WebGL + WebGPU Meetup

Winter 2023

January 31, 2023

KHRONOS GROUP / WEBINARS & MEETUPS

WebGL & WebGPU Updates

Ken Russell (Google) and Kelsey Gilbert (Mozilla)
On Behalf of the WebGL WG and WebGPU CG

KHRONOS GROUP / WEBINARS & MEETUPS

Agenda

Join WebGL & WebGPU Communities

WebGL Updates

- General WebGL Updates
- Shader Pixel Local Storage Extension Proposal
- Provoking Vertex Extension
- Clip Distance Extension Draft
- Depth Bias Clamp Extension Proposal
- ANGLE/Metal Progress

WebGPU Updates

- Standardization
- Implementations
- Resources and Contributing

khr.io/web202301

Join WebGL & WebGPU Communities

- The WebGL and WebGPU APIs are supported by vibrant online communities!
- If you're developing with these APIs, we would like to hear from you!
- On the WebGL side:
  - Please consider joining the WebGL Dev List: announcements of products, demos, new tools, job postings, questions, discussions - all are welcome!
  - Khronos' public_webgl mailing list hosts lower-traffic spec announcements
  - The WebGL Matrix chat room offers a way to talk with browser implementers and other developers
  - You can find a lot of cool stuff by searching #webgl on Twitter 😎

khr.io/web202301

KHRONOS GROUP

Join WebGL & WebGPU Communities

- On the WebGPU side:
  - If you have feedback on the API, please see the main WebGPU "gpuweb" repository for options to communicate it to the community group
  - The WebGPU Matrix chat room (#WebGPU:matrix.org) also offers a great way to talk directly with browser implementers and other developers
  - There's an increasing amount of cool stuff showing up on #webgpu on Twitter 😎
- We all look forward to hearing from you!

khr.io/web202301

General WebGL Updates

Many additions to the WebGL extension specs and conformance tests in the following areas:

- WEBGL_provoking_vertex (formerly EXT_)
- getUniformIndices
- ANGLE_shader_pixel_local_storage
- Immutable textures with compressed formats
- OES_fbo_render_mipmap sample code
- FBO encoding and component type queries enabled by sRGB and floating-point extensions

Thanks to everyone who contributed these improvements!

khr.io/web202301

General WebGL Updates

- Various smaller bug fixes to all browsers' WebGL implementations
- Work ongoing in multiple browsers to support wide-color-gamut with WebGL-rendered canvases

khr.io/web202301

Shader Pixel Local Storage Extension Proposal

- Chris Dalton (Rive) is developing an ANGLE_shader_pixel_local_storage extension for WebGL
- This extension abstracts over many underlying implementation primitives and provides programmable blending functionality to applications
- Implementation is complete in ANGLE; ready for prototyping in browsers
- Follow ANGLE bug 7279 if you're interested in progress on this extension

Provoking Vertex Extension

- WEBGL_provoking_vertex provides control over which vertex initiates a primitive
  - OpenGL convention = last vertex
  - Most other APIs = first vertex
  - Governs the behavior of flat shading
- Availability of this extension implies that it should be used for best performance
- Extension has been community approved and is present in WebKit and Chromium; please use it in your apps

khr.io/web202301

Clip Distance Extension Draft

- WEBGL_clip_cull_distance provides support for hardware clip and optional cull distances
  - New ESSL 3.00 vertex shader outputs and fragment shader inputs
    - gl_ClipDistance[]
    - gl_CullDistance[]
  - New toggleable state CLIP_DISTANCEi_WEBGL
- Enables per-primitive clipping and culling, avoids expensive fragment discard
- Extension is present in draft form (behind a flag) in WebKit and Chromium
  - WebKit Nightly
  - Chromium 112.0.5571.0
- Please try in your apps and provide feedback to browser implementers

khr.io/web202301

Depth Bias Clamp Extension Proposal

- EXT_polygon_offset_clamp adds a new parameter to the polygon offset function that clamps the calculated offset to a minimum or maximum value
- Prevents shadow artifacts by limiting depth bias values, which may get too high for steep polygons
- Trivial adoption
  - **polygonOffset**(factor, units) ⇨ **polygonOffsetClampEXT**(factor, units, **clamp**)

khr.io/web202301

ANGLE/Metal Progress

- Work is still ongoing in ANGLE's Metal backend
- Used by WebKit's WebGL implementation on macOS/iOS, and soon, Chromium's on macOS
- Team proceeding toward shipment in Chromium
- Major recent fixes:
  - Performance improvements for key benchmarks (MotionMark)
  - Disabling Metal backend on unsupported older GPUs
  - Progress on dual-GPU support on Intel/AMD MacBook Pros
- Nearly there!
- Remaining concern and blocker is a startup time regression due to inability to cache compiled shaders with Metal APIs
- Collaborating with Apple engineers on this

khr.io/web202301

WebGPU

An upcoming "modern" graphics API for the Web:

