INSTRUCTIONS MANUAL

Vendaval is an audio synthesis plugin written in Csound with the Cabbage framework, developed by Caio M. Jiacomini.

If you have any questions after reading this, feel free to contact me at caiojmini@gmail.com. You can also access the Github repository for the source code here: https://github.com/CaioMJ/Vendaval

Vendaval is made of six components:

GLOBAL:

Controls attributes that govern the whole plugin.

It's important to note that the values for the ADSR envelope must be set before a MIDI note on message triggers the instrument. If you change the decay, sustain, or release value after the instrument is triggered, the new values will not be applied, therefore, **you must set all your envelope values as you want them before triggering a MIDI note on message.**

WOOING:

Produces filtered noise to simulate the wind whistling sound. Pink noise is being filtered with a bandpass filter with its center frequency being modulated to random values at random times.

BACKGROUND:

Works almost identical to the wooing component, except it is limited to a larger bandwidth range and has a more limited frequency range.

GUSTS:

Also works similarly to the wooing and background components, the key difference being that the center frequency of the bandpass filter is also modulated by an LFO, providing a more undulating sound.

RUMBLE:

Produces low-pass filtered noise. Instead of adjusting the center frequency of a bandpass, there's a parameter to adjust the cutoff frequency of the low-pass filter, which is not being randomly modulated like the other filters.

REVERB:

Applies reverb to all other components.

PARAMETERS:

Volume: controls the volume of the individual component
Frequency/Cutoff: sets the center frequency of the primary bandpass filter or the cutoff frequency of the low pass filter
Range: controls the frequency range that the bandpass filter can be modulated by both above and below the center frequency. Acts as a multiplier to a hard coded number
Rate: controls how fast the bandpass filter is modulated. Act as a multiplier to a hard coded range of time values
Bandwidth: controls the bandwidth of the bandpass filter.
Resonance: controls the filter resonance
Distortion: controls distortion on the Rumble component

Harmonizer Vol: controls the volume of a secondary bandpass filter that harmonizes with the primary filter. The secondary filter is supposed to be used to emulate the resonant sound that wind makes when blowing through small gaps
Harmonizer Freq: multiplies the frequency of the primary filter by the number defined in this parameter to set the frequency of the secondary filter. More realistic results can be achieved by keeping the value between 1.1 and 2

Attack, Decay, Sustain, Release: sets the amplitude envelope for all components

Mix: sets the dry/wet reverb mix
Size: sets the size of the reverb effect

PRESETS

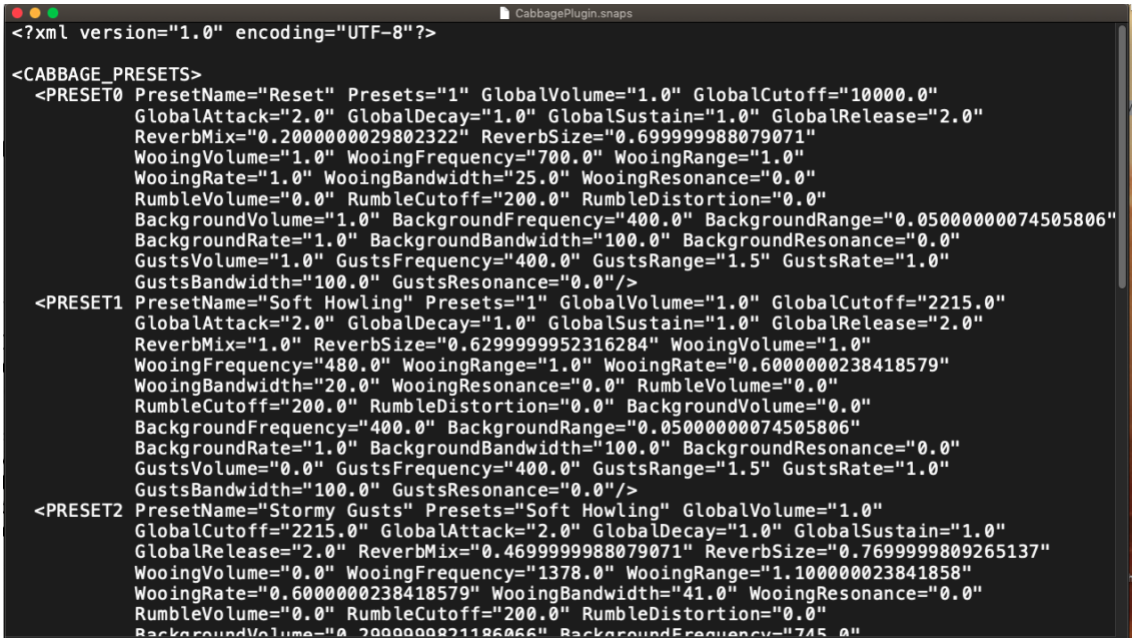
Click the “Save” button to change the current configuration as a preset

