Recent Focus of Working Group

● Continuing to fix corner cases in WebGL 2.0
  ○ Most of these were reported by customers!
  ○ Longstanding OpenGL and GLSL spec issues
  ○ Bugs and missing cases in conformance tests
  ○ Bugs in WebGL implementation and graphics drivers
  ○ All focusing on improving portability and consistency

● Since last year, worked on:
  ○ 57 Github issues, ~220 pull requests, ~660 Chromium issues, 160 Firefox issues, ...
Recent Focus of Working Group

- ANGLE team subsumed all WebGL 2.0 validation
  - WebGL 2.0 is as easy as binding ANGLE to JavaScript
- Microsoft Edge’s WebGL implementation now runs on top of ANGLE!
  - Should be easy to support WebGL 2.0
- Progress on integrating ANGLE into Apple’s WebKit
- Significant ANGLE performance optimizations
Non-Blocking Readback from GPU

- Collaboration between Mozilla and Google
- Helps with readPixels and GPGPU computations
- Synchronous GPU readbacks now async in WebGL 2.0 via getBufferSubData and fences
- Documented in getBufferSubData section of spec
- Benchmark shows improvements
WEBGL_multiview extension

- Major contribution by NVIDIA
- Significantly reduces CPU overhead for WebVR/XR
- Can test in Chromium today
  - Enable WebGL draft extensions in about:flags
  - Make sure passthrough GPU command buffer is enabled
    (--use-cmd-decoder=passthrough)
- See WebGL Aquarium and VR version in WebGLSamples repository (Aquarium link with demo settings)
- Preliminary integration into Three.js too!
WebGL 1.0 Universally Available

Thanks to Florian Bösch, creator of webglstats.com
WebGL 2.0 Becoming Universally Available

Thanks to Florian Bösch, creator of webglstats.com
WebGL Used in Surprising New Fields

Neuroglancer: visualizing neurons’ connectivity (demo)
Many contributions from Intel

- Intel’s Advanced Web Technology Team has made many contributions to the WebGL ecosystem
Intel’s Efforts on Web Graphics

Presenter: Yang Gu (Software Engineering Manager)
Intel Advanced Web Technology Team
August 15, 2018
Why Does Intel Focus on Graphics?

- Graphics has been growing very fast
- Intel has about 70% market share on desktop GPU
- Intel has a good plan on discrete GPU
Team History

- Nov 2011: Began to focus on web
- Nov 2012: Began to work on Chrome, mostly on performance tuning
- Nov 2014 - Jan 2017: Implemented WebGL 2.0 in Chrome
- Jan 2017 - present: WebGL 2.0 Compute
- July 2018: Began to contribute to WebGPU
- Contributed 1400+ patches to Chrome and related projects
Standards Participation

- Intel rep on Khronos WebGL Working Group
- Intel rep on W3C WebGPU Community Group
WebGL 2.0 Compute

- **Desc**
  - Bring OpenGL ES 3.1 (mostly Compute Shaders) to the web

- **Status**
  - Compute Shaders are almost ready for both OpenGL and D3D backends in Chrome behind command line flag “enable-webgl2-compute-context”
  - TF.js can almost run

- **Plan**
  - Implement all compute shader features by end of this year
  - Support WebGL 2.0 Compute in Emscripten
  - Port native dEQP ES 3.1 cases to web
  - Implement the rest of ES 3.1 features according to requirements
Parallel Shader Compilation

- **Desc**
  - Implement KHR_parallel_shader_compile extension so that shaders can be compiled in the background

- **Status**
  - All the work has been accepted by ANGLE

<table>
<thead>
<tr>
<th>Case</th>
<th>Time w/o extension (ms)</th>
<th>Time w/ Extension (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>After the Flood (PlayCanvas)</td>
<td>6223.5</td>
<td>2074.6</td>
</tr>
</tbody>
</table>

