



Grasswhistle

Aqeel Adam Sound

User manual (Revision 1.0.0)

Introduction

Thank you for using Grasswhistle. Grasswhistle is a free arpeggiator with some tricks up its sleeve. You can load it as an instrument or as a MIDI effect in any DAW, so you can play it on its own, or you can sequence your favorite instruments with it.

Grasswhistle is adapted from the arpeggiator in my other plugin, Waymaker. It also carries some newer features that I have been developing since Waymaker first came out, so you may notice a few things here that are not in Waymaker yet. If you enjoy Grasswhistle's workflow, you should definitely check out Waymaker.

I hope Grasswhistle finds a place in your sessions. And if you are new to Aqeel Aadam Sound, welcome – it's great to have you here.

-Aqeel

Overview

At its core, **Grasswhistle** is a customizable arpeggiator. You create a sequence of notes – a series of stages – and Grasswhistle plays through that sequence whenever it receives input. The twist is that a stage does not choose a specific pitch. Instead, it chooses which note of the chord you are holding to play: the lowest note, the second lowest, the third lowest, and so on. Because the sequence is defined relative to the chord rather than in absolute pitches, the arpeggio keeps its shape no matter what chord you play into it.

Around that sequencer sit a handful of other tools. Performance controls let you hold chords and decide when the sequence resets. A chord generator can build a chord from a single note. A note range filter keeps the output within a chosen register. When Grasswhistle is loaded as an instrument, a built-in synth voice lets you hear the results immediately.

Installation and usage

When installing Grasswhistle you will see two builds: **Grasswhistle**, which is the instrument version, and **Grasswhistle MIDI**, which is the MIDI effect version.

Load the instrument build on an instrument track and play it with MIDI; it includes a built-in synth voice, so it makes sound on its own. Load the MIDI build ahead of another instrument on that instrument's track, so that Grasswhistle's output drives the instrument instead. If your DAW does not automatically identify which version should be loaded in a particular context, just remember: MIDI is the effect version, and the other one is the instrument.

Contents

- 5. User interface
- 7. The sequencer
 - 7. Stages
 - 7. Notes beyond the chord
 - 8. Beats
 - 8. Quick-set buttons
 - 9. Sequencer options
- 10. Performance
- 11. Chord generator
- 12. Synth voice
- 13. Output view
 - 13. Range limiting
- 14. Settings menu
- 16. Presets
- 18. Techniques and ideas
- 19. Installation
 - 19. Compatibility
 - 19. Uninstall locations
- 20. Support

User interface



This is the primary screen that you will see when using **Grasswhistle**. Each section is described in depth in this manual.

Here's a brief overview:

1. **Sequence stages**: This defines the arpeggio in terms of which note to play on which stage.
2. **Beats**: This defines how long to stay on each stage.
3. **Quick set buttons**: These buttons can conveniently set the volume of all sequence stages.
4. **Sequencer settings**: Sets the speed and length of the sequence, as well as the behavior of the notes generated.
5. **Hold**: Holds one chord until the next is received.
6. **Reset**: Defines how and when the sequence resets to the first stage.

7. **Chord generator:** Optionally generate a full chord within a given scale.
8. **Voice menu:** A built-in synth voice can be used when loading Grasswhistle as an instrument.
9. **Settings:** This is where all global configuration (including default settings) for Grasswhistle can be found.

The sequencer

Stages



A **stage** is one step of the arpeggio. Each stage chooses a note from the chord you are holding – not a specific pitch, but a position within the chord: the lowest note, the second lowest, the third lowest, and so on. When you hover over or drag a stage's slider, Grasswhistle shows you which note that stage will play.

By default, Grasswhistle has a sequence of first, second, and third notes. Hold a C major chord and it plays through C, E, and G. Switch to the **output view** on the right side to watch the input and the output of the arpeggiator side by side.

