Vulkanised 2024

The 6th Vulkan Developer Conference
Sunnyvale, California | February 5-7, 2024

# Everything you need to know about the Vulkan SDK

Karen Ghavam, LunarG

Presentation:

https://bit.ly/420QuRQ

# Who is LunarG?

- An Independent, private company with Khronos membership
  - Specializing in 3D graphics software solutions for our clients
- Developing Vulkan Ecosystem components since 2015
  - Generous sponsorship from Valve and Google
- Vulkan Ecosystem Projects
  - Vulkan SDK
  - Vulkan Loader
  - Vulkan Validation Layers
  - Vulkan Profiles Toolset
  - Vulkan Extension Layer
  - GFXReconstruct
  - glslang
  - …

Today's Presentation:

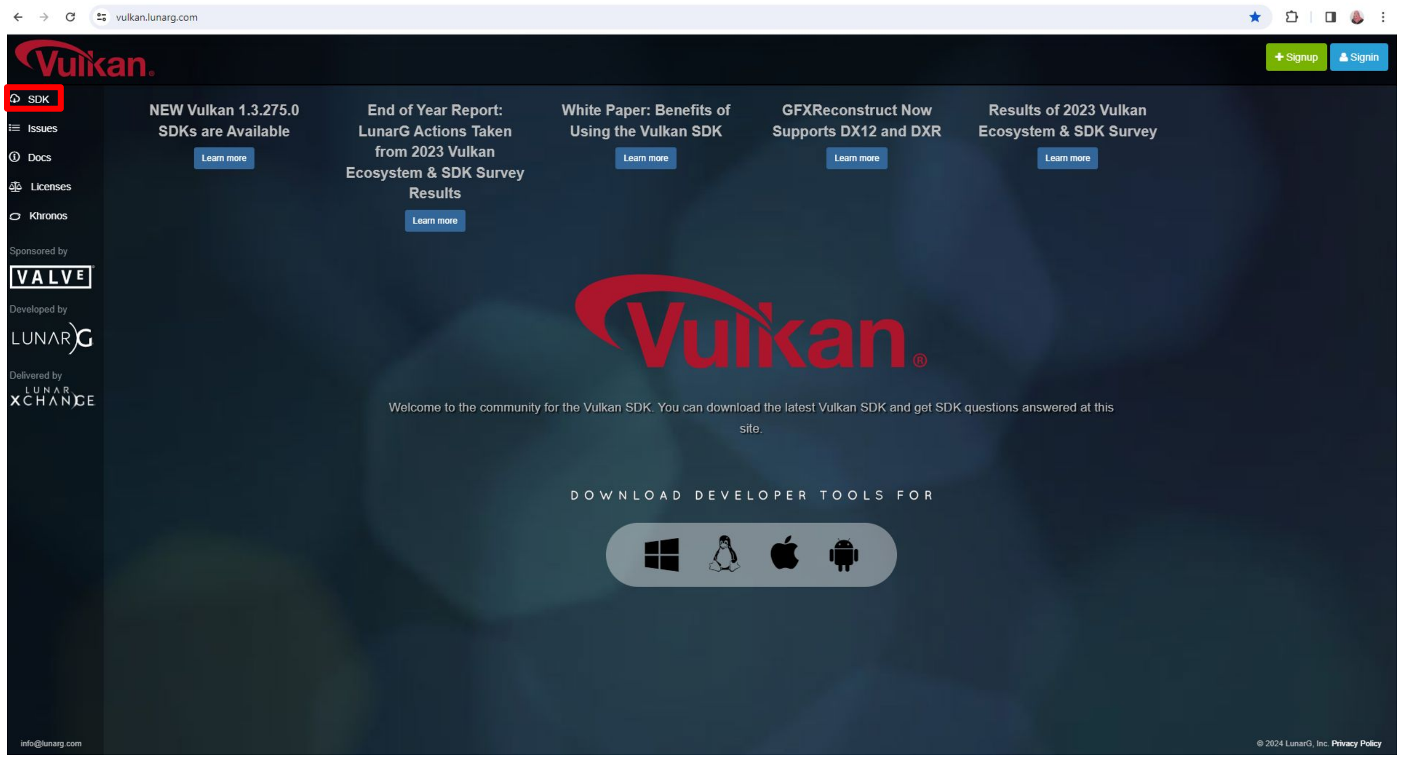
https://bit.ly/420QuRQ

# 2023 Ecosystem Survey Highlights

- 275 respondents. 48.3/50.7% split between self-study/commercial developers
- Amount of released content increased from 28% to 36%
- Themes:
  - Shader tool chain needs more development and maintenance
    - DXC usage was 20% of population
    - glslangValidator/shaderc (glsl->SPIR-V) was 60+% of the population
  - Validation Layers
    - Invaluable!
    - Continue to increase coverage
    - Error messages are very verbose and could be formatted better for easier reading
    - Interpreting errors (finding my root cause) is difficult (Synchronization & GPU-AV in particular)
    - Improve the performance
  - Would like to have MoltenVK to move forward more quickly

