Introduction to zenAud.io ALK

Congratulations on purchasing zenAud.io ALK. With ALK, and a little bit of ingenuity, you will be able to create stunning live shows and dexterous musical transitions.

ALK is a **sequenced** looper. Unlike a traditional looper, where the signals to record or play a loop are triggered via a pedal, in ALK you can structure loops on a timeline – before you play your gig. Then, when it's time to perform the song in front of an audience, ALK takes care of triggering the appropriate actions depending on where you are in the song.

ALK is a VST and AudioUnit audio host, which means you can use any of your plugins during recording in your show, However, you'll need to watch out for plugin latencies when creating your effects chains.

Unlike traditional loopers, ALK is capable of looping MIDI signals, and comes with two useful built in MIDI effects: quantize and note repeat.

ALK has unique and powerful mixing capabilities. Unlike other sequencers, ALK has no need for mixer or effects return track types. This is because in ALK, any track (audio or MIDI) is able to mix any number of inputs, and feed into any number of outputs. **Symmetric routing** technology makes it incredibly fast and efficient to create submixes or effect sends.

zenAud.io ALK takes a slightly unconventional approach to automation. Philosophically, we believe that operating a rotary encoder on stage *is itself* a kind of performance. To reflect this, we've elevated automation to its own track, decoupled from the track(s) and plugins that modulate it. This allows a single hardware encoder or fader to control different destinations at different times, simply by drawing regions and assigning the input to the desired VST parameter within that region. Of course, it's still possible to draw automation using curves, and ALK supports trackpad gestures to minimize the number of clicks and mouse drags needed to perform common curve manipulations.

Although ALK's unique look and feel and its innovative sequenced looping system are relevant and powerful features, the real meaning of ALK is what you can do with it: for the first time, you can realistically perform complex arrangements in a live setting. We hope you enjoy using ALK as much as we enjoyed designing and coding it.

How ALK Works

Using *ALK* is a three step process:

1. Set up any Tracks you need for your performance, connect them to I/Os, and set any VST or AU plugins used. This can be done for a particular track using its Track Panel. For instructions on how to create a track, see Creating And Deleting Tracks, and for instructions on how to add and delete plugins to a track, see this section.
2. Draw Loops represent the parts of your song in the appropriate track lanes. Loops come in two flavours:

    - Record Loop, which allows you to create loops representing parts of the song you intend to perform yourself, and create when clicking and dragging in *Record Pencil Mode*.
    - Play Loop, which allows you to create loops that refer to material – recorded during the current "take" – stored in prior *Record Loops*. *Play Loops* can play back the MIDI or audio data of any *Record Loop* that occurs prior to the current loop on the same track.

3. Perform the song by hitting the Record Button. When you do this, the track time begins to advance. As the cursor moves over recorded loops, any signal, whether MIDI or audio, is recorded into the record loops clip object, allowing it to be looped during the performance through *Play Loops* later occurring in the track.

Of course, you won't follow these steps exactly while your actually writing you're songs. Instead, you'll probably draw in little bits at a time, and perform them, bit by bit (using Lock Performances to allow you to concentrate on one thing at a time). The steps above serve to illustrate that, unlike production sequencers, the job ends with a gig rather than with an "Export As…" command!

The trick to using *ALK* is understanding that record loops represent areas to be performed *in the future* – that is, when you hit the *Record* button to engage *Record Mode*. So whereas a tape reel is an appropriate metaphor for traditional timeline oriented sequencers, for *ALK*'s the Arrangement View the more appropriate metaphor is a *script* – a *recipe* which controls the actions of the underlying, manual looper(s) throughout the progress of the song.

One of the advantages of being freed from manually triggering overdubs (not to mention MIDI looping) is that you can get a song off the ground much faster than with a traditional looper. The *Arrangement View* encourages you to be creative and allocate the most precious asset you have onstage – your concentration – wisely.

That way, you will rise to the musical nirvana of achieving a big sound, quickly, while remaining *in the flow*.

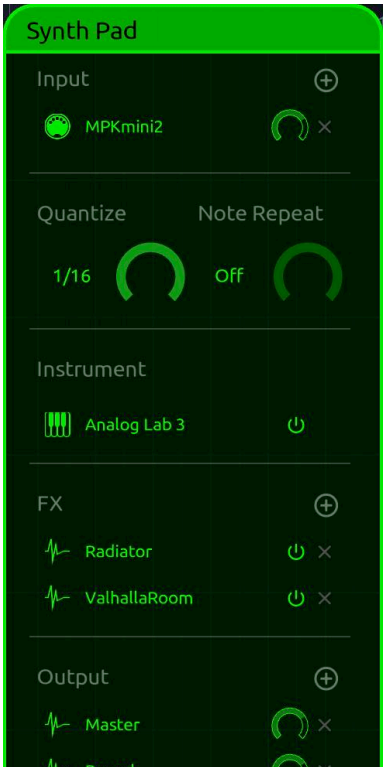
Track Panel

The **Track Panel** is analogous to a physical *mixer*. It is the low-level nerve center of ALK. Low-level, because it ignores everything that has to do with time. In particular, a *mixer* has no knowledge of a sequence or loops — it simply combines a bunch of raw audio or MIDI signals with each other and spits them back to its outputs.

Every track in *ALK* has its own virtual mixer, with its own view of the physical inputs and outputs. It can be located inside each track's **Track Panel**.

User Interface

The track panel has the following items (in the order of signal flow, from input to output).

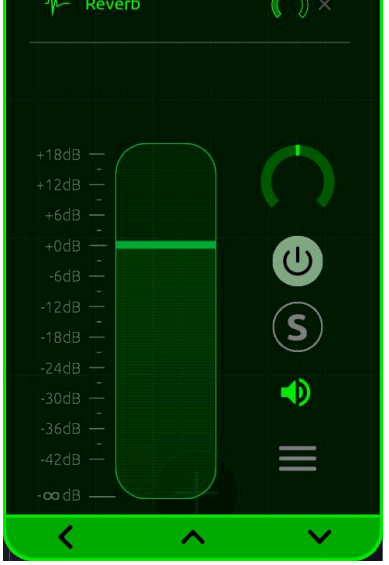


Input Terminal

This is a list of the current inputs to the track. These can be physical inputs, such as an audio interface channels and MIDI keyboards, or Track inputs which feed from the output of other tracks in the session (ie: a Master track which takes its inputs from many individual tracks). We call these connections cables in *ALK*.

MIDI Effects (Instrument and MIDI tracks only)

We've included two useful MIDI effects right in the track panel. They are **Quantize** and **Note Repeat**. **Quantize** effectively shifts any out-of-time notes into time. A drop down menu allows the user to



choose the resolution of quantization. For example: choosing 1/16 results in recorded notes playing back aligned to the nearest 16th note. The rotary control is used to control the *amount* of quantization. This decides by how much notes are shifted to the chosen quantize resolution. Setting a *quantization amount* lower than 100% is favourable to some as it results in a more human and less rigid correction of note timing. **Note Repeat** essentially translates held MIDI notes into a stream of repeated notes. A drop down menu allows the user to choose the resolution of these notes. For example, choosing 1/8 results in held notes transforming into a stream of 8th notes. The rotary control here determines the amount of *swing* in the repeated notes.

Instrument (Instrument Track only)

Here you can insert your VST or AU instrument of choice. Clicking on the Instrument will open it in a new window where you can view the user interface and edit parameters and fine tune your sound. We've also included two handy utility instruments: **Test Note** and **Metronome**.

Note

Test Note and *Metronome* do not have a plugin editor, so unlike unlike other VSTs and AU plugins, clicking on the label has no effect.

FX

Here you can insert VST or AudioUnit effect plugins to process the audio input (Audio track), Instrument audio (Instrument track) or MIDI data (MIDI track) of your track. Clicking on the insert effect opens its user interface in a new window where you can edit parameters and fine tune the effect. Inserts can be powered off (bypassed) using the power button. Bypassed inserts do not consume CPU: once bypassed, they no longer participate in the processing chain until they are re-enabled.

Note

If you're project gets large enough that it starts to stutter, finding CPU-hungry plugins, and bypassing them, can be a good way to fix the problem. Bypassing can also be useful to determine which plugins are responsible for high CPU use, simply by comparing the CPU use before and after bypassing.

Tip

Instrument and FX plugins can be **copied**, **cut** and **pasted** between tracks using commands **C**, **X** and **V**. FX plugins can be reordered by cutting and pasting a plugin to its desired position.

Output Terminal

This is a list of the current outputs of the track. Again, these can be physical outputs such as those on your audio/MIDI interface or Track outputs which feed to the input of other tracks in the session.

Pan Rotary

This acts the same as pan controls on physical mixers, except that in addition to controlling the pan, it also displays the current pan amount. This is determined in real-time by comparing the levels on the left and right channels.

Volume Slider

This acts the same as a typical volume slider and LED meter combination on a physical mixer.

Solo Button

Hitting *solo* powers off every other track, allowing you to hear only the current track. Note that multiple tracks can be soloed. Solo state is engaged if there is at least one track which is soloed. Once the last soloed track is unsoloed, auditioning behaviour reverts to normal.

Power Button

Hitting *power* powers off the current track, rendering it silent.

Audition Indicator/Button

Auditioning (monitoring) in *ALK* controls how the user hears the output of the track. Auditioning has two modes, *manual* and *hover* which can be switched in the track menu under *Disable/Enable Hover Auditioning*. In manual mode, auditioning can be turned on and off via the Audition button. In hover mode, auditioning is turned on by hovering the mouse over a track. This is a useful way to quickly audition the sound of tracks in session without having to manually turn on and off their auditioning.

Track Menu

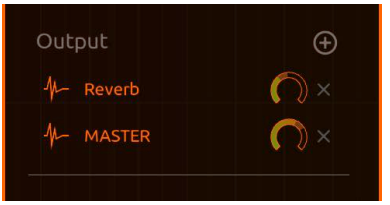
Here there are options for *Enabling/Disabling Hover Audition*, Lock/Unlock Performances, *Duplicate Track* and *Delete Track*.

Note

The *inputs* and *outputs* are also called *I/Os*.

Cables

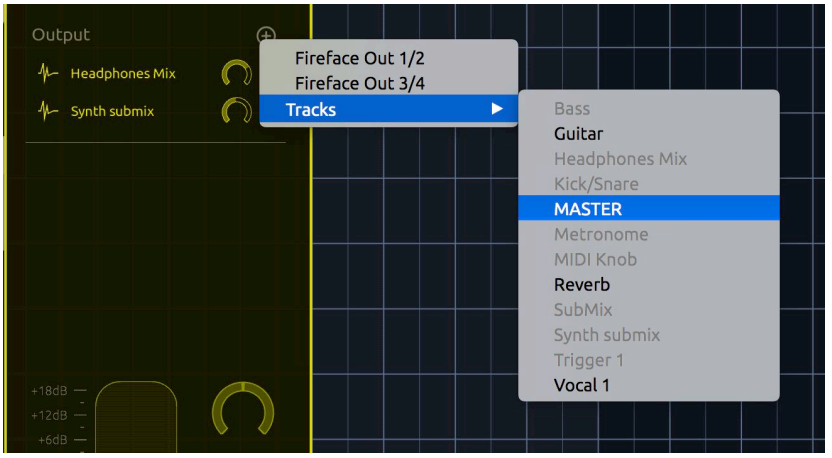
A cable is roughly analogous to a physical cable. In other words, it serves to connect, say, an microphone input to the input of a track, or the output of a track to the submix track. However, unlike physical cables, or the connections in other sequencers, *ALK's* cables have two differences:



1. Cables in ALK have a **volume knob** between their ends. This means you can control the individual amount of signal flowing from the input to the output.
2. Cables in ALK have a **level meter** built in, which allows the user to obtain visual feedback for the signal passing through the cable.

Note

The level meter currently only works for audio tracks. This limitation will be removed in a future version of the program.



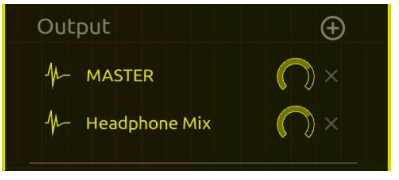
Cables come in two flavours: MIDI and Audio. You never have to choose the type of **cable** used: you simply connect the track to wherever you want, and the correct cable type is selected for you.

A *cable* is represented visually in *ALK* by an *object* in the appropriate input or output

terminal. In the image below, *Synth submix* and *Headphones Mix* are both cables. The surrounding element labeled "Output" in the image, is the *Terminal* they belong to, which we will discuss next.

Terminals

A terminal represents the input or output of the current Track. In *ALK*, a terminal can have *more than one* Cable attached to it. This means that every track in ALK can:



1. act as a "mixer", using multiple cables as inputs, and using the cable's volume knobs as the "mixer's" individual volume sliders;
2. act as a signal "splitter", directing the signal flowing through this track to other tracks or physical outputs;
3. act as an FX track: this can be achieved by connecting the appropriate input cables to an audio track with your desired effect plugin inserted in the track FX;

You can add a cable to a terminal clicking on the *Add* (+) button at the top right corner of the the *Terminal*.

