W3C Audio Working/Community Group

# TPAC 2024 Agenda

Contacts hongchan@google.com, matthew.paradis@bbc.co.uk, padenot@mozilla.com, mjwilson@google.com
Last updated Sep 25, 2024
Self-link bit.ly/audiowg-tpac-2024
Status In-Progress

## Key Information

- TPAC 2024 Overview | TPAC 2024 Schedule
- The meeting will be hybrid supporting both in-person and virtual attendance.
- Date and time: September 23-27, 2024 (time zone; 9am PDT → 6pm CEST)
- Place:
  - In-person: Hilton Anaheim
  - **Virtual: Zoom links in the daily agenda**

## Schedule

| Date | Time (PDT) | Location | Zoom Link | Notes |
|---|---|---|---|---|
| **9/23 Mon** | 2:00pm - 4:00pm | 4 Concourse Level - Manhattan | Link (passcode: 2024) | Audio WG TPAC Kick-off<br>Focused discussion<br>- Handling problems in AudioContext<br>- AudioContext interrupted state<br>- Playout Statistics API<br>- Redesigning issue labels in the issue tracker |
| | 4:30pm - 6:00pm | 4 Concourse Level - Manhattan | | Open discussion on issues<br>- How to triage issues in Audio WG<br>- Chromium's "default" audio device label<br>- What to do with proposed change markup |
| **9/24 Tue** | 2:00pm - 4:00pm | 4 Concourse Level - La Jolla | Link (passcode: 2024) | Audio WG: What's next?<br>Focused discussion<br>- The future of Web Speech API<br>- Web MIDI API status report<br>- The journey of Render Capacity API<br>- Web MIDI API issues resolution |
| | 4:30pm - 6:00pm | 4 Concourse Level - La Jolla | | Open discussion on issues |

| Date | Time (PDT) | Location | Zoom Link | Notes |
|---|---|---|---|---|
| | 7:00pm - 9:00pm | "A restaurant" | N/A | Audio WG/CG social event |
| 9/25 Wed | | N/A | N/A | No Audio WG/CG activities |
| 9/26 Thu | | N/A | N/A | No Audio WG/CG activities |
| **9/27 Fri** | 2:00pm - 4:00pm | 4 Concourse Level - San Clemente | Link (passcode: 2024) | Audio CG Discussion |

# Daily Agenda

## 9/23 Audio WG Day 1 (W3C event, minutes)

- Audio WG TPAC Kick-off – Hongchan Choi (14:00-14:20)
  - Safety reminder: https://www.w3.org/2024/09/TPAC/health.html
  - Code of conduct: https://www.w3.org/policies/code-of-conduct/
  - Audio WG introduction: https://www.w3.org/groups/wg/audio/
    - Web Audio API
    - Web MIDI API
    - Audio CG
  - Audio WG members/participants introduction
  - Audio WG Announcements
    - Paul Adenot (Mozilla) is a new WG co-chair!
    - Steve Becker and Gabriel Brito (Microsoft) joined Audio WG. Welcome!
  - The overview of TPAC 2024 Audio WG activities: agenda and schedule
- Focused discussion (14:20-16:00)
  - Handling problems in AudioContext – Michael Wilson
    - Next step: Use `onerror` event handler when reporting erroneous situations from the AudioContext constructor. The WG decided not to use other models (e.g. exceptions).
  - AudioContext interrupted state (explainer) – Gabriel Brito
    - Next step:
      - The state transition from "suspended" to "interrupted" will be stalled until subsequent resume() is called. All other state transitions in the proposal look great.
      - The AudioSession API (Media WG) is the best place to specify the "interrupted" state handling. The Web Audio API can use the spec by linking to it.

            - ~~[DONE] Youenn (Apple) will open an issue to the AudioSession API issue tracker.~~ – issue
    - Playout Statistics API – Guido Urdaneta
        - Next step:
            - The WG agrees that this API (along with the Render Capacity API) solves real problems.
            - We need to understand why this is not as problematic privacy-wise as Render Capacity API.
                - Chromium implementers might want to reach out to the internal privacy/security experts for further clarification.
                - Check with PING and get their view on the API.
    - Redesigning issue labels in the Web Audio API issue tracker – Michael Wilson
        - TPAC 2024 Web Audio Github Label Proposal
            - References:
                - https://www.w3.org/Guide/documentreview/#working_with_horizontal_review_labels
                - https://speced.github.io/spec-maintenance/
                    - https://speced.github.io/spec-maintenance/WebAudio/web-audio-api/
                    - https://speced.github.io/spec-maintenance/about/
        - Next steps:
            - The proposed label system looks good.
            - [DONE] Michael Wilson (Google) will implement the change. – labels
- Open discussion on issues (16:30-18:00)
    - How to triage issues in Audio WG (~20m)
        - Moving forward the WG will be using Michael's new plan.
        - Utilizing Issue Tracker, Discussions, and Projects?
            - https://docs.github.com/en/discussions
            - https://docs.github.com/en/issues/planning-and-tracking-with-projects
        - WG discussion VS Incubation
        - Work size estimate is important: let's make the contribution easier!
        - Next steps:
            - Use the Issue tracker for all WG discussions with proper labeling.
                - For incubation issues, we can use milestones.
                - For any "Needs edits" issues, the work side estimate is required.
            - Any developer questions will be routed to "Discussions".
    - Chromium's "default" audio device label (~30m)
        - References:

