Download Report to File

Graphics Feature Status

- Canvas: Hardware accelerated
- Canvas out-of-process rasterization: Enabled
- Direct Rendering Display Compositor: Disabled
- Compositing: Hardware accelerated
- Multiple Raster Threads: Enabled
- OpenGL: Enabled
- Rasterization: Hardware accelerated on all pages
- Raw Draw: Disabled
- Skia Graphite: Disabled
- Video Decode: Hardware accelerated
- Video Encode: Hardware accelerated
- Vulkan: Disabled
- WebGL: Hardware accelerated
- WebGL2: Hardware accelerated
- WebGPU: Hardware accelerated

Driver Bug Workarounds

- avoid_consecutive_keyframes_for_vp9
- disable_accelerated_av1_encode
- disable_d3d11_vp9_ksvc_decoding
- enable_webgl_timer_query_extensions
- exit_on_context_lost
- unpack_overlapping_rows_separately_unpack_buffer
- disabled_extension_GL_KHR_blend_equation_advanced
- disabled_extension_GL_KHR_blend_equation_advanced_coherent
- disabled_extension_GL_MESA_framebuffer_flip_y

Problems Detected

- Some drivers are unable to reset the D3D device in the GPU process sandbox
  Applied Workarounds: exit_on_context_lost
- Unpacking overlapping rows from unpack buffers is unstable on NVIDIA GL driver: 596774
  Applied Workarounds: unpack_overlapping_rows_separately_unpack_buffer
- Disable KHR_blend_equation_advanced until cc shaders are updated: 661715
  Applied Workarounds: disable(GL_KHR_blend_equation_advanced), disable(GL_KHR_blend_equation_advanced_coherent)
- Expose WebGL's disjoint_timer_query extensions on platforms with site isolation: 808744, 870491
  Applied Workarounds: enable_webgl_timer_query_extensions
- Disable GL_MESA_framebuffer_flip_y for desktop GL: 964010
  Applied Workarounds: disable(GL_MESA_framebuffer_flip_y)
- Disable hardware MFT Av1 encoder on machines with multiple GPUs except Intel alchemist GPUs: 1367038
  Applied Workarounds: disable_accelerated_av1_encode

- Corruption when consecutive VP9 keyframes are requested from MFVEA on Intel.: 1473665
  Applied Workarounds: avoid_consecutive_keyframes_for_vp9
- Disable VP9 k-SVC video decoding on non-Intel GPUs: 1508379
  Applied Workarounds: disable_d3d11_vp9_ksvc_decoding

ANGLE Features

- **allowCompressedFormats** (Frontend workarounds): Enabled: true
  Allow compressed formats
- **alwaysRunLinkSubJobsThreaded** (Frontend features) anglebug:8417: Disabled
  If true, sub tasks of the link job are always threaded, regardless of GL_KHR_parallel_shader_compile
- **cacheCompiledShader** (Frontend features) anglebug:7036: Disabled
  Enable to cache compiled shaders
- **compileJobIsThreadSafe** (Frontend features) anglebug:8297: Enabled: true
  If false, parts of the compile job cannot be parallelized
- **disableAnisotropicFiltering** (Frontend workarounds): Disabled
  Disable support for anisotropic filtering
- **disableDrawBuffersIndexed** (Frontend features) anglebug:7724: Disabled
  Disable support for OES_draw_buffers_indexed and EXT_draw_buffers_indexed
- **disableProgramBinary** (Frontend features) anglebug:5007: Disabled
  Disable support for GL_OES_get_program_binary
- **disableProgramCaching** (Frontend features) anglebug:1423136: Disabled
  Disables saving programs to the cache
- **disableProgramCachingForTransformFeedback** (Frontend workarounds): Disabled
  On some GPUs, program binaries don't contain transform feedback varyings
- **dumpShaderSource** (Frontend features) anglebug:7760: Disabled
  Write shader source to temp directory
- **dumpTranslatedShaders** (Frontend features) anglebug:8280: Disabled
  Write translated shaders to temp directory
- **emulatePixelLocalStorage** (Frontend features) anglebug:7279: Enabled: true
  Emulate ANGLE_shader_pixel_local_storage using shader images
- **enableCaptureLimits** (Frontend features) anglebug:5750: Disabled
  Set the context limits like frame capturing was enabled
- **enableProgramBinaryForCapture** (Frontend features) anglebug:5658: Disabled
  Even if FrameCapture is enabled, enable GL_OES_get_program_binary
- **enableShaderSubstitution** (Frontend workarounds) anglebug:7761: Disabled
  Check the filesystem for shaders to use instead of those provided through glShaderSource
- **enableTranslatedShaderSubstitution** (Frontend workarounds) anglebug:8280: Disabled
  Check the filesystem for translated shaders to use instead of the shader translator's
- **forceDepthAttachmentInitOnClear** (Frontend workarounds) anglebug:7246: Disabled: isAMD
  Force depth attachment initialization on clear ops
- **forceGlErrorChecking** (Frontend features) https://issuetracker.google.com/220069903: Disabled
  Force GL error checking (i.e. prevent applications from disabling error checking
- **forceInitShaderVariables** (Frontend features): Disabled
  Force-enable shader variable initialization

