



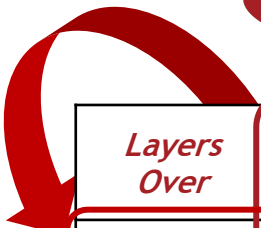
Vulkan Portability Update

January 2021



Open-Source Layering Projects

Fighting Platform Fragmentation



| <i>Layers Over</i> | Vulkan | OpenGL | OpenCL | OpenGL ES | DX12 | DX9-11 |
|--------------------|-----------------------------|-----------------------|-------------------------|-------------------|-----------------------|------------------------|
| Vulkan | | Zink | clspv clvk | Angle GLOVE | vkd3d-Proton vkd3d | DXVK WineD3D |
| OpenGL | gfx-portability Ashes | | | Angle | | WineD3D |
| DX12 | gfx-portability | Microsoft 'GLOn12' | Microsoft 'CLOn12' | | | Microsoft D3D11On12 |
| DX9-11 | gfx-portability Ashes | | | Angle | | |
| Metal | MoltenVK gfx-portability | | clspv + SPIRV-Cross? | MoltenGL Angle | | |

Vulkan as
porting target for
multiple APIs

ROWS:
Bring more APIs
to Platforms

Vulkan functionality
available everywhere

COLUMNS: Making APIs
available across platforms

Vulkan Portability

TODAY



Open-source tools, SDKs and libraries bring Vulkan applications to Apple platforms over Metal



Multiple Implementations of Layered Vulkan Functionality

Enable Vulkan applications on diverse platforms and APIs

Khronos Portability Extension

Portable queries for what functionality is available through Vulkan layering

Extended Vulkan Conformance Tests

Gracefully handle partial functionality to enable Vulkan Portability conformance - what is present must work!

Enhanced Vulkan Layers and Tools

DevSim/Validation Layers flag or simulate queries for features that are not present



+more!





Fighting fragmentation by enabling Vulkan applications on platforms without native drivers through layered implementations of Vulkan functionality over Metal, DX12 and other APIs

<https://github.com/KhronosGroup/Vulkan-Portability>

KRONOS®
GROUP



Example Vulkan Portability Implementations

| | | |
|--------------------------------|---|---|
| |  |  |
| Baseline Vulkan Version | Vulkan 1.1 with Vulkan 1.2 in development | Vulkan 1.0 with extensions |
| API/Platforms Supported | Metal on macOS, iOS, tvOS and Mac Catalyst (Shipping) macOS 10.11 / iOS 9.0 and up | Metal on macOS and iOS (in Beta) DirectX 12 (in development) UWP platforms such as Windows 10 S, Polaris, Xbox. GPUs without native drivers e.g., Intel Haswell/Broadwell OpenGL (ES)/D3D11 (in development) |
| Implementor | Community and Brenwill Workshop | Community and Mozilla |
| Licensing | Royalty-free Apache 2.0 | Royalty-free Apache 2.0 / MIT |
| Notes | Used by multiple production applications on Mac and iPhone | Implemented using Rust Language |
| More Information | https://github.com/KhronosGroup/MoltenVK | https://github.com/gfx-rs/gfx https://github.com/gfx-rs/portability https://github.com/gfx-rs/portability/releases |

Example beta and prototype implementations that are expected to be Vulkan Portability conformant once the Vulkan Portability Extension and conformance tests are complete

Vulkan Portability Timelines



Upcoming Deliverables
Finalized Conformance Tests
Portability Extension 1.0



MoltenVK released
in open source for
macOS and iOS



Vulkan CTS
ported to macOS



Vulkan CTS
ported to macOS



gfx-rs DX11
backend in alpha -
joins DX12 and
Metal backends



Vulkan Portability
Provisional
Extension released
and shipped in
MoltenVK and gfx-rs



MoltenVK 1.1 ships
with support for
Vulkan 1.1 features,
48 Vulkan extensions,
Metal 3.0, Apple
Silicon, Mac Catalyst
and tvOS



Vulkan SDK Validation
Layers support
Portability Extension.
Vulkan Configurator
manages DevSim and
config files

GDC
2018

August
2018

September
2020

Fall
2020

January
2021

Vulkan Apps on macOS and iOS using MoltenVK



"I was initially skeptical about the use of a Vulkan translation layer to Metal, but it turns out MoltenVK enables applications using the real-time [NAP Framework](#) to run as well on macOS as they do on Linux and Windows. This is something I'd like to highlight and stress the importance of - without the portability of Vulkan we would have been forced to implement a special Metal rendering backend just for macOS users, something not doable for a relatively small company such as ours. Plus switching to Vulkan dramatically improved render-times for Mac users."