Notes beyond the chord

What happens if a stage asks for a note that does not exist in the chord – such as the fourth note of a chord that only has three? In this case, Grasswhistle automatically transposes notes up an octave to reach the necessary note. Ask for the fourth note of a triad and you will hear the lowest note an octave up; ask for the fifth and you will hear the second lowest note an octave up. Stages can also play down from the root note, in which case notes are transposed down into lower octaves.

The end result is a custom arpeggio that maintains its shape no matter how many notes are in a given chord. You are free to play a triad, then a seventh chord, then just two notes, or even a single note all

in a row, and the arpeggio keeps its character and its contour no matter what you throw at it.

This is the heart of Grasswhistle: the sequence describes a shape, and the chord supplies the notes.

Beats

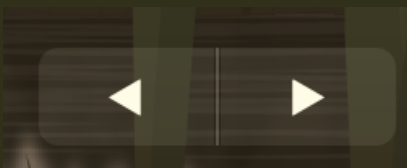


Each stage also has a number of **beats**, which sets how many steps need to pass before the sequence moves on to the next stage. As Grasswhistle sits on a stage, you can see its beats filling up on the slider. Hover over any of the beat sliders and the total number of beats in the sequence appears on the left side, so you can check that everything adds up to, say, a multiple of four or eight.

Quick-set buttons



Underneath the stages are a number of **quick-set buttons** for convenience. The first group sets every stage at once: zero all of the stages, set them all to go up, all to go down, up then down, or down then up.



Next are two options for **rotating** the sequence, which shift every stage one position to the left or to the right. Sometimes you might come up with a sequence and find yourself hearing the downbeat in a different place than you expected. Rotating lets you quickly shift where the sequence starts and ends without rebuilding it.

Finally, there are two **randomization options**. One randomizes the steps in the sequence, and the



other randomizes the number of beats. Both are tuned to stay fairly musical and provide instant inspiration.

Sequencer options



At the bottom of the UI, you can set the **speed** of the sequence. By default it is synchronized to your DAW's BPM, but you can desynchronize it as well. Next to it, you can set the **number of stages** in the sequence, up to 16.



The last controls along the bottom determine how **velocity** and **note length** are handled when Grasswhistle generates notes. By default, the output notes have a fixed velocity, which you can set here. Turn **fixed velocity** off and the velocity of each note matches the input instead – so if you play a chord with mixed velocities, you get an arpeggio with mixed velocities.

Note length is a percentage of the sequence rate. The **Sustain** toggle controls whether a note is held through a stage that has multiple beats. By default it is off, and each stage plays a short, staccato note. Turn it on and notes are held for the full duration of their stage.

Performance

Up in the top left corner of the UI are Grasswhistle's performance and play settings.

 Hold

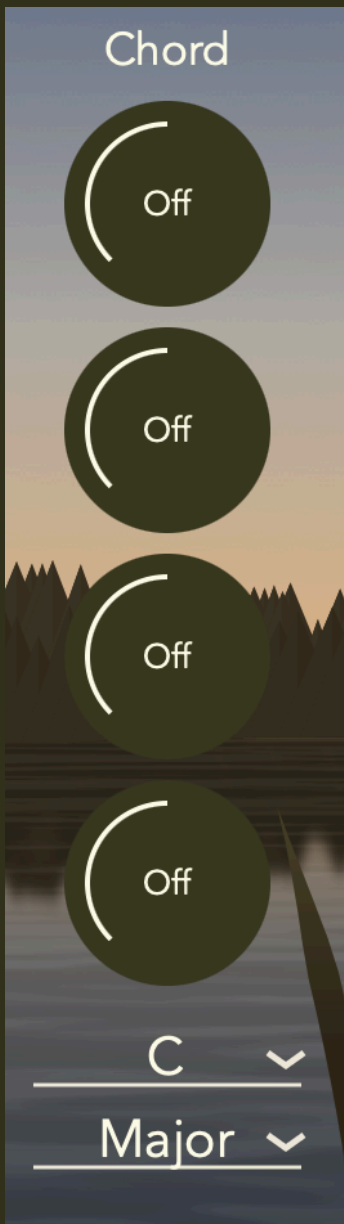
Hold does what it says. When enabled, Grasswhistle keeps holding a chord until another chord is played, or until you pause your DAW's transport. This lets you play a chord, let go, and keep the arpeggio running while your hands are free.

