

S *snap* **heap**

OPERATOR'S MANUAL

Table of contents

Glossary	3
System requirements	4
Overview	5
User interface	5
Signal path.....	6
Operating the controls	6
Working with patches	7
Building your sound.....	8
Lanes	8
Snapins.....	9
Modulation	9
Macro knobs	9
LFOs	10
Envelopes	11
Pitch tracking	12
MIDI	13
Acknowledgements	15

Glossary

DAW

Snap Heap is not a standalone program, it needs to be used in conjunction with a host application where it is loaded as a plugin. These programs are usually called "DAWs" (Digital Audio Workstations) or simply "hosts". There are many different DAWs available on the market, and Snap Heap works with those that support either the VST 2, AAX or the Audio Unit plugin architecture.

Patch

A patch is a configuration of Snap Heap that gives it a certain sound. Whenever you pull a slider or turn a knob you modify the current patch.

Preset

Presets are ready-made patches which ship with Snap Heap. Presets also make good starting points for your own patches. Presets are grouped into banks. More preset banks may be available to download using the Kilohearts installer.

Snapin

Similarly to how a DAW will load Snap Heap as a plugin, Snap Heap itself can also load small audio effect modules called **snapins**. Snapins can be used either directly in your DAW as a VST or Audio Unit, or used as building blocks in a Snap Heap patch.

System requirements

These are the minimum recommended system requirements for running Snap Heap.

CPU

2 GHz or faster

Memory

1 GB or more

Operating System

Windows (7 or newer) or Mac OS X (10.7 or newer)

Software

A DAW supporting VST 2, AAX, or Audio Unit plugin standards.

If you have many instances of Snap Heap running, or use a lot of snapins at the same time in your patch the CPU usage will increase accordingly. Thus, we cannot guarantee that Snap Heap will work flawlessly in all use cases even if your system does meet the minimum recommended system requirements.

Overview

Snap Heap is a modular effect plugin with many uses. At the heart of the plugin is four lanes of modular effect plugins known as **snapins**. Snapins come in many different flavors, and implements many classic effects like distortion or chorus as well as more unusual ones like formant filtering and frequency shifting.

In addition to being loaded into Snap Heap, snapins can also be loaded as plugins directly in your DAW. Snap Heap is bundled with a few different snapins, and more can be bought from the Kilohearts website.

User interface



This is an overview of the plugin UI. It may look a bit overwhelming at first, but let's try to break it down a bit.

1. In the top bar the name of the **current patch** is shown. By clicking the patch name you will open up the **preset browser** from where you can save and load presets.
2. The area below the top bar is the **modulation area**. From here you can modulate parameters of your patch using **Macro knobs**, **LFOs** (low frequency oscillators), **Envelopes** or **MIDI** input. The different modulation options are described in detail below.
3. The big space below the modulation area holds the **snapin lanes**. Each lane can hold several snapins. Adding new snapins is as easy as clicking the empty space at the bottom of a lane.
4. Under each lane, at the very bottom of the window, sits the **lane mixing** controls. Here you can adjust the gain and pan of each lane independently, among other things.

Signal path

The input audio is passed through all of the enabled effect lanes from left to right. Each lane is processed in sequence, unless the "parallel routing" button is clicked. When parallel routing is enabled, the two adjacent lanes will be linked together and processed in parallel instead, and the result mixed together before being passed into the next lane. If multiple parallel routing buttons are enabled, all linked lanes will be processed in parallel.



Operating the controls

Most parameters of Snap Heap are controlled by the knobs and the sliders seen in the UI. To move a knob or slider simply click on it, and while holding the mouse button down move the mouse up or down.

