

Vulkan SDK Tools to Use and Create Vulkan Profiles, a Tutorial

Christophe Riccio, LunarG, Inc

KHRONOS
GROUP

WEBINARS
& MEETUPS

Vulkan®

Agenda

- A brief presentation of the Vulkan SDK Profiles Tools
- A tutorial on how to use the Vulkan SDK Profiles Tools

Please, ask your questions in the chat. (Žiga will moderate it)

- Link for this presentation
- Based on The Vulkan Profiles Toolset solution whitepaper
 - <https://www.lunarg.com/wp-content/uploads/2022/03/The-Vulkan-Profiles-Toolset-Solution-FEB2022.pdf>

Vulkan Profiles

What's a Vulkan Profile?

- Released with Vulkan 1.3
 - But it's not really a part of the Vulkan specification, it's essentially developer tools.
- A collection of Vulkan Capabilities
 - Extensions
 - Features
 - Properties
 - Queue properties
 - Formats
 - Etc...
- A way to increase the minimum requirements of Vulkan specification
 - Based on our Vulkan developer needs

Vulkan Profiles usages:

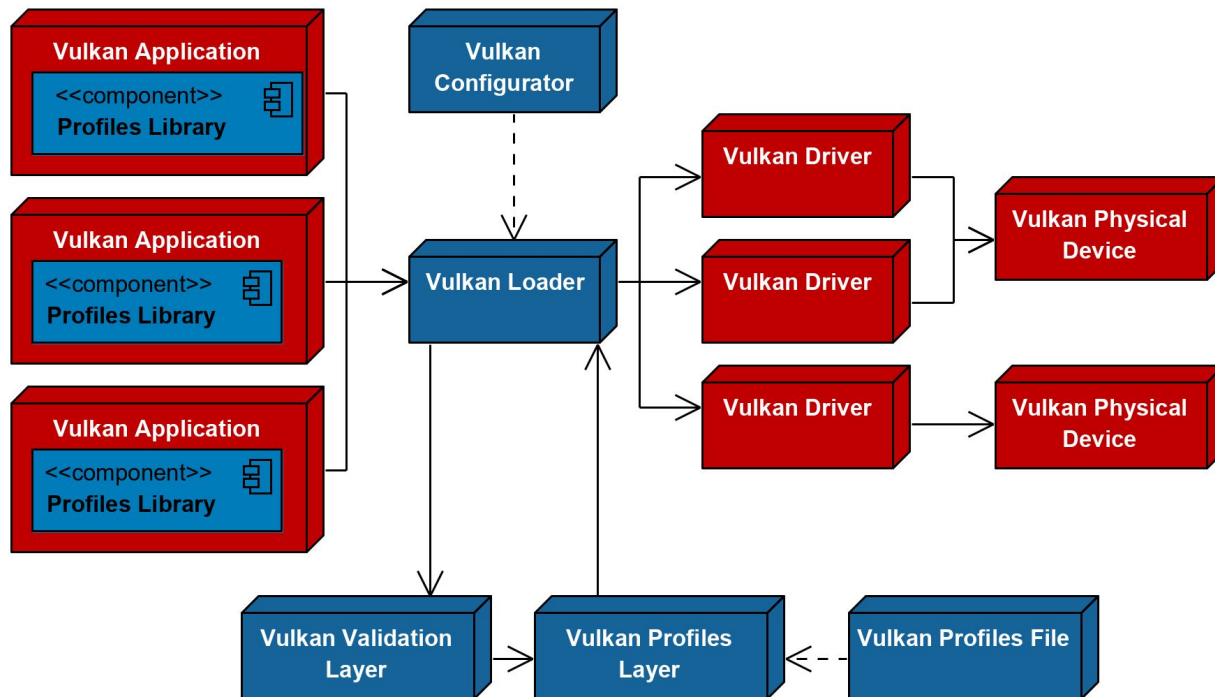
- *Roadmap profiles*: to express guidance on the future direction of Vulkan devices or projects. Eg: Vulkan Roadmap 2022.
- *Platform profiles*: to express the Vulkan support actually available on a platform. Eg: Android Baseline 2021.
- *Device profiles*: to express the Vulkan support of a single Vulkan driver for a Vulkan device. Eg: [GPUinfo.org reports](https://gpuinfo.org/reports)
- *Architecture profiles*: to express the Vulkan support of a class of GPUs. Eg: D3D12 Feature Level 12.1
- *Engine profiles*: to express some rendering code paths requirements of an engine. (Eg: VKD3D and VK_EXT_mutable_descriptor_type)
- Etc.