- **Plan**
  - Parallelize shader translation (GLSL to HLSL) in ANGLE
WEBGL_video_to_texture

● Desc
  ○ Implement `WEBGL_video_texture` extension to optimize video uploads to WebGL (for 360 degree VR videos in particular)

● Status
  ○ Patches in progress for Skia and Chromium
  ○ First performance improvements (>33%) have been shown
  ○ Still work-in-progress

● Plan
  ○ Land initial patches and continue analyzing performance
WebGL Geometry Shader Extension

● Desc
  ○ Bring Geometry Shader to the web

● Status
  ○ Geometry Shader OpenGL backend has been implemented in ANGLE

● Plan
  ○ Finish the D3D backend
  ○ Enable it in Chrome if there is any requirement
  ○ Add WebGL CTS
WebGPU

● Desc
  ○ Collaborate with W3C WebGPU CG to bring Metal, D3D12 and Vulkan to web

● Status
  ○ Warm up

● Plan
  ○ Contribute to Dawn
  ○ Port Aquarium to Dawn
Web Machine Learning

● Desc
  ○ Bring Machine Learning to web

● Status
  ○ Compute Shader solution
    ■ TensorFlow.js can be accelerated with both OpenGL and D3D backends
  ○ WebML API
    ■ Discussed with some OS vendors on how to define the APIs
    ■ Lots of prototypes and experimental data based on machine learning APIs provided by OS

● Plan
  ○ Compute Shader solution
    ■ Optimize TF.js to show its power
  ○ WebML API
    ■ Work with OS vendors and browser vendors to define the APIs
    ■ Create a W3C CG to move things forward
Native vs. Web Performance comparison

- **Desc**
  - Understand the performance gap between WebGL and native in a quantitative approach
  - Understand the overhead of web platform in depth, like JS engine, security including validation, D3D conversion on the fly

- **Status**
  - OpenGL backend can run on Windows, macOS and Linux
  - Some performance optimizations at native side

- **Plan**
  - Port to other backends
    - Web: WebGL 2.0, WebGPU
    - Native: D3D11, D3D12, Vulkan, Metal, Dawn
  - Explore other workloads

<table>
<thead>
<tr>
<th>Fish Number</th>
<th>Optimized OpenGL Aquarium (FPS)</th>
<th>OpenGL Aquarium (FPS)</th>
<th>Chrome (FPS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100000</td>
<td>18</td>
<td>14</td>
<td>8</td>
</tr>
<tr>
<td>50000</td>
<td>30</td>
<td>26</td>
<td>16</td>
</tr>
<tr>
<td>30000</td>
<td>36</td>
<td>36</td>
<td>25</td>
</tr>
<tr>
<td>10000</td>
<td>60</td>
<td>60</td>
<td>45</td>
</tr>
</tbody>
</table>
Quality Gate Keeper on Intel Platforms

- Google helped to integrate Intel platforms into Chrome buildbots
- Intel driver team integrated WebGL CTS into CI system
- Monthly conformance test (WebGL CTS) with latest hardware, OSes, Chrome and WebGL CTS
- Monthly performance test with various hardware, OSes, browsers and selected benchmarks
Call for Feedback

- We need your feedback to see if some features are important on web
  - Compute Shader
  - Geometry Shader
  - All the ES 3.1 features
  - Machine Learning
  - … …
Today’s Presenters

Gary Li, AGI
Ib Green and Georgios Karnas, Uber
Mauricio Vives, MerchLogix
Paul Chambers, Sketchfab
Ricardo Cabello, Google

Simon Kallweit and Adrian Blumer, Esri R&D Center Zurich
Tarek Sherif, Biodigital
Trevor Baron, Microsoft
Will Eastcott, PlayCanvas
Come Back at **5:30** for the **NETWORKING RECEPTION**

Sponsored by:

[Logos of Nvidia, LunarC, Cesium, AMD]

[Website: www.khronos.org]