 Reset (Once)

Reset controls when the sequence returns to its first stage. There are three options:

- **Once** (the default). If you start playing while Grasswhistle is not currently running, the sequence resets to the start. In practice, whenever you start playing from silence, the arpeggio begins at the first stage.
- **Always**. Every time you trigger a new chord, the sequence resets to the start.
- **Off**. Grasswhistle never resets and simply picks up wherever it left off.

Chord generator



Below the performance controls is the **chord** section. Grasswhistle has a built-in chord generator, which can conveniently create a chord from just one note.

The chord generator is controlled by a set of interval sliders. When all of the sliders are set to off, nothing happens and your notes pass through as played. Once you enable any of them, Grasswhistle starts generating a chord on top of every note you play. You can also play multiple notes to stack multiple chords – for example, you might add a third to any note played, and then play two or more notes to hear each of them harmonized.

At the bottom of the section, you can set the scale within which chords are generated. Keep in mind that this does nothing while all of the chord intervals are set to off.

By default, the note you play is quantized to the chosen scale as well, so that both the root and its chord stay in key. If you would rather keep the chord in the scale but leave the root exactly as you played it, see [Quantize root](#) in Settings.

Synth voice



When using Grasswhistle as an instrument, you also have the option of using a built-in synth voice. This voice is loosely based on the Ocarina sound from The Legend of Zelda, and it has four controls.

- **Bloom** is an envelope that controls how quickly notes come in and out.
- **Vibrato** adds vibrato to notes, increasing in intensity the longer a note is held – similar to something like a violin.
- **Noise** adds a bit of noise to the signal. It is introduced at the start of notes, like the breathy sound you get when you play a flute.
- **Reverb** smooths everything out, via a reverb that increases in length and volume as you turn it up.

At the bottom of the section, you can toggle between the synth being **polyphonic** or **monophonic**. By default it is polyphonic.

The synth voice is only available in the instrument build. In Grasswhistle MIDI, the output is MIDI only, to be played by whatever instrument comes next in the chain.

Output view

On the right side of the UI, you can switch from the stage view to the **output view**, which shows the input to the arpeggiator alongside the notes it generates. When you hover over this view, you can see the actual note names of any note that is generated.

Range limiting

The output view also contains a global **note range** filter, which ensures that every generated note falls within a chosen range. Any note outside the range is transposed up or down an octave until it fits. This is convenient for keeping everything in the same register, or for avoiding something like block chords when a sequence would otherwise reach too far.



Settings

The Settings menu holds Grasswhistle's additional options – the ones you might not need that often. Volume. When Grasswhistle is loaded as an instrument, you can control the volume of its synth voice here.

MIDI channels

Set the input and output MIDI channels that Grasswhistle listens to and sends on.

Transport

This controls how Grasswhistle reacts to your DAW's transport. By default, it stops when your host stops. You can instead set it to play only while the host plays, or to ignore the host entirely.

Quantize root

When the chord generator is active, Grasswhistle quantizes any note you play to the chosen scale by default. If you are generating a simple chord in C major and play a C, you get a C with a chord in the key of C. Play a C sharp and you get exactly the same thing, because the C sharp is quantized to the scale. Turn this off and Grasswhistle still creates the chord within C major, but it no longer quantizes the original note – so C, C sharp, D, and D sharp each get a chord of their own.

Turning Quantize root off is an interesting way to explore options that are chromatic but still musically related to the given scale.

Scenic mode

Hides the sliders by default so you can simply enjoy the scene. You can still see the sliders when you hover over them, and if you drag something like a beat slider, all of the other beats appear as well.

