

# v3dv: Status Update for Open Source Vulkan Driver for Raspberry Pi 4

Alejandro Piñeiro Iglesias <[apinheiro@igalia.com](mailto:apinheiro@igalia.com)>



# Contents

- Fosdem 2021 recap
- Current status
- Performance
- Path to 1.1 conformance
- Contributing



# Fosdem 2021 recap



# Fosdem 2021 recap

- [Aug 20] Minimal Vulkan 1.0 feature set
- [Oct 20] Moved development to Mesa upstream
- [Nov 20] Vulkan 1.0 conformant
- [Nov 20] Started to test on real world apps
- [Dec 20] Tested on 64-bit, started to work on performance



# Current status



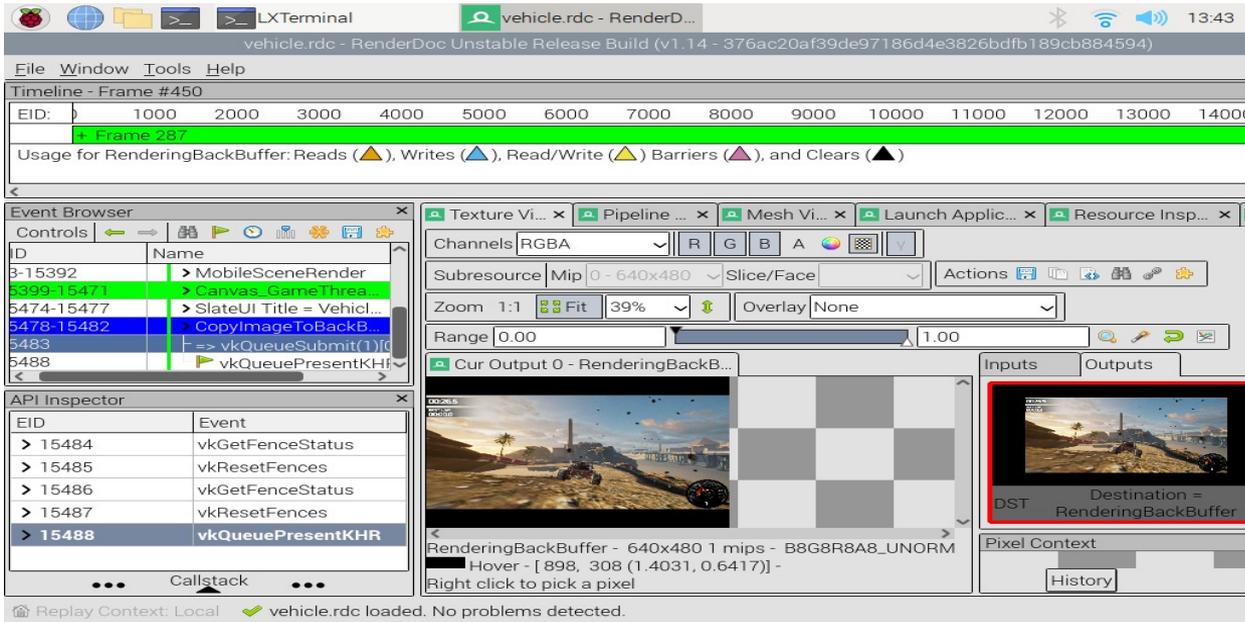
# Current Status

- Upstream gitlab CI integration
- Better WSI platform support (Wayland, Display)
- Significant performance improvements
- Vulkan 1.1 conformant



