WebGL Working Group Updates

WebGL Meetup, November 2020

Agenda

Cool WebGL Stuff

WebGL 2.0 Universally Available

Improved WebGL Sandboxing in Firefox

Better WebXR Performance in Edge

Higher-Quality Universal Compressed Textures

Higher-Bit-Depth Textures

Improved Geometry Batching

Future Directions

Cool WebGL Stuff

- Lots of great web sites and products using WebGL are released every day!
- Khronos' WebGL working group tries to find and highlight these to the community
- They're sent via email every couple of weeks to the WebGL Dev List
  - Also archived on this blog
- Please join the community, and share your own creations and findings!

WebGL 2.0 Universally Available

- WebGL 2.0 is finally coming to all major operating systems
  - iOS in particular!
- Collaboration with Apple since June 2019
- Integrated ANGLE project into WebKit as the WebGL backend
  - OpenGL / OpenGL ES backend initially
  - Hope to switch to ANGLE's brand-new Metal backend soon
- Available for testing now (November 2020):
  - Safari Technology Preview on macOS
  - Safari in iOS 14.x betas
- Significantly improved WebGL 1.0 conformance
- WebGL 2.0 passing >98% of conformance; aiming for 100%
- Follow progress of the project
- You can now rely on availability of WebGL 2.0 essentially everywhere!
- Upgrade your applications to WebGL 2.0 now!

Improved WebGL Sandboxing in Firefox

- Firefox is strengthening its WebGL security by adding a GPU sub-process
- WebGL API calls are sent to this process for validation and execution
- The JavaScript process no longer accesses the GPU directly
- Shipping today in Firefox Nightly on Windows and macOS!
- Try it and report any issues via product feedback!

Better WebXR Performance in Edge

- OVR_multiview is on by default
  - Improves performance of XR content on the web
- Microsoft is adding support for multiple GPUs to Chromium on Windows for XR scenarios. Try out the experimental support!
  - --enable-blink-features=webxr-multi-gpu command line argument

Higher-Quality Universal Compressed Textures

- **Basis Universal released as open-source!**
  - The advantages of GPU compressed textures, with the file size of JPEGs!
  - Author chooses medium-quality or high-quality, according to application's requirements and file size constraints
- WebAssembly module - works in all browsers
- KTX 2.0 incorporates Basis Universal's technology!
- Using Basis Textures in Three.js
- KHR_texture_basis extension for glTF
- WebGL now supports BPTC (BC6H / BC7) and RGTC (BC4 / BC5) compressed textures
  - Enables higher-quality supercompressed textures on the web!
  - 3D Formats WG preparing Best Practices document on usage - watch for its release!

Higher-Bit-Depth Textures

- EXT_texture_norm16
- Community approved
  - Shipping in Chromium
- Provides new 16-bit signed/unsigned normalized fixed point renderbuffer and texture formats for better memory usage and precision
- Widely available on devices
  - Several driver bugs have been worked around and reported to hardware vendors
- Available in emscripten
- Support for image decoding with 16 bits per channel is being implemented

Improved Geometry Batching

- WEBGL_multi_draw
  - Community approved
  - Reduce the CPU overhead of issuing draw calls
  - Widely supported across devices
  - Results from microbenchmarks are impressive: 3-6x improvements in common case, up to 70x (!) in some situations
  - Available in emscripten
- WEBGL_draw_instanced_base_vertex_base_instance
- WEBGL_multi_draw_instanced_base_vertex_base_instance
  - Help reduce CPU overhead with static batching and provide better instancing functionality
  - Available on most desktop/latest mobile platforms, with emulation when necessary
  - Good results from microbenchmarks; also provide better functionality and simplify applications
  - Test in Chromium with --enable-webgl-draft-extensions
  - Available in emscripten

Future Directions

- WebGPU is coming soon from the W3C. What does this mean for WebGL?

Future Directions

- WebGL will be supported - forever!
  - It is an integral part of the web ecosystem
- Browsers will continue to maintain and improve WebGL implementations
  - Performance and quality are the WebGL WG's highest priority
- WebGL 2.0 is a viable, universal deployment target
  - Graphical applications that need to ship today should target WebGL 2.0
- Encourage you to start developing with the in-progress WebGPU implementations in browsers
  - Babylon.js and Three.js are making tremendous progress on WebGPU backends!

Future Directions

- At the same time, browsers are aiming to transition WebGL implementations to maintenance mode
  - Aiming for WebGL 1.0.4/2.0.1 conformance & spec snapshots first
- Devoting less effort to new features, and more effort to WebGPU
  - Customers' requests for their highest-priority WebGL extensions will be considered on a case-by-case basis

Future Directions

- WebGPU is being developed by all browser vendors at the W3C
- Khronos and the W3C collaborate closely, and will continue to do so
- WebGPU offers high performance, portable GPU compute, and an advanced feature set
- WebGPU paves the way for feature parity with native platforms
  - As an example, see this ray tracing prototype
- Aiming to integrate with best-of-breed tooling to improve the developer experience
- Future meetups are likely to cover both WebGL and WebGPU!
- All are welcome to participate in the W3C's WebGPU community group!

Presentations

Great group of presenters today!

- PlayCanvas
- BlackSmithSoft
- xeokit
- Sketchfab
- vis.gl
- model-viewer
- Babylon.js

We'll answer your Q&A live at the end of the session!