Full report: https://www.lunarg.com/wp-content/uploads/2023/04/2023-Ecosystem-Survey-Public-Report-06APR2023.pdf
End of year report (results):
https://www.lunarg.com/wp-content/uploads/2024/01/2023-Ecosystem-Survey-Public-Report-End-of-Year-Update.pdf

# The Vulkan SDK (Vulkan.lunarg.com)



vulkan.lunarg.com

NEW Vulkan 1.3.275.0 SDKs are Available

Learn more

End of Year Report: LunarG Actions Taken from 2023 Vulkan Ecosystem & SDK Survey Results

Learn more

White Paper: Benefits of Using the Vulkan SDK

Learn more

GFXReconstruct Now Supports DX12 and DXR

Learn more

Results of 2023 Vulkan Ecosystem & SDK Survey

Learn more

Welcome to the community for the Vulkan SDK. You can download the latest Vulkan SDK and get SDK questions answered at this site.

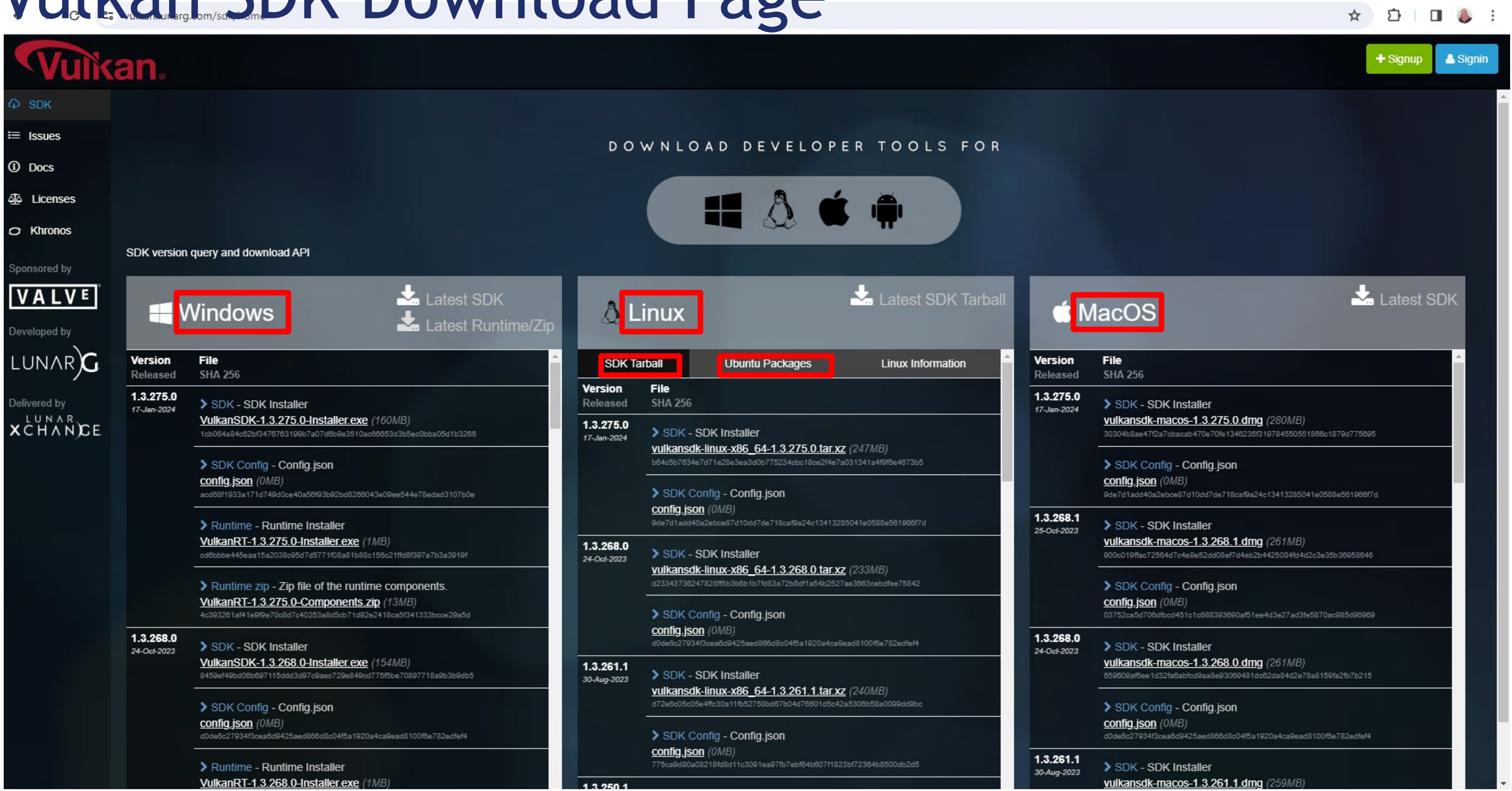
DOWNLOAD DEVELOPER TOOLS FOR

info@lunarg.com

© 2024 LunarG, Inc. Privacy Policy

Delivered by LunarG in close coordination with the Khronos Vulkan working group

# Vulkan SDK Download Page



DOWNLOAD DEVELOPER TOOLS FOR

SDK version query and download API

## Windows

Latest SDK
Latest Runtime/Zip

| Version<br>Released | File<br>SHA 256 |
|---|---|
| 1.3.275.0<br>*17-Jan-2024* | SDK - SDK Installer<br>VulkanSDK-1.3.275.0-Installer.exe *(160MB)*<br>1cb064a84c62bf3476763199b7a07d6b9e3510ac66653d3b5ec0bba05d1b3266 |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>acd68f1933a171d749d0ce40a56f93b92bd8266043e09ee544e78edad3107b0e |
| | Runtime - Runtime Installer<br>VulkanRT-1.3.275.0-Installer.exe *(1MB)*<br>cd6bbbe445eaa15a2038c95d7d5771f08a81b88c156c21ffd8f397a7b3a3919f |