            - https://github.com/WebAudio/web-audio-api/issues/2569#issuecomment-2110286417
            - https://w3c.github.io/mediacapture-main/#creating-a-list-of-device-info-objects
            - https://github.com/w3c/mediacapture-output/issues/133
        - Next steps:
            - `AudioContext('')` will behave the same with `AudioContext()`.
                - WG agrees that MediaElement also should be the same.
            - Each UA can use the sinkId enumerated by itself. For example, if the UA generates "`default`" sinkId, the AudoContext in the same UA can use it.
                - Chromium has the "`default`" label for a historical reason (currently unknown), but the WG believes that it is not necessary to clarify it on the spec.
            - ~~[DONE] Youenn will file an issue to the "mediacapture-output" spec to clarify what the empty string means; the Web Audio API then can link to it.~~ – issue
    - What to do with proposed change markup in the spec document
        - Chris Lilly suggested two options for the WG:
            - Simplifying the Updatable REC Process (breakout session idea)
            - Publish a FPWD of Web Audio 2.0 and use it for future iteration/incubation
        - Next steps:
            - Soft WG consensus on using FPWD for the spec update that is faster and easier.
            - The WG should consider how to message this change carefully. For example, calling it "V2" might mislead people to assume this is a breaking change.
- TPAC Triage Session: TPAC 2024: Web Audio/MIDI API Issue Triaging (16:30-18:00)
    - WG Goal: For any issues with "Ready for editing" or "Needs Edit", update the issue labels and produce work estimates (S, M, L).

## 9/24 Audio WG Day 2 (W3C event, minutes)

- Audio WG: What's next? (14:00-14:30)
    - The current charter has been extended until December, and this is to help the rechartering process.
    - The new charter draft is currently in the grace period after gathering feedback from ACs:
        - The notable change is how the WG will handle the incubation of the new projects, and the WG has seen no objections so far.

    - Next steps:
        - This decision is based on the W3C guideline.
        - The WG made a resolution to move forward with FPWD of the new version, v1.1.
        - Hongchan will do:
            - Keep the current spec source as is.
            - ~~After the approval of the issue~~ (approved), publish the FPWD to https://www.w3.org/TR/webaudio-11.
                - The WG will communicate this change via w3c mailing lists (public-audio@, etc) and social media outlets.
                - ~~[DONE] A repo tag created.~~
                - [DONE]
            - Make a duplicate of the spec source and remove all the proposed addition/corrections markup.
                - V1.1 doesn't need a change log, because it's the "new beginning".
- Focused discussion (14:30-16:00)
    - The future of Web Speech API (slide, breakout session)– Evan Liu (Google)
        - The WG showed general consensus on the proposal and the ownership of this API work.
            - There is more discussion to be done, and the WG agreed that it can be explored in the Audio CG. (e.g. fingerprinting issues)
        - Next steps:
            - Hongchan asked Evan to join the WG to drive this project.
            - When the Audio CG finishes the spec design discussion, the Audio WG can start the rechartering process to include it to the WG's scope/deliverables.
    - Web MIDI API status report – Michael Wilson (Google)
        - The WG acknowledged the lack of proper testing infrastructure, but this shouldn't be blocking the publication. The WG can simply mention the concern in the implementation report.
        - Next steps:
            - Publish the next working draft, and call for horizontal review. After addressing/responding to any issues arise and then go to CR.
    - The journey of Render Capacity API – Hongchan Choi (Google)
        - The presentation discussed the journey of launching this controversial feature in the Chrome team.
        - Next steps:
            - If PING review on Playout Statistics API gives green light, the WG will remove this API from the specification.
    - Web MIDI API issues resolution

        - TPAC 2024: Web Audio/MIDI API Issue Triaging (MIDI tab): discussed 4 issues.
- TPAC Triage Session: TPAC 2024: Web Audio/MIDI API Issue Triaging (16:30-18:00)
    - WG Goal: For any issues with "Ready for editing" or "Needs Edit", update the issue labels and produce work estimates (S, M, L).
        - Over 2 days the WG discussed 34 issues and closed 2 issues.

### 9/27 Audio CG Day (W3C event, Minutes)

- Agenda
    - New advanced Web Audio API application roundup
        - Wwise on the Web, first game shipped!
        - …
        - Please add below
    - New Web technologies that pair well with the Web Audio API
        - WASM multi-memory
        - emscripten AudioWorklet integration
    -

# Minutes

Legend:

- 📡remote participation

### 9/23 Audio WG Day 1

- Attendees
    - Hongchan Choi (Google)
    - Paul Adenot (Mozilla)
    - Michael Wilson (Google) - Volunteered to scribe. Thanks!
    - Ryoya Kawai (NTT)
    - Steve Becker (Microsoft)
    - Gabriel Brito (Microsoft) 📡
    - Jean-Yves Avenard (Apple)
    - Youenn Fablet (Apple)
    - Chris Lilley (W3C)
    - Guido Urdaneta (Google)
    - Elad Alon (Google)
    - Matt Paradis (BBC) 📡
    - Christoph G. (W3C Invited Expert) 📡
    - Evan Liu (Google) 📡