The Vulkan SDK Profiles Tools

The Vulkan SDK Profiles Tools:

- Vulkan Profiles JSON schema
 - A JSON format to exchange Vulkan capabilities programmatically
 - One JSON schema per Vulkan Header revision
- Vulkan Profiles file generation
 - Vulkaninfo and [GPUinfo.org](https://gpuinfo.org) export *Device Profile JSON files*
 - Command line tool for multiple profiles intersection or union of capabilities
 - VP_LUNARG_desktop_baseline_2022 provided as a profile example
- Vulkan Profiles layer
 - A layer to emulate/clamp profile capabilities on Vulkan developer system
- Vulkan Profiles API library
 - C++ code ; Header-only or Header + Source
 - A library for Vulkan applications code to check profiles support, to create `VkDevice` with features enabled
 - [A KhronosGroup/Vulkan-Samples sample](https://github.com/KhronosGroup/Vulkan-Samples) is available on github for demonstrating the library usage
- Vulkan Profiles comparison table
 - [Markdown documentation](#), to easily read, search, compare capabilities across profiles

Deployment of the tools



Vulkan Layers Management

- ☐ Layers Fully Controlled by the Vulkan Applications
- ☒ Overriding Layers by the Vulkan Configurator
- ☐ Apply only to the Vulkan Applications List
- ☐ Continue Overriding Layers on Exit

Edit Applications...

Vulkan Layers Configurations

- ☐ API dump
- ☐ Frame Capture
- ☒ Portability
- ☐ Synchronization
- ☐ Validation

New...

Edit...

Duplicate

Remove

Vulkan Application Launcher

> Application vkcube

☒ Clear log at launch

Clear

Vulkan Loader Messages:

none

Launch

Vulkan Development Status:

```
- Layers override: "Portability" configuration
- VULKAN_SDK environment variable: E:\VulkanSDK\1.3.216.0
- Vulkan Loader version: 1.3.204
- User-Defined Layers locations:
  - VK_LAYER_PATH variable: None
  - Per-configuration paths:
    - E:\Github\khronos\Vulkan-Profiles - Master\build\bin\Debug
  - VK_ADD_LAYER_PATH variable: None
- `vk_layer_settings.txt` uses the default platform path:
  C:\Users\Piranha\AppData\Local\LunarG\vkconfig\override
- Available Layers:
  VK_LAYER_KHRONOS_validation
  VK_LAYER_KHRONOS_profiles
```

Portability Settings

Vulkan Applications

> VK_LAYER_KHRONOS_validation

v VK_LAYER_KHRONOS_profiles (BETA)