| | Runtime zip - Zip file of the runtime components.<br>VulkanRT-1.3.275.0-Components.zip *(13MB)*<br>4c393261af41e9f9e70c8d7c40253a8d5cb71d92e2418ca5f341333bcce29a5d |
| 1.3.268.0<br>*24-Oct-2023* | SDK - SDK Installer<br>VulkanSDK-1.3.268.0-Installer.exe *(154MB)*<br>8459ef49bd06b697115ddd3d97c9aec729e849cd775f5be70897718a9b3b9db5 |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>d0de6c27934f3cea6d9425aed866d8c04f5a1920a4ca9ead8100f6e782edfef4 |
| | Runtime - Runtime Installer<br>VulkanRT-1.3.268.0-Installer.exe *(1MB)* |

## Linux

Latest SDK Tarball

SDK Tarball | Ubuntu Packages | Linux Information

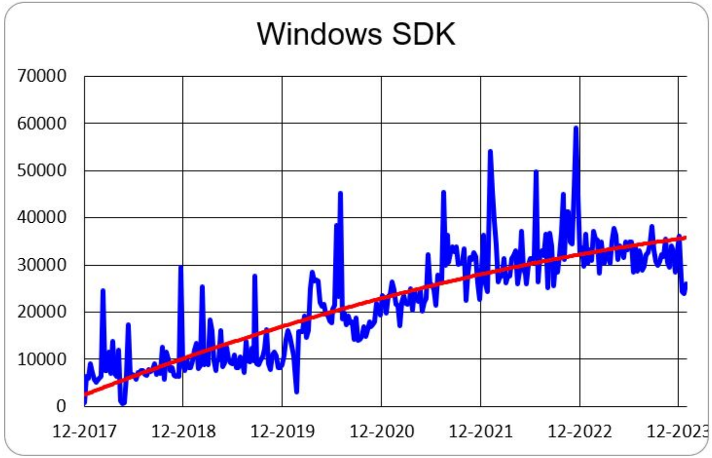
| Version<br>Released | File<br>SHA 256 |
|---|---|
| 1.3.275.0<br>*17-Jan-2024* | SDK - SDK Installer<br>vulkansdk-linux-x86_64-1.3.275.0.tar.xz *(247MB)*<br>b64c5b7634e7d71e28e3ea3d0b775234cbc18ce2f4e7a031341a4f9f6e4673b5 |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>9de7d1add40a2ebce87d10dd7de718caf9a24c13413285041e0588e561966f7d |
| 1.3.268.0<br>*24-Oct-2023* | SDK - SDK Installer<br>vulkansdk-linux-x86_64-1.3.268.0.tar.xz *(233MB)*<br>d23343736247828ff5b3b6b1b7fd83a72b5df1a54b2527ae3663cebdfee75842 |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>d0de6c27934f3cea6d9425aed866d8c04f5a1920a4ca9ead8100f6e782edfef4 |
| 1.3.261.1<br>*30-Aug-2023* | SDK - SDK Installer<br>vulkansdk-linux-x86_64-1.3.261.1.tar.xz *(240MB)*<br>d72e6c05c05e4ffc30a11fb52758bd67b04d76601d5c42a5306b58a0099dd9bc |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>775ca9d90a08218fd8d11c3091ea97fb7ebf64b607f1823bf72364b8500db2d5 |
| 1.3.250.1 | |