    - Thomas Dobbs (freelance developer) 📡
    - Nishita Burman (Microsoft) 📡
- Notes
    - Introduction
        - We will scribe notes here instead of IRC
        - Agenda: three presentations with discussion, open discussions motivated by ~83 open issues remaining on the specification issue tracker
        - TPAC safety information and code of conduct (see links in agenda section above)
        - Introductions of attendees
        - Paul Adenot is a new Working Group co-chair!
        - Welcome Steve Becker and Gabriel Brito to the working group!
    - **Focused Discussion #1: AudioContext exceptions**
        - Paul: We will need a two-step error reporting mechanism.  First: if we can statically determine if the deviceID is not corresponding to a device on the machine we can throw synchronously.  There is an unsolvable problem where if you make an asynchronous call and then remove the device physically then it will fail, so next best thing would be to fire onError.
        - Paul: No backwards-compatibility problem because it is only for non-default device which was not possible previously.
        - Paul: Some implementations will, when all audio devices is removed, will switch everything to a system thread and if you find a new audio device then it will reopen it and continue silently (Chromium and Firefox implement this).  This is when you use the default constructor.
        - Hongchan: We use low-priority fake audio worker thread for the silent thread in Chromium.
        - Michael: What is returned by the construction?
        - Paul: Nothing, when an exception is thrown nothing is returned and the line doesn't execute.
        - Youenn: Is it vital for applications to set sink ID on the constructor?
        - Paul: Important to set the initial sample rate, and to fetch attributes from the device, where we want to set these from the start
        - Youenn: Alternative is to attempt to set the ID, and receive onSetSinkChanged event
        - Hongchan: Sudden device disconnection will be reported by onerror
        - Youenn: We will reuse it for audio context constructor
        - Paul: Yes. And onsinkchange is also for changing environment if device changes
        - Near consensus:
            - Throwing exception from Constructor for synchronous failure
            - also dispatch onerror for asynchronous failure; the instance will non-null value, but it won't be functioning

            - not closing the context, it will be suspended state
            - sinkid property = null but resources are not acquired
        - Discussion 2/2
            - Youenn: we need to spec how things are cached otherwise different implementations will see different behavior here
            - sinkid is per origin
            - Thomas: Should we get the same ID every time you plug in the device?
                - Yes, you should (there may be a bug on Linux Chrome though)
        - Skipping the cache altogether, and just raise onerror
            - Christoph: It would be better than having to refresh the cache
        - Summary
            - Using `onerror` only. Treat all the problems asynchronously: don't use other models.
- **Focused Discussion #2: AudioContext interrupted state**
    - Gabriel presenting slides
    - Issue 2392 on the WebAudio repository
    - Youenn: There is the AudioSession API that exposes interrupted state. If the AudioContext is not running, then the user agent will not interrupt it. If the AudioSession is not playing sound, it will not be interrupted. If we specify it this way, we may not need the boolean flag. Puts burden on user agent. Resuming applies the same rules.
    - Steve: there is a state where the context is running but silent and interrupted and if starts to play audio we will let it play silently?
    - Youenn: The system will not see the page as playing audio, so will not interrupt, at least on iOS. For instance, may want to use WebAudio to do things on the microphone track and not render anything.
    - Paul: In Firefox we have something similar that if it is silent for some time we close the hardware to save power, and emergency reopening, but that is transparent to the webpage. It is a power optimization.
    - Hongchan: No special treatment for that kind of audio stream?
    - Paul: If it's silent and suddenly not silent we quickly reopen the stream
    - Hongchan: We have a system in Chromium called silent sink suspension, after 30 seconds we change to a worker thread. Invisible to the API.
    - Paul: What about immediately changing to interrupted?
    - Gabriel: We can do it, but I changed it because of the privacy concern.
    - Youenn: The privacy considerations are valid, and we have to handle them in AudioSession, but it is the responsibility of the user agent.
    - Paul: Can specify it in the simpler way with an informative note about the privacy concerns
    - Hongchan: What is the privacy concern?

        - Youenn: If the webpage is not playing audio and you get a phone call, the page will know.  If you are playing audio, then it's unavoidable that the webpage will know.
        - Summary
            - If suspended no transition happened, the resume is called either it doesn't resolve immediately
            - suspended -> interruption: transition to interruption when it resume()
            - AudioSession may be the best place to specify interrupted state handling, and WebAudio can hook into it (Media WG / WebRTC WG)
            - Youenn will file an issue on the AudioSession spec
    - **Focused Discussion #3: Playout Statistics API**
        - Guido: Example where this was helpful: there was a performance regression in an OS API, and glitch metrics enabled application developers to understand and work around this
        - Youenn: Is this observable in an AudioWorklet?  Is it something where the definition of a glitch might be different with different user agents?
        - Guido: definition of a glitch is when we send the audio to the system, if we don't have frames provided and have to provide a fallback frame we will increment this counter
        - Hongchan: So audio graph overload, that would include AudioWorklet overload
        - Christoph: You cannot see anything from the AudioWorklet, since time keeps increasing
        - Paul: There's not going to be holes, because by spec there can't be holes. You will see the currentTime incrementing slowing, one second will not be one second
        - Youenn: You will see a drift
        - Paul: What is the problem we are attempting to solve?
        - Guido: Let applications be able to detect regressions in audio quality and give them the flexibility to do something about it
        - Guido: It will not be practical to do something live, but if they observe in telemetry they could do something about it in the application
        - Paul: Sources could be: application is doing too much processing.  This is actionable by removing processing, use higher latency, etc. Others you cannot do anything in the application level but you would like to know, such as OS bugs.
        - Guido: Sometimes you can work around it, or fix the OS bug
        - Paul: Webpage showing a popup for instance if Bluetooth device is too far away and can notify the user