Note

Currently, for VST and AU plugins, only a single mono/stereo terminal is allowed. In other words, while plugins like *BFD* are supported, you currently cannot use their extra outputs; only the main stereo output can be used by the program. This limitation will be removed in the future, which will also allow sidechaining. We apologize for the inconvenience!

Tracks

Tracks are an important concept in ALK, so in this section, we'll cover what they are and the different types of tracks that exist. Tracks have two visual representations in *ALK*: Track Leaders and Track Panels.

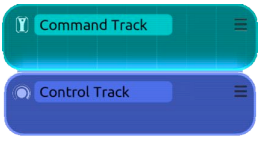
Like Loops, tracks have *names*, which can and should be changed to something meaningful, like "lead guitar". This is described in more detail in Using Tracks. (Giving tracks good names brings tangible benefits in ALK: track colors are grouped by according to their name, so e.g. *Synth Bass* and *Synth Pad* would automatically get similar colors, whereas *Lead Vocal* and *Backing Vocals* would get a different color.)

Tracks operate at a *higher level* than the Track Panel's mixer: they know the *sequence* of Loops in the arrangement. To use an analogy, if the mixer represents the raw, low-level, *immediate*, realtime signals, then the track is like an *orchestrator* or *sound man*, which controls the operation of the varous audio and MIDI processor, such as the loop functionality, and FX, *over time*, i.e. throughout the live performance.

The script or sequence this metaphorical orchestrator follows is determined by the sequence of Loops contained in the track.

Track Types

There are five types of tracks in ALK. Three *performance tracks*: **Audio**, **Instrument** and **MIDI**, and two *automation tracks*: **Command**, and **Control**.



Track types can be distinguished through their Track Type Icon. In the image on the left, *Analog Synth* is an **instrument track**, *Main Vocals* is an **audio track**, and *Keys* is a

MIDI track; on the right, we see a *Command Track* and *Control Track* — note that the icons are look like a foot switch and a rotary encoder, respectively.

Track Type	Input Type	Output Type	Plugin Types
Audio	Audio Data	Audio Data	Audio FX
Instrument	MIDI Data	Audio Data	Audio FX
MIDI	MIDI Data	MIDI Data	MIDI FX
Command	MIDI Data	Automation Data (On/Off/Trigger)	None
Control	MIDI Data	Automation Data (Continuous)	None

Using Tracks

- **Track Names** - Track names can be changed by clicking on the track name in the Track Panel or Track Leader. ALK approach to **coloring tracks** is unique, and depends on you providing **good names**. Basically, ALK uses a so-called *clustering algorithm* on a *string-distance metric* to give tracks that *have similar names similar colors*. This removes the redundancy of giving good names *and* choosing colors. To help make this processs seamless, ALK automatically sets initial track names in the following situations:
  1. For instrument tracks, where the name of the track *has not* been changed manually (i.e. by typing in the edit field in the track leader or panel), the name of the track is automatically changed to the track instrument.
  2. For audio tracks, the name is changed to the name of the first input. Since it is possible to change the visual name of hardware inputs in the Preferences window, this allows you to create semantically meaningful inputs (say, *Jim's Vocals* — which could override the default hardware input name *Babyface Input 1*) which, when added as first cable to an audio track, automatically change the name of the track to its input. (So the track would now be called *Jim's Vocals*.) Note that, as above, this is *only the case* if track hasn't already been renamed "by hand".

- **Creating Tracks** - Tracks can be created using the add track button.
- **Deleting Tracks**: Tracks can be deleted by clicking the delete track menu item in the track options menu on the relevant track's panel or leader.
- **Options Menu**: Through the track options menu, you can also access features such as hover auditioning and lock performances.
- **Reordering Tracks**: Tracks can be reordered by dragging and dropping.

Loops

Loops and regions lie at the heart of zenAud.io. Both loops and regions describe musical objects that occupy a *range* of time; that is, they have a *beginning* and an *end*, and are visually represented in ALK by a rounded rectangle whose fill color is determined by the color of the track it lives in, and its stroke or border color is determined by the type of loop or region.

Note

Loops are **regions**, but the converse is not true. In other words: region is the *more general* term, whereas loops are specific to audio, instrument, and MIDI tracks. It's actually easy to remember the difference: if it *loops*, it's called a loop, otherwise, it's called a *region*!

Note

This section will focus on *loops* rather than (automation) regions. Automation regions are covered in the section on automation.

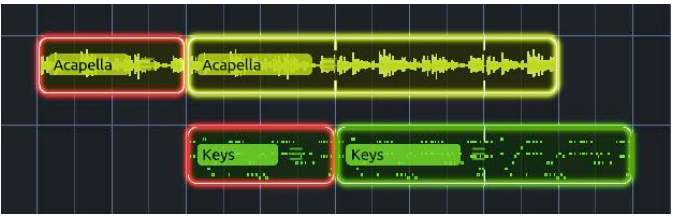
Loops are the vehicle by which you pre-plan the specific sequence of record/play/mute/unmute commands for a performance. Loops are ordered by time: each loop represents a range of time. Loops in a single track can't overlap in time, so for a given track, at any one time, there can only be one loop.

Loops come in two flavors: **record loops** and **play loops**. You can tell loop types apart by their outline or border color: record loops are *red*, while play loops are *green*. This mirrors the typical color scheme used to represent the transport, and reminds us that the loops function in a similar way.

Note

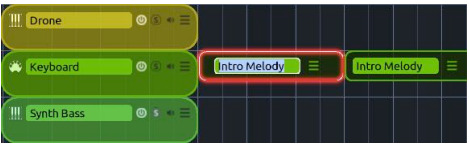
When a track is locked, its loops take on the same border color as the track itself

(i.e. its Track Leader). This emphasizes the fact that recording doesn't affect the track in the locked performances state.



Record Loops

Record loops represent commands for the looper to begin and end recording. Record loops have *names*; no two record loops on the same track can have the same name. You can change the name for a loop by clicking on its name text field in the *Arrangement View*, which makes the text field editable. This is shown in the screenshot below.



Record loops contain **clips**, which can be seen as a buffer that stores individual performances. If you record/perform the same song several times, the last eight performance takes will be saved as *clips* which are accessible by right-clicking the loop.

A **record loop** behaves in different ways depending on whether we are **recording** or in **play mode**. Assume that at the current position in the song, or the **playhead**, is within the range of the loop in question. In other words, we're in the loop's region.

1. If the **transport** is in the **record mode**, then any incoming signal is recorded into a loop's **clip** which subsequent **play loops** will use as a reference to play back from.
2. If the **transport** is in **play mode**, then the current loop clip is played back. The incoming signal is passed through for monitoring depending on the combination of **mute**/**solo**/**track audition** settings on the loop's track leader, however it is not recorded.
3. Otherwise, if the **transport** is in **stop mode**, incoming signal is passed through for monitoring depending on the combination of **mute**/**solo**/**track audition** settings on the loop's **track leader**. No signal is recorded and no pre-recorded loops are played back.

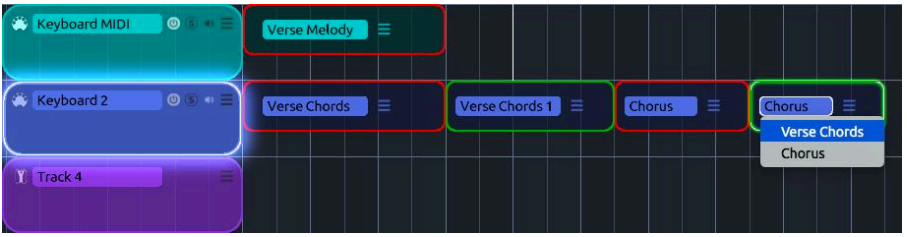
Play Loops

Like a **record loop**, a **play loop** stretches over a range of time – it has a beginning and an end. However, a **play loop** represents a command to play back or *loop* previous material.

A **play loop** doesn't store any loops. Instead, it links to **record loops** occurring before the (play) loop on the current track, and asks *them* what their current clips are.

Play loops are named by the **record loop** that they link to. To change which record

loop a play loop links to, click on its name in the arrangement view.



A **play loop** behaves in the same way, regardless of whether we are in **recording** or in **play mode**.

Assume, again that we're within the loop's region:

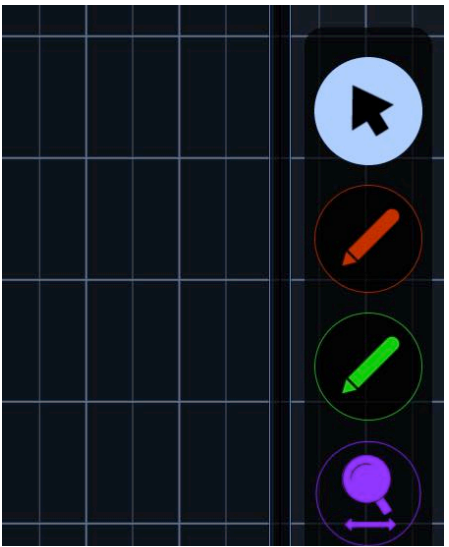
1. If the **transport** is in **record mode** and the loop refers to a valid **record loop**, then that **record loop's current clip** is played back during the **play loop**. The incoming signal is passed through only within the **record loops**.
2. If the **transport** is in **play mode** and the loop refers to a valid **record loop**, then that **record loop's current clip** is looped to fill the **play loop's** loop length. The incoming signal is passed through for monitoring depending on the combination of **mute/solo/track audition** settings on the loop's **track leader**.
3. Otherwise, if the **transport** is in **stop mode**, no pre-recorded loops are played back. The incoming signal is passed through for monitoring depending on the combination of **mute/solo/track audition** settings on the loop's **track leader**.

Loop Editing

In this section, we discuss the tools *ALK* provides in order to create arrangements using **loops** in the Arrangement View. The tools are described under *drawing modes* in the *User Interface* drawing modes section.

Drawing Operations

Loops can be drawn using one of two **pencils**: the **Record Pencil**, or the **Play Pencil**, both of which appear in the right hand tool bar. Drawing is accomplished by pressing the mouse down in a track (which determines the beginning of the loop), and subsequently dragging the mouse to the right to determine the length of the new loop. When the mouse button is released, the new event will be added to the sequence.



The **draw mode** used for the operation can also be determined by modifier keys pressed at the start of the drawing operation:

1. No modifier keys pressed: the drag operation is a select range, and is therefore outlined in blue.
2. **Cmd key** pressed: the **record pencil** is activated, and a record loop is drawn.
3. **Alt key** pressed: the **play pencil** is activated, and a play loop is drawn.

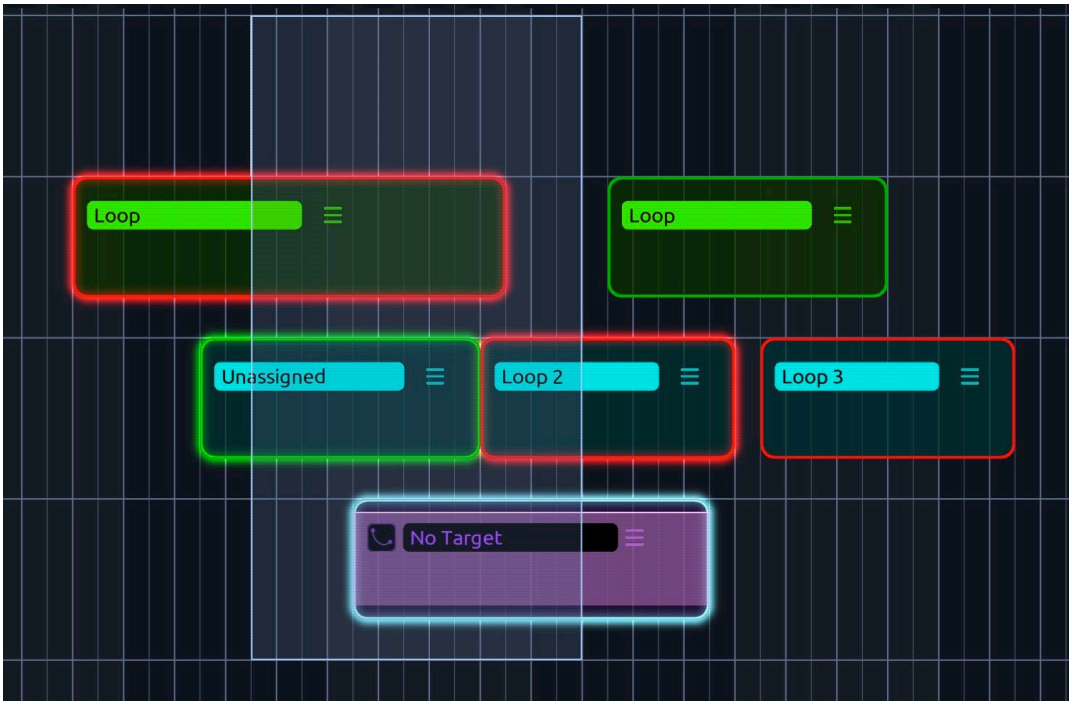
Note

The modifier keys must be pressed before the beginning of the drag operation.