## MacOS

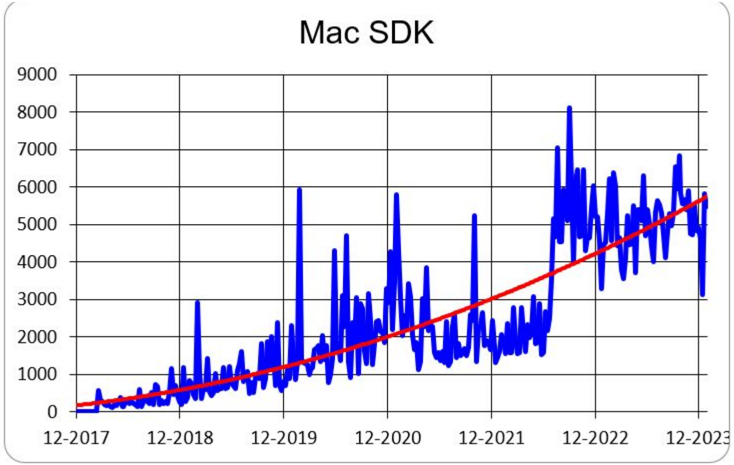
Latest SDK

| Version<br>Released | File<br>SHA 256 |
|---|---|
| 1.3.275.0<br>*17-Jan-2024* | SDK - SDK Installer<br>vulkansdk-macos-1.3.275.0.dmg *(280MB)*<br>30304b8ae47f2a7cbacab470e70fe1346235f319784550551866c1879d775695 |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>9de7d1add40a2ebce87d10dd7de718caf9a24c13413285041e0588e561966f7d |
| 1.3.268.1<br>*25-Oct-2023* | SDK - SDK Installer<br>vulkansdk-macos-1.3.268.1.dmg *(261MB)*<br>900c019ffac72564d7c4e9e52dd08ef7d4eb2b4425084fd4d2c3e35b36958646 |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>03752ca5d706dfbcd451c1c688393690af51ee4d3e27ad3fe5870ac985d96969 |
| 1.3.268.0<br>*24-Oct-2023* | SDK - SDK Installer<br>vulkansdk-macos-1.3.268.0.dmg *(261MB)*<br>659609af6ee1d32fa6abfcd9aa8e93069481dc62da84d2e78a8159fa2fb7b215 |
| | SDK Config - Config.json<br>config.json *(0MB)*<br>d0de6c27934f3cea6d9425aed866d8c04f5a1920a4ca9ead8100f6e782edfef4 |
| 1.3.261.1<br>*30-Aug-2023* | SDK - SDK Installer<br>vulkansdk-macos-1.3.261.1.dmg *(259MB)* |

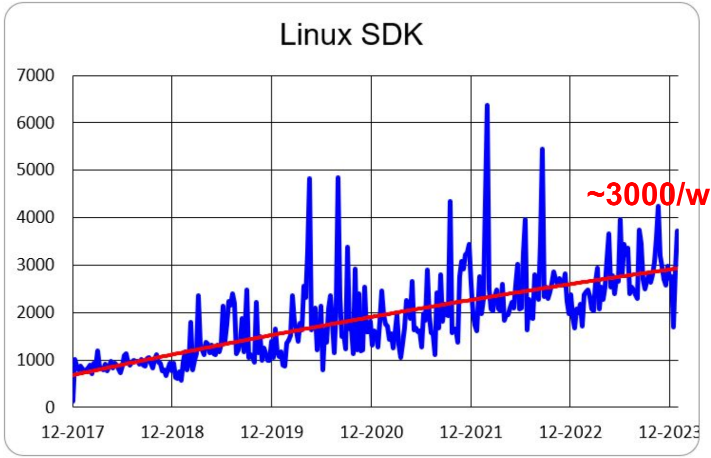
# Vulkan SDK Downloads are Healthy



~38,000/week



~5800/week



~3000/week

Note: Numbers are for Linux "Tarball" only and don't include Ubuntu packages also available from LunarG