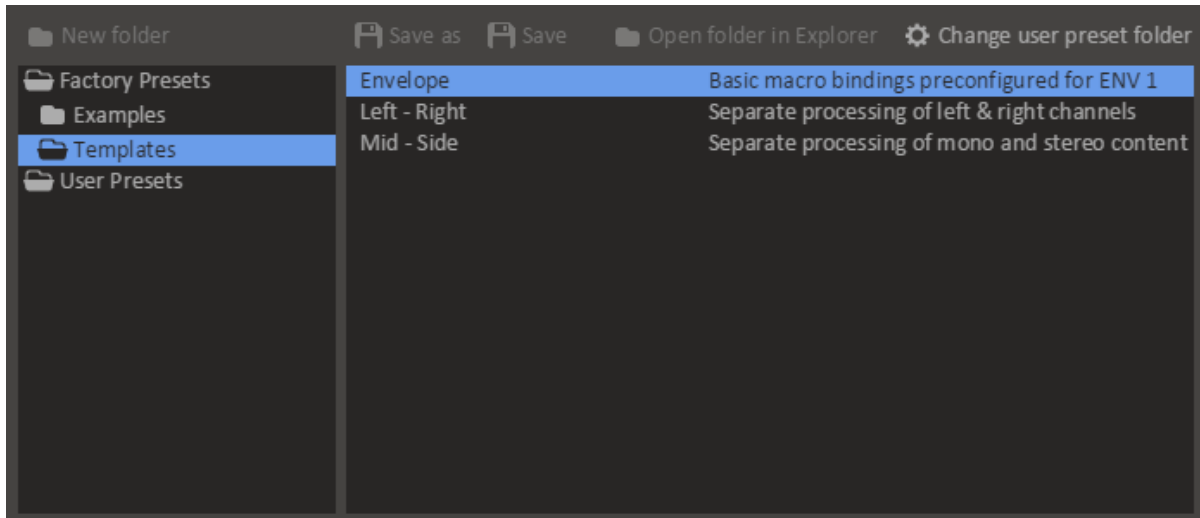
Sometimes you might want more precise control when tuning a parameter. Hold the **shift** key while moving a knob or slider to enter **fine tuning** mode, where the knob or slider will move more slowly.

You can reset a knob to its default position by double clicking it.

Finally, most controls support entering the value using your keyboard by right-clicking on them.

Working with patches

The first thing you want to do after installing Snap Heap is probably to try out some of the presets it comes with. To reveal the preset browser, click on the name of the current patch in the top most part of the UI.



Loading a preset in the preset browser is as easy as clicking on it. The preset browser also allows saving presets, and organizing them into folders. Presets are divided into two categories, **factory presets** and **user preset**.

Factory presets are included with Snap Heap or downloaded using the Kilohearts installer. Factory presets are write protected to protect them from accidental corruption.

The user presets folder is where you as the end user should store your own presets. By default this folder is located in your documents folder, but you are free to relocate it elsewhere.

Building your sound

Once you have checked out some of the presets you are probably eager to get your hands dirty and create your own patches. We designed Snap Heap to be easy and fast to work with, and we hope you will enjoy experimenting and discovering the possibilities. Still, knowledge is power, so in this section we will go through all the features of Snap Heap in detail.

Lanes

Lanes are divided into three parts. The header, where you will find a few toggle buttons, the main part of the lane, which holds the snapins in the lane, and the footer where you will find some mixing controls.

In the lane header you will find the following controls:



Enable button

This toggle button will turn on or off the whole lane.

When the lane is turned off no sound will pass through it at all, and any snapins sitting in it will have no effect. It is advisable that you turn off all lanes that you are not using, to spare some CPU cycles.

Mute button

A toggle switch for muting a lane. Audio will still send audio to the lane, but all effects and lane mixing will be bypassed.

Solo button

This toggle button will mute all other lanes. It is useful during sound design if you temporarily want to listen to the audio that passes through just one lane.

In the lane footer you will find some knobs which impact how the output of the lane is mixed down during lane mixing:

Gain

Changes the volume of the output of the lane.

Pan

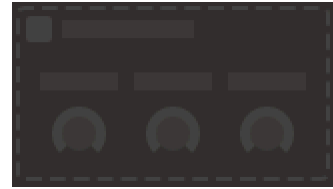
Pans the output of the lane left or right.