Selecting and Moving Loops

In addition to drawing operations, loops can be modified in a few ways. First, you can move a loop's position and track by dragging the loop with the mouse to the desired location.

Note that when a loop is moved, *any other loops* that are currently selected are moved in exactly the same way. Also, if the **Shift key** is pressed while the loop is being dragged, the move operation becomes a *copy* operation – and all the selected loops are copied rather than moved.



Creating a selection area by clicking and dragging the mouse allows a number of loops to be moved at the same time. Clicking and dragging on part of a loop *within* the selection **moves everything within this selection**, whereas clicking and dragging on part of a loop *outside* but *touching* the selection will **move all loops touching the selection**. The same can be achieved by pressing the **Esc key** which removes the

selection area while maintaining loop selection. Pressing *Esc* once more deselects all loops.

Resizing Loops

You can adjust the beginning and end of a loop (*resize* the loop) by hovering the mouse close to the beginning and ends of the loop, which changes the mouse cursor into the resize icon. If you now click and drag in any direction, the loop's begin/end will be adjusted appropriately.

Tip

Hitting **FF** after any record loop automatically draws a play loop after it which continues to the end of the arrange window. Hitting **FF** after any play loop extends the loop to the end of the arrange view.

Select, Cut And Paste Operations

In addition to the *mouse modify operations*, *ALK* supports the standard cut, copy, paste and undo operations for loops. Unlike in other programs, *ALK*, for ergonomic reasons, uses the following keys for cut, copy, paste, and undo:

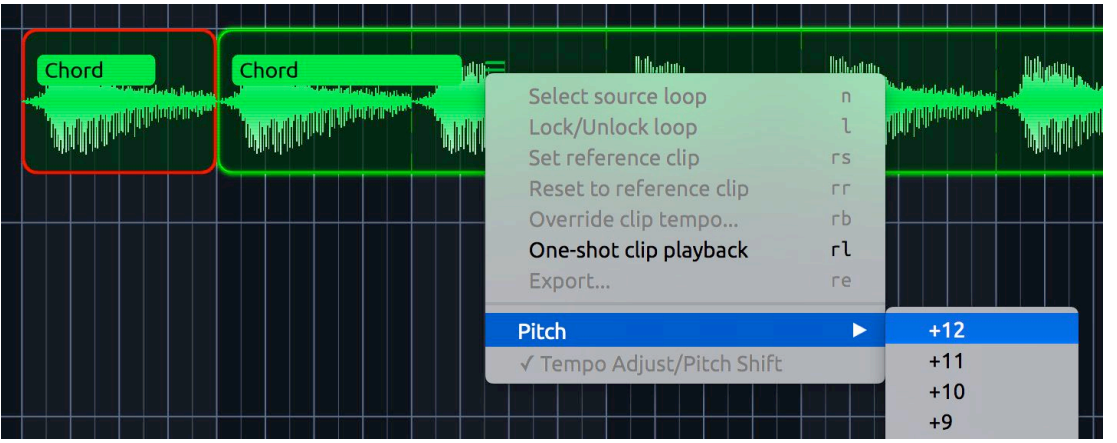
Select all	a key	Press repeatedly to cycle between all regions in track, documer
Delete	delete key	
Cut	x key	Like delete, but copies the loops/regions before deleting.
Copy	c key	
Paste	v key	Pastes the loops verbatim: record loops are pasted to (new) rec
Paste special	⇧ +v	Converts record loops to play loops linking to original record lo
Undo	z key	
Redo	⇧ +z	
Deselect	Esc	First press seselects selection area. Second press deselects loop

Note

These keys are the same as the standard key combinations *but without* the **Cmd key** being pressed. *Adieu*, carpal tunnel syndrome!

Pitch Shifting Loops

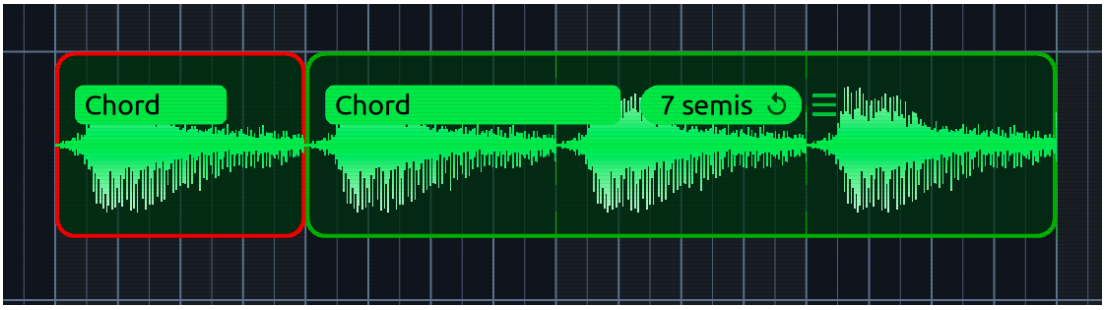
ALK allows for the pitch shifting of any play loop by up to +12 or -12 semitones. Pitch-shifting can be accessed by right-clicking the play loop and choosing the desired semitone of the pitch-shift.



Tip

Pitch-shifting can also be changed by hovering the mouse over the play loop, holding **Cmd** and **scrolling up and down** on the **trackpad**.

Once pitch-shifted, the semitone value appears on the play loop. To reset the semitone back zero, click the reset button next to the value.



Changing Loop Offset

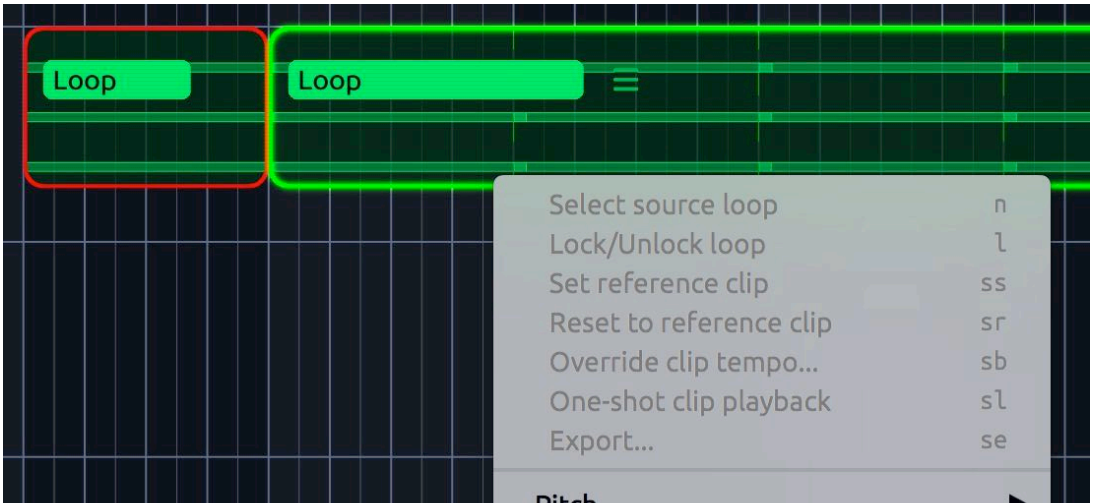
It is possible to offset the start-point of any play loop in *ALK* by any denomination of bars, beats and fractions of beats. The loop offset can be adjusted by hovering the mouse over the desired play loop, holding down **Cmd** and **scrolling left and right** on the trackpad.

The way the loop offset is displayed dynamically changes depending on the value. For example, an offset of two bars would be displayed as "**2 bars**" whereas an offet of **one bar**, **three beats** and **one quarter beat** would be displayed as "**1.3.25**" as per the image below:

To reset the offset of a loop back to zero, simply click the reset button next to the offset value.

Retrigger Overlapping Notes

When recording and looping MIDI, ALK intelligently extends held notes that overlap with the next repeat of the loop. This is useful for creating long extended notes, but can also be an unwanted behaviour. When retriggering of overlapping notes is preferred, a menu item is available by right-clicking the playback loop and choosing *Retrigger overlapping notes*.




Pitch
✓ Tempo Adjust/Pitch Shift
Retrigger overlapping notes
rt

Automation

Automation refers to the ability to dynamically control mixing parameters such as volume, panning, etc., throughout the course of a song or performance. The word 'automation' harks back to the days of analogue mixing desks. Mixing engineers, in a bid to add interest to a track, would manually alter the fader levels of individual tracks. Since this was limited by the fact that a human being has only two hands, mixing desk manufacturers came up with a solution: they incorporated motorized faders into their desks, which allowed the positions of the faders to change 'automatically'; hence, automation.

Automation in the era of the DAW has a taken on a broader meaning: essentially, it involves altering any sound-influencing parameter (including VST/AU effect plugin parameters). The parameter can be altered in two ways. It can be either programmed beforehand using so-called automation curves, or it can be performed using e.g. a MIDI controller (such as a fader, rotary controller, or pedal). zenAud.io ALK has direct support for both types of automation, in the form of Scripted Automation and Captured Automation.

In ALK – unlike other DAWs – all automation, whether scripted or captured, lives in Automation Tracks. Automation tracks look like normal tracks, and just like with normal tracks, you can draw "loops" inside them, but the correct terminology in ALK for this kind of "loop" is an Automation Region, because it **doesn't loop**. We'll discuss automation regions in greater detail below; for now, let's discuss the "why" of zenAud.io's automation, rather than the "how".

The Philosophy Of Automation Tracks

The idea that automation should live in its own track type is somewhat different from other DAWs. Typically, captured automation is treated differently from scripted automation: usually, only the latter is visible in the arrangement view, and even then it usually appears as a sub-lane of the track that it affects. Captured automation, on the other hand, is usually accessible separately from some global view.

zenAud.io's philosophy is that **all relevant information** about what you intend to do

when you perform should be available directly in the arrangement. With the increasing inportance of pattern based music – particularly in the domain of electronic dance music –, nowadays, in many types of performances, DJ-style automation of parameters such as filter cutoff and similar lie at the very heart of the performance, so it makes sense to treat automation as a *first-class citizen*.

As such, the right way to think of an automation track is simply as another instrument in the performance. You can imagine each fader, or each rotary, as a separate instrument in your rig, in much the same way that you may allocate a track to your guitar, vocals, synths, and drum machines. From that point, it's up to you whether you want to perform the automation *live* – i.e. to use captured automation – or to have it play back automatically using scripted automation.

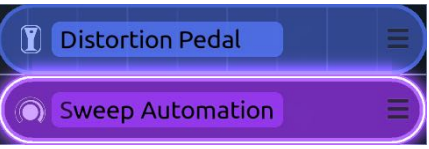
The Fader Sweep *Is* The Performance

As alluded to above, once you take the view that auotomation is a *first-class citizen*, and that the *fader itself* is the instrument, it makes sense that a single fader can control *different things at different times*. This is accomplished with automation regions, which allow the destination plugin and parameter to be set for that specific region, independently of all others.

This allows a single fader to control different parameters, and even different plugins on different tracks, at different times during the performance. At that point it, a fader acquires a *higher-level* meaning than in other DAWs. For example, rather than buying an expensive controller with a large number of faders or rotaries, you can instead use a controller with, say, four faders. The first fader can be used for overall, say, glitch-style effects – and in each song (or part of some song) in your set, you can draw a region assigning that fader to specific parameter on a specific plugin. The second fader could be used as a "breakdown fader". In the first song it may control a low-pass filter cutoff parameter to give you the classic filter-sweep effect. In the second song, it could control the decay of a reverb, to give you an effect with a similar *purpose*, but not exactly the same.

Automation Tracks

Automation tracks come in two flavors:



- Command Track: this represents an on/off parameter or a trigger. For example, you would use a command track to automate the *power* (or *mute*) track parameter. Command tracks have track type icons that resemble a foot pedal.
- Control Track: this represents a parameter which takes on continuous values. If you wanted to automate, say, the volume of a track, you would use a control track. Control tracks have track type icons that represent a rotary control.

Both automation tracks can be **duplicated** and **deleted** via the track menu.

Both automation tracks can be **duplicated** and **deleted** via the track menu.

A scripted control region set up to control the volume of a track.

Automation Track Panel

Just like other track types, automation tracks have track panels that give you access to functionality that impacts the track *as a whole*. Automation track panels look a little different than normal track panels, and this reflects the fact that, by itself, an automation track does not generate its own audio output; instead, it alters another track.