        - Guido: Application could do this kind of thing, but main motivation is reports in aggregate
        - Paul: Privacy issue, why do we need a number instead of an API that just says glitches have occurred
        - Guido: Number gives you an indication of the magnitude of the problem
        - Youenn: With bluetooth, can have a glitch for a small amount of time, how often to fire event
        - Elad: If you set a threshold before firing event it could not be low enough to find issues in aggregate such as 3% increase in issues. This could be missed by the event.
        - Paul: what if you listen to glitches on two different origins.
        - Michael: Thresholding, the only threshold that makes sense since different applications have different thresholds is no glitches or some glitches
        - Hongchan: This passed Chromium internal review
        - Guido: PING is also informed of this
        - Youenn: You can already do this with performance.now and AudioWorklet
        - Hongchan: Guido has a good point that some applications do not use an AudioWorklet
        - Guido: AudioWorklet is unergonomic
        - Paul: Tell the people who rejected the proposal
        - Paul: What if the audio graph worked fine but the bluetooth device for example has glitches. Is that included in the counter?
        - Guido: Yes.
        - Paul: So it is exposing new information, since this can't be detected by WebAudio
        - Hongchan: Does this belong somewhere outside of WebAudio then?
        - Guido: This is for the device used by WebAudio
        - Youenn: We are interested in stats that are actionable by the user that is using the webpage, less interested in stats that are only for other users.
        - Paul: Want to better understand privacy implications, how is this different from RenderCapacity from a privacy standpoint?
        - Paul: Do we know any developers that have tried already with the existing API?
            - Nobody knows of this
        - Summary:
            - There is a problem that this solves
            - Unknown: why isn't this as problematic privacy-wise as RenderCapacity
        - Next steps:
            - Check with PING and internally with browser privacy teams
    - **Open Discussion #1: How to triage issues in Audio WG**

- General consensus is implement label plan: TPAC 2024 Web Audio Github Label Proposal
- How to use discussion board?
- Chris: Most things should be issues, some groups push discussions when issues pile up but there are some things that should be discussions (just topics)
- Hongchan: That's our intention, not to replace issues. More of a Q&A thing.
- Hongchan: Can we do some incubation of features / ideas? Why not use the CG for that? But we don't have any repository / place to do this.
- Paul: People tend to file on the main tracker.
- Hongchan: CG / incubation can happen in discussions instead of the issue tracker.
- Paul: At the end of the day doesn't change much, but if we have clear labels
- Hongchan: Different buckets: discussion bucket and issue bucket. Then have a feature to elevate from discussion to issue tracker.
- Michael: What about using milestones?
- Hongchan: I also thought about that, have a milestone for charter deliverables and a milestone for incubation. Or we could have a project.
- Paul: I'm not picky as long as we have a decision. Michael's labels are good.
- Hongchan: How to start discussion of new idea?
- Paul: We used to start an issue labeled as discussion / future development. What difference does it make? Discussion is more geared toward getting answers. You can vote up an answer.
- Hongchan: we can point people to discussions when they have usage questions instead of closing the issue. And use milestones for incubation.
- Michael: Can we transfer issues to discussions?
    - Yes, it's on the right side of the panel in the issue UI
- Michael: what's the Projects tab for?
    - Hongchan: can use Gantt chart, etc. maybe not so useful for us
- Hongchan: What to do about the markup?
- Chris: Works well for simple textual spec, such as changing SHOULD to MUST. But when you have a complicated cross-linked spec it doesn't work well and links go to the wrong place, and they don't work, and it's a mess. Some people propose writing a standalone spec and tooling will figure it out, but it's not clear how that will work.
- Chris: Goal was to simplify things and make recs that are less out of date. I think it's slowing us down, it would be simpler to copy the spec and slap