- **forceMinimumMaxVertexAttributes** (Frontend features): Disabled: false
  Force the minimum GL_MAX_VERTEX_ATTRIBS that the context's client version allows.
- **forceRobustResourceInit** (Frontend features) anglebug:6041: Disabled
  Force-enable robust resource init
- **linkJobIsThreadSafe** (Frontend features) anglebug:8297: Enabled: true
  If false, parts of the link job cannot be parallelized
- **loseContextOnOutOfMemory** (Frontend workarounds): Enabled: true
  Some users rely on a lost context notification if a GL_OUT_OF_MEMORY error occurs
- **singleThreadedTextureDecompression** (Frontend workarounds): Disabled
  Disables multi-threaded decompression of compressed texture formats
- **uncurrentEglSurfaceUponSurfaceDestroy** (Frontend workarounds) https://issuetracker.google.com/292285899: Disabled
  Make egl surface uncurrent when calling eglDestroySurface(), if the surface is still bound by the context of current render thread
- **addMockTextureNoRenderTarget** (D3D workarounds) anglebug:2152: Disabled: isIntel && capsVersion >= IntelDriverVersion(160000) && capsVersion < IntelDriverVersion(164815)
  On some drivers when rendering with no render target, two bugs lead to incorrect behavior
- **allowClearForRobustResourceInit** (D3D workarounds) 941620: Enabled: true
  Some drivers corrupt texture data when clearing for robust resource initialization.
- **allowES3OnFL100** (D3D workarounds): Disabled: false
  Allow ES3 on 10.0 devices
- **allowTranslateUniformBlockToStructuredBuffer** (D3D workarounds) anglebug:3682: Enabled: IsWindows10OrLater()
  There is a slow fxc compile performance issue with dynamic uniform indexing if translating a uniform block with a large array member to cbuffer.
- **borderColorSrgb** (D3D workarounds): Disabled
  Some drivers expect sRGB border color for sRGB texture formats
- **callClearTwice** (D3D workarounds) 655534: Disabled: isIntel && isSkylake && capsVersion >= IntelDriverVersion(160000) && capsVersion < IntelDriverVersion(164771)
  Using clear() may not take effect
- **depthStencilBlitExtraCopy** (D3D workarounds) anglebug:1452: Disabled: (part1 <= 13u && part2 < 6881) && isNvidia && driverVersionValid
  Bug in some drivers triggers a TDR when using CopySubresourceRegion from a staging texture to a depth/stencil
- **disableB5G6R5Support** (D3D workarounds): Disabled: (isIntel && capsVersion >= IntelDriverVersion(150000) && capsVersion < IntelDriverVersion(154539)) || isAMD
  Textures with the format DXGI_FORMAT_B5G6R5_UNORM have incorrect data
- **disableRasterizerOrderViews** (D3D workarounds) anglebug:7279: Disabled
  Disable ROVs for testing
- **emulateIsnanFloat** (D3D workarounds) 650547: Disabled: isIntel && isSkylake && capsVersion >= IntelDriverVersion(160000) && capsVersion < IntelDriverVersion(164542)
  Using isnan() on highp float will get wrong answer
- **emulateTinyStencilTextures** (D3D workarounds): Disabled: isAMD && !(deviceCaps.featureLevel < D3D_FEATURE_LEVEL_10_1)
  1x1 and 2x2 mips of depth/stencil textures aren't sampled correctly

