. Warranties of merchantability, satisfactory quality, fitness for a particular purpose, or compatibility with third-party Software are not provided.

5.2 foreverloops assumes no liability whatsoever for damages, in particular not for personal injury, property damage, consequential damage, indirect damage or loss of profit nor from warranty or product liability unless mandatory consumer protection provisions stipulate otherwise.

5.3 If foreverloops becomes liable for damages under the statutory provisions, this liability shall be limited to intent and gross negligence. The liability is limited to the amount of the actual contract. Any further liability is excluded unless consumer protection provisions stipulate otherwise.

5.4 foreverloops assumes no liability for the topicality, correctness, completeness and content of the information provided.

#### **6. Exclusion of the right of revocation**

The End User takes note and expressly acknowledges that the right of revocation is excluded.

#### **7. Termination**

7.1 The End User may terminate the agreement at any time by destroying the Application and all components.

7.2 foreverloops may terminate the Agreement at any time if the End User does not comply with the terms of use of this EULA. In this case, the End User may no longer use the application and must destroy it as well as all of its components.

#### **8. Written form requirement**

This agreement is final.

There are no oral collateral agreements.

Changes to this agreement must be agreed upon in writing.

#### **9. Severability clause**

Should any provision of this EULA be or become invalid, the validity of the remaining provisions shall remain unaffected.

The invalid provision shall be replaced by a valid provision which most closely reflects in the commercial sense the provision to be replaced.

In the event of a contradiction with the General Terms and Conditions of foreverloops, the provisions of the EULA shall apply.

#### **10. Jurisdiction and applicable law**

In the case of disputes, the court of jurisdiction for commercial customers is the relevant court at the registered office of foreverloops.

Austrian law is expressly applicable;

the application of UN purchasing law is expressly excluded unless mandatory consumer protection regulations do not provide otherwise.

## **11. Extrajudicial settlement of disputes**

The following applies to End Users within the European Union:

In the case of a dispute concerning the interpretation, fulfillment or validity of the EULA, an amicable solution shall be sought before any legal action is taken.

Complaints may be submitted to [contact@foreverloops.com](mailto:contact@foreverloops.com).

If a complaint fails, the End User can file an online complaint within one year after the failure of the complaint on the online dispute resolution website of the European Commission:

<https://webgate.ec.europa.eu/odr/main/index.cfm?event=main.home.chooseLanguage>

or on the website of the European Consumer Center:

<http://www.europe-consommateurs.eu/en/home/>.

If extrajudicial redress procedures fail, proceedings may be brought before the competent court.