



OpenGL / OpenGL ES Update



Piers Daniell
July 2019



OpenGL ES Update



- Updated OpenGL ES 3.2 API and shader language specs
- OpenGL ES 3.2 adoption expands
 - 20+ new conformance submissions

arm



Imagination

Qualcomm



BROADCOM®



NVIDIA®

- Android OpenGL ES 3.2 adoption increases

OpenGL ES version	Distribution
2.0	23.8%
3.0	31.7%
3.1	13.9%
3.2	30.6%



OpenGL ES Version	Distribution
GL 2.0	21.1% ↓
GL 3.0	29.8% ↓
GL 3.1	13.6% ↓
GL 3.2	35.5% ↑

Data collected during a 7-day period ending on July 23, 2018.

Data collected during a 7-day period ending on May 7, 2019.

OpenGL ES Extensions



- Newly ratified extensions
 - GL_KHR_shader_subgroup
- New multi-vendor extensions
 - GL_EXT_depth_clamp
 - GL_EXT_texture_query_lod
 - GL_EXT_texture_shadow_lod
 - GL_EXT_multiview_timer_query
 - GL_EXT_multiview_texture_multisample
 - GL_EXT_multiview_tessellation_geometry_shader



New Vendor OpenGL ES Extensions

- **NVIDIA**

- GL_NV_memory_attachment
- GL_NV_compute_shader_derivatives
- GL_NV_scissor_exclusive
- GL_NV_mesh_shader
- GL_NV_shader_texture_footprint
- GL_NV_representative_fragment_test
- GL_NV_shading_rate_image
- GL_NV_fragment_shader_barycentric
- GL_NV_shader_subgroup_partitioned

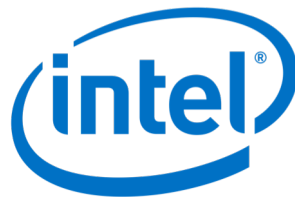
- **Qualcomm**

- GL_QCOM_shader_framebuffer_fetch_rate
- GL_QCOM_texture_foveated_subsampled_layout
- GL_QCOM_YUV_texture_gather



OpenGL Update

- Updates to OpenGL 4.6 API and shader language specs
 - Bug fixes and clarifications
 - Many of which were contributed through GitHub
- New OpenGL 4.6 conformant product submissions from:



- Conformance test coverage improvements and fixes
- Development of new multi-vendor extensions



OpenGL Extensions

- Newly ratified extensions
 - GL_KHR_shader_subgroup
- New multi-vendor extensions
 - GL_EXT_texture_sRGB_R8
 - GLX_EXT_context_priority
 - GL_EXT_texture_shadow_lod
 - GL_EXT_multiview_timer_query
 - GL_EXT_multiview_texture_multisample
 - GL_EXT_multiview_tessellation_geometry_shader



New Vendor Extensions

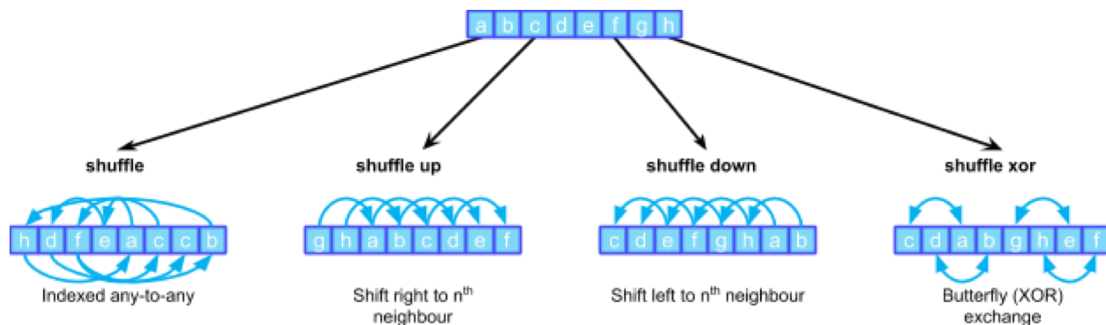
- **NVIDIA:**
 - GL_NV_memory_attachment
 - GL_NV_compute_shader_derivatives
 - GL_NV_fragment_shader_barycentric
 - GL_NV_mesh_shader
 - GL_NV_representative_fragment_test
 - GL_NV_scissor_exclusive
 - GL_NV_shader_texture_footprint
 - GL_NV_shading_rate_image
 - GL_NV_vdpau_interop2
 - GL_NV_shader_subgroup_partitioned





