HOW THE VULKAN VK_EXT_EXTENDED_DYNAMIC_STATE EXTENSION CAME TO BE

Ricardo Garcia
rgarcia@igalia.com

WHAT THE EXTENSION DOES

VK_EXT_extended_dynamic_state adds some more dynamic state to support applications that need to reduce the number of pipeline state objects they compile and bind.

GRAPHICS PIPELINE

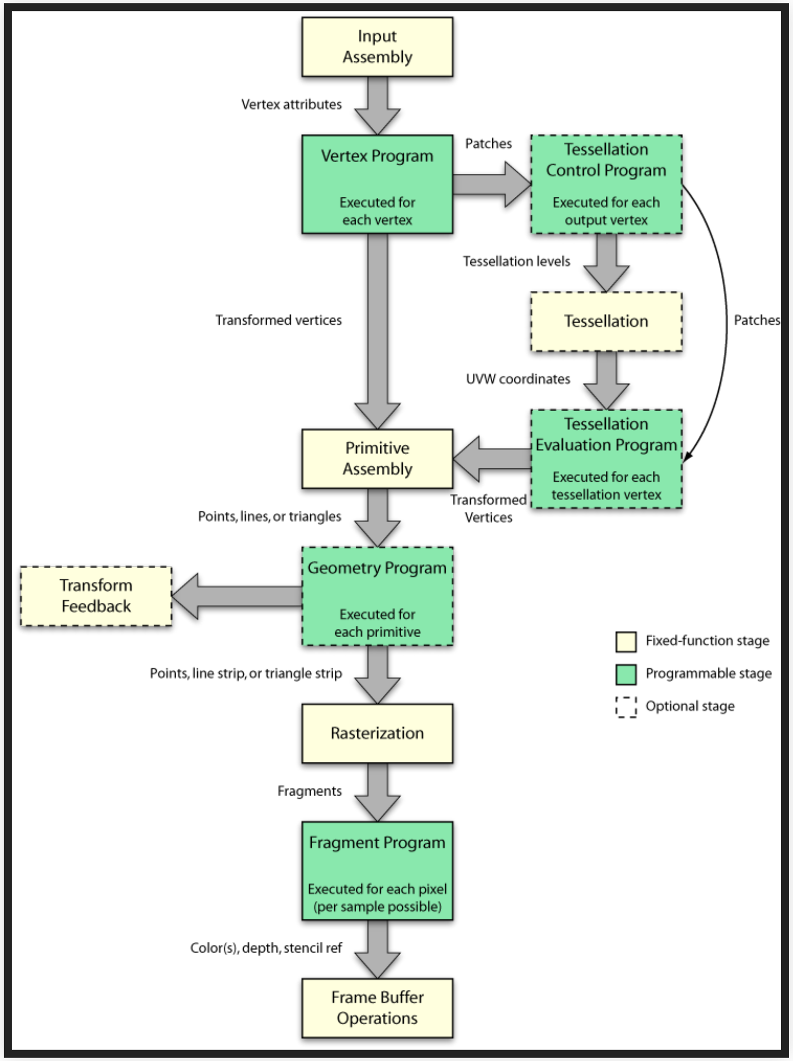
