

GPU Web 2018-06-13

Chair: Corentin

Scribe: Dean

Location: Google Hangout

[Minutes from last meeting](#)

TL;DR

- Still waiting on the CLA.
- Next F2F may or may not be at Apple Park end of September
- We'll wait to have resolved most of the graphics issues before filing for a TAG review.
- [PR 57](#): Usage validation rules
 - Only races are UAVs in render passes and UAVs in a single dispatch.
 - Sounds ok to people, Apple to review.
- [PR 58](#): Changes to WebGPU device creation
 - Use a mixin on the window object to add ``webgpu.getAdapter``
 - Might need a promise for `getAdapter` or `createDevice` at some point.
 - Unsure how to move data between adapters, especially a problem with video.

Tentative agenda

- PR 57: usage validation rules
- PR 58: Changes to WebGPU device creation.
- Agenda for next meeting

Attendance

- Apple
 - **Dean Jackson**
- Google
 - **Corentin Wallez**
 - **Dan Sinclair**
 - **James Darpinian**
 - **Kai Ninomiya**
- Microsoft
 - **Chas Boyd**
 - **Rafael Cintron**
- Mozilla

- **Dzmitry Malyshau**
 - **Jeff Gilbert**
 - **Markus Siglreithmaier**
- **Yandex**
 - **Kirill Dmitrenko**
- **Elviss Strazdiņš**
- **Joshua Groves**

Administrative updates

- Skip upcoming meeting on July 4th. (US Holiday)
- CW: Most companies have responded to the CLA review. Just waiting on that to happen.
- CW: Google have sent an “intent to implement” to the blink list for WebGPU. Ian Kilpatrick suggested to us that we should file for a TAG review now, rather than later. But we think it might still be a bit early since much of the API hasn’t been set. Any thoughts?
- JG: Sounds a bit early, but that might just be a difference in process between Google and Mozilla. We’d get lots of questions about how things would be done, with “I don’t know yet” answers. I’m thinking of waiting until the MVP spec is mostly done and a couple of implementations.
- KN: Some Google folks are saying we would benefit from early TAG review. I disagree and think it would just be more noise. We want to agree more on core concepts before getting review on API shape.
- RC: Waiting is fine as long as we’re ok with things possibly changing later. My TAG rep isn’t a graphics person so would appreciate lots of explainer text. If we do it now we’d just get a bunch of questions about terminology
- CW: I don’t think the TAG will have an opinion on core graphics... hopefully.
- CW: So we’ll wait before sending to the TAG for review.

PR 57: usage validation rules

- <https://github.com/gpuweb/gpuweb/pull/57>
- CW: Is everyone clear on proceeding along these lines?
- RC: Is the plan to have barriers done manually?
- CW: The first iteration has compute passes with implicit UAV barriers. The explicit barrier discussion was moved to the bottom of the document, and left out of MVP (deferred to V1).
- RC: Does that mean for an MVP, during a pass there can be data races, but between the implementation will add barriers?
- CW: No. In-between passes there can be no races. The only race in the MVP can be between a compute pass and its dispatch.
- RC: Everything else is handled by the implementation?

- CW: Yes, except UAVs inside render passes.
- RC: I will read the pull request and reply.
- CW: Apple input?
- DJ: We'll respond within a day or two.

PR 58: Changes to WebGPU device creation.

- <https://github.com/gpuweb/gpuweb/pull/58>
- CW: I started to put blink code in for implementing device creation. I noticed a few things that don't make sense in the IDL.
- JG: I was doing something similar in Gecko.
- RC: I see you commented out the limits on Adapter and createDevice, but not on the Device
- CW: You cannot configure the limits, but you can query them via the device.
- CW: The driver behind this was to provide a way to create a device. The current trend is a mixin on Window or Navigator. There were also some statics that didn't make sense.
- JG: Yeah. It seems more trendy at the moment to have an instance of WebGPUDevice rather than a bunch of static methods on the class.
- CW: Yes, gamepad and performance are like this.
- CW: Do we want to make some of the getters (createDevice, etc.) async (via promises)? They are expensive.
- JG: No, we don't. I don't think they are that expensive.
- CW: Creating a Vulkan device can take maybe 100s of ms.
- JG: Then maybe.
- KD: Yeah, I don't want to block for that long.
- JG: Generally if it can take longer than a frame, then we should make a promise. Please use a separate pull request.
- RC: How about multiple GPUs?
- JG: Some way to choose between multiple adapters.
- RC: Will we expose adapters that the page isn't using yet?
- JG: Yes.
- RC: What will we do if the page is using Adapter A, but we request Adapter B. How will the data get between A and B?
- JG: Yes, we should move it.
- C: You can't share DXGI Surfaces between adapters on D3D.R
- CW: Well..."bummer"
- JG: Ideally once Firefox has DirectComposition running, we want to enable WebGPU / WebGL on discrete and the rest on integrated.
- RC: Works when WebGPU doesn't interact with the rest. But causes problems when you interact with video and images. Especially video.

- JG: Suspect eventually we will need to have a “video bitmap” concept where the Web app pulls video frames itself with a specified adapter. We assume it is fast at the moment, but that won’t always be true.
- CW: Hopefully we won’t need to do CPU readbacks, but maybe we can’t avoid it.
- CW: OK, accept pull request 58. Make a new one for promises. Worry about video/dom to texture later (for multiple adapters).
- JG: What does replaceable mean? I suggest making it SameObject.

Next F2F Meeting

- CW: When should we meet again? At the next Khronos F2F.
- CW: Let’s meet around the end of September.
- DJ: I’ll look into getting space at Apple Park.

Suggested agenda for next meeting

- JG: Shading?????? Languages???????