PRESETS MANAGEMENT

Managing and renaming presets is a bit tricky with the framework I used to develop **Vendaval**, so here’s my best attempt to break it down

Presets configuration are stored inside a text file with a *.snaps* format.
On Windows, this file lives in the same folder as the plugin files
On MacOS, this file can be accessed by right clicking the plugin and selecting “Show Package Contents”. The *.snaps* file is inside the Contents folder

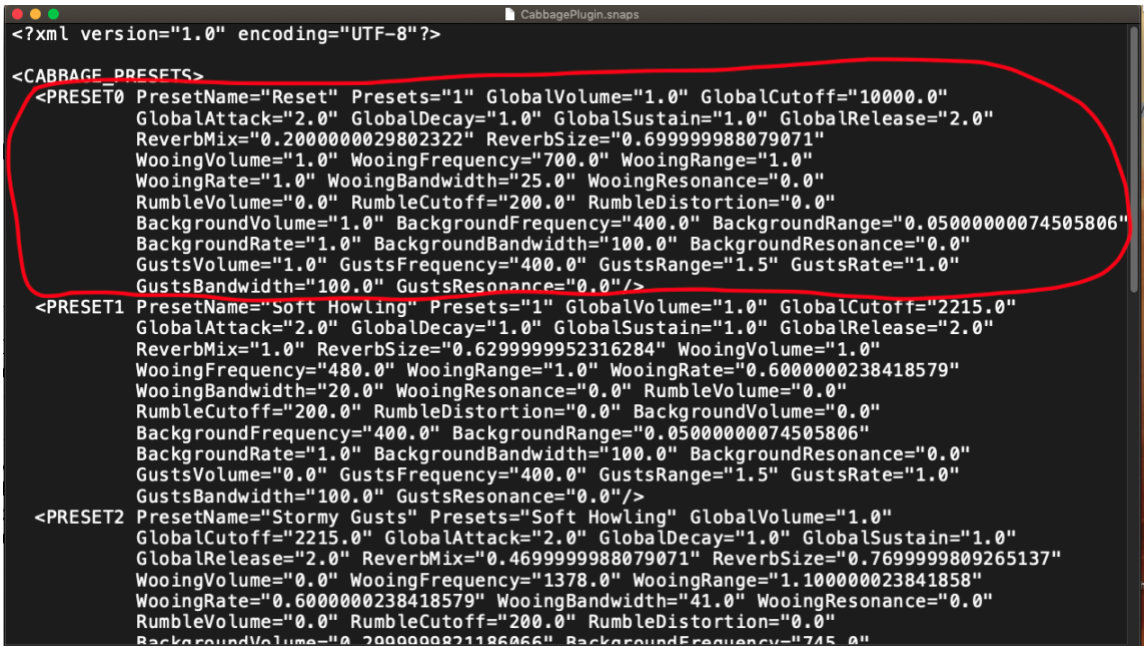
Here’s what that file looks like:



What this file essentially does is: sort the order of the presets, define a name for the preset, and store the value of each parameter for the preset

Each preset starts with **<PRESET#** and ends with **/>**

I highlighted below the boundaries of the **PRESET0** for more clarity:



After defining the preset number (PRESET0, PRESET1, PRESET2), the text defines its name with **PresetName=“name”**. The first preset will be called “Reset” since it’s declared as **PresetName=“Reset”**

To change the name, you just have to substitute whatever text between the =“ ” signs with the desired name. If I wanted to rename the first preset to Wirt, the text would read as **PresetName=“Wirt”**

The same thing applies if you want to change a parameter value, just find the parameter you want and change the value between the =“ ” signs