LunarG Desktop Baseline 2022 Preset

v Profile Selection

LUNARG_desktop_portability_2022.json

LUNARG_desktop_portability_2022

☐ Schema Validation☒ Emulate VK_KHR_portability_subset

v Simulate Profile Capabilities

☒ Version☒ Features☒ Properties☐ Device Extensions☐ Formats

Exclude Device Extensions

Exclude Formats

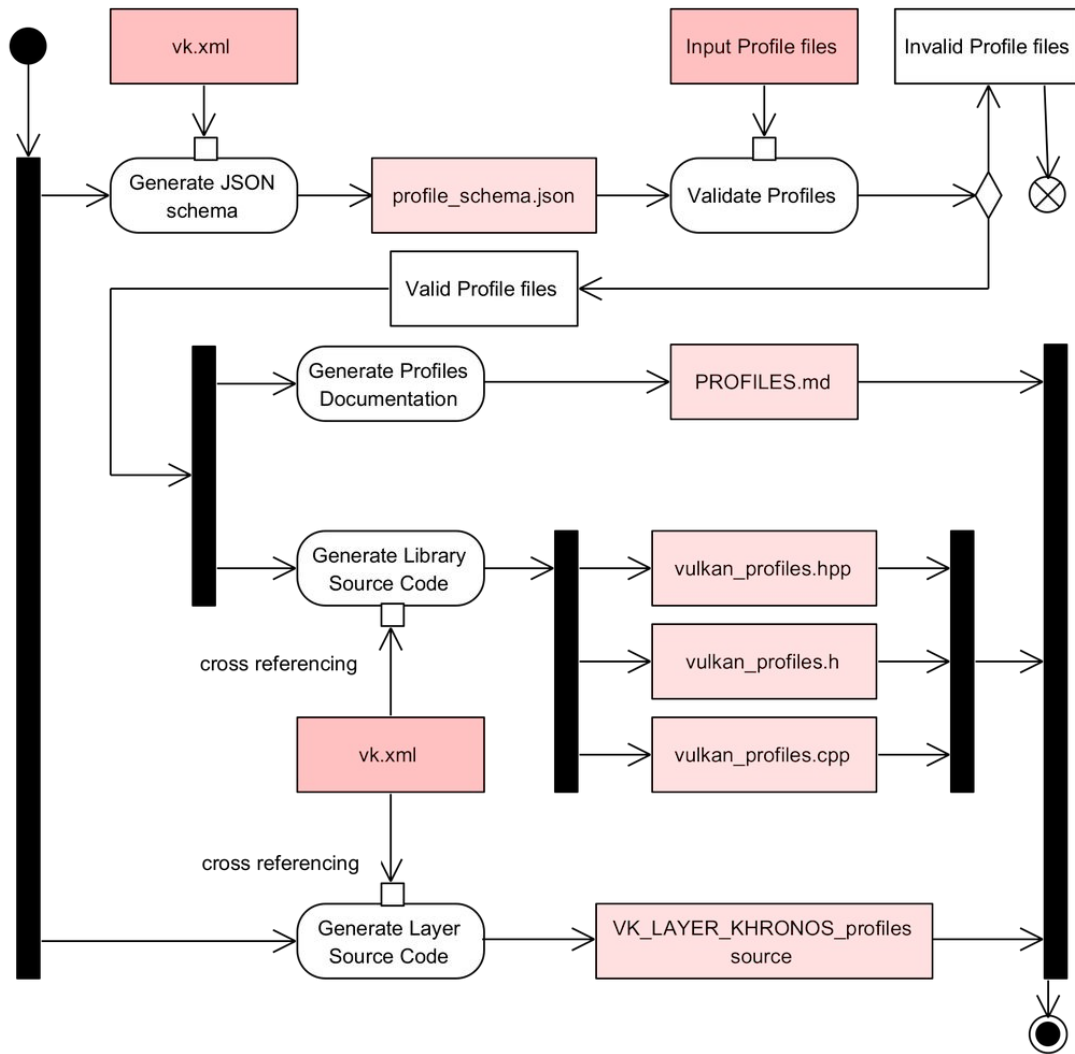
v Debug Actions

☒ Log to stdout☐ Log to OutputDebugString☐ Log to File

v Log Filename

profiles_layer_log.txt

☒ Clear Log at Launch



A full moon is visible in the upper right quadrant of the image, appearing as a bright, detailed sphere against the starry background. The sky is a deep blue, densely populated with stars of varying brightness. In the lower left, the dark silhouette of a large, leafless tree stands prominently, its branches reaching upwards. The horizon is a thin, dark line at the bottom of the frame.

Tutorial!

A full moon is visible in the upper right corner of a deep blue night sky filled with numerous stars. In the lower left, the dark silhouette of a large, leafless tree stands against the horizon. The overall scene is a serene night landscape.

Questions?