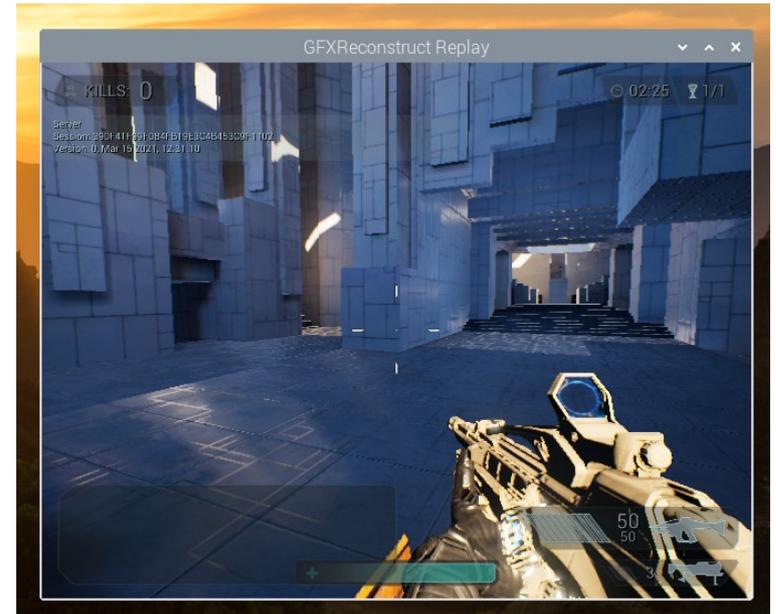
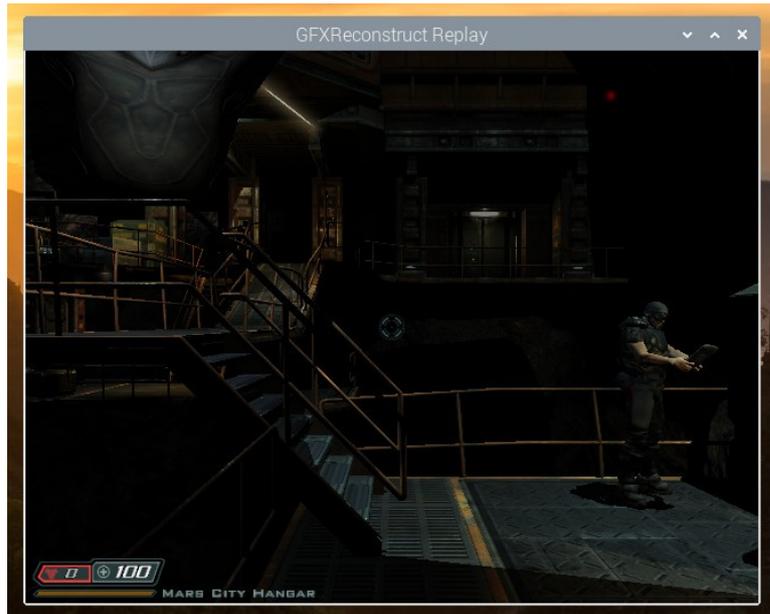
# 2021 status

- RenderDoc



# 2021 status

- GFXReconstruct



# 2021 status

- Various UE4 samples running successfully



# Performance



# Performance

- Driven by native Unreal Engine 4 samples
  - Generally GPU limited
  - Very expensive shading
- Focus on backend shader code optimizations
  - OpenGL/ES driver also benefited

# Performance

- Process:
  - Capture generated shader code
  - Identify non optimal code traces
  - Figure out how that code is generated
  - Design & Implement optimizations
  - Verify results:
    - shader-db + GFXReconstruct + manual testing