new features into it and go for a new first working draft. Say this was an interesting failed experiment.
    - Paul: What about if we have a rec and want to change the rec, do we kick off a new horizontal review?
    - Chris: Candidate vs proposed. Proposed triggers an AC review. Not normative / covered by patent policy until it's made normative. Nowadays as soon as we go to candidate recommendation we get patent protection. So go for first working draft.
    - Paul: It's subjective how many changes are necessary?
    - Chris: Yes, it's up to the group.
    - Paul: We could have time-based and look at list of things.
    - Chris: Some people maintain different levels of specs.
    - Paul: For us it's fairly linear, and we work together fairly closely.
    - Michael: I don't think there's anybody working off the Recommendation version of the spec.
    - Paul: They shouldn't be. It's just for patent protection.
    - Hongchan: Updatable spec is just an idea?
    - Chris: It has been used, but fairly small-scale. But even there, there were problems with things having to sync up.
    - Hongchan: so our idea is to move forward to publishing a new working draft. 1.0, 2.0?
    - Chris: As long as we message it properly, that there's no semantic difference, then it's ok.
    - Chris: What do editors think about the markup?
    - Paul: It's too complicated.
    - Hongchan: It's a mental blocker for me, even if I have a small change I want to make I don't want to deal with the markup
    - Chris: I was worried about this.
    - Hongchan: How many groups are using the markup?
    - Chris: Quite a few.
    - Chris: The way the charter is worded, anything that is part of an existing Audio WG project is fine to include but new functionality may need different language.
    - Hongchan: This is something we can discuss tomorrow, since speech is not under any WG and we may want to include it in Audio WG
    - Chris: My preference is to not hold up the charter for that.
        - General agreement with this
- **Open Discussion #2: "Default" label**
    - Yoan: We're using empty string
    - Guido: On Windows Chromium exposes "default" and on windows "communications"

- Guido: WebAudio spec says it is an exposed device by enumerateDevices so it should be allowed, not that "default" is a special string
- Yoan: proposal for Jean-Yves that if the default device is not exposed then add another device that would be the default if it's not already exposed.
- Yoan: Should we specify that all browsers have this special device that tracks the system default device?
- Paul: plugging / unplugging, and follows the system default.
- Yoan: issue is exactly this: we have two entries: default, and the actual device. But other browsers that don't expose it need to add empty string themselves and pass empty string to setSinkId.
- Guido: System-level default device is different from "use this device which is currently default" which does not track the system device changes. And the UI default could be either concept.
- Paul: The issue is distinguishing between the system default and the device that happens to be the current system device
- Yoan: Is it important to expose?
- Paul: I'm more concerned about input. We don't do that.
- Yoan: We don't do that either.
- Guido: I don't know if Chrome does that, but Chrome does what the system does. I don't think the OS tracks changes.
- Yoan: Webpages have a hard time recovering if the microphone is gone. We added a case where if the default microphone is gone we switch to the new default microphone. In general we don't want to do that much.
- Paul: We push that out, but for Mac we consider the internal microphone and TRS as the same device.
- Paul: For output only it's a bit less problematic. We could consider having a system default that tracks. Do we want it in the enumerateDevices?
- Yoan: That's an ongoing discussion. We want to use the empty string.
- Guido: Empty string in setSinkId media element, means work as if you never called setSinkId.
- Yoan: Also don't have to check if it's a valid device ID, which is nice.
- Michael: I think we should follow another spec.
- Michael: Should we try to standardize the string "default"?
- Guido: We don't need to, that is a different discussion. It's in Chromium for historical reasons.
- Yoan: It's not a UUID
- Guido: Yes, it was before that proposal.
- Yoan: I prefer that we do not specify "default".
- Guido: We could do some experiments. We could spec that IF a default is exposed, it has the ID "default".

        - Yoan: You are doing the same for microphone; there it's harmful for device selection because you have the same entry twice.
        - Guido: Yes, but for historical reasons it's there.  Especially on Windows, don't know the exact history.
        - Resolution:
            - Empty string means "will behave in the same way as creating an AudioContext without anything." meaning, if the output changes it drops to that default.
            - Should be the same for mediaElement.
            - WebAudio should use the list of devices from enumerateDevices.
            - MediaCapture output will file an issue to define what the "default device" means, and then WebAudio spec will link to this definition.
    - **Open Discussion #3: Open Issues**
        - (resolutions noted in bug descriptions)

## 9/24 Audio WG Day 2

- Attendees
    - Hongchan Choi (Google)
    - Paul Adenot (Mozilla)
    - Ryoya Kawai (NTT)
    - Michael Wilson (Google) - Volunteered to scribe. Thanks!
    - Evan Liu (Google)
    - Steve Becker (Microsoft)
    - Jean-Yves Avenard (Apple)
    - Youenn Fablet (Apple)
    - Eric Carlson (Apple)
    - Chris Lilley (W3C)
    - Guido Urdaneta (Google)
    - Matt Paradis (BBC) 📡
    - Thomas Dobbs (freelance developer) 📡
    - Christoph G. (W3C Invited Expert) 📡
    - Nishitha Burman (Microsoft)
    - Chris Wilson (Google)
    - Robert Warren (Glengarry Forestry and Agriculture)
- Notes
    - **Audio WG: What's next?**
        - Currently in extended charter status until December 2024
        - Draft section 3 (Deliverables) has a new section, that we can incubate new features in the Audio Community Group which may later be adopted into new deliverables