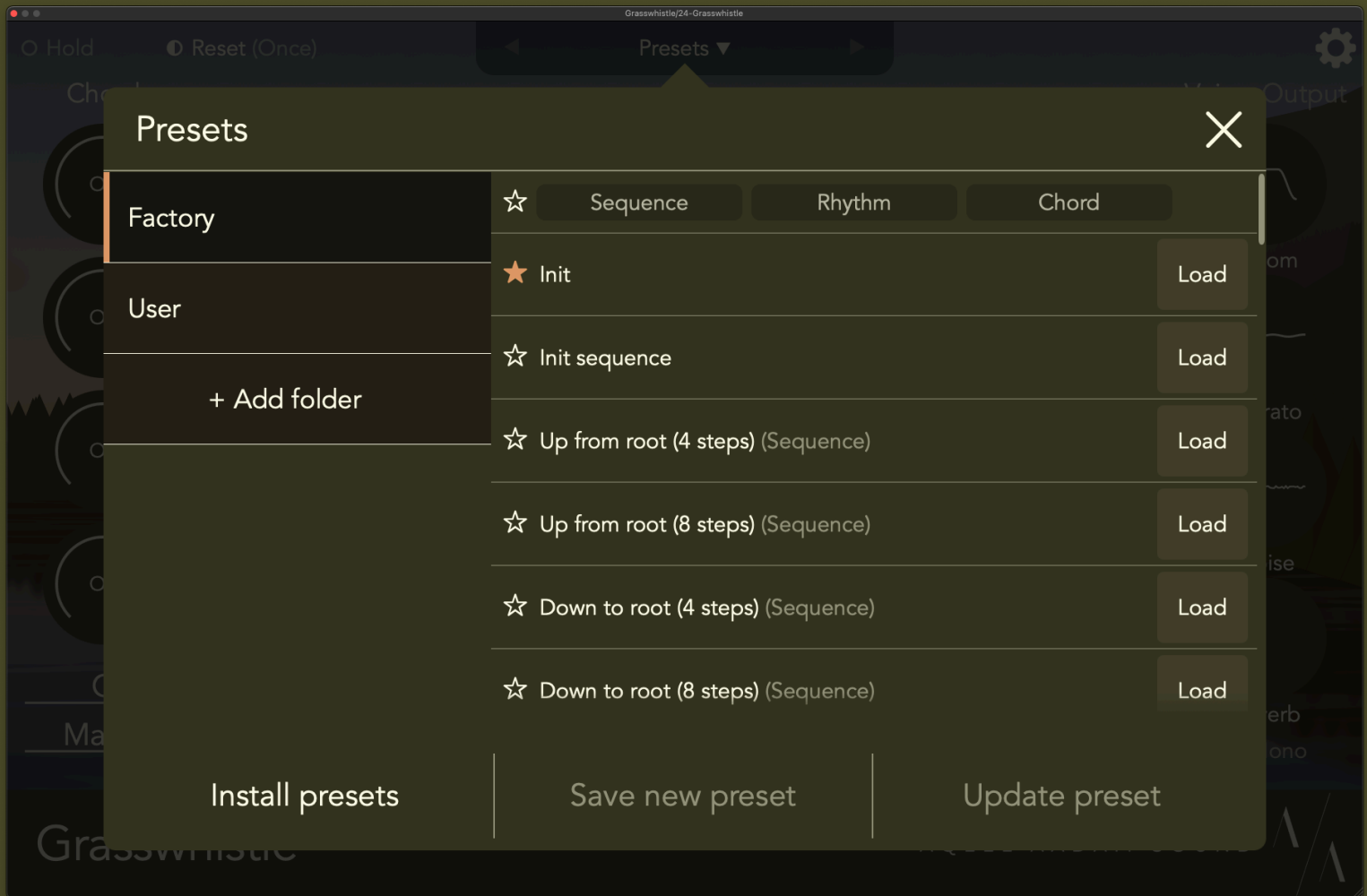
Reduce motion

If you find the scene distracting, or you just want to save on CPU, this freezes the scene and shows a simplified version of it.

