# Noiseless Web Audio Tests

Christoph Guttandin
Media Codings

## ABSTRACT

There is a reason why the sentence "bring your own headphones" is part of every invitation to an audio related hackathon. And of course the whole point of working with audio engines like the Web Audio API is to create impressive sonic experiences. But while building such an experience it can quickly become cumbersome to test the same sound over and over again. A hackathon project might not need any tests at all, but more mature projects will benefit from tests which can be run automatically. This is even more important for projects which deal with ever evolving APIs like the Web Audio API. Every browser update may introduce a breaking change. If you don't have automated tests to check if everything still works as expected, you might need to keep your headphones at hand.

I would like to introduce a new category of tests which I call expectation tests. The idea behind expectation tests is to separate code which deals with browser anomalies from the business logic. This opens up the possibility to write code as if you dealt with one single (perfect) browser. Whenever there is a shortcoming in a supported browser which needs to be patched with a polyfill, this should be accompanied by an expectation test which checks if the browser still behaves wrongly. Later on this test will break when the polyfill can be removed again. This makes sure that only necessary patches are part of the production code.

In addition to that, the polyfills should be completely separate from any business logic. Instead the polyfills should make sure to provide a uniform API in each supported browser, which then can be used by the business logic.

Another benefit from using expectation tests is that it's not necessary anymore to test your business logic with the "real" Web Audio API. I would like to present a framework which mocks the Web Audio API and allows to advance the currentTime of an AudioContext in a controlled way to make sure scheduling and AudioParam automations work correctly.