        - It has a typo (GC should be CG), we will ping Chris to fix this *(Note: fixed as of Sep 27, 2024 )*
        - This is how we already have been operating, now it is formalized
- How to continue working on the spec? Leaning toward a new First Public Working Draft.
    - Paul: Chris said yesterday to not say version 2, since that implies a semantic difference
    - Hongchan: CSS group is using level-1 level-2 scheme, but I also don't like version 2 which
    - Paul: WebGL2 was incompatible
    - Yuen: WebRTC is adding new stuff, clarifying, fairly heavyweight flow. We are trying to avoid it because it's heavyweight?
    - Paul: Yes, it's unmanageable for us having overlapping things
    - Chris L: (FPWD) requires taking what we have, stripping out the markup, and applying for FPWD, takes a couple of days, then we publish it
    - Hongchan: There is a website which shows possible next steps, which has FPWD if your recommendation doesn't allow it
    - Chris: That's not a requirement, we can still do it
    - Hongchan: Then we just go through the process listed?
    - Chris L: Yes, and it's simple I do it all the time when you decide tell me and I will validate it and commit back or pull request anything necessary
    - Hongchan: What about naming?
    - Chris L: WebAudio API NG, you could call it 2, you could call it 1.1, we can deal with confusion by messaging. We can call it 1.1 for small updates
    - Hongchan: Everyone is opposed to 2, so 1.1 sounds good.
    - Chris: If we call it 1.1 messaging should be ok.
    - Hongchan: What is the recommended forum for communication?
    - Chris L: Announce to public lists, including CG, and social media
    - Chris L: This will start a patent review, only for what was not in the previous version
    - Paul: We have a changelog
    - Chris L: For first public working draft you don't have to put a changelog in the spec. Should have changes since the previous spec though.
    - Hongchan: I can do the markup cleanup and new repo under the audio working group.
    - Chris L: New repo, or new filename under same repo?
    - Hongchan: Can they coexist?
    - Chris L: Yes.

- Paul: Yes, we have that for codecs. There is a lot of tuning but it works.
- Hongchan: What do we show on the landing page?
- Paul: Latest one
- Chris L: It's worth squirreling away a copy of it with all the markup, just in case.
- Chris L: Can we take a resolution now to publish First Public Working Draft? Just need a URL to point to. When it's ready let me know and I will do the transition.
- Chris L: For CSS it's mostly backporting things from later levels, which is fairly simple (doesn't involve IDL, crosslinking, etc.)
- Matt: I'm happy with everything that's being proposed
- Hongchan: Developer perspective?
- Chris L: It's all fine for me
- Thomas: I think it makes a lot of sense. What would be the way to detect which version they're running?
- Paul: In code do the feature detection like any other standard that evolves.
- Thomas: Ok.
- Paul / Hongchan: Most of the work / maintenance will go on the new version
- Hongchan: Management on github?
- Michael: There is a standard set of milestones, so I will set them up.
- Summary:
  - We will make version 1.1
  - github issue linked from agenda is tracking the resolution

- **Discussion #1: New ideas around Web Speech API**
  - Evan: Web Speech API is implemented on Firefox behind a flag, although it didn't work a few months ago.
  - Paul: Firefox supports synthesis but not recognition right now.
  - Evan: Web Speech API is in draft state, want to see if Audio WG could adopt the spec and get it on track. It is quite widely used despite being in draft state.
  - Evan: Not sure if anybody at Apple / Mozilla is working on it
  - Yuen: We are excited about offline speech recognition
  - Paul: For now it was for internal usage, that's what we're looking at right now
  - Evan: Last group was closed after 10 years of inactivity, so nobody is actively working on this API right now. Github issues are out there, nobody has proposed changes to the spec recently.
  - Yuen: Looks good, MediaStreamTrack support is good as well.

- Paul: For us it's marked for internal usage
- Evan: Andreas from Mozilla proposed that
- Paul: He's still around and has the same opinion
- Chris L: How does this relate to other speech APIS? epub speech, css speech level 1, etc.
- Evan: epub speech includes text-to-speech
- Chris L: But no xml, so this is unrelated?
- Evan: Yes.
- Chris L: Speech recognition grammar? (2004)
- Evan: I'm not sure what that is
- Chris L: These questions come up, overlap with other groups, but given the dates on these probably not.
- Jean-Yves: Why would the website decide where speech processing would happen instead of the user?
- Evan: We considered adding browser features, but there are definitely websites that don't want data sent to Google's servers
- Yuen: They are using WebAudio for that.
- Hongchan: Is this method proposal discussed extensively yet?
- Evan: I sent it to Apple and Mozilla, but I think the people who looked at it weren't actively working on it
- Hongchan: Language selection might be a privacy issue
- Evan: In Chrome we are combatting that by requiring user consent to install language packs, which would disincentive use for fingerprinting since it would alert users
- Paul: Always issue with installed languages
- Hongchan: Once the process starts we would get feedback from PING
- Paul: MediaStreamTrack seems really useful
- Hongchan: Issue with removing grammar is no venue for discussion?
- Evan: Right.
- Chris L: Bias is teaching it new stuff?
- Evan: Give it a string and a boost to that string, for instance my last name is Liu so you could bias that and it would be more likely to transcribe than Lu.
- Evan: Depends on the model though
- Paul: Pretty common in models, seems useful
- Yuen: It would be good to check if it's implementable on MacOS and iOS
- Paul: Same, it would take some time to look at the gap but generally in support
- Evan: Next steps to officially adopt the spec?
- Hongchan: We need a discussion, and I already see strong interest from Apple and Mozilla.
- Paul: What other working groups would be suitable, as an exercise?