Tooltips

On by default. Hover over a control and you will see some helper text covering much of what is in this manual.

Presets



Grasswhistle comes with a collection of handcrafted factory presets and samples as well as the ability to save and create your own presets. Presets can be accessed from the Presets button at the top of the main screen.

When loading a preset, a couple of options at the bottom of the manager make things a little more convenient. **Sequence Only** loads just the configuration of the sequencer itself and leaves everything else alone. **Keep Performance** makes sure that your Hold, Reset, and Transport settings do not change.

Saving presets

All Grasswhistle presets must exist within a folder. By default, you will have a **User** folder available, though new folders can be created easily via **+ Add folder**. A new preset can be saved by clicking **Save new preset**, which will give you the ability to name and save a preset within the current folder.

When saving a preset, you will have the option to optionally add a category as well: atmosphere, texture, reverb, or distortion.

If a preset is currently loaded, it can be updated via the **Update preset** button. This will overwrite the current preset, which cannot be undone (when in doubt, it may be safer to save a new preset instead).

Importing and Exporting

Presets can be imported into Grasswhistle's preset system in two ways: using the **Install Presets** button in the bottom left of the Presets menu or simply dragging and dropping them into Sightlines.

Presets can be imported as **.grasswhistle** files, a folder with **.grasswhistle** files (and samples), or a **.zip** containing **.grasswhistle** files (and samples).

A preset folder can be exported directly as a **.zip** file from the bottom of the folder's list of presets. In this case, Grasswhistle will automatically create a **ZIP archive** of all presets for easy sharing or distribution.

Techniques and ideas

Here are some starting points that show off different sides of Grasswhistle.

One shape, many chords

Build a custom sequence – a few stages climbing past the top of a triad, then one dipping below the root. Now play a triad, a seventh chord, and a single note one after another. The contour stays the same while the harmony changes underneath it, which makes Grasswhistle especially forgiving of chord changes with different numbers of notes.

One-finger harmony

Enable a third and a fifth in the chord generator, pick a scale, and play single notes. With Quantize root on, every key gives you a diatonic triad through the arpeggiator. Turn Quantize root off and the chromatic notes in between get chords of their own – strange, but still related to the scale.

Sequencing another instrument

Load Grasswhistle MIDI ahead of a favorite synth, turn fixed velocity off, and play chords with uneven velocities. The arpeggio inherits the dynamics of your playing, so a soft inner voice stays soft while the top note rings out.

Rhythm first

Start from a simple Up preset, randomize the beats, and listen for where the natural downbeat lands. Rotate the sequence left or right until the accent falls where you want it, then shape the notes from there.

Installation

macOS: Open the .pkg installer and follow the provided instructions.

Windows: Open the .exe installer and follow the provided instructions.

After the above instructions, you will need to restart your DAW and/or re-scan for new plug-ins.

Compatibility

macOS 10.13 or later, Intel and Apple M1/M2 chips supported. VST3/AU/AAX format.

Windows 10 or later, 64-bit. VST3/AAX format.

Sightlines requires an internet connection for initial authorization. Internet connection will not be necessary afterward.

Uninstall locations

macOS:

- AU: Macintosh
HD/Library/Audio/Plug-Ins/Components/
- VST3: Macintosh
HD/Library/Audio/Plug-Ins/VST3/
- AAX: Macintosh HD/Library/Application Support/Avid/Audio/Plug-Ins/
- Presets and data: Macintosh
HD/Users/<username>/Application Support/AqeelAdamSound/Sightlines/

Windows:

- VST3: C:\Program Files\Common Files\VST3\
- AAX: C:\Program Files\Common Files\Avid\Audio\Plug-Ins\
- Presets and data:
C:\Users\<username>\AppData\Roaming\Aqeel
AdamSound\Sightlines\

Support

For any support inquiries, please feel free to [reach out!](#)