

Designing Synthesizers with Web Audio

Luke Teaford

4100 Quentin Boulevard
#310
Columbus, OH 43230
luketeaford@gmail.com

ABSTRACT

Synthesis is an important part of Web Audio, and the native audio nodes give developers the power to build innovative and unique synthesizers. This talk will share knowledge and insight into synthesizer design and application architecture from the approach of making musical instruments gained from developing my Web Audio Synthesizers.

1. WHAT IS A SYNTHESIZER?

A synthesizer is a machine for creating new sounds. Synthesizers do not need to be musical or to be played in traditional ways. Although Web MIDI and musical keyboards are nice, a sound could be triggered by any browser events. I will call these types of synthesizers "sound engines". Sound engines would be useful in video games, user interfaces, and other highly interactive applications.

2. MORE INTERESTING USES OF THE NATIVE AUDIO NODES

In TANGUY, "Oscillator One" is built of multiple OscillatorNodes that are treated as one: they receive the same pitch information. Each OscillatorNode is connected to a GainNode, which creates a mixer so the various types can be blended. My EVE synth uses eight GainNodes to give the user control of the amplitude of eight sine waves. Each oscillator receives the same detune value, but the frequency is doubled each time. OscillatorNodes can be connected to AudioParams to modulate them. Hardware synthesizers usually have one or more Low Frequency Oscillators (LFOs) to do precisely this. Typically, LFOs would have a frequency less than 20Hz, but experimentation is encouraged. It is useful to vary the intensity of an effect. TANGUY uses a modulation wheel that is associated with the amplitude of a GainNode so that the effect of the LFO can be attenuated or stopped entirely. EVE has an LFO with a delay stage in addition to the attack decay sustain release envelope pattern. This can be used to shape vibrato and tremolo effects.

3. ARCHITECTURE TIPS AND RECOMMENDATIONS

Use JSON to store the values of parameters. I recommend using simple values in presets (0-1 is a common range in TANGUY and EVE), and taking those values from HTML5 range inputs. Mathematical operations happen elsewhere in the JavaScript using values without changing them. This makes it easy to keep the UI synchronized. Create CustomEvents and use a Pub/Sub pattern to simulate the workings of hardware synths. Consider placing a GainNode between each connection, and using multiple GainNodes connected to another GainNode to be used as a mixer. Doing so will give you more flexibility for fine tuning the depths of modulation and the relative amplitudes of your sources. Remember the UI does not need to reflect all possible operations that happen in the code. Feedback loops are permissible. Simplify the choices you're giving the user. Try providing one control that does many things. Do not be afraid to experiment.

WEB LINKS

TANGUY Web Audio Synthesizer: <https://tanguysynth.com>

EVE Web Audio Synthesizer: <http://evesynth.com>



Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** owner/author(s).

Web Audio Conference WAC-2016, April 4-6, 2016, Atlanta, USA.

© 2016 Copyright held by the owner/author(s).