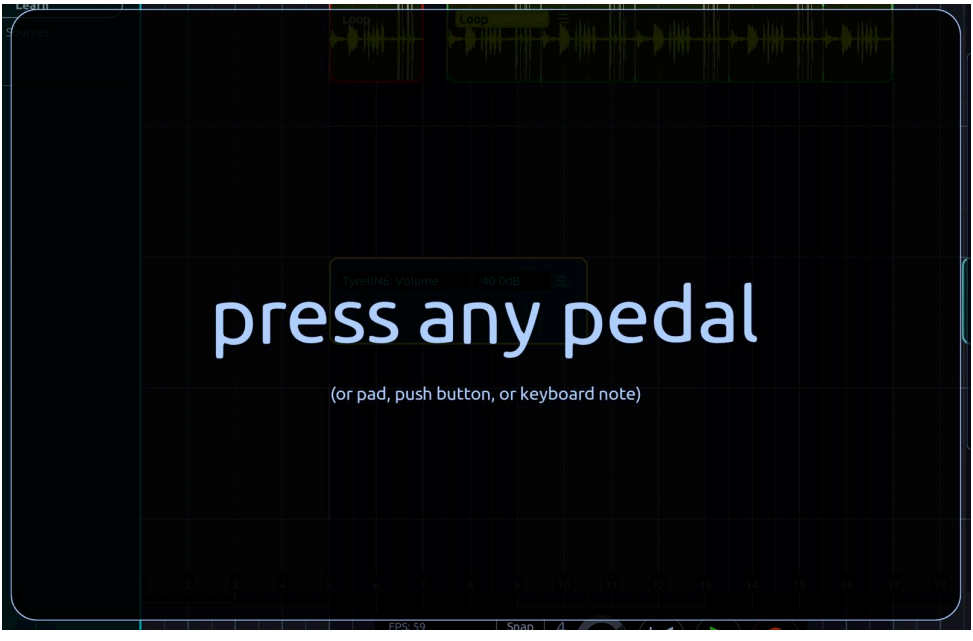
Following are the user interface elements available in an automation track.

- **Learn Button**: clicking this button initiates the Learn dialog. Learning associates a given *input* – say, the top-right fader – with the current automation track. Once a fader or rotary has been learned, ALK will forward that type of messages to any destinations for the duration of any captured automation regions.
- **Control/Command Sources**: this is a list containing any input sources (i.e. faders, rotaries, or keys) that the current automation track listens to. For command tracks, this list is labeled *Command Sources*; for for control tracks, it is labeled *Control Sources*. Sources can be removed by clicking the **delete button** on the right-hand side of the source, which means that the input in question no longer triggers the destination during a captured automation region. Note that multiple *sources* can trigger the same automation regions (and thus destinations). This allows, for example, several band members in a band to control the same automation destination. A typical use-case for this is to allow several band members to control the global panic mode.

Learn Source

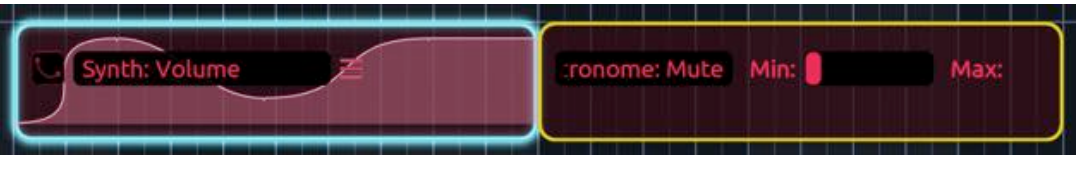
In order to use our MIDI controller's physical controls to manipulate functions in *ALK*, an input source must be determined. Within the Automation Tracks' Track Panels. Once a MIDI controller is connected, *MIDI Control Change* messages (CC's) can be assigned to the current track by clicking the **Learn** button. Once the *Learn dialog* opens, the user manipulates a physical MIDI control (fader, rotary, pedal etc.) and the control's *CC* number is learned by the automation track, allowing it to be routed to

Captured Automation Regions.



Automation Regions

Similarly to automation track types, automation regions also have two variants.



On the left, a *scripted control automation region*. On the right, a *captured control automation region*.

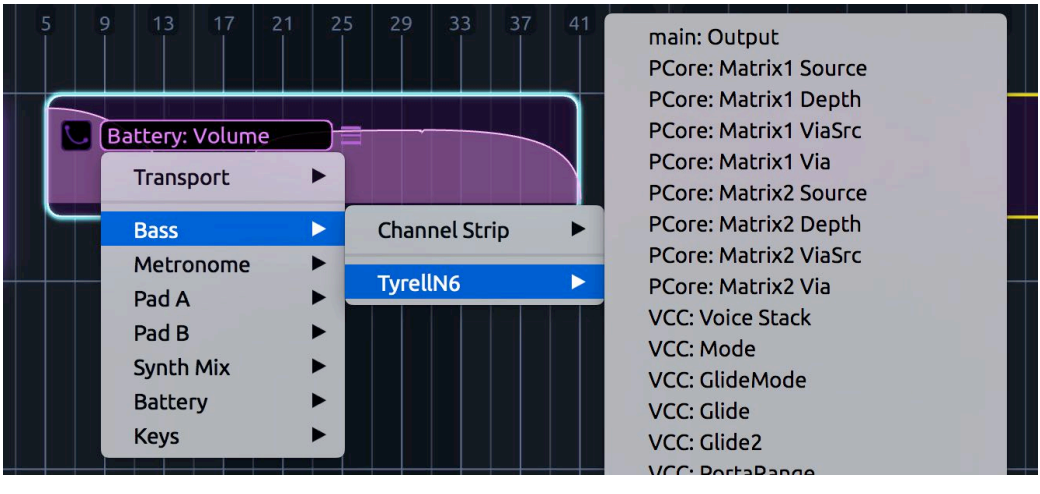
- Captured Automation Region: this is a region in which you intend to **perform** some automation. For example, you may use the learn function to associate a particular fader (say, the *left-most fader* on your MIDI controller, (with MIDI CC 25, say) with a particular track. This is analogous to setting the input to, say, channel 1 (guitar) on your *Rhythm Guitar* track. Next, you draw a captured automation region, which allows to determine what fader 0 controls in for the duration of the region. *Scripted automation regions* always have a **cyan border**, while their fill color takes on the hue of the track they reside in.
- Scripted Automation Region: this region represents automation which is **pre-planned**. For command tracks, scripted automation regions allow typing in the value taken on by the destination when the region is entered. For control tracks, scripted automation regions allow you to draw a curve using control points which will determine the values taken on by the destination over the course of the region. *Captured automation regions* always have a **yellow border**, regardless of the track they reside in.

Since there are two types of automation tracks, and two types of automation regions, there are in effect four variants of automation regions; these are summarized in the table below.

Track	Kind	Name	Notes
Control	Scripted	Scripted Control Region	Used to draw automation curves
Control	Captured	Captured Control Region	Control ranges can be controlled
Command	Scripted	Scripted Command Region	Destination value allows precise c
Command	Captured	Captured Command Region	Used to trigger effects. Allows de

Automation Destinations

All *automation regions* have the concept of a **destination**. This is essentially the *output* of the automation track. Note the difference with other track types: in, say, an instrument track, the output is controlled by adding cables on the output side (i.e. at the bottom) of that track's track panel. Assuming you don't add or remove cables during your performance, the output does not change. On the other hand, *each* captured automation region can output to a *different destination*. This **destination** is always the first text field that appears on the automation region (unless the region is so small that all of its controls are hidden, in which case zooming in will reveal them). Clicking on the destination field reveals a popup-menu which allows you to choose where to send the learned inputs for the **duration of the region**.



For example, in the second image down there are two captured command regions side by side. During a performance (i.e. when recording) or during playback, when the current position in the song is inside the first region, pressing the *Bass* track's *ENV1 Trigger* will be set to *Gate*.

Latch to Last Value vs. Revert to Initial Value

Upon exiting an *automation region*, the current value of the region's target can either **latch to the last given value** while inside this region, or **revert to the initial value prior to entering the region**.

Both modes have their use; for example the temporary control of an effect may be required in the chorus of a song. Once the next verse is entered, the effect will jump back to the previous value before it was manipulated in the chorus. This is achieved by setting the automation region to *Revert to Last Value*.

Captured Automation Regions

Captured Automation Regions are the analogue of Record Loops for automation tracks, in that their contents will be filled out during your performance, rather than when you create your arrangement. Similar to ALK's handling of record loops during a performance, any input (from control sources that are learned) is routed to the appropriate automation destination *during the region* in question.

Furthermore, you can create any number of regions in a single automation track, and each can send its output to a different destination – even if the destination is on a different track or plugin. This allows for much more efficient use of a limited number of physical MIDI controls during a song or set.

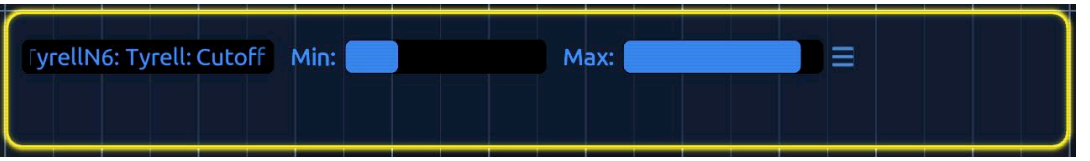
Tip

Maximize your hardware! A controller may have only four rotary knobs but each one can be assigned to a different target at different points in time, allowing complex performance control with even the most limited of hardware controllers.

Captured Control Regions

Captured Control Regions are used for *live* control of continuous targets, such as track volume or plugin parameters. They take input from the *learned MIDI inputs* of their *Control Tracks* (eg: MIDI controller fader CC12) and output the data to the target assigned within the region (eg: volume, synth filter cutoff etc.). The minimum and

maximum values of the control can be set inside the region which allows a sweet spot of control to be honed in on. These values can also be be reversed which allows physical controls to be flipped. For example, pulling a physical fader down could now turn up the volume of a track.

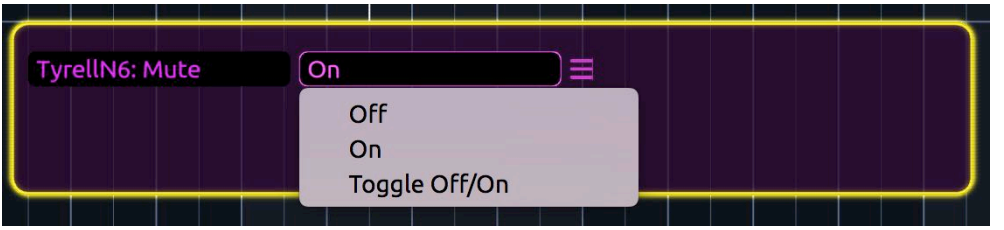


Note

The min/max sliders only become visable when zoomed in to a reasonable degree. If they are not visable, increase the level of zoom.

Captured Command Regions

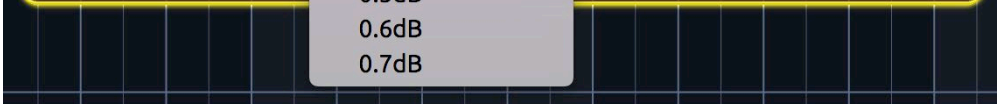
Captured Command Regions are used for *live* triggering of fixed values, such as on/off or delivering set values such as a triggering a tempo BPM change. They are triggered by the *learned MIDI inputs* of their *Command Tracks* (eg: MIDI controller button or pedal CC9) and pass that trigger onto to the target assigned within the region (eg: power, effect on/off etc.).



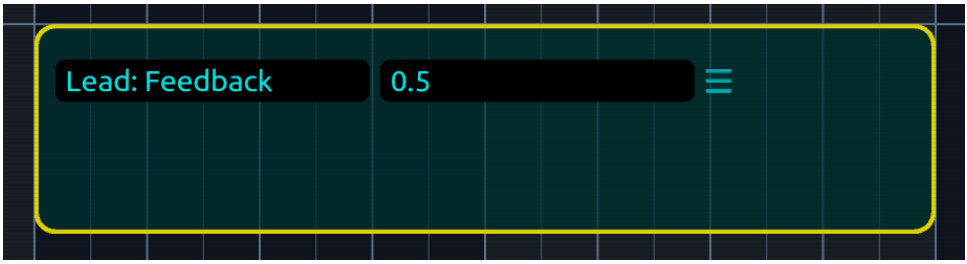
A target destination with a value of *on* or *off* can be triggered in the ways seen in the above image.

- **Off**: Triggering within the region sends an *off* message to the target.
- **On**: Triggering within the region sends an *on* message to the target.
- **Toggle Off/On**: Triggering within the region sends consecutive *on* and *off* messages to the target.

A target destination such as *Volume* provides a choice of values which can be triggered. They can be accessed by clicking the value box which opens a drop-down menu.

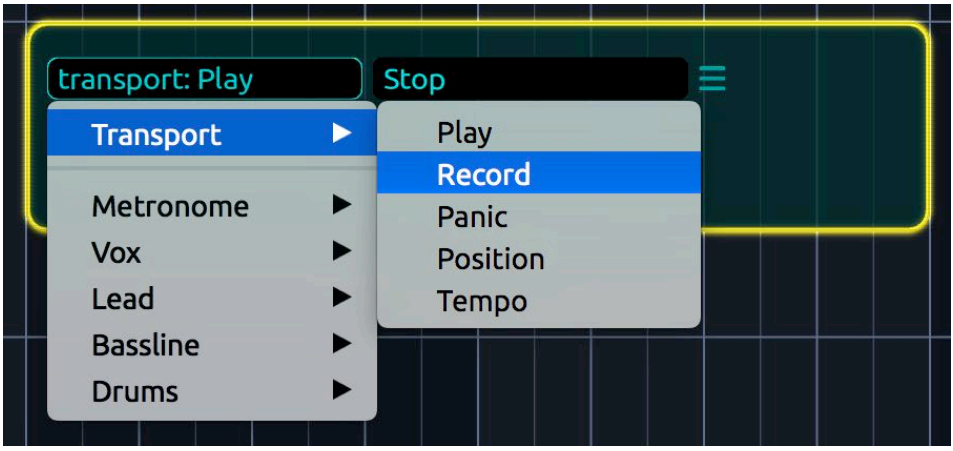


Other continuous values such as plugin controls can be triggered by entering a value between 0 and 1 into the *Command region*. For example, entering "0.5" into a *Command region* targetting the "Feedback" parameter of a delay plugin will set the parameter at 50%.



