

# Develop WebAudio Plugins in a Web Browser

Shihong Ren, Stéphane Letz, Yann Orlarey, Romain Michon, Dominique Fober  
GRAME, 11 cours de Verdun LYON  
renshihong@hotmail.com  
(letz, orlarey, michon, fober)@grame.fr

Michel Buffa, ElMehdi Ammari, Jerome Lebrun  
Université Côte d'Azur  
CNRS, INRIA  
(buffa, lebrun)@i3s.unice.fr,  
ammarielmehdi@gmail.com

## ABSTRACT

We propose to demo an online IDE based around the FAUST DSP audio language [1], that includes a source code editor, embedded compiler and GUI editor allowing to directly test, generate and deploy WebAudio Plugins (WAP). The tool is available online<sup>1</sup>.

## 1. INTRODUCTION

When audio effects or audio/MIDI instruments have to be shared between several DAWs or audio environments, a plugin model is usually preferred.

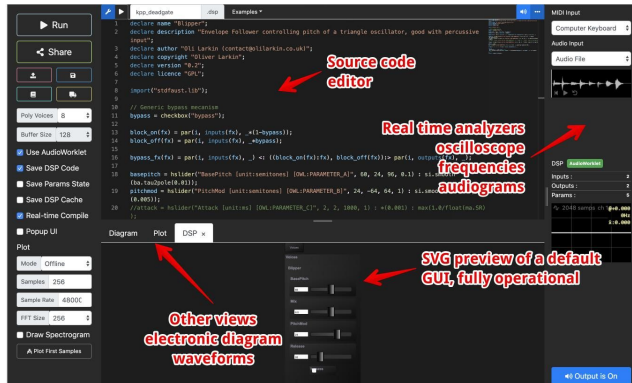


Figure 1: the FAUST IDE provides many embedded tools: oscilloscopes, spectroscope and spectrogram, functional default GUI, schema preview, etc.

Several native audio plugin formats are now popular, including Steinberg's VST format (Virtual Studio Technology, created in 1997 by Cubase creators), Apple's Audio Units format (Logic Audio, GarageBand), Avid's AAX format (ProTools creators) and the LV2 format from the Linux audio community. In the much newer WebAudio API (2011), there was no standard format for high-level audio plugins. With the emergence of Web-based audio software such as digital audio workstations (DAWs) developed by

<sup>1</sup> <https://faust.grame.fr/ide/>, experimental version with the WAP GUI Builder available at <https://mainline.i3s.unice.fr/idewap>



Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0).

Attribution: owner/author(s).

Web Audio Conference WAC-2019, December 4–6, 2019, Trondheim, Norway.

© 2019 Copyright held by the owner/author(s).

companies such as SoundTrap, BandLab or AmpedStudio, it is desirable to have a standard to make WebAudio instruments and effects interoperable as plugins compatible with these DAWs and more generally with any compatible host software.

Such a plugin standard needs to be flexible enough to support these different approaches, including the use of a variety of programming languages. New features made possible by the very nature of the Web platform (e.g., plugins can be remote or local and identified by URIs) should also be available for plugins written in different ways. To this end, some initiatives have been proposed [2, 3] and with other groups of researchers and developers we proposed [4] a standard for WebAudio plugins called WAP (WebAudio Plugins), which includes an API specification, an SDK, online plugin validation tools, and a series of plugin examples written in JavaScript but also with other languages<sup>2</sup>. These examples serve as proof of concept for developers and also illustrate the power of the Web platform: plugins can be discovered from remote repositories, dynamically uploaded to a host WebApp and instantiated, connected together etc. The reader can get a "multimedia" idea of this work<sup>3</sup>. Since the last year, WAP now includes support for pure MIDI plugins (a GM midi synthesizer, virtual midi keyboards, a MIDI event monitoring plugin, etc). We propose to demo a new online IDE, that is well suited for coding, testing, publishing WAP plugins written in FAUST, directly in a Web browser. The IDE includes a GUI editor that allows developers to fine-tune the look and feel of the plugins. Once complete (DSP + GUI) the plugins are packaged in the form of standard W3C WebComponents and published on remote WAP plugin servers. The plugins will then be directly usable by any compatible host software, using their URIs.

## 2. THE ONLINE IDE

We embedded a WebAssembly version of the FAUST compiler in the IDE (Fig 1.) created by the Emscripten transpiler, to dynamically generate WebAudio nodes from FAUST DSP codes.

<sup>2</sup> <https://github.com/micbuffa/WebAudioPlugins>

<sup>3</sup> <https://www.youtube.com/watch?v=pe8zg8O-BFs>