LUNARG

# Why Use the SDK?

- An installation process that is easy and fast
  - Windows, Linux, and macOS versions
- Pre-built tools installed into the correct system locations, ready for use.
- Vetted and curated content to ensure compatibility and seamless integration
- Ready-to-use versions of the Vulkan Configurator
- SDK release notes and user documentation
- License Registry
  - Details ALL of the open-source licenses present in the SDK

# Vulkan Configurator

- Vulkan Configurator
  - Greatly simplifies experience with layer enablement and configuration!
- Multiple preset default configurations
  - Ability to create your own layer configurations as well
- Recent addition: physical device selection
- Next major release preview
  - Better control of the layers
    - Multiple versions of a layer
    - Full ordering of the layers
      - Including implicit layers
  - Improved UI: Tab based redesign
  - Diagnostics tab driven by Vulkan loader diagnostic information
- Resources:
  - Munich Vulkanised 2023:
    - Using the Vulkan Configurator for Daily Vulkan Development
      - Khronos Youtube Video

# Developer tools in the Vulkan SDK

- VK_LAYER_KHRONOS_validation - validate correct API usage
  - GPU Assisted Validation
  - Best Practices for
    - Nvidia (new as of August 2022), ARM, Imagination, and AMD
  - Synchronization Validation
  - Debug Printf - “printf inside a shader”
- VK_LAYER_LUNARG_api_dump
  - Ascii output of Vulkan API calls
- Vulkaninfo
  - Show GPU device properties and extensions, installed layers, supported image formats, properties…
- Emulation Layers
  - VK_LAYER_KHRONOS_synchronization2
  - VK_LAYER_KHRONOS_shader_object

New as of May 2023

LUNARG

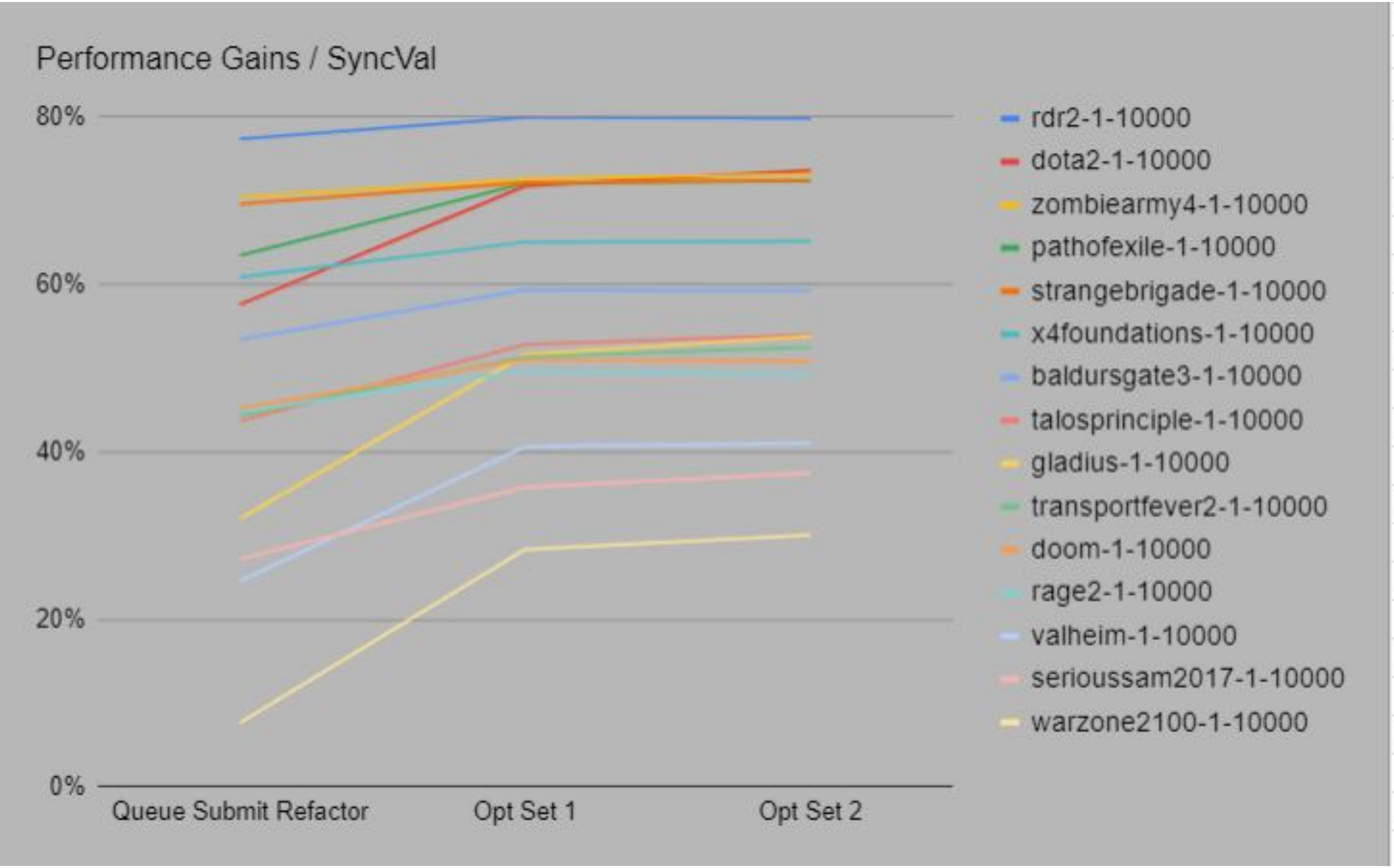
# Validation Layer Performance Improvement