        - Evan: Media WG
        - Paul: Lot of playback
        - Jean-Yves: Media they do a lot for sight stuff, this seems quite far (from Audio)
        - Yuen: It's another action item to work on, question is if the Media WG would have a better position on these topics
        - Evan: There will be a breakout session tomorrow to go over these
        - Paul: Another candidate is RTC, about the same distance
        - Eric: Probably a bit further
        - Jean-Yves: What about (ptt?)
        - Eric: It was a different group then picked up by (?)
        - Paul: It doesn't seem far-fetched.
        - Chris L: Group has expressed desire to have this as a deliverable. Slightly awkward given the timing. Current charter says "specific functionality for speech" already in "out of scope"
        - Chris L: This would need a complete rechartering, everyone would need to rejoin the group, etc. Do we want to do that now, or get the existing charter under our belt now?
        - Paul: What's the timeline?
        - Chris L: I make a charter fairly quickly, then staff review 4 weeks, then AC review another 4 weeks, then 2 weeks to respond to comments.
        - Hongchan: We're not in a hurry though, just need to find the right home. We can go ahead with the current charter and use the Audio CG to incubate. Not sure about the explicit out of scope.
        - Chris L: Not sure why, maybe because of Time Text Markup to not tread on toes? Copied since the previous charters.
        - Chris L: If we're not in a hurry let's finish the rechartering which gives us two years. When this is ready we can start the process.
        - Hongchan: Can we incubate in the CG?
        - Chris L: Incubation location is not specified, CG is "for example"
        - Hongchan: Is that good? You can join the group too (CG / WG).
        - Evan: Ok.
        - Hongchan: What about speech synthesis? Can we get the MediaStreamTrack from that?
        - Evan: Possibly. There was a github issue
        - Paul: There was, before that the features were siloed
        - Hongchan: By inspecting the PCM data generated maybe you can do fingerprinting?
            - (we will consider this later)
    - **Discussion #2: Web MIDI API Status Report**
        - Chris L: interleaved
        - Chris W: supported

        - Chris L: implementation report: testing needs hardware but there's not much to test
        - Chris L: Implementation report could be WPT report, or it could be a separate document.  For PNG there are some with "colors must be visually distinct" so just wrote out as a document and as a snapshot it's enough.
        - Hongchan: Browser automation group?  They work on test infrastructure.
        - Chris L: absolute next steps are publish a next working draft, call for wide review and give them a certain amount of time, satisfy any issues they raise, then once you're clear of that you can go to CR.
        - Michael: Can you add new things to CR?
        - Chris L: You can.  Just that patent is not covered until Recommendation
        - Chris L: Get rid of issues we can, get horizontal review.
    - **Discussion #3: RenderCapacity API**
        - Hongchan: want consensus from group about what to do with this API (deprecate, continue to pursue, other?)
        - Yuen: What are the issues?
        - Hongchan: Leaky resource attack (link to research paper is on the slide)
        - Yuen: It seems like the first value is the issue, what about the underrun ratio?
        - Hongchan: The underrun ratio might be ok.  But if you carefully threshold the value between a certain range you can get usable data from underrun ratio as well.  Constantly adding nodes until we see different values.
        - Chris L: Another option from a popup is to make a setting where the user has to set a pref in the browser, but that's only useful for developers debugging their own things.
        - Hongchan: One application is videoconferencing
        - Chris L: Example from CSS spec webfonts: at some point in Safari if the font isn't installed you have to use webfonts.  It's good for privacy, but breaks Web for people with minority languages.  That has been deadlocked for five years.  Can consume a huge amount of time.
        - Hongchan: I experienced many times in the process where nobody knew what to do next, that's what I meant by a framework.
        - Chris L: When you go to Candidate Recommendation they will look if there are labeled issues open, it's possible to proceed anyway even if there is an identified issue.  As long as you can justify it.
        - Guido: Have you considered tying it to the microphone permission?
        - Hongchan: That was definitely discussed, and we actually got approval twice but the security team reverted their decision.
        - Guido: The attacks were to find who the user is, but if you have the microphone you can hear their voice so you know who they are.
        - Hongchan: Not disagreeing with you.