- A successor to WebGL, not a replacement.
- Compute shaders on the Web!
- Lower overhead API
- Foundation for future features (bindless, ray tracing, multithreading …)

Development happens on GitHub and at the W3C

- Anybody can join and participate in the development.
- Thanks to Khronos for hosting us here!

khr.io/web202301

WebGPU standardization updates

Standardization of v1.0 is nearing completion. Blockers are being addressed, and we're polishing the spec and reaching a decent amount of conformance testing. V1.0 specs in 2023Q1 hopefully!

WGSL standardization tackling hopefully its final few major issues:

- Allowing for implementations to trap/discard instead of e.g. clamp out-of-bounds access
- Adding opt-outs for uniformity analysis errors/warnings in fragment shaders
  - Uniformity analysis is still required in compute shaders
- Finalization of syntax for pseudo-generics (e.g. vec3<f32>), leaving the door open to real (and user-defined!) generics in the future

API standardization is also trying to finish up:

- GPUExternalTexture for automagically sampling rgb from e.g. <video> sources
- Nailing down aliasing behaviors
- Other final blockers for shipping our v1! **(last call!)**

khr.io/web202301

KHRONOS GROUP

WebGPU - Implementation status

Firefox

- In Nightly, set **`dom.webgpu.enabled`** to **`true`** in **`about:config`**
- Not yet suitable for general secure browsing with this flag enabled, but expect it to be on by default in Nightly very soon for Windows! Other OSes to follow.
- Shipping to Release closer to the middle of the year!

Chromium

- Windows, ChromeOS and Mac (Linux and Android later)
- The **WebGPU Origin Trial** allows publishing WebGPU apps on the Web today!
  - Breaking API / shading languages are happening by design. You must fix warnings surfaced in the devtools, as these will become hard errors!
  - web.dev/gpu
- Will be released in or around **Chromium 113** this year!

Implementations are mostly interoperable already!

khr.io/web202301

WebGPU - Using it in JS without a browser!

Many reasons to use WebGPU outside of a browser:

- Automated testing.
- Offline rendering using the same tech stack.
- "Native" frameworks like Electron, BabylonNative, etc.

Deno:

- Deno is a Javascript runtime with built-in WebGPU support
- Uses wgpu under the hood.

Node.js

- Dawn has a dawn.node Node.js module.
- In a WIP but fairly good state (99% on par with Chromium for tests)

WebGPU - Partnerships and Collaboration

Steady progress on WebGPU backends for popular web 3D libraries

Three.js, Babylon.js

Ongoing partnerships with teams including Intel, TensorFlow.js, Google Meet, MediaPipe, and more

PlayCanvas has been undertaking a major refactor of their engine in support of WebGPU

Tracking bug: https://github.com/playcanvas/engine/issues/3986

Fantastic feedback and collaboration with Unity, as they investigate porting existing shaders to WGSL and our new Uniformity Analysis requirements!

khr.io/web202301

WebGPU - Resources

Tutorials:

- Get started with GPU Compute on the web by Francois
- WebGPU - All of the cores, none of the canvas by Surma
- Raw WebGPU by Alain
- WebGPU Best Practices by Brandon

Samples

- Check out the up-to-date WebGPU Samples repo (Github) by Austin

khr.io/web202301

WebGPU - Contributing!

Many ways to engage!

- Try the API and provide feedback via one of the channels mentioned
- Try out publishing sites using WebGPU using Chrome's WebGPU Origin Trial
  - Could use WebGPU support in popular frameworks like Three.js Babylon.js and TF.js
- Help with conformance testing
- Contribute sample / demos / articles using WebGPU
- Join the conversations on the Matrix chat!

khr.io/web202301

A recording of this presentation will be available at
https://www.khronos.org/events/webgl-webgpu-meetup-january-2023

For more information on WebGL, please visit
https://www.khronos.org/webgl

Email: public_webgl@khronos.org

KHRONOS GROUP / WEBINARS & MEETUPS

WebGL™ WebGPU