Mix

Adjust the output mix of the lane between completely unprocessed at 0% to fully processed with snapins at 100%.

Snapins

To add a new snapin to a lane, click the **add snapin icon** which appears when you hover the empty space in a lane. This will make a dialog appear giving you a selection of all the snapins you have installed. Simply click one to add it to the lane.



Snapins can be reordered or moved between lanes by clicking their title bars and dragging them. If you hold the **ctrl** key when dropping the snapin you will make a copy of it instead of moving it

To remove a snapin simply click the little X icon in the top right corner of the snapin.

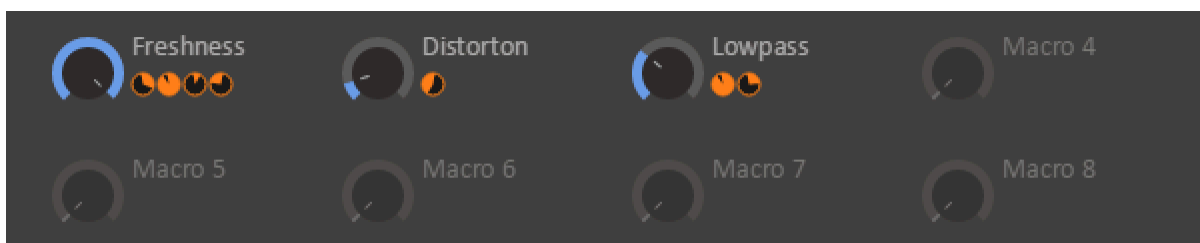
Modulation

Almost all parameters in Snap Heap and in snapins can be modulated. In Snap Heap you will find four different **modulation sources** that can be used for modulation; **macro knobs**, **LFOs**, **envelopes** and **MIDI**.

Hooking up a modulation source to a target parameter is done in pretty much the same way for all modulation sources. Look for the little link icon appearing when you are hovering your mouse pointer over a modulation source. Clicking the link icon selects the modulation source and switches the UI over to **modulation target selection mode**. In this mode a small orange **modulation knob** will appear next to each possible modulation target. Click and drag on the modulation knob to connect the modulation source to the target parameter and set the modulation level. The modulated control will turn an orange color to indicate that it is being modulated.

After a modulation source has been connected to a target parameter the modulation knob will be visible next to the modulation source at all times. The modulation knob will also appear next to the modulation target when the target is hovered. The modulation knob can be dragged to adjust the level of modulation. To disconnect the modulation, double click the modulation knob.

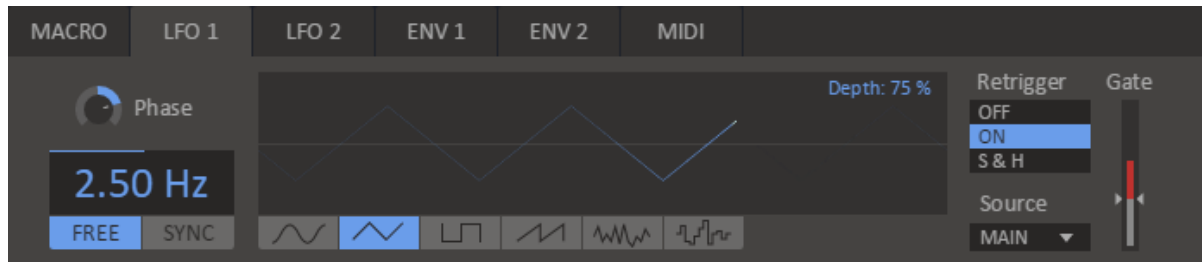
Macro knobs



The macro knobs can be routed to any other parameter in Snap Heap or in Snapins using the modulation system. This lets you control many different aspects of the sound of the patch using a single macro knob. The macro knobs can be renamed by clicking on their label. The new name is stored in the preset.

The macro knobs are probably the first thing you should take a look at when trying out new presets, since the preset maker may have routed them to allow quick adjustment of some key parameters in the preset.