# Performance

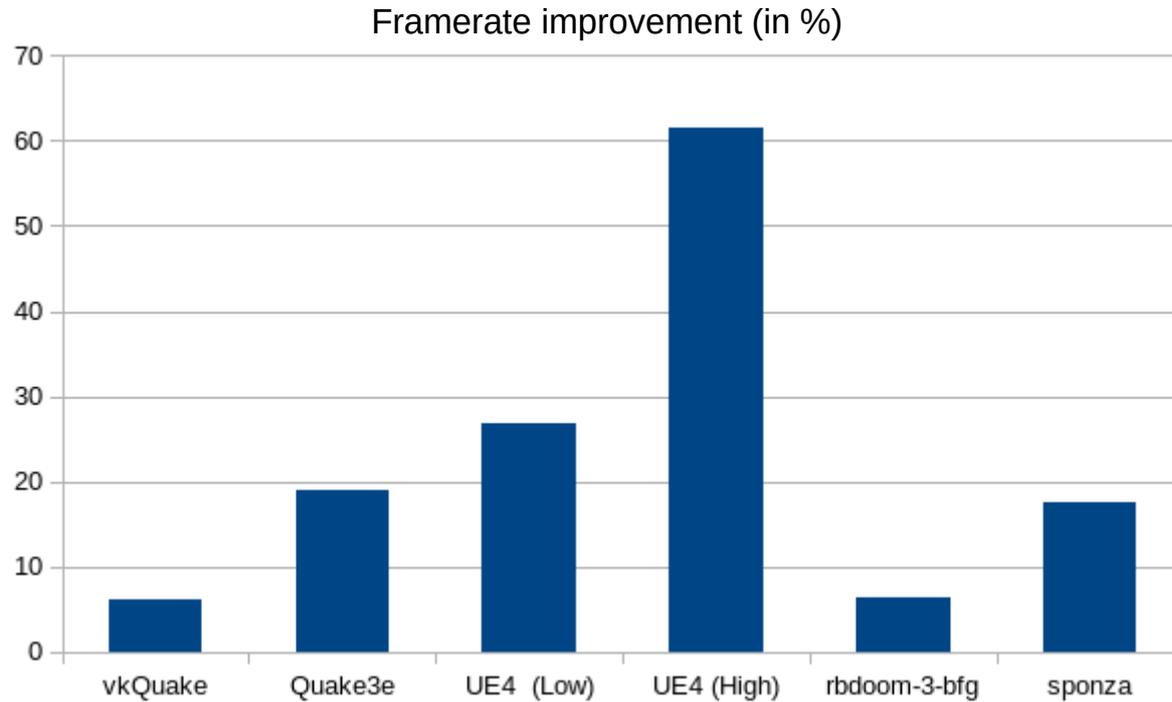
- Better pipelining of TMU operations
- Non-TMU access for uniform UBO reads
- QPU code scheduling improvements
- Better instruction-level parallelism
- Better pipelining of varying setup instructions
- And many more...

# Performance

Shader-db improvement (in %)

Stat	Improvement
Threads	+1.06%
Instructions	-9.00%
Uniforms	-6.86%
Spills / Fills	-4.82% / -8.33%

# Performance



# Performance

- Many optimizations increase register pressure
- Recompile with opts disabled to avoid spills
- UE4 samples compile at run time → stutter
- Implemented a disk cache to mitigate this
  - Vulkan only for now

# Performance

- NIR scheduling useful but very limited
  - Might want to consider a VIR scheduler
- Better VIR doesn't always lead to better QPU
  - Room for improvement in RA and QPU scheduling
- Compile strategies not always optimal

# Path to 1.1 conformance



# Path to 1.1 conformance

- 1.1 was included as “Maybe?” on last FOSDEM
- Started adding the most common 1.0 optional extensions
- As the progression was good, focused on pending extensions for 1.1

# Path to 1.1 conformance

VK\_KHR\_get\_display\_properties2

VK\_KHR\_multiview

VK\_KHR\_copy\_commands2

VK\_KHR\_shader\_non\_semantic\_info

VK\_KHR\_descriptor\_update\_template

VK\_KHR\_storage\_buffer\_storage\_class

VK\_KHR\_relaxed\_block\_layout

VK\_KHR\_device\_group

VK\_KHR\_dedicated\_allocation

VK\_KHR\_get\_memory\_requirements2

VK\_KHR\_bind\_memory2

VK\_KHR\_maintenance3

VK\_KHR\_maintenance2

VK\_KHR\_create\_renderpass2

VK\_KHR\_swapchain\_mutable\_format

VK\_KHR\_surface\_protected\_capabilities

VK\_EXT\_custom\_border\_color

VK\_KHR\_display



# Path to 1.1 conformance

- Several required little work, thanks to common mesa implementation
- More tricky VK\_KHR\_multiview, as involved a custom passthrough geometry shader using layered rendering

# Future Plans



# Future Plans

- Continue adding extensions and features
  - Maybe start experimenting with fp16
- Resume performance work
- Improve kernel interface (multisync)
- Maybe start work on Vulkan 1.2

# Contact

- IRC: #videocore@OFTC
- Mailing list: [mesa-devel@freedesktop.org](mailto:mesa-devel@freedesktop.org)
- Gitlab: <https://gitlab.freedesktop.org/mesa/mesa>
- Blogs:
  - <https://blogs.igalia.com/itoral>
  - <https://blogs.igalia.com/apinheiro>



# Q&A

We are hiring: [www.igalia.com/jobs](http://www.igalia.com/jobs)