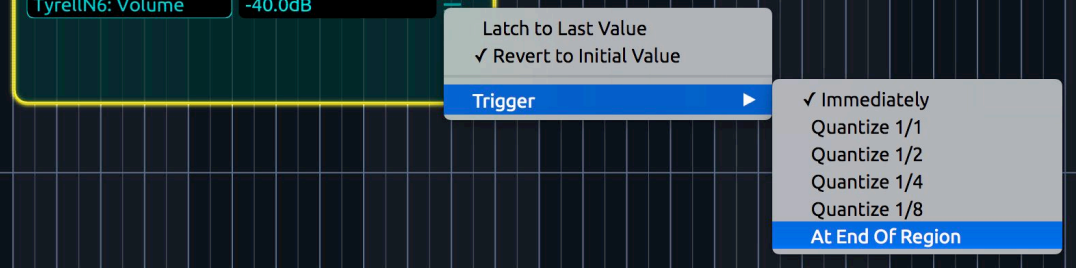
Command regions can also be used to send **Transport Commands**, allowing major first degree functions of *ALK* to be controlled automatically or with physical controls.

- **Play**: Triggers playback.
- **Record**: Triggers Recording *(Performance)*.
- **Panic**: Triggers the Panic Mode.
- **Position**: Jumps to the *bar number* set in the *region value*.
- **Tempo**: Jumps to the *tempo* set in the *region value*.



Every *Captured Command Region* has its own trigger mode. When the *Learned MIDI input* for the *Command track* is triggered, the following happens based on each setting:

- **Immediately**: Triggering within the region is sent to the target immediately.
- **Quantize**: Triggering within the region is sent to the target at the end of the chosen bar measure (1/1, 1/2, 1/4 or 1/8).
- **At End Of Region**: Triggering within the region is sent to the target at the end of the *Captured Command Region*.

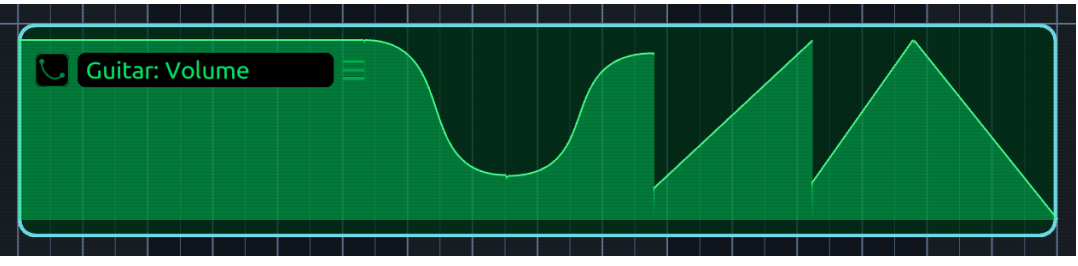


Scripted Automation Regions

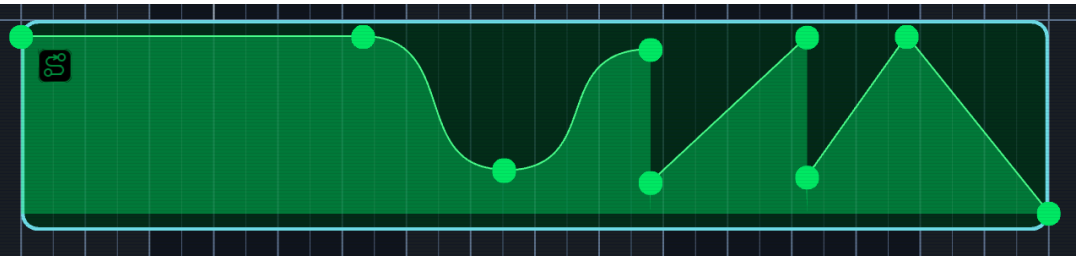
Scripted Automation Regions differ from *Captured Control Regions* in that they require no MIDI input source. Instead **the input is derived from the region itself**. This comes in two forms: *automation triggers* and *automation curves*. These pre-programmed regions allow a vast array of functions in *ALK* to be automated. Lets take a look:

Scripted Control Regions

As with *Captured Control Regions*, *Scripted Control Regions* deal with a continuous and changing value. They do this through the use of an *automation curve* which looks like this:



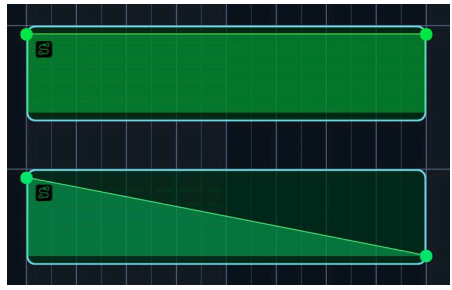
In the example above, the target is the *Volume* of our *Guitar* track. The height of the *automation curve* controls the value sent to this target across time. Clicking on the left hand *Edit Curve* buttton opens up the region for curve editing:

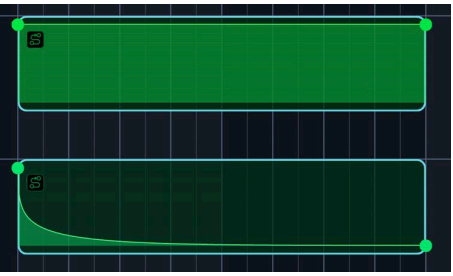


Editing an automation curve can be done in a number of ways. Once the region is open, clicking on the area inside creates *automation nodes*. Nodes can be moved by

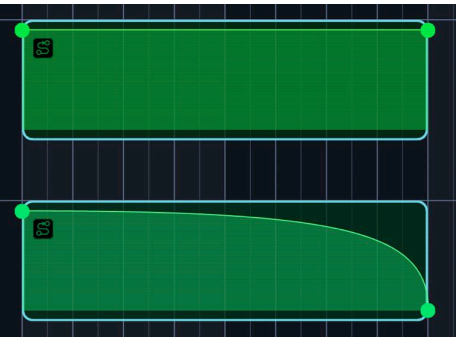
simply clicking and dragging them the mouse, and then deleted by clicking on them. There are also a number of *trackpad gestures* which can be used to generate more complex curves:

cmd + scroll down: Hovering over the mid point between two nodes, holding the *cmd* key, and scrolling down with two fingers on the trackpad moves down the whole section between the nodes. Scrolling down nearer to one of the nodes moves only that node, enabling *diagonals* to be drawn.

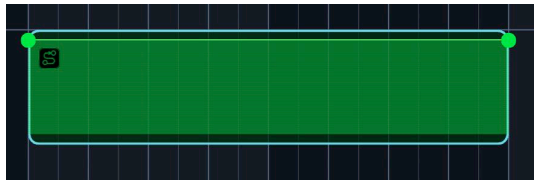




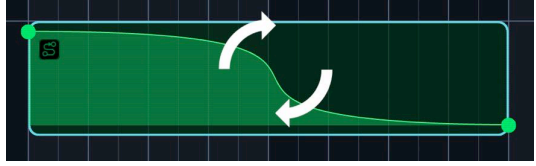
cmd + scroll left/right: Hovering over a diagonal, holding the *cmd* key, and scrolling left or right with two fingers on the trackpad changes the shape of the diagonal to convex or concave.



cmd + rotate: Hovering over a diagonal, holding the *cmd* key, and making a rotate gesture with two fingers on the trackpad changes the shape of the



trackpad changes the shape of the diagonal between straight and S-shaped.



Scripted Command Regions

Scripted Command Regions deal with on/off messages or trigger fixed values. As an example, they can be useful for triggering the *power* function (mute) of a Metronome, allowing it to be silenced after a specific point in the performance.

As explained above, *scripted automation* is used to pre-program changes in your performance that you would rather not control live. In this sense, *scripted command regions* could be used to automatically turn on and off the pedals of a virtual pedalboard, such as what is found in most guitar effects plugins. This allows the musician to truly focus on their instrumentation.

Panic Mode

So you messed up, what now?

Loopers have been a great addition to the performing musician's live set up, and have allowed complex multi-layered arrangements to be created by one or two people. However, a problem arises in the permanence of these recordings when a musician makes a mistake. Any non-looping musician can easily brush over a timing mistake or a dud note and carry on with their song. A looping musician is stuck with their mistake for as long as the loop is repeating.

Once a mistake is made, the looping musician is generally left with three options:

- **Continue**: They continue the song, brushing over the mistake. However the audience hears it, they hear it, and it loops for the entirety of the performance.
- **Undo**: With some modern loopers, there is the option to *undo* with a simple foot press. The musician is given a second chance to replay the part before continuing with the rest of the song. However, this disturbs the structure of the song and the audience clearly sees they messed up.
- **Restart**: Probably the least favorable option. In a worst case scenario when neither of the above are viable options, the musician stops the song and restarts over. Unfortunately, this is usually what happens when things go wrong in a live looping gig.

There Is Another Way…

The problem with the above options is that none of them allow the musician to continue undisturbed and keep their *flow*. Continuing with a badly played layer can easily psych them out on stage, as can having to undo and then replay their part. Stopping is simply not an option at a gig. So what now?

The fourth option: trigger the **Panic Mode**.

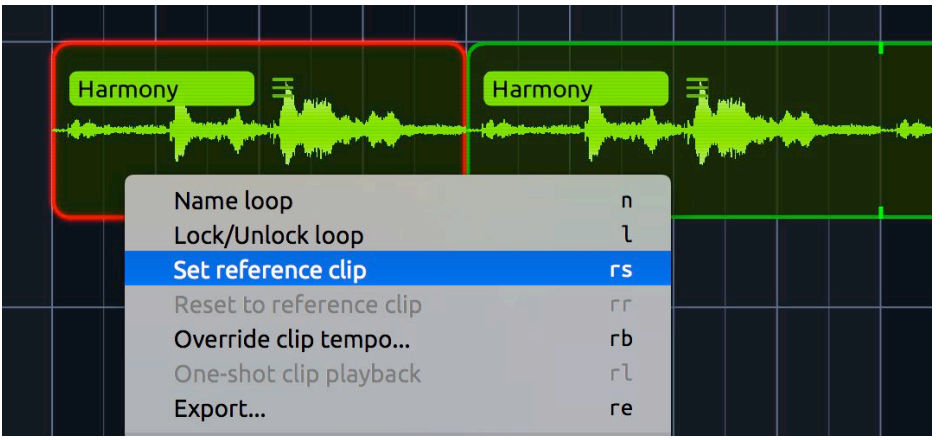
Imagine you've messed up a one of your guitar lines, and instead of options 1-3, you simply hit a pedal which instantly inserts a pre-recorded perfectly played *reference clip*, performed by yours truly at a time before the gig. This allows for the song to continue

completely undisturbed. You keep your *flow*, and the audience doesn't know any better.

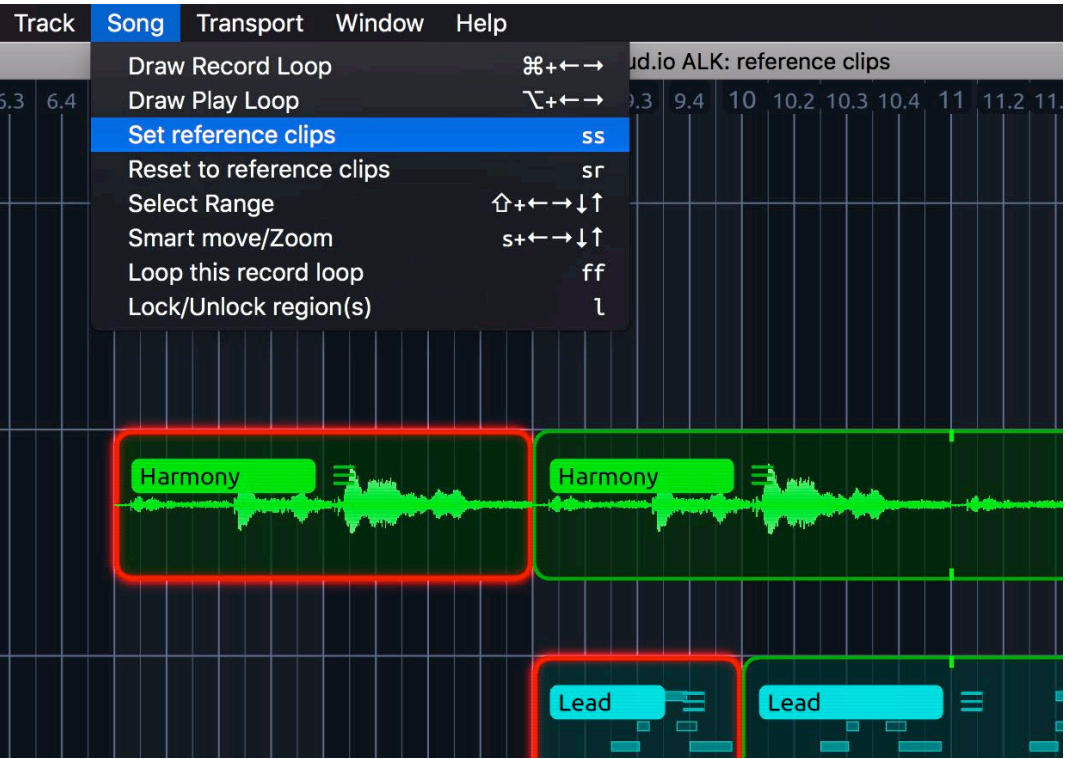
ALK's Panic Mode can deliver exactly that.