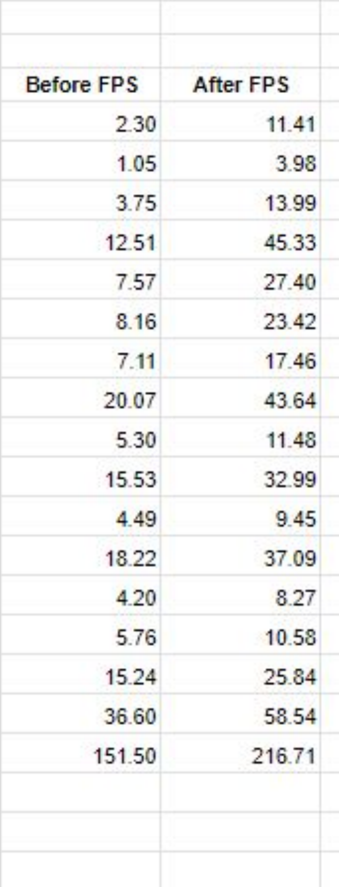
- Now completed! Improved descriptor indexing validation performance
    - Existing CPU side implementation had performance and correctness problems
        - Couldn’t determine which descriptors are ‘dynamically used’ and therefore need validation
    - Moved descriptor validation to be executed on the GPU
        - Refactored to better scale for 1M+ descriptor arrays
    - Significant performance gains
        - Gebben will elaborate more in his upcoming presentation:
- Some past performance improvements
    - Linear Memory Mapping in GPU-AV (18% to 314% improvement)
    - Fine grained locking (60% to 250% improvement)

# Synchronization Validation

- Synchronization Validation
  - Identifies resource access conflicts due to incorrect synchronization operations between (draw, copy, dispatch, blit) reading or writing the same regions of memory.
    - Within a single buffer
    - Within and between queue submissions, and across multiple queues
- Implementations complete with some limitations
  - Limited aliasing detection (More than one resource using same memory)
  - Binary Semaphore only
    - No support of timeline semaphore (very large effort)
  - Not GPU Assisted (doesn't know shader execution time information)
  - Limited extension support
  - No Host side resource tracking (CPU read/write to a Vulkan resource)
  - No swizzle and write mask support
    - Only operating at the pixel granularity

# Synchronization Validation Performance Tuning





| Before FPS | After FPS |
|---|---|
| 2.30 | 11.41 |
| 1.05 | 3.98 |
| 3.75 | 13.99 |
| 12.51 | 45.33 |
| 7.57 | 27.40 |
| 8.16 | 23.42 |
| 7.11 | 17.46 |
| 20.07 | 43.64 |
| 5.30 | 11.48 |
| 15.53 | 32.99 |
| 4.49 | 9.45 |
| 18.22 | 37.09 |
| 4.20 | 8.27 |
| 5.76 | 10.58 |
| 15.24 | 25.84 |
| 36.60 | 58.54 |
| 151.50 | 216.71 |