LFOs



Snap Heap is equipped with two **low frequency oscillators**, or LFOs for short, which are ideal for modulating parameters in an oscillating and rhythmic fashion.

Frequency

Controls the speed of the oscillator, either in Hertz when in free running mode, or as a note length when running in synced mode.

Shape

Oscillator shape can be one of **sine**, **triangle**, **square**, **saw tooth**, **soft noise**, **hard noise**.

Depth

Adjusts the amplitude of the LFO. Can be modulated in order to, for example, make the LFO modulation fade in using an envelope.

Phase

Adjusts the phase of the LFO. Is especially useful when the LFO is in synced mode, or when retriggering is enabled. The phase parameter can also be modulated, for example by the other LFO, for interesting effects.

Retrigger

The LFOs in Snap Heap offers two different retrigger modes. Setting retrigger to **on** causes the LFO to restart whenever the trigger condition is met. The **sample & hold** mode instead cases the value if the oscillator to be sampled whenever the trigger condition is met.

Source

This drop down selects which input signal controls triggering. When this setting is set to **note on** the LFO will retrigger whenever a MIDI note on event is sent to Snap Heap (this requires that your DAW sends MIDI events to Snap Heap in the first place). The other option is to retrigger whenever the audio level goes over a certain threshold. You can select audio signals for retriggering between **main** (the main input to Snap Heap), **external** (secondary / sidechain input to Snap Heap) or **band** for input signal to a certain band.

Gate

If the trigger source is set to one of the audio sources the threshold for when triggering should occur can be set using this fader. The current peak level for the selected audio channel is shown on the encoder of the fader.

Envelopes



Snap Heap is equipped with two **envelopes**, which can be used to modulate parameters using the audio volume or MIDI note events.

Mode

Three envelope modes are available. In **peak** and **RMS** modes the volume of the audio is tracked and used as the modulation value. There is a subtle difference in how peak and RMS tracks the audio volume. Peak mode tracks peaks in the audio volume and is thus very responsive to transients such as drum hits. RMS mode tracks the power of the signal using the root mean square method, which can give more stable results than peak mode, though the tracking may be less responsive. Finally, in **ADSR** mode the envelope will act like an attack / decay / sustain / release envelope of the kind that can often be found on synthesizers. The ADSR envelope can be triggered either by MIDI input or by audio volume.

Attack

When you trigger a note the envelope will start to climb up from zero to its maximum level. This is known as the **attack phase**, the length of which is decided by the attack parameter of the envelope. Keep the attack low for a short snappy sound, or make it high for a slow sound which builds up over time.

Decay

Next follows the **decay phase**, where the envelope drops down from its maximum level down to the sustain level. The time this takes depends on the decay parameter. The decay parameter is only available if the envelope is running in ADSR mode.

Sustain

After the envelope has dropped down to the sustain level comes the **sustain phase**, which lasts until the note has ended. Thus, the sustain parameter does not influence the length of this phase but rather how high the sustain level is. The sustain parameter is only available if the envelope is running in ADSR mode.

Release

When the note is released the envelope moves on to the **release phase**, where the envelope drops back down to zero. The time this takes is dictated by the release parameter.

Source

This drop down selects which input signal to follow or to use for retriggering. When this setting is set to **note on** the envelope will retrigger whenever a MIDI note on event is sent to Snap Heap (this requires that your DAW sends MIDI events to Snap Heap in the first place). The other option is to follow the volume level of an audio signal. You can select audio signals between **main** (the main input to Snap Heap), **external** (secondary / sidechain input to Snap Heap) or **band** for input signal to a certain band.

Gate

If the mode is set to ADSR and the source is set to one of the audio sources the threshold for when the envelope should be triggered can be set using this fader. The current peak level for the selected audio channel is shown on the encoder of the fader.

Curve

Selects between **exponential** and **linear** falloff in the decay and release phases when the envelope is in ADSR mode.

Legato

Selects whether the envelope should restart from zero or if it should continue from its current value when it is triggered in ADSR mode.