        - Steve: Seems similar to compute pressure API, but they have just 4 buckets
        - Hongchan: (security reviewers) said in this case since it's just one real-time high-priority thread instead of general CPU usage, it's much more sensitive.
        - Yuen: Is the spec clear about the privacy issues?
        - Hongchan: There was a review, but (?)
        - Thomas: Is the only alternative now to measure the current time versus the expected time? Do we need to know the percentage if it's under 100% or do we just need to know it's not running?
        - Hongchan: There will be some inaccuracy involved, because of the messageport.
        - Paul: We can do the demo you showed (of the exploit) today. We don't need a new API.
        - Yuen: We can always make post message longer
        - Paul: You don't need post message, you can do shared array buffer. More accurate.
        - Hongchan: Shared array buffer is gated
        - Paul: Even if you jitter, you can run longer and statistically you can recover the information. Slows it down, but 40-minute WebRTC call…
        - Christoph: Would it be negative or positive to prove that we can do it already, in terms of existing APIs?
        - Paul: If you have a resource, someone who is data-mining can see there is less of that resource. It's just another shared resource.
        - Michael: I think we don't need both PlayoutStats and RenderCapacity
        - Paul: PlayoutStats is basically underrun ratio, right?
        - Guido: It's basically the same information
        - Hongchan: It can be computed
        - Guido: Was RenderCapacity just webaudio or all audio?
        - Hongchan: Just webaudio
        - Hongchan: Next steps is see what PING review for PlayoutStats API?
        - Yuen: That would mean removing it from the spec?
        - Hongchan: Yes. That is the decision I want to make.
        - Yuen: (agree)
        - Paul: (agree)
        - Decision: remove it after we get green light from PING on PlayoutStats
    - **Open Issues**
        - (see results in each bug's comments)
    - .

# 9/27 Audio CG Day

- Attendees
    - Paul Adenot (Mozilla)
    - Matthew Paradis (BBC)📡
    - Hongchan Choi (Google)
    - Michael Wilson (Google)
    - Ryoya Kawai (NTT)
    - Christoph Guttandin (W3C Invited Expert)📡
    - Philippe Milot (AudioKinetic)📡
- Discussion topics
    - New advanced Web Audio API application roundup
        - Wwise on the Web, first game shipped! 🎉
            - We can follow up with Philippe Milot (AudioKinetic) later.
            - Unity, Steam (web version is free)
            - Potions: A Curious Tale (steam)
                - https://kozee.games/
            - Web port took ~3-4 weeks
            - Unity is single-threaded, so loading large assets can cause audio to skip (not Web Audio API issue, limitation of Unity)
            - The SAB gatekeeping is still a problem for game developers.
            - Questions
                - What's the next for AudioKinetic on the Web?
                    - 1) first class support for downloading assets from the server (vs relying on unity or etc)
                    - 2) native JS API/binding of Wwise API
                - any WebCodecs? (no)
        - Web Audio Modules (Michel Buffa)
            - Recently ported Prophet-5 to the web (!)
            - Worked out sample-accurate automation, keeping things in phase, full-featured MIDI features (recording, editing) with latency compensation
        - Big names in Audio Effects going to WebAudio API
            - Antares Auto-Tune shipping in MStudio
        - Ableton Tuning
        - Interest in continuing the Web Audio Weekly newsletter in CG?
            - Chris Lowis is probably busy (last edition Jun 22, 2023)
            - Can CG continue this effort? It has been useful for many people.
        - KORG Katokatone (Japanese news release)
            - Educational software for tablets, including MIDI support
            - Using WASM
        - Rust Web Audio API implementation with JS bindings (Otto Rottier)
            - Version 1.0 two weeks ago! 🎉

                - (mostly?) feature-complete / spec compliant
                    - Includes RenderCapacity, convolver
                - Used in production in IRCAM installation, works well for long periods of time
                - Can run WPT almost without modification
    - New Web technologies that pair well with the Web Audio API
        - WASM multi-memory:
            - Can help skip a significant number of copies
        - SIMD in WebAssembly: audio transcription
            - 32-bit 4x4 SIMD on the web also (compile emscripten SIMD to SIMD WASM)
        - GC integration in WebAssembly:
        - emscripten AudioWorklet integration
        - WebCodecs: copy into / out of various sample types
            - Safari looking to ship soon
            - Moving to CR 🎉
        - WebRTC has standardized how to get audio input latency
        - Questions
            - "Growing SAB" is it in-place growing or reallocation/cloning?
    - Deficiencies / Missing Technologies
        - Not much progress on read-only memory for WebAudio / WebCodecs (for sharing between threads)
            - https://www.w3.org/2021/03/media-production-workshop/session-1.html#webcodecs-copies
            - https://www.w3.org/2021/03/media-production-workshop/session-2.html#webassembly-copies
    - CG operation moving forward
        - How to handle the new feature incubation in CG
            - Web Speech API (Evan Liu)
                - The discussion will happen in CG, and we might be able to change the meeting schedule to accommodate the discussion speed.
            - Sync on Web, now and next of realtime media services on web in Audio CG?
                - What do we need to synchronize everything?
                - Still in prototyping / discussion stage
                - Is Audio CG the right place?
                    - Multi-device Timing Community Group ?
                - Can come to Audio WG for new APIs if necessary
                - Clear requirements will help with designing (latency, behavior under packet loss, etc.)

        - Next steps: Kawai-san will follow up with the CG contacts after the internal team discussion.
    - Meeting cadence and other logistics
        - Web Speech API - needs a dedicated meeting cadence (coordinate on CG meeting list)
        - Sync on Web - circle back when the Kawai-san's team has a proposal/pitch deck?
        - logistics:
            - For scheduling: W3C event scheduling system
            - Agenda/artifacts handling: TBD, make decisions at the first meeting

# References

## Meeting Room Map (PDF)

## Other Notes

- https://w3c.github.io/charters-dashboard/