# Synchronization Validation Tutorials

- John Zulauf's at Vulkanised 2024 presentation
  - Using Vulkan Synchronization Validation Effectively

https://bit.ly/3U5PtWU

- Synchronization white paper:
  - https://www.lunarg.com/news-insights/white-papers/using-vulkan-synchronization-validation-effectively/
- Siggraph Birds of a Feather Presentations:
  - Ensure Correct Vulkan Synchronization by Using Synchronization Validation
    - Khronos Youtube video
  - Correct Vulkan Synchronization with Extended Synchronization Validation
    - Khronos Youtube video

# Developer tools in the Vulkan SDK

- VOLK

  New as of June 2022

  - A Vulkan entry point meta-loader
- HW Capabilities Viewer from gpuinfo.org

  New as of Oct 2022

- Vulkan Portability Solution

  Vulkan PORTABILITY

  - Vulkan® Portability™ enables the consistent use of layered implementations of Vulkan functionality over Metal and other APIs
  - Vulkan Loader: VK_KHR_portability_enumeration
  - API: VK_KHR_portability_subset
  - Vulkan Profiles Layer enables simulating the portability subset

# The macOS SDK

- iOS is now a supported platform for the Vulkan Loader and Layers (beta).
    - Enables running Vulkan layers on your iOS device with your app
        - Validation Layer
        - Apidump
        - Synchronization2 and shader object emulation layers
- More details with Richard’s presentation later today:
    - [Vulkan Development for Apple Desktops and Devices](https://bit.ly/3Hngbm9)

https://bit.ly/3Hngbm9

- All mac SDK components now support both Apple Silicon and Intel Architectures
    - No longer need Rosetta emulation environment
        - Except during SDK installation. Installer needs Rosetta
- Resources:
    - LunarG White Paper - The state of Vulkan on Apple Devices

LUNARG

# GFXReconstruct

- Capture Vulkan API calls in a file with VK_LAYER_LUNARG_gfxreconstruct
  - Replay with gfxrecon-replay
- Linux, Windows, Android
- API-agnostic; Vulkan and Direct3D 12 so far!
- Use cases
  - Bug reporting
  - App debugging
  - Silicon development
  - Driver quality testing
- Resources:
  - Siggraph 2023:
    - Capture & Replay with Vulkan & DX12: GFXReconstruct
  - Munich Vulkanised 2023:
    - GFXReconstruct- Tools to Capture and Replay Graphics API Calls
      - Youtube Video

2023 Siggraph
BoF:

# Vulkan Profiles

**Essential for Roadmap 2024**

- A mechanism that enables the precise specification of capabilities
  - Communication of capabilities to participants in the Vulkan ecosystem
  - Easier Vulkan development for a selected range of actual ecosystem devices
- Khronos Roadmap 2024
  - Next SDK: Config\VK_LAYER_KHRONOS_profiles\VP_KHR_roadmap.json
    - Will be updated to include the 2024 Roadmap in addition to the 2022 Roadmap

# Example Profiles Usage

- Roadmap profiles: To express guidance on the future direction of Vulkan devices
  - In the SDK: Config/VK_LAYER_KHRONOS_profiles\VP_KHR_roadmap.json
  - Will include any Khronos Roadmap profiles
- Platform profiles: To express Vulkan support available on different platforms
  - In the SDK:
    - Config\VK_LAYER_KHRONOS_profiles\VP_ANDROID_15_minimums.json
    - Config\VK_LAYER_KHRONOS_profiles\VP_LUNARG_desktop_baseline_2024.json
- Device Profiles: To express Vulkan support of a single Vulkan device
  - Gpuinfo.org provides device profiles
- Engine Profiles: To express requirements of the rendering code path
  - Prevent application from requiring unavailable features on devices

# Vulkan Profiles Toolset

- Profiles Schema -  A JSON data format to communicate about Vulkan capabilities
  - Extensions, features, properties, formats, and queue properties
  - Schema for each Vulkan API revision (KhronosGroup/Khronos-Schemas)
- VK_LAYER_KHRONOS_profiles
  - A layer used during application development to ensure adherence to the requirements of a chosen Vulkan Profile.
  - It simulates Vulkan capabilities. It works together with the Validation layer which reports errors when using capabilities not exposed by the Vulkan developer system.
  - The layer requires a Vulkan 1.1 driver.
- Vulkan Profiles Library
  - A header-only C++ library to use Vulkan Profiles in Vulkan applications
  - Checking Profile support on a device.
    - Create a vkDevice instance w/ features/extensions enabled
  - The library requires a Vulkan 1.0 driver that supports the VK_KHR_get_physical_device_properties2 extension.
- The Vulkan Profiles JSON file generation
  - Generate profiles file by combining multiple existing profiles
  - Union and intersection of Vulkan capabilities