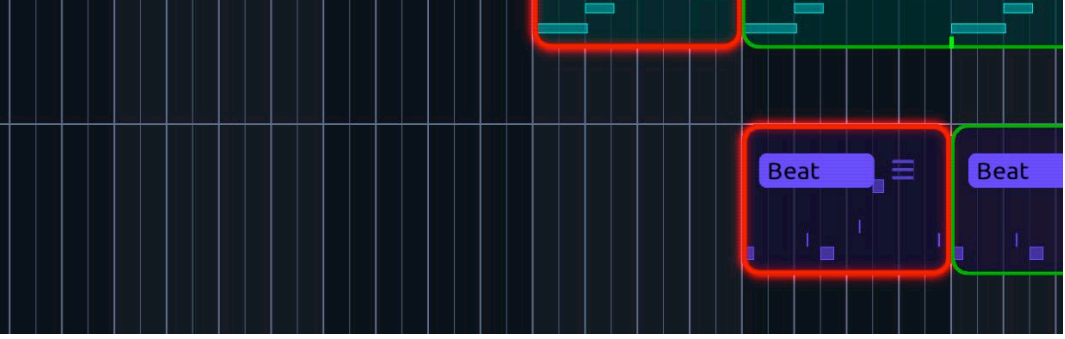
Reference Clips

In order to set up *ALK* for to use **Panic Mode** we first need to set the *reference clips* of the song. *Reference clips* are essentially the perfectly played versions of each audio, instrument and MIDI record loop. Once the performance of a record loop is to a satisfactory level, the reference clip can be set by right-clicking on the loop and clicking *Set reference clip*:



Multiple *reference clips* can be set at the same time by selecting the specified record loops, then choosing *Set reference clips* from the *Song menu*





The last eight clips of a record loop (your performance takes) along with the reference clip can be viewed by right-clicking the loop to reveal the loop menu. After performing new takes of the song, the original reference clips can be restored at any time by:

- right-clicking a record loop and choosing *reset to reference clip*
- selecting the appropriate record loops and choosing *Reset to reference clips* from the *Song menu*

Tip

Quickly set or reset all reference clips by pressing "AA" to *Select ALL* and then choosing *Set/Reset to reference clips* from the *Song menu*.

Panic Mode Trigger Setup

Now that satisfactory clips have been set as the reference clips of the song, we need to determine how to trigger the quick replacement of our bad performance takes with these clips. As explained in the Automation section, the triggering of functions with physical triggers (such as a MIDI foot pedal or button) is performed with Captured Command Regions. This is how a physical controller can be used to trigger the *Panic Mode*.

The setup is as follows:

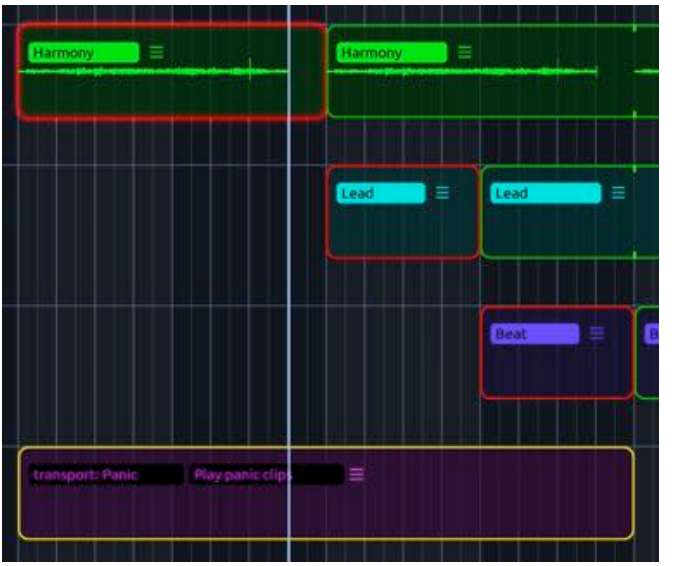
1. Create a Command Track.
2. In the track's *Track Panel* click Learn and then press the desired pedal or button on the MIDI controller. This will become the trigger for the *Panic Mode*.
3. Draw a Captured Command Region for the length of time that is desired for the function to be available. If you require it for a whole song, make the region as long as the song.
4. Set the region to target Transport -> Panic Mode, by clicking on region's *target box*.

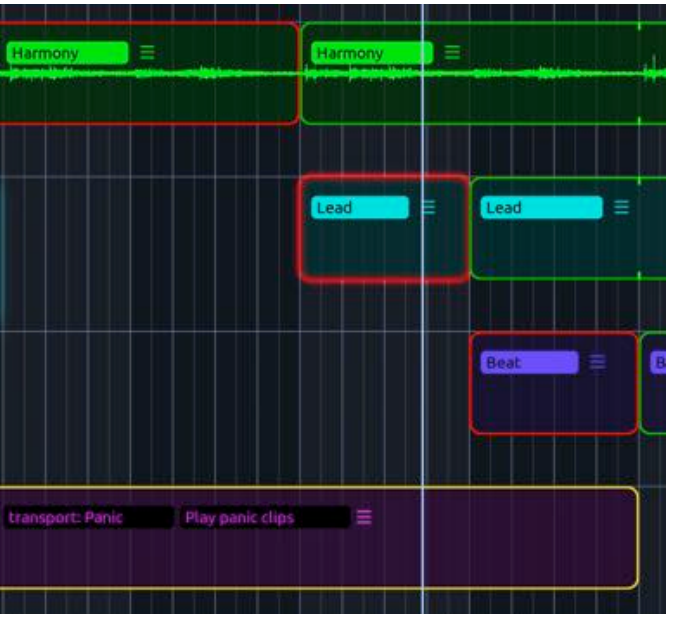
Panic Mode In Performance

At this point, the *Panic Mode* is set up for use. While in a record (performance) state, triggering *Panic* resets the current clip to the *reference clip*. Below is a scenario

explaining what happens when the Panic Mode is triggered at different times:

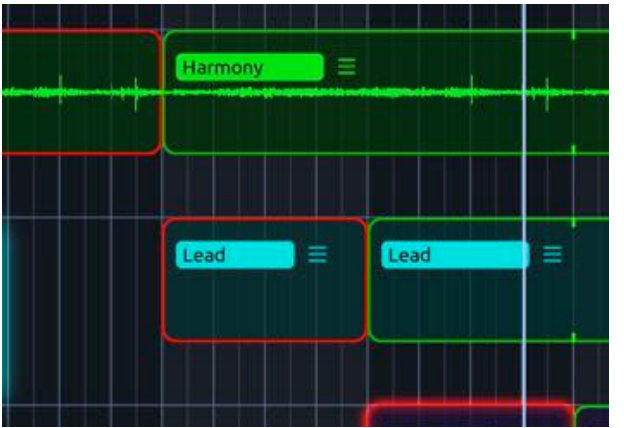
Hitting the pedal here inserts the reference clip of “Harmony”.





Hitting the pedal here inserts the reference clip of “Lead”.

Hitting the pedal here inserts the reference clip of “Beat”.



In simple terms, any time a loop is performed incorrectly, hitting the trigger inside that loop will instantaneously insert the reference clip, allowing you to freely move on from any mistakes without sacrificing any part of the song.

Importing And Exporting

ALK allows for the importing and exporting of both *audio* and *MIDI*. Pre-recorded material can easily be dropped into parts where only playback is required. Performance data such as audio and MIDI from record loops can also be exported for use in other audio applications and in other DAWs.

ALK supports *import* of the following file formats:

- **WAV** (audio)
- **MP3** (audio)
- **AIFF** (audio)
- **FLAC** (audio)
- **OGG** (audio)
- **MID** (MIDI)

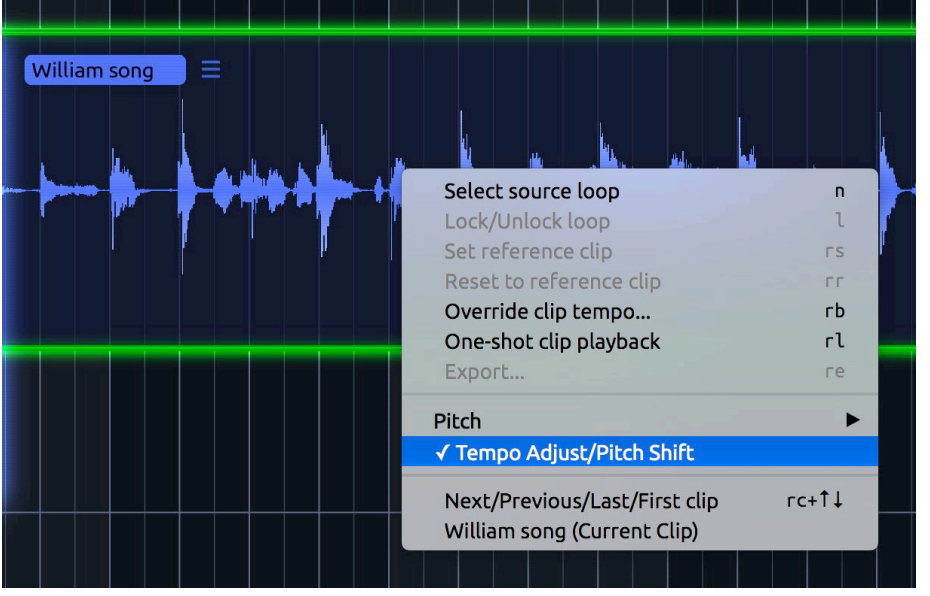
The following file formats are supported for *export*:

- **WAV** (audio)
- **MID** (MIDI)

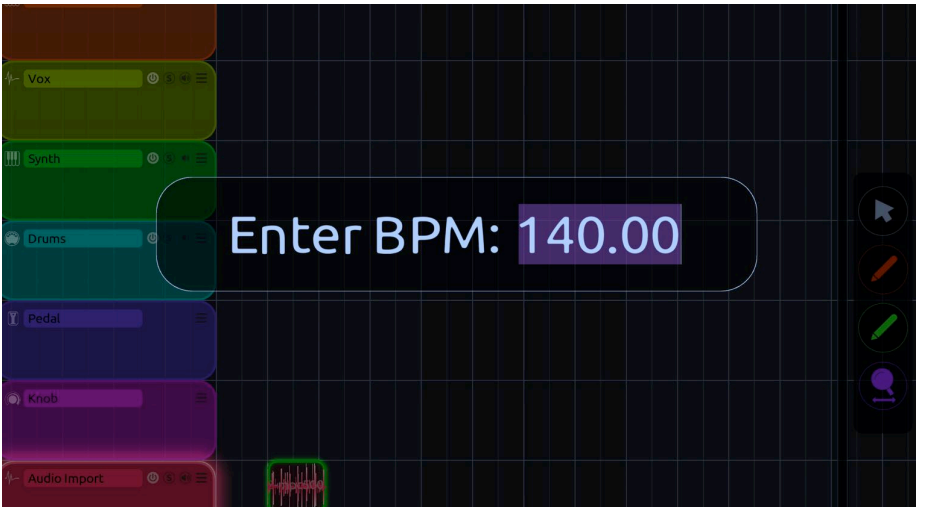
Importing

Audio and MIDI files can be imported in two ways. A file can simply be dropped into the application window from the file browser of the operating system, or can be imported via the top menu *File -> Import*. In both cases, importing a new file creates a new audio or MIDI track depending on the file type. Importing multiple files at the same time creates multiple tracks.

When importing *audio*, *ALK* **automatically detects and shifts the tempo of the audio content** to match the *tempo* of the session. This can be disabled by right clicking the audio region and unticking *Tempo Adjust/Pitch Shift*.