GL_KHR_shader_subgroups

- Brings Vulkan 1.1 subgroup functionality to OpenGL/ES
- Synchronized cross-thread operations without shared memory
 - Reducing latency and increasing bandwidth
 - Accelerates parallelizable algorithms like reduction and sorting
- Available with GLSL and SPIR-V (OpenGL 4.6)
- Vote, arithmetic, ballot, **shuffle**, clustered and quad operations



- Beta driver available: <https://developer.nvidia.com/opengl-driver>

OpenGL's Missing Vulkan Features

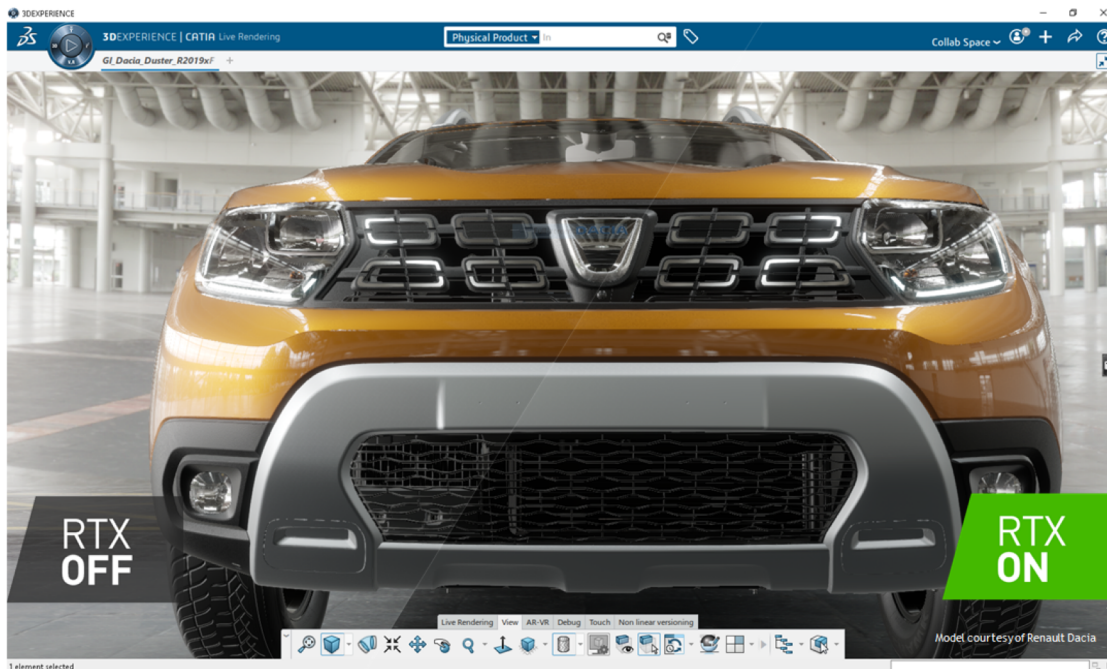


- Like ray tracing, for example
- Can use interop extensions
 - `GL_EXT_memory_object{,_win32,_fd}`
 - `GL_EXT_semaphore{,_win32,_fd}`
 - `GL_EXT_win32_keyed_mutex`
- Enables OpenGL applications to incrementally leverage Vulkan functionality
 - Shared explicit memory objects

3DEXPERIENCE CATIA



- Dassault Systèmes is using Vulkan/OpenGL interop to achieve interactive object space AO in CATIA, an OpenGL application
 - See the demo at the NVIDIA booth