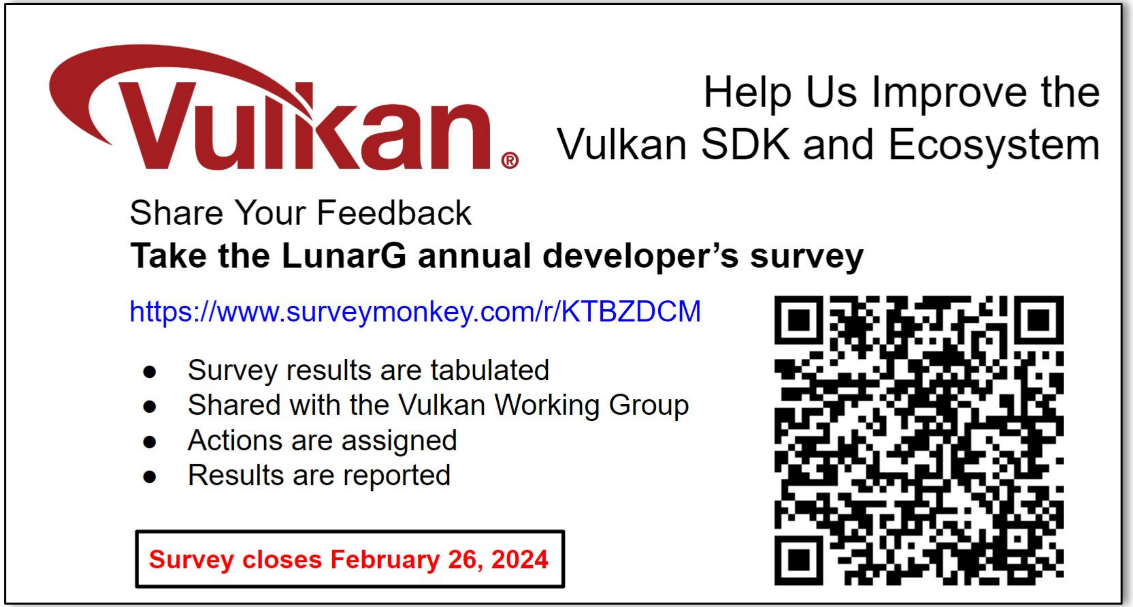
# Profiles Tutorials

https://bit.ly/3SkZZIe

- 2024 Vulkanised
  - Better Vulkan Applications Deployment Thanks to Vulkan Profiles
  - LunarG White Paper:
    - Better Vulkan Applications Deployment Thanks to Vulkan Profiles
- 2023 Munich Vulkanised
  - Creating Vulkan Profiles
  - Khronos Youtube Video
- 2022 Khronos Vulkanised
  - Vulkan SDK Tools to Use and Create Vulkan Profiles
  - Khronos Youtube Video

# Shader Tool Chain

- Offline executables and API libraries for:
  - SPIRV-Tools
    - Validator, optimizer, assembler, disassembler, diff, Remapper
  - GLSL->SPIR-V
    - glslang SPIR-V generator
  - HLSL->SPIR-V
    - Glslang SPIR-V generator (up to shader model 5)
    - DXC (Microsoft DirectX Shader Compiler)
  - Shaderc
    - Glslang and SPIRV-Tools wrapper for better integration with build tools
  - SPIRV-CROSS
    - SPIR-V shaders -> HLSL/Metal/GLSL shaders
  - SPIRV-Reflect
    - Provides a C/C++ reflection API for SPIR-V shader bytecode
- Did you know? A really handy tool to visualize your SPIR-V
  - https://www.khronos.org/spir/visualizer/



Vulkan®

## Help Us Improve the Vulkan SDK and Ecosystem

Share Your Feedback

**Take the LunarG annual developer's survey**

https://www.surveymonkey.com/r/KTBZDCM

- Survey results are tabulated
- Shared with the Vulkan Working Group
- Actions are assigned
- Results are reported

**Survey closes February 26, 2024**

Today's Presentation:

https://bit.ly/420QuRQ

# Thank you!

# QUESTIONS?

Get A FREE Tumbler at the LunarG Sponsor Table!

LUNARG