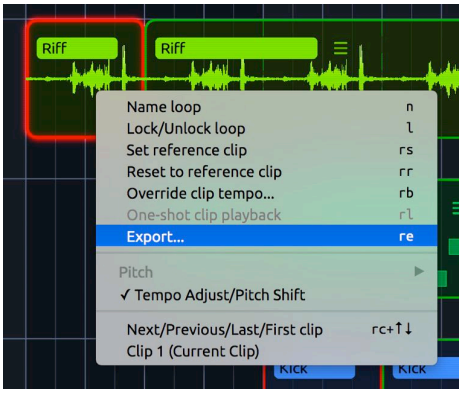
The **tempo can also be overridden** with a specified tempo by choosing *Override clip tempo* in the same menu. Clicking this option displays the following dialog where another tempo value can be entered:

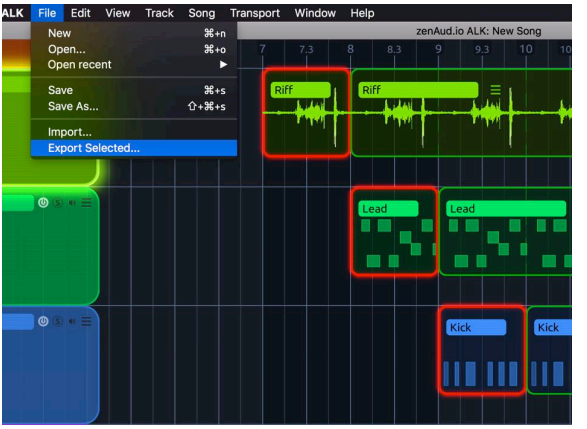


An imported file in *ALK* behaves much like a play loop in that clicking the end of the loop and dragging to extend it allows the contents to be looped. Below is a short audio file that has been extended into a loop. This function can also be disabled by toggling *One-shot clip playback* in the loop's menu.

Exporting

Individual audio and MIDI loops can be exported from *ALK* simply by clicking *Export* from the loop's right-click menu. Multiple loops can be exported at the same time by selecting the desired loops and choosing *Export Selected* from the *File* menu.

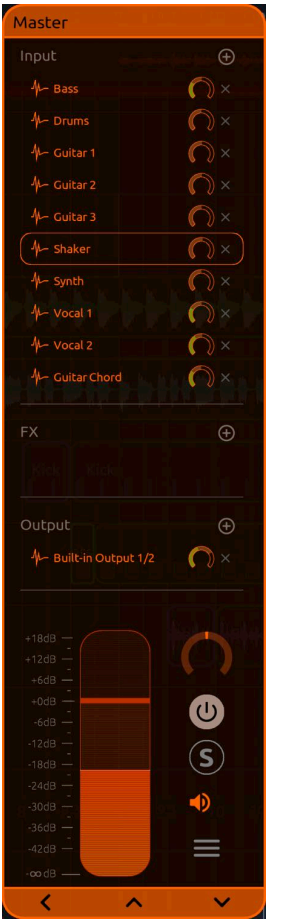




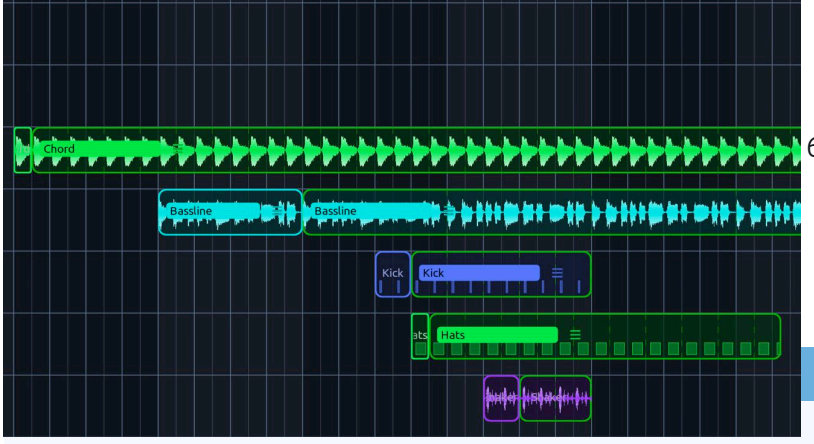
Exporting A Mixdown

Once you're satisified with the performance of a song you may wish to create a *mixdown* to a stereo .WAV file in order to share your performance. Below are the steps to create a mixdown:

1. Create an audio track and open it's track panel.
2. Connect all instrument and audio tracks in the session to this track's input terminal.
3. On this track, draw a record loop spanning the entire song.



4. Lock all record loops in the session except the loop being used for the mixdown.
5. Hit Record. This mixdown



loop is now making a stereo recording of the song.

6. When recording has finished, right-click the mixdown loop and choose *Export*.

Note

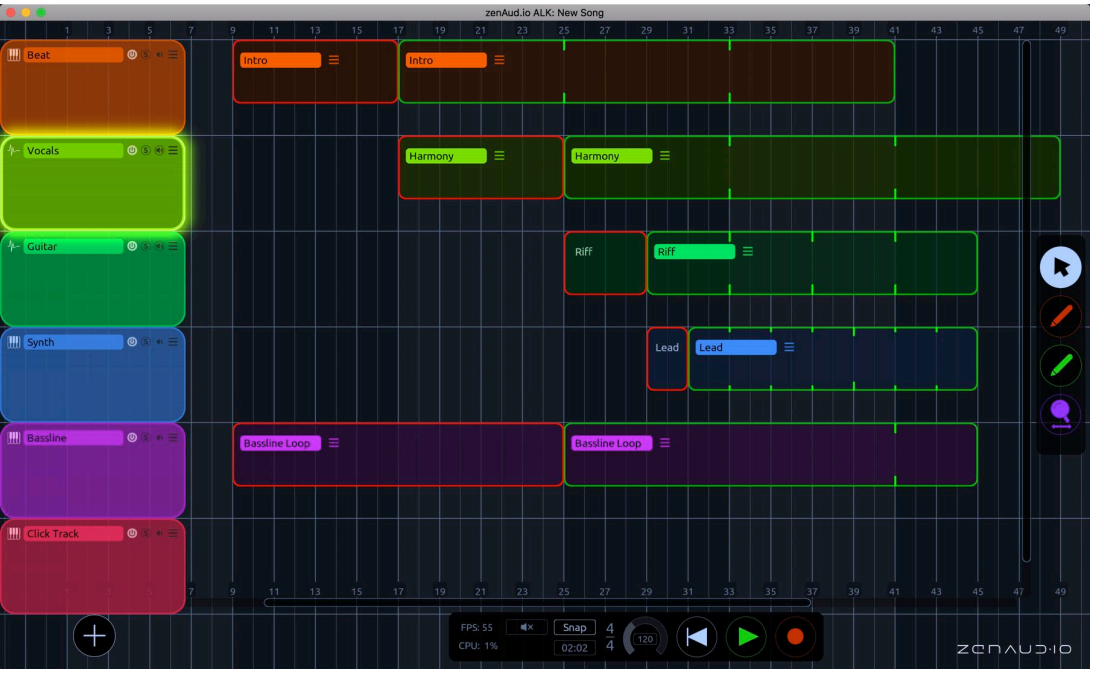
The levels of the individual inputs may need to be adjusted on the input terminal to avoid **clipping**.

User Interface

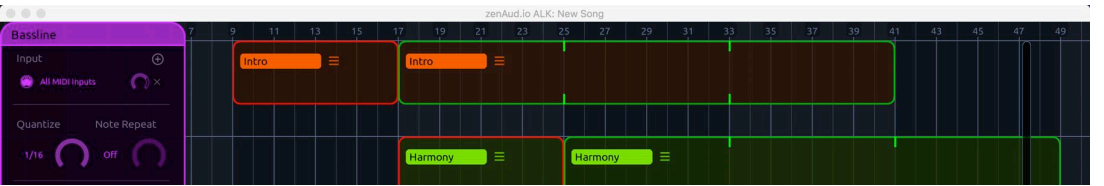
Before we delve into the details, let's take a step back and look at the overall interface.

The ALK user interface contains the following basic elements:

Arrangement Window: This window houses the current ALK session, displaying the *loop arrangement*, *transport bar*, *toolbar* and track leaders (which give access to the various track panels).



Plugin Editor: Clicking on any VST/AU plugin within the track panel opens the plugin's user interface in a new window where its settings can be edited.



The document follows a hierarchical structure: *songs* contain tracks, each *track* contains loops, while its track panel contains *inputs*, *outputs*, *plugin instruments* and *plugin effects*, and so on.

Arrangement View

The arrangement view is the main view in ALK. Before we enumerate the main parts of the interface, note that the arrangement view is essentially a stretchable *Surface* that allows a look at the song at arbitrary levels of detail. You can *Zoom* in on a part of the document by using the *Pinch Gesture* and scroll using the *Scroll Gesture* or using the mouse.

Below is an annotated screenshot of ALK's arrangement view. In the arrangement view, you can:

- manipulate the *Transport*, allowing you to control the *tempo* and *length* of the song as well as the standard play/record buttons;
- create and delete tracks, and modify their:
  - volume/power/solo/… functionality: these can be controlled in the Track Leader widget, and in the Track Panel.
  - inputs and outputs: these are accessible in the *Inputs* and *Outputs* sub-panels of the track's Track Panel;
  - add and remove plugins: VST or AU instrument and FX plugins can be loaded in the *Instrument* and *FX* sub-panels of the Track Panel.
- create and delete loops, and modify which record loops the play loops will link to.

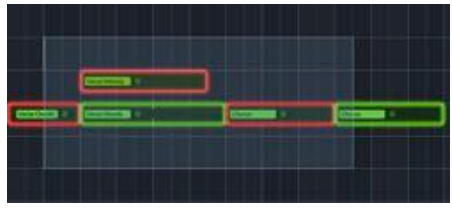
Below is an annotated screenshot of the ALK's arrangement view.

We shall now go into more detail on each of the image's annoted elements:

Drawing Modes

The toggle buttons in this panel modify user actions on the main *Surface*.

- *Select Mode*: in this mode, you can select groups of loops, by pressing and dragging the mouse over the loops you want to select. You can add to a selection by pressing *Shift* while drag-selecting. Loops can be unselected by hitting the *Esc* key.



Selecting using *Select Mode*

- *Record Pencil Mode*: in this mode you can draw record loops. You can also enable this mode by pressing and holding down the *Cmd* key while clicking and dragging.
- *Play Pencil Mode*: in this mode you can draw play loops. You can also enable this mode by pressing and holding down the *Alt/Opt* key while carrying out the draw operation.
- *Zoom Mode*: this mode you to zoom in and out, both vertically (making the individual tracks larger or smaller) and horizontally (zooming in and out on the time axis). You can enable this mode by pressing and holding down the *E* key and dragging vertically for time zooming.

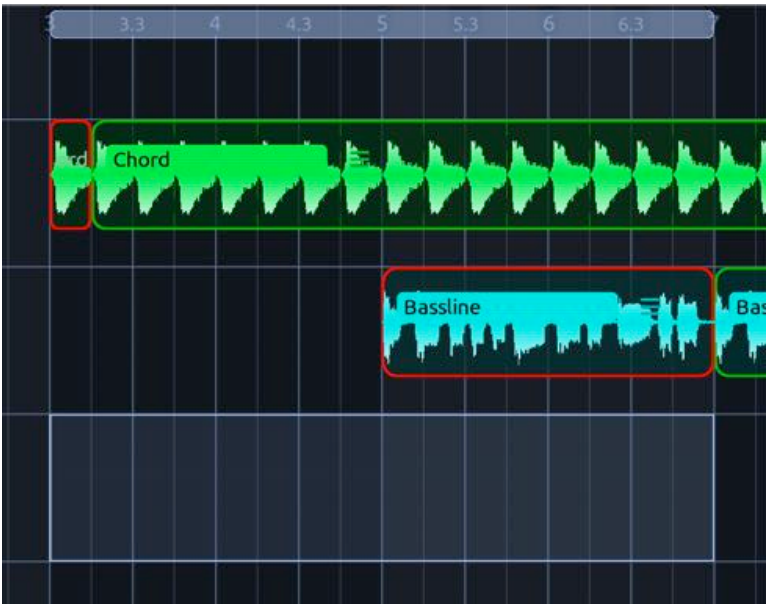
Transport

This panel contains controls that dictate the tempo and length of the song, as well as buttons to engage *Record* mode – that is, to perform the song – or *Play* mode – to playback the preview performance). Finally, there is the *Rewind Button*, which brings the playhead back to the start.

- *Track Length Control*: the track length control on the left hand side of the of the *Transport* dictates the length of the song *in minutes and seconds*. Songs default to length 00:30 (i.e. 30 seconds). Clicking inside this control allows you to change the length through typing.
- *Tempo Rotary*: the tempo rotary allows you to change the tempo of the song. Note that that the tempo of a song cannot be changed while recording; only in *Play Mode* (or *Stop Mode*). The tempo can be changed via mouse and scroll gestures, as well as with the keyboard by clicking on the BPM text field. - *Time Signature*: the time signature determines the number of beats in each bar (top) and the note value equivalent to one beat. Both values can be changed by clicking the appropriate number.
- *Snap*: switching off the *Snap* function allows for free drawing, editing, resizing and moving without being locked to the grid. By default this option is left on.
- *Silence MIDI Notes*: Clicking here silences all currently playing MIDI notes stops playback. Handy for killing occasional stuck notes.
- *FPS*: displays the current visual frame-rate per second
- *CPU*: displays the current usage of the computer's processor as a percentage

Global Loop

Global loop is used for continuously repeating a section of the song. It can be activated by selecting an area of the *arrangement* and pressing "GG" or choosing *Toggle Global Loop* from the *Transport Menu*.