Vulkan's Missing OpenGL Features



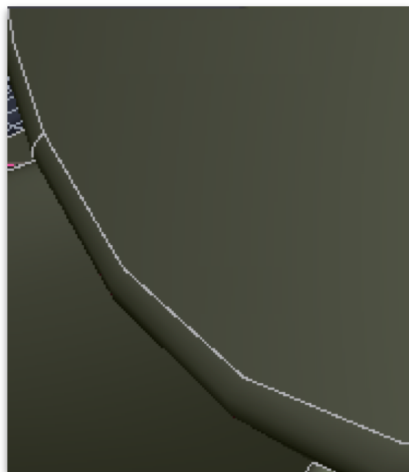
- **Fixing functionality gaps between Vulkan and OpenGL**
 - VK_KHR_draw_indirect_count
 - VK_KHR_shader_draw_parameters
 - VK_EXT_buffer_device_address / VK_EXT_descriptor_indexing
 - VK_EXT_conditional_rendering
 - VK_EXT_depth_range_unrestricted
 - VK_EXT_fragment_shader_interlock
 - VK_EXT_index_type_uint8
 - VK_EXT_line_rasterization
 - VK_EXT_transform_feedback
 - VK_EXT_vertex_attribute_divisor
- **Remaining functionality gaps**
 - Quad primitives
 - GL_NV_command_list



OpenGL Line Quality on Vulkan

- **VK_EXT_line_rasterization**
 - Initiated by leading CAD software vendor
- **Bresenham-style line rasterization**
- **Smooth rectangular lines (coverage to alpha)**
- **Line stipple for all line rasterization modes**

Non-multisampled
rectangular lines:



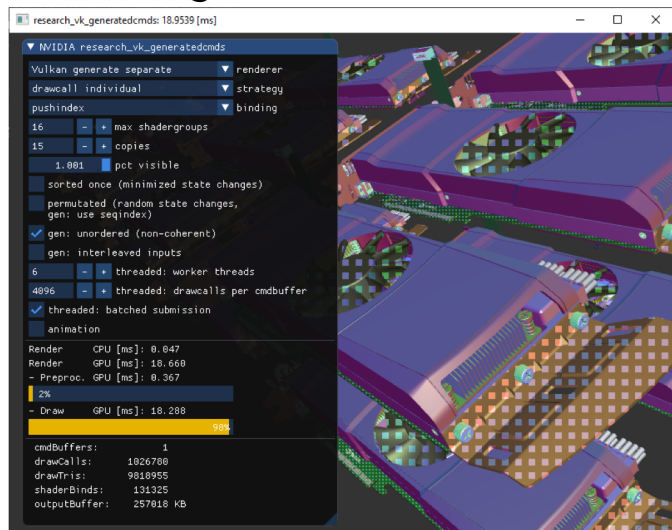
Non-multisampled
Bresenham lines:



- Beta driver available: <https://developer.nvidia.com/vulkan-driver>

GL command list → VK device generated cmds

- OpenGL extension allows recording commands on the device
 - vbo, ibo, ubo bindings and drawcalls
 - Useful for in-frame occlusion culling, or other specialized work creation
- Vulkan extension is work in progress
 - NVX_device_generated_commands design was significantly overhauled
 - Now NV_dgc uses real addresses like GL, and shader groups like ray tracing
 - Exposed in NDA driver - looking for feedback to refine design before public EXT
- RTX 6000 preliminary results:
 - Recording 1M drawcalls
 - With 131k shader group changes
 - Commands per draw:
 - shader group, vbo, ibo, 2x push
 - 6 threads on i7-6850K: ~8.7ms
 - GPU-generated: ~0.36ms





OpenGL/ES and DX on Vulkan

- Improved Vulkan functionality makes emulation easier
 - Zink - OpenGL on Vulkan
 - ANGLE - OpenGL ES on Vulkan
 - dxvk - DirectX 10/11 on Vulkan
 - vkd3d - DirectX 12 on Vulkan



Thanks!

- Questions? Come see me after at the Khronos Networking Reception

