

# Recent and future evolution of the Web Audio API

[Extended Abstract]

Paul Adenot  
Mozilla Corporation  
padenot@mozilla.com

## ABSTRACT

There have been 358 commits in 125 pull requests to the Web Audio API specification repository since the Web Audio Conference 2018 in Berlin, with a number of important changes.

This yearly talk at the Web Audio Conference aims to demystify the world of web specifications, as well as informing authors of the current specification status, new features, bug fixes, and what the group has planned for the future.

In consequence, this talk is structured in three parts:

- What's new in the spec since WAC'18?
- Where are we in the standardization process?
- What's next?

### *New in the spec since WAC'18.*

As usual, a taxonomy of those changes can be established, with four general categories:

- Features
- Bug fixes
- Specification holes
- CR process compliance

This time around, no real features were introduced (with the possible exception of the various new bits of normative behaviour about auto-play of `AudioContext`).

Most of the time was spent finding and fixing specification bugs, as the specification is now very deep in the Candidate Recommendation process. Because there only is a single implementation of `AudioWorklet`, we can't really proceed further. However this is only temporary, and Firefox's implementation is coming.



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).

A lot of care was taken to avoid making breaking changes this year, and only stabilize the prose for a V1 release. However, changes are always necessary, sometimes to solve a real problem, sometimes because of mistakes made by the group in the past. Sometimes, those specification changes didn't make it into implementation, so the specification change can go through as-is. Other times, some data is required, or engagement with authors, to understand whether or not the change will break existing content.

Another set of issues was the aforementioned *specification holes*. Most of them have been resolved, especially when it comes to the text related to `AudioWorklet`, that was brand new. Additionally, this new bit of specification interacts a lot with script. Interacting with script and execution contexts in a specification requires lots of additional text, very often at the intersection of `ECMAScript` and `WebIDL`.

In the same vein, nailing constructors and converting a number of concepts from using English prose to using an algorithm in pseudo-code with clear steps was a clear win, for specifying edge cases: consider a JavaScript statement using a Web Audio API function, that contains two errors that would result in throwing an exception: which exception should be thrown?

Concepts that were too handwavy for a proper specification document did get clearer. In particular, the normative behaviour for the getter for the `value` attribute of an `AudioParam` finally got specified. Cycle handling, and cycles containing `AudioParam` also got specified. The relationship between the `AudioListener` and the `PannerNode` was also clarified.

Various other interactions between parts of the API also got defined: the text for the garbage collection observability issue was finally laid down, including a rewritten section about `AudioNode` lifetime.

The decision was made to punt on a few edge cases or features, and leave things for V2, so that the current document could be finished and stable.

### *Where are we in the standardization process.*

The groups has extended its charter (3 years originally, ex-

tended for 6 months), to allow finishing some work, and also to wait for Firefox's implementation of the `AudioWorklet`. We're still waiting for all PRs to merge, and all problems to be fixed, before continuing. At time of writing, there are 19 issues opened, all of them with resolution (waiting for the prose to be written). Another round of security and privacy review will take place before becoming a recommendation (the last step of the W3C process).

The Audio Working Group has moved all the issues that were not in scope for V1 to another repository called `web-audio-api-v2`, preserving all comments and links. The group has triaged the different issues using the `Project` tab, and is currently seeking input, both for features that should be included, and for discussing the details and specifics of features already included in this new version.

The Audio Community Group is a good way to engage with the standard body, with a low barrier to entry, and a monthly conference call to discuss ideas and issues.

The group expects to be done with the current issue this year, and then change the process to put more emphasis on the future development of the API.

### *What's next.*

Multiple efforts have started that are linked to the Web Audio API specification:

- Web Codecs
- Web Audio API V2
- Audio Device Client

The Web Codecs effort aims at providing low-level primitives to have a more flexible way to decode and encode media data on the Web. It will compose well with `MediaStreams` and `Streams`, and will allow real-time and non-real-time processing of media data.

For the Web Audio API, it will be an improvement over `decodeAudioData`, that always has been the source of problems (no progress information, no chunk decoding, no encoding support, etc.). All the issues related to encoding and decoding of media files have been subsequently closed, as they are best addressed there.

The new repository Web Audio API V2 will be used for all new feature. It is still an unknown if the current text will be copied over and forked, or if the new features will be developed referencing the previous version of the standard.

`Audio Device Client` is a different proposal, that is related to the Web Audio API, but aims at being lower-level, while being capable of interfacing to the existing APIs on the Web platform, such as `MediaStreams`, `AudioContexts` and `HTMLMediaElements`. It is in the early stages, and its usefulness is still debated, compared to just adding the few missing features to the `AudioContext`, that already has the capabilities to interface with the Web Platform.

In any case, input from the greater community of Web Audio API users is going to be a critical component of the

success of a new iteration of the specification. The first iteration direction was in large part decided by the existing implementation, on top of which a few necessary features were added. The next version will address the problems found during the long period of time authors have experimented with the API.

### *Conclusion.*

The Audio Working Group is currently at a pivotal point. After a number of years putting all new features in the backlog, a coherent text is being published, and new feature request will be considered (along with all the features and modification that were proposed in the past, and have been triaged).

An essential part of this new effort will be in other working groups, and non-members are encouraged to participate, via the newly formed Audio Community Group.