



COLLABORA

Vulkan Video Encoder

@ndufresne

Nicolas Dufresne

nicolas@collabora.com

Open First



NOT a Linux Media Project

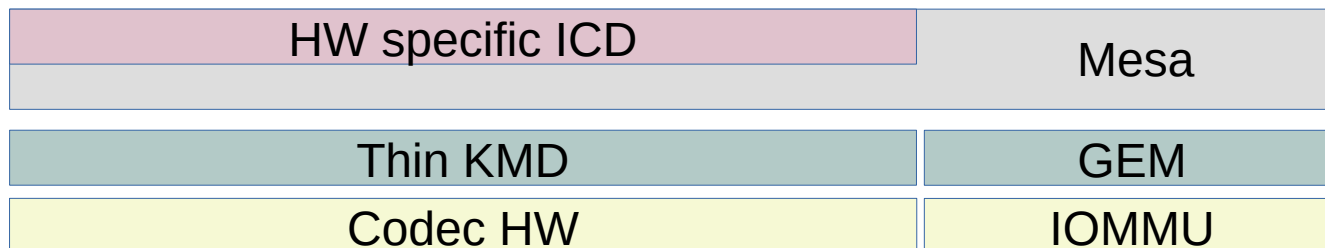
VV on Linux DRM/Accel based driver

Pros:

- Amazon **Proof of Concept** [KMD](#) and [ICD](#)
- **Thinner** Linux driver**
- **Rust** language supported
- **Flexible** memory model
- **Built-in synchronization** including foreign synchronization object

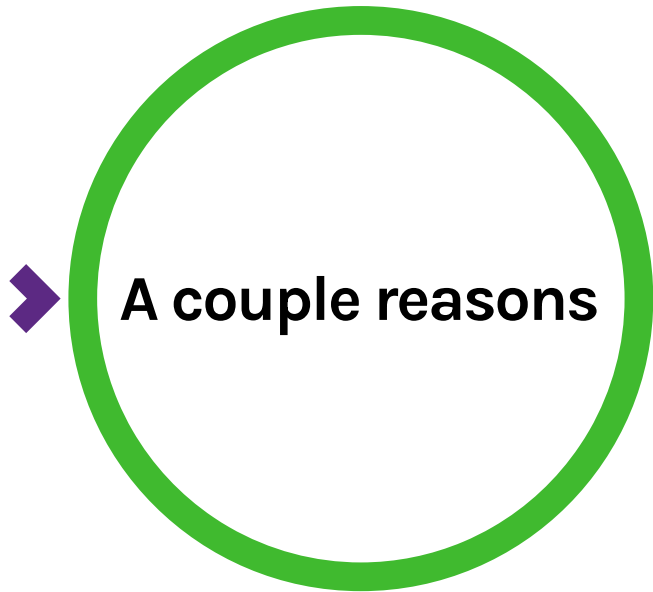
Cons:

- **CPU accessed hardware drivers are thinner as long as you have memory protection such as an **IOMMU**
- No VPU only drivers upstream Linux yet





Why not Linux Media?



- Not wanting to **add a standard** behind the standard, saving **time and effort**
- The uAPI is made to support the **fixed HW interface**, a clear goal that won't evolve over time
- Vulkan has **atomic submission model**, V4L2 split that over to then reassemble (lower complexity)
- Vulkan let you expose **all the resources**, while V4L2 only show you the data on endpoints
- The HW **can protect itself** from bad memory access, we don't need as much protection at parameter level





What DRM say about it so far...

- Its **not a new thing**, AMD and Intel drivers for VPU exists, they just accidentally also host a GPU.
- **No free for all**, pretty much **not accel**
- **OSS** support as the main target (**mesa**)
- Linux Media and DRM subsystem share that same ambition for OSS, but implements it differently





Why do we care then ?



Why I care

- Regardless what comes next, regardless if decoders also move to that model, I'm still going to make codec driver
- Some of the principle that motivate DRM style drivers could pretty much exist in Linux Media, perhaps even shared





Ideas

- Introduce our own flexible buffer object (or share GEM?)
- Move away from homogeneous buffer queues, introduce generic command queues, maybe we already have that (request, parameter formats)
- Recenter M2M around their media driver, drop video drivers





Thank you!



COLLABORA

Open First