JZZ.js - a unified API for MIDI applications

Sema Kachalo
Jazz-soft
Bethesda MD
sema@jazz-soft.net

ABSTRACT

There are multiple ways to access and emulate MIDI devices in JavaScript. This includes real MIDI access like Web-MIDI API or Jazz-Plugin, Web-Audio synths like MIDI.js or Timbre.js, or HTML gadgets for MIDI input like Qwerty-Hancock. When used all together, these methods allow good OS/browser coverage; however, it is cumbersome to use them all in the same project because of their API differences.

We introduce JZZ.js - a JavaScript library that provides unified interface for all possible MIDI implementations and has the following key features:

- Hiding the details of asynchronous calls behind the developer-friendly chaining syntax.
e.g. `JZZ().openMidiOut(/Yamaha/).send(0x90,60,127).wait(500).send(0x80,60,127);`

- Treating MIDI inputs and outputs as graph nodes that can be directly connected (inspired by the Web-Audio API design).
e.g. `JZZ().openMidiIn().connect(JZZ().openMidiOut('Microsoft GS Wavetable Synth'));`

- Numerous helpers for converting MIDI messages to/from the human-readable form.
e.g. `console.log(JZZ.MIDI.aftertouch(0,'D#5',127).toString());`

- Extensibility. We provide examples how to adapt the developers’ own libraries and gadgets into the framework.

JZZ.js works with Node.js and all major browsers in Windows, MacOS, and Linux. Limited support is available for Android and iOS.

The library is free to download at https://github.com/jazz-soft/JZZ

ACKNOWLEDGMENTS

Special thanks to Chris Wilson and Daniel van der Meer for valuable discussions, great ideas and encouragement.

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