- **enableTimestampQueries** (D3D workarounds): Disabled
  Enable timestamp on GL_EXT_disjoint_timer_query extension
- **expandIntegerPowExpressions** (D3D workarounds): Enabled: true
  The HLSL optimizer has a bug with optimizing 'pow' in certain integer-valued expressions
- **flushAfterEndingTransformFeedback** (D3D workarounds): Enabled: isNvidia
  Some drivers sometimes write out-of-order results to StreamOut buffers when transform feedback is used to repeatedly write to the same buffer positions
- **forceAtomicValueResolution** (D3D workarounds) anglebug:3246: Enabled: isNvidia
  On some drivers the return value from RWByteAddressBuffer.InterlockedAdd does not resolve when used in the .yzw components of a RWByteAddressBuffer.Store operation
- **getDimensionsIgnoresBaseLevel** (D3D workarounds): Enabled: isNvidia
  Some drivers do not take into account the base level of the texture in the results of the HLSL GetDimensions builtin
- **mrtPerfWorkaround** (D3D workarounds): Enabled: true
  Some drivers have a bug where they ignore null render targets
- **preAddTexelFetchOffsets** (D3D workarounds): Disabled: isIntel
  HLSL's function texture.Load returns 0 when the parameter Location is negative, even if the sum of Offset and Location is in range
- **rewriteUnaryMinusOperator** (D3D workarounds): Disabled: isIntel && (isBroadwell || isHaswell) && capsVersion >= IntelDriverVersion(150000) && capsVersion < IntelDriverVersion(154624)
  Evaluating unary minus operator on integer may get wrong answer in vertex shaders
- **selectViewInGeometryShader** (D3D workarounds): Disabled: !deviceCaps.supportsVpRtIndexWriteFromVertexShader
  The viewport or render target slice will be selected in the geometry shader stage for the ANGLE_multiview extension
- **setDataFasterThanImageUpload** (D3D workarounds): Enabled: !(isIvyBridge || isBroadwell || isHaswell)
  Set data faster than image upload
- **skipVSConstantRegisterZero** (D3D workarounds): Enabled: isNvidia
  In specific cases the driver doesn't handle constant register zero correctly
- **supportsNonConstantLoopIndexing** (D3D workarounds): Enabled: !isFeatureLevel9_3
  Whether the API supports non-constant loop indexing
- **useInstancedPointSpriteEmulation** (D3D workarounds): Disabled: isFeatureLevel9_3
  Some D3D11 renderers do not support geometry shaders for pointsprite emulation
- **useSystemMemoryForConstantBuffers** (D3D workarounds) 593024: Disabled: isIntel
  Copying from staging storage to constant buffer storage does not work
- **zeroMaxLodWorkaround** (D3D workarounds): Disabled: isFeatureLevel9_3
  Missing an option to disable mipmaps on a mipmapped texture

Version Information

Data exported	**2024-04-12T13:00:25.625Z**
Chrome version	**Chrome/123.0.6312.105**
Operating system	**Windows NT 10.0.22621**
Software rendering list URL	**https://chromium.googlesource.com/chromium/src/+/399174dbe6e**