Coen Klosters, Lead Developer of NAP Framework and Technical Director at Naivi



Multiple iOS and macOS apps shipping
e.g., Forsaken Remastered



<https://google.github.io/filament/>

Google Filament PBR
Renderer on Mac



<https://www.winehq.org/>

Initial ports of DX
games in progress
using Vulkan on macOS



ARTIFACT

Artifact ships on
Steam for macOS -
first Valve Vulkan-
only app on macOS

RPCS3

<https://rpcs3.net/>

RPCS3 PlayStation 3
Emulator on macOS



<https://dolphin-emu.org/>

Dolphin GameCube
and Wii Emulator
working on macOS



<https://diligentgraphics.com/diligent-engine/>

Diligent Engine
runs on macOS



CodeWeavers and
Square Enix ship Final
Fantasy XIV on macOS



Underlords ships on
Steam - first Valve
Vulkan app on both
macOS and iOS



CODE
WEAVERS
SOFTWARE LIBRARIANS

Crossover Windows
compatibility layer

N A P



<https://www.napframework.com/>

Naivi ports NAP
real-time
performance
engine to macOS



Production Dota 2 on
macOS with up to
50% more perf than
Apple's OpenGL

First iOS Apps using
MoltenVK ship
through app store



<https://www.qt.io/blog/2018/05/30/vulkan-for-qt-on-macos>

Qt Running on macOS
through MoltenVK

June
2018

September
2018

November
2018

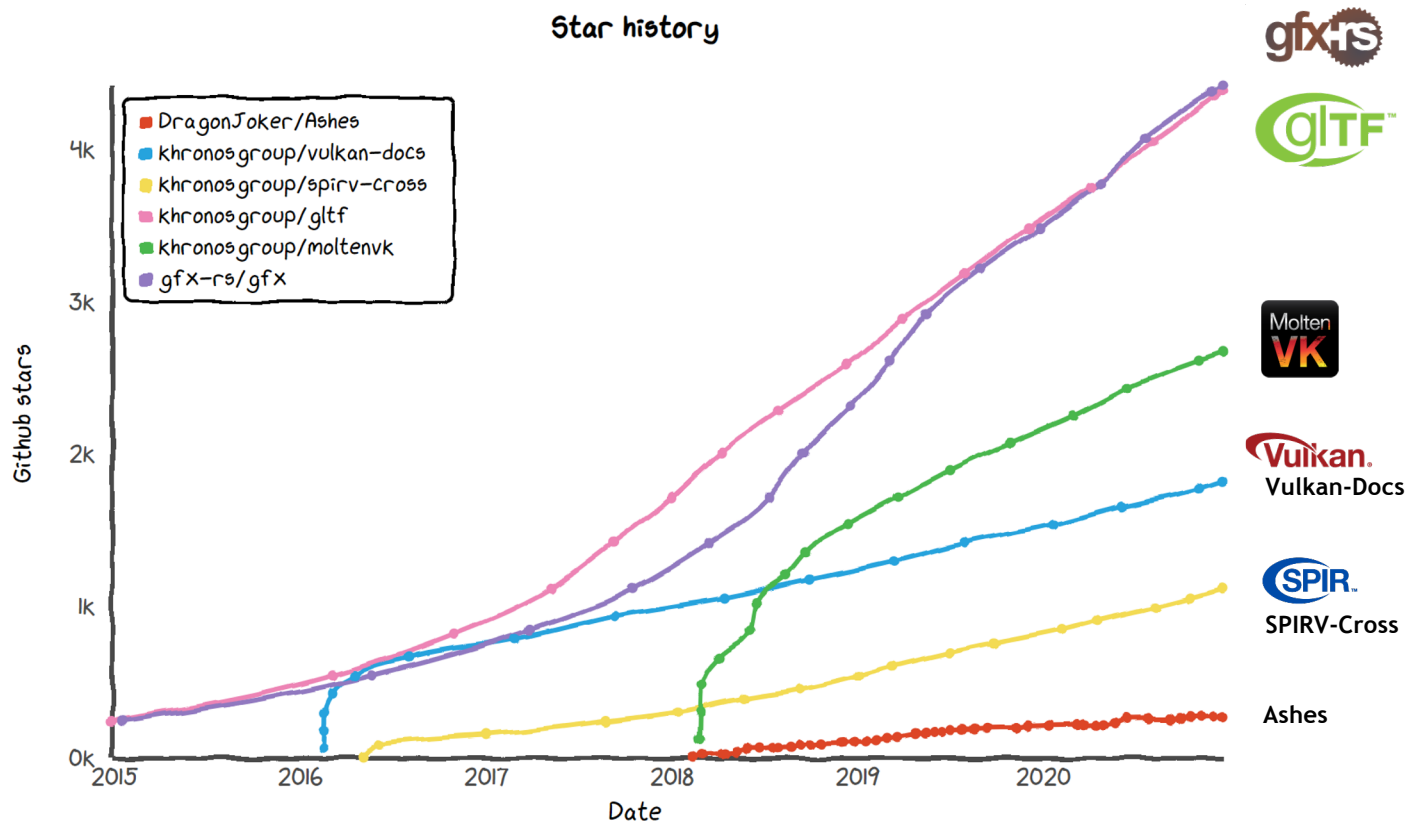
January
2019

June
2019

January
2021

Khronos-Related GitHub Projects

Star history



<https://star-history.t9t.io/#khronosgroup/moltenvk&khronosgroup/gltf&khronosgroup/vulkan-docs&khronosgroup/spirv-cross&gfx-rs/gfx&DragonJoker/Ashes>