Pitch tracking



The pitch tracker allows modulation using the pitch of the input signal. The modulation output of the pitch tracker is set to match any frequency based parameters at 100%. For example, this means modulating the cutoff of the Filter snapin with 100% pitch, the cutoff frequency will follow the tracked frequency of the input similarly to how 100% keytracking in a synth behaves.

Lower and upper note limit

Two vertical handles can be used to select the expected pitch range of the input. This hint helps the algorithm perform better and reduces the risk of glitches in the pitch tracking.

Root

The root parameter acts as an offset in the modulation output. If the input signal has the same note as root, the modulation output will be zero. Lower notes will yield negative modulation values and higher notes will yield positive modulation values.

Sensitivity

This parameter adjusts the sensitivity of the algorithm, and higher values result in faster tracking but is more prone to glitches. The default value of 50% is usually a good compromise of speed versus stability, but this can be tweaked to improve the tracking depending on the input.

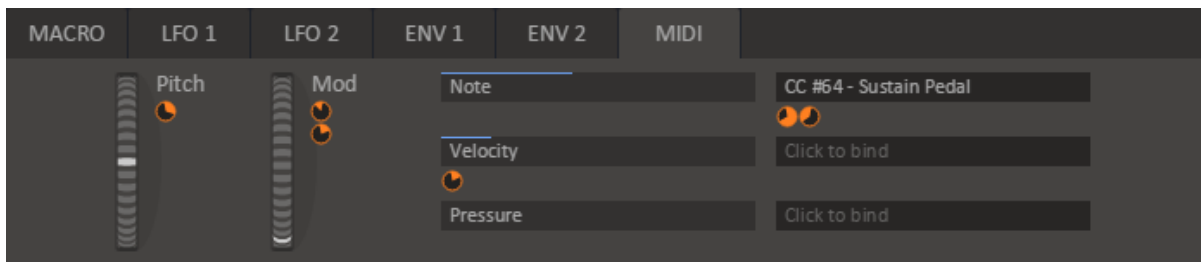
Source

This drop down selects which input signal to track the pitch of.

Gate

The gate parameter sets a lower limit of when to run the tracking, to help avoid trying to tracking quiet unpitched sounds like noise.

MIDI



The MIDI tab allows modulation using MIDI input. Since Snap Heap is an effect rather than an instrument, many DAWs will not route MIDI events to it by default. Check the documentation for your DAW for details on how to set up MIDI routing.

Pitch

This wheel is connected to the pitch wheel commonly found on MIDI keyboards. The pitch wheel has its zero position in the middle of the wheel, and spans in range from -1 to +1. Whenever the wheel is released it snaps back to the zero position.

Mod

This wheel is connected to the modulation wheel commonly found on MIDI keyboards. In contrast to the pitch wheel, this wheel ranges from 0 to +1 and does not snap back when released.

Note

This modulation source is based on the last played note on the keyboard. Higher notes give higher modulation values.

Velocity

This modulation source is based on the velocity of the last played note, i.e. the force with which you hit the key on the MIDI keyboard when playing a note.

Pressure

This modulation source is based on the channel pressure, i.e. the force with which you hold down the keys on the MIDI keyboard.

MIDI CC slots

To the right of the slots for note, velocity and pressure are three general purpose MIDI CC slots which can be bound to any MIDI controller. To bind a slot to a controller, simply click on it and then move the controller on your MIDI keyboard.

Acknowledgements

This development of this product was helped by the following pieces of excellent open source software:

Boost C++ Libraries

Skia Graphics Library
Copyright © 2011, Google Inc.

Symbiosis AU/VST
Copyright © 2010-2013, NuEdge Development / Magnus Lidström

LodePNG
Copyright © 2005-2015, Lode Vandevenne

C++ optimized SHA1 algorithm
Copyright © 2011, Micael Hildenborg

miniz
By Rich Geldreich

FastDelegate
By Don Clugston

FFTReal
By Laurent de Soras