Driver bug list URL	https://chromium.googlesource.com/chromium/src/+/399174dbe6e
ANGLE commit id	bbf1e1ea6bcf
2D graphics backend	Skia/123 3d4e45907f9b239a54957001d619d2d4a6ca06b4
Command Line	"D:\project\video-composition-optimization\.cache\puppeteer\chrome\win64-123.0.6312.105\chrome-win64\chrome.exe" --allow-pre-commit-input --disable-background-networking --disable-background-timer-throttling --disable-backgrounding-occluded-windows --disable-breakpad --disable-client-side-phishing-detection --disable-component-extensions-with-background-pages --disable-component-update --disable-default-apps --disable-dev-shm-usage --disable-extensions --disable-field-trial-config --disable-hang-monitor --disable-infobars --disable-ipc-flooding-protection --disable-popup-blocking --disable-prompt-on-repost --disable-renderer-backgrounding --disable-search-engine-choice-screen --disable-sync --enable-automation --export-tagged-pdf --generate-pdf-document-outline --force-color-profile=srgb --metrics-recording-only --no-first-run --password-store=basic --use-mock-keychain --disable-features=Translate,AcceptCHFrame,MediaRouter,OptimizationHints, --enable-features=NetworkServiceInProcess2 --headless=new --hide-scrollbars --mute-audio --disk-cache --enable-gpu --ignore-gpu-blacklist --enable-gpu-rasterization --enable-zero-copy --gpu-rasterization-msaa-sample-count=16 --enable-gpu-memory-buffer-video-frames --enable-native-gpu-memory-buffers --video-capture-use-gpu-memory-buffer --video-threads=14 --remote-debugging-port=0 --user-data-dir="C:\Users\ASUS\AppData\Local\Temp\puppeteer_dev_chrome_p XXXXXXLJTPiL" --noerrdialogs --flag-switches-begin --flag-switches-end about:blank

Driver Information

Initialization time	117
In-process GPU	false
Passthrough Command Decoder	true
Sandboxed	true
GPU0	VENDOR= 0x10de, DEVICE=0x28e0, SUBSYS=0x20bd1043, REV=161, LUID={0,81145}, DRIVER_VENDOR=NVIDIA, DRIVER_VERSION=31.0.15.5176 *ACTIVE*
GPU1	VENDOR= 0x8086, DEVICE=0xa78b, SUBSYS=0x20bd1043, REV=4, LUID={0,85228}, DRIVER_VERSION=31.0.101.4502
GPU2	VENDOR= 0x1414, DEVICE=0x008c, LUID={0,85028}, DRIVER_VERSION=10.0.22621.2506
Optimus	false

AMD switchable	false
Desktop compositing	Aero Glass
Direct composition	true
Supports overlays	true
YUY2 overlay support	SCALING
NV12 overlay support	SCALING
BGRA8 overlay support	SOFTWARE
RGB10A2 overlay support	SOFTWARE
Driver D3D12 feature level	Not supported
Driver Vulkan API version	Not supported
GPU CUDA compute capability major version	0
Pixel shader version	5.0
Vertex shader version	5.0
Max. MSAA samples	8
Machine model name	
Machine model version	
GL implementation parts	(gl=egl-angle,angle=d3d11)
Display type	ANGLE_D3D11
GL_VENDOR	Google Inc. (NVIDIA)
GL_RENDERER	ANGLE (NVIDIA, NVIDIA GeForce RTX 4060 Laptop GPU (0x000028E0) Direct3D11 vs_5_0 ps_5_0, D3D11-31.0.15.5176)
GL_VERSION	OpenGL ES 2.0.0 (ANGLE 2.1.22631 git hash: bbf1e1ea6bcf)
GL_EXTENSIONS	GL_AMD_performance_monitor GL_ANGLE_base_vertex_base_instance GL_ANGLE_base_vertex_base_instance_shader_builtin GL_ANGLE_client_arrays GL_ANGLE_depth_texture GL_ANGLE_framebuffer_blit GL_ANGLE_framebuffer_multisample GL_ANGLE_get_serialized_context_string GL_ANGLE_get_tex_level_parameter GL_ANGLE_instanced_arrays GL_ANGLE_lossy_etc_decode GL_ANGLE_memory_size GL_ANGLE_multi_draw GL_ANGLE_pack_reverse_row_order GL_ANGLE_polygon_mode GL_ANGLE_program_cache_control GL_ANGLE_provoking_vertex GL_ANGLE_request_extension GL_ANGLE_robust_client_memory GL_ANGLE_texture_compression_dxt3 GL_ANGLE_texture_compression_dxt5 GL_ANGLE_texture_usage