Global loop can also be set to repeat any loop in the arrangement by clicking the loop and pressing "GG" or choosing *Toggle Global Loop* from the *Transport Menu*.

The *global loop* can be deactivated by pressing Esc, pressing "GG", clicking the global

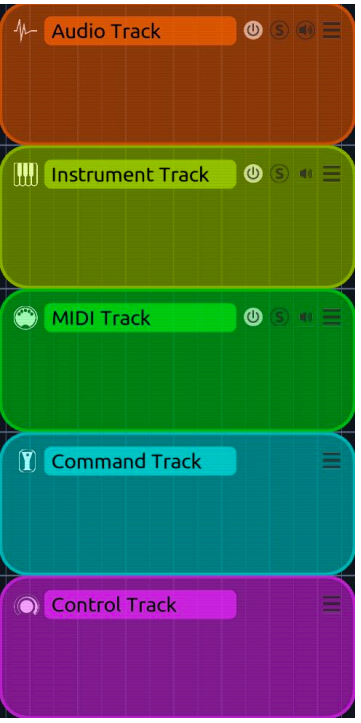
loop or choosing *Toggle Global Loop* from the *Transport Menu*.

Add Track Button

Use this button to create new tracks. Clicking on the button pops up a menu on which you choose one of the 5 track types available in ALK: *Audio*, *Instrument*, *MIDI*, *Command* and *Control*. As more tracks are available in ALK, they will appear here.

Track Leaders

Track leaders give you quick access to some of the most commonly used functions for tracks. The track leader has the following elements, from left to right:



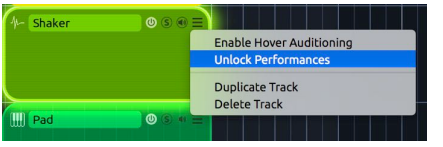
Track Type Icon indicates the Track Type, and is not interactive in any way

Track Name Text Field displays the track name. To edit a track name, click within the text field, then type to change the now-selected text.

Power Button: The power button controls the overall state of the track in question. If the power is off for a track, no signal comes out of a track. It behaves in the opposite way as a regular mixer's *mute* function.

Solo: The solo button behaves exactly the same as its analogue in the physical mixing world. Pressing solo allows a track to be heard unaccompanied.

Audition On` (button/icon): controls – in the *Stop* and *Play* modes – whether the live input to a track is heard for monitoring purposes. (During *Record* mode, record loops are automatically monitored, so you always hear what you are playing.) Note that while *Hover Auditioning* is enabled, this button becomes an non-interactive icon, since the audition state is now controlled by the mouse location rather than the manual state of a button.



Options Menu: has the following items:

Disable/Enable Hover Auditioning: when hover auditioning is enabled, a track is auditioned by hovering over it. When hover auditioning is disabled (and we are not recording), the *Audition On* turns from an icon into a button which can be toggled on or off.

Lock Performances: this function temporarily locks all record loops on the track. In this mode, entering a record state will not overwrite the current clips. This can be

this mode, entering a record state will not overwrite the current clips. This can be useful for isolating and practicing difficult areas of a song, without having to play the neighbouring loops each time.

A record loop in **unlocked state**.

A record loop in **locked state**. Locked loops are assigned the color of the track.

Duplicate Track: clicking on this duplicates the track, its settings and all its loops.

Delete Track: clicking on this pops up a confirmation dialog, asking if you're really sure you want to delete the track. If you click "Delete" on this dialog, the track will be deleted.

Keyboard Commands

ALK contains many keyboard shortcuts to help you speed up your workflow. ALK key commands assume US keyboard key locations so the position of commands remains fixed with international keyboards (ie: On a German keyboard, Z to undo becomes Y to undo).

Transport Commands

Key	Function	Notes
<Spacebar>	Play/Stop	
<Enter>	Record	
r	Rewind	Cycles: rewind to beginning of loop/beginning of song.
cmd+,	Open Preferences	
fq	Snap on/off	Toggles on/off snap to grid
b+↓↑←→	Change tempo	Changes tempo by +1/-1 (↓↑) and +10/-10 (←→)

Movement Commands

Key	Function
←	Move left
→	Move right
↑	Move up
↓	Move down
E+←→	Smart move

Primary Gestures

For maximum ergonomics, zenAud.io ALK uses modifier keys to allow quick navigation and operation of useful functions, enabling a speedy workflow.

Modifier	Trackpad	Keyboard	Function
cmd	Pinch in and out	↓↑	Zoom in time axis.
cmd	Drag horizontally	← →	Draw record loop.
alt (⌥)	Pinch in and out	↓↑	Zoom in track axis.
alt (⌥)	Drag horizontally	← →	Draw play loop.
cmd+alt		↓↑	Zoom in time and track axis.
E	Drag vertically		Zoom in time axis.
E	Drag horizontally		Scroll in time.
E		↓↑	Zoom in time and track axis.

Copy, Paste, and History Commands

Key	Function
z	Undo
⇧+z	Redo
x	Cut
c	Copy

v	Paste

Selection Commands

Key	Function	Notes
a	Select All	Cycles between selecting all in song, all in track, and none.
⇧+←	Select left	
⇧+→	Select right	
⇧+↑	Select up	
⇧+↓	Select down	

Track Commands

Key	Function	Notes
g+↓↑	Track Gain	After pressing this key, use movement keys to set volume
l	Lock	Locks loop if mouse/caret hovers over, otherwise locks tr
ts	Solo	Toggles solo of current track
tp	Power	Toggles power of current track
tna	Create audio track	
tni	Create instrument track	
tnm	Create MIDI track	
tnc	Create control track	
tnt	Create command track	
ta	Audition	Cycles: audition off, audition on, hover auditioning.
tt	Toggle Track Panel	
t+↓↑	Change Track Panel	
tii	Open instrument editor	
ss	Set reference clips	Sets the currently selected clips as reference clips

sr	Reset to reference clips	Resets the currently selected clips to the reference clips
p+⌫	Empty clip	Requires selection or hover of record loop
tr	Rename track	
tv	Duplicate track	
t+⌫	Delete track	
t+Del	Delete track	

Play Loop Commands

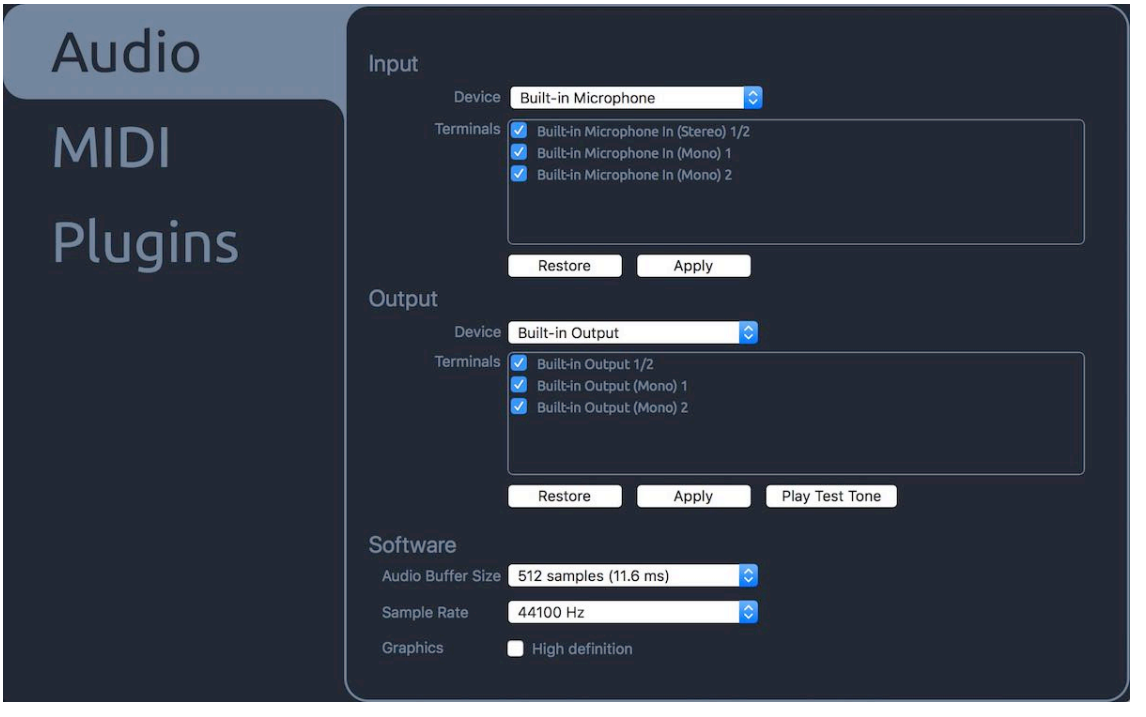
Mouse must be hovering over a play loop.

Key	Trackpad	Function
cmd	Scroll up/down	Adjusts the pitch shift in semitones.
cmd	Scroll left/right	Adjusts the offset in bars, beats and beat fractions.
ff	Hover over/after loop	Extend loop to length of arrangement.

Preferences

Preferences can be accessed via the "*cmd+,*" key command, or in the *ALK->Preferences…* menu item.

Audio Preferences



The audio preferences pane encapsulates settings related to the audio interface and audio callback buffer size. For both inputs and outputs, you can choose from the following settings:

- *Device*: this combo box allows you to choose which audio interface is used for input purposes.
- *Terminals*: this list view allows you view, and modify the names for the set of terminals detected for the current *device*. Name changes for terminals are reflected

elsewhere in the user interface. For example, if you change the name of "Input 1" on, say, your *RME Fireface* sound card to "Roby's Guitar DI", it will be much clearer and simpler to select inputs correctly for your guitar tracks.

- *Restore Button*: pressing this button returns the audio system back to the default state: each terminal is restored to its original name.

- *Audio Buffer Size*: an audio host such as *ALK* works by processing audio in chunks; so-called *audio frames*. The size of the buffer impacts ALK in two ways:
  - **latency**: as the buffer size increases, the latency of the entire audio system increases commensurately. If the buffer size is too high, there could be a noticeable delay between the original and monitored signal.
  - **load on the system**: as the buffer size decreases, the load on the CPU is increased. This is because the buffer – which is now smaller and thus represents less time – needs to now be filled more often, and this in turns increases the CPU load. If the load on the audio thread is too high (for example, if you are running many or CPU-hungry audio plugins), you will hear stuttering and glitches. If stuttering or dropouts become a problem and you're not able to increase the buffer size (e.g. because the latency becomes unplayable), then you must reduce the CPU load in the audio thread by removing instances of certain plugins, or even tracks.

- *Sample Rate*: Here the sample rate can be changed to any rate supported by a connected audio interface.
- *Graphics*: This check box switches on and off graphical glow FX and limits the frame rate to 30 FPS. Uncheck this option if continual CPU problems are encountered.

Tip

It's a good idea to experiment with the Audio Buffer Size in your program. Your goal during experimenting should be to dial down the buffer size as much as possible *without* inducing stuttering or glitching in a project *somewhat larger and heavier* than the typical project you use. This way, you still have the proverbial extra headroom if your track needs that extra gear change!

MIDI Preferences

The MIDI

device list displays the currently detected MIDI hardware devices (such as USB keyboards or drum pads). For each detected device, a row appears in the list view. Click on the device name to change its name (for example, changing “nanoPAD2 PAD” to “Ray’s Drums Pads”).

In addition, you can disable or enable the MIDI device by clicking on the check box on the device row. If a MIDI device is disabled, you will not receive any input from the device.

Plugins

The plugins pane allows you to **scan** for VST/AU plugins, based on a list of **search directories**. The scanning process looks at each search path in turn, so if the same plugin appears in two search paths, the plugin in the *first* search path will be used in *ALK*.

- *Plugin Directories* - this list view displays the directories that will be searched for plugins when you click Rescan. The controls below the this list view are, from left to right:
  - *Add Directory* (‘+’) - this adds a new place to search for plugins;
  - *Remove Directory* (‘-’) - this deletes the currently selected directory from the list;
  - *Increase Priority* (**up arrow**) - this moves the directory up on the list;
  - *Decrease Priority* (**up arrow**) - this moves the directory down the list;
  - *Restore Defaults* - pressing this button causes the search paths to be reset to the default AU and VST search paths on *macOS*.
- *Plugins* - this area consists of:

- a table view which displays the currently detected VST and AU plugins, along with some of their properties, such as *Plugin Vendor*, plugin type (*Instrument* or *FX*), etc.

- *Rescan* - this button causes *ALK* to scan for plugins, which is described in more detail below.

Scanning For Plugins

When you click Rescan, *ALK* begins traversing the plugin directories displayed above. When it finds a file that looks like it could be a plugin, it tries to load the plugin. If successful, the plugin is added to the list of detected plugins (note that this is only reflected in the listview when the entire scan operation has completed.

Scanning is a process that can take a while, so while *ALK* scans for plugins in the background, you can continue using the program. Scanning progress is displayed in a panel showing the plugin currently being scanned, and the overall progress.

Plugin scanning progress.

ALK failed to load “Tassman VST Effects” plugin.