	GL_ANGLE_translated_shader_source GL_APPLE_clip_distance GL_CHROMIUM_bind_generates_resource GL_CHROMIUM_bind_uniform_location GL_CHROMIUM_color_buffer_float_rgb GL_CHROMIUM_color_buffer_float_rgba GL_CHROMIUM_copy_compressed_texture GL_CHROMIUM_copy_texture GL_CHROMIUM_lose_context GL_CHROMIUM_sync_query GL_EXT_EGL_image_external_wrap_modes GL_EXT_base_instance GL_EXT_blend_func_extended GL_EXT_blend_minmax GL_EXT_clip_control GL_EXT_color_buffer_half_float GL_EXT_debug_label GL_EXT_debug_marker GL_EXT_depth_clamp GL_EXT_discard_framebuffer GL_EXT_disjoint_timer_query GL_EXT_draw_buffers GL_EXT_draw_elements_base_vertex GL_EXT_float_blend GL_EXT_frag_depth GL_EXT_instanced_arrays GL_EXT_map_buffer_range GL_EXT_multi_draw_indirect GL_EXT_multisampled_render_to_texture GL_EXT_occlusion_query_boolean GL_EXT_polygon_offset_clamp GL_EXT_read_format_bgra GL_EXT_robustness GL_EXT_sRGB GL_EXT_shader_texture_lod GL_EXT_texture_border_clamp GL_EXT_texture_compression_bptc GL_EXT_texture_compression_dxt1 GL_EXT_texture_compression_rgtc GL_EXT_texture_compression_s3tc_srgb GL_EXT_texture_filter_anisotropic GL_EXT_texture_format_BGRA8888 GL_EXT_texture_mirror_clamp_to_edge GL_EXT_texture_norm16 GL_EXT_texture_rg GL_EXT_texture_storage GL_EXT_texture_type_2_10_10_10_REV GL_EXT_unpack_subimage GL_KHR_debug GL_KHR_parallel_shader_compile GL_NV_EGL_stream_consumer_external GL_NV_fence GL_NV_framebuffer_blit GL_NV_pack_subimage GL_NV_pixel_buffer_object GL_OES_EGL_image GL_OES_EGL_image_external GL_OES_compressed_EAC_R11_signed_texture GL_OES_compressed_EAC_R11_unsigned_texture GL_OES_compressed_EAC_RG11_signed_texture GL_OES_compressed_EAC_RG11_unsigned_texture GL_OES_compressed_ETC2_RGB8_texture GL_OES_compressed_ETC2_RGBA8_texture GL_OES_compressed_ETC2_punchthroughA_RGBA8_texture GL_OES_compressed_ETC2_punchthroughA_sRGB8_alpha_texture GL_OES_compressed_ETC2_sRGB8_alpha8_texture GL_OES_compressed_ETC2_sRGB8_texture GL_OES_depth24 GL_OES_depth32 GL_OES_draw_elements_base_vertex GL_OES_element_index_uint GL_OES_fbo_render_mipmap GL_OES_get_program_binary GL_OES_mapbuffer

	GL_OES_packed_depth_stencil GL_OES_rgb8_rgba8 GL_OES_standard_derivatives GL_OES_surfaceless_context GL_OES_texture_border_clamp GL_OES_texture_float GL_OES_texture_float_linear GL_OES_texture_half_float GL_OES_texture_half_float_linear GL_OES_texture_npot GL_OES_vertex_array_object GL_WEBGL_video_texture
Disabled Extensions	GL_KHR_blend_equation_advanced GL_KHR_blend_equation_advanced_coherent GL_MESA_framebuffer_flip_y
Disabled WebGL Extensions	
Window system binding vendor	Google Inc. (NVIDIA)
Window system binding version	1.5 (ANGLE 2.1.22631 git hash: bbf1e1ea6bcf)
Window system binding extensions	EGL_EXT_create_context_robustness EGL_ANGLE_d3d_share_handle_client_buffer EGL_ANGLE_d3d_texture_client_buffer EGL_ANGLE_surface_d3d_texture_2d_share_handle EGL_ANGLE_query_surface_pointer EGL_ANGLE_window_fixed_size EGL_ANGLE_keyed_mutex EGL_ANGLE_surface_orientation EGL_ANGLE_direct_composition EGL_NV_post_sub_buffer EGL_KHR_create_context EGL_KHR_image EGL_KHR_image_base EGL_KHR_gl_texture_2D_image EGL_KHR_gl_texture_cubemap_image EGL_KHR_gl_renderbuffer_image EGL_KHR_get_all_proc_addresses EGL_KHR_stream EGL_KHR_stream_consumer_gltexture EGL_NV_stream_consumer_gltexture_yuv EGL_ANGLE_stream_producer_d3d_texture EGL_ANGLE_create_context_webgl_compatibility EGL_CHROMIUM_create_context_bind_generates_resource EGL_CHROMIUM_sync_control EGL_EXT_pixel_format_float EGL_KHR_surfaceless_context EGL_ANGLE_display_texture_share_group EGL_ANGLE_display_semaphore_share_group EGL_ANGLE_create_context_client_arrays EGL_ANGLE_program_cache_control EGL_ANGLE_robust_resource_initialization EGL_ANGLE_create_context_extensions_enabled EGL_ANDROID_blob_cache EGL_ANDROID_recordable EGL_ANGLE_image_d3d11_texture EGL_ANGLE_create_context_backwards_compatible EGL_KHR_no_config_context EGL_KHR_create_context_no_error EGL_KHR_reusable_sync
Direct rendering version	unknown

Reset notification strategy	0x8252
GPU process crash count	0
gfx::BufferFormats supported for allocation and texturing	R_8: not supported, R_16: not supported, RG_88: not supported, RG_1616: not supported, BGR_565: not supported, RGBA_4444: not supported, RGBX_8888: not supported, RGBA_8888: not supported, BGRX_8888: not supported, BGRA_1010102: not supported, RGBA_1010102: not supported, BGRA_8888: not supported, RGBA_F16: not supported, YVU_420: not supported, YUV_420_BIPLANAR: not supported, YUVA_420_TRIPLANAR: not supported, P010: not supported

Compositor Information

Tile Update Mode	Zero-copy
Partial Raster	Enabled

GpuMemoryBuffers Status

R_8	Software only
R_16	Software only
RG_88	Software only
RG_1616	Software only
BGR_565	Software only
RGBA_4444	Software only
RGBX_8888	GPU_READ, SCANOUT
RGBA_8888	GPU_READ, SCANOUT
BGRX_8888	Software only
BGRA_1010102	Software only
RGBA_1010102	Software only
BGRA_8888	Software only
RGBA_F16	Software only
YVU_420	Software only
YUV_420_BIPLANAR	Software only
YUVA_420_TRIPLANAR	Software only
P010	Software only

Display(s) Information

Info	Display[2528732444] bounds=[0,0 2560x1440], workarea=[0,0 2560x1392], scale=1, rotation=0, panel_rotation=0 external detected
Color space (all)	{primaries:BT709, transfer:SRGB, matrix:RGB, range:FULL}
Buffer format (all)	BGRA_8888

Color volume	{name:'srgb', r:[0.6400, 0.3300], g:[0.3000, 0.6000], b:[0.1500, 0.3300], w:[0.3127, 0.3290]}
SDR white level in nits	203
HDR relative maximum luminance	1
Bits per color component	8
Bits per pixel	24
Refresh Rate in Hz	170

Video Acceleration Information

Decoding	
Decode h264 baseline	64x64 to 4096x4096 pixels
Decode h264 main	64x64 to 4096x4096 pixels
Decode h264 high	64x64 to 4096x4096 pixels
Decode vp9 profile0	64x64 to 8192x8192 pixels
Decode vp9 profile2	64x64 to 8192x8192 pixels
Decode hevc main	64x64 to 8192x8192 pixels
Decode hevc main 10	64x64 to 8192x8192 pixels
Decode av1 profile main	64x64 to 8192x8192 pixels
Encoding	

Vulkan Information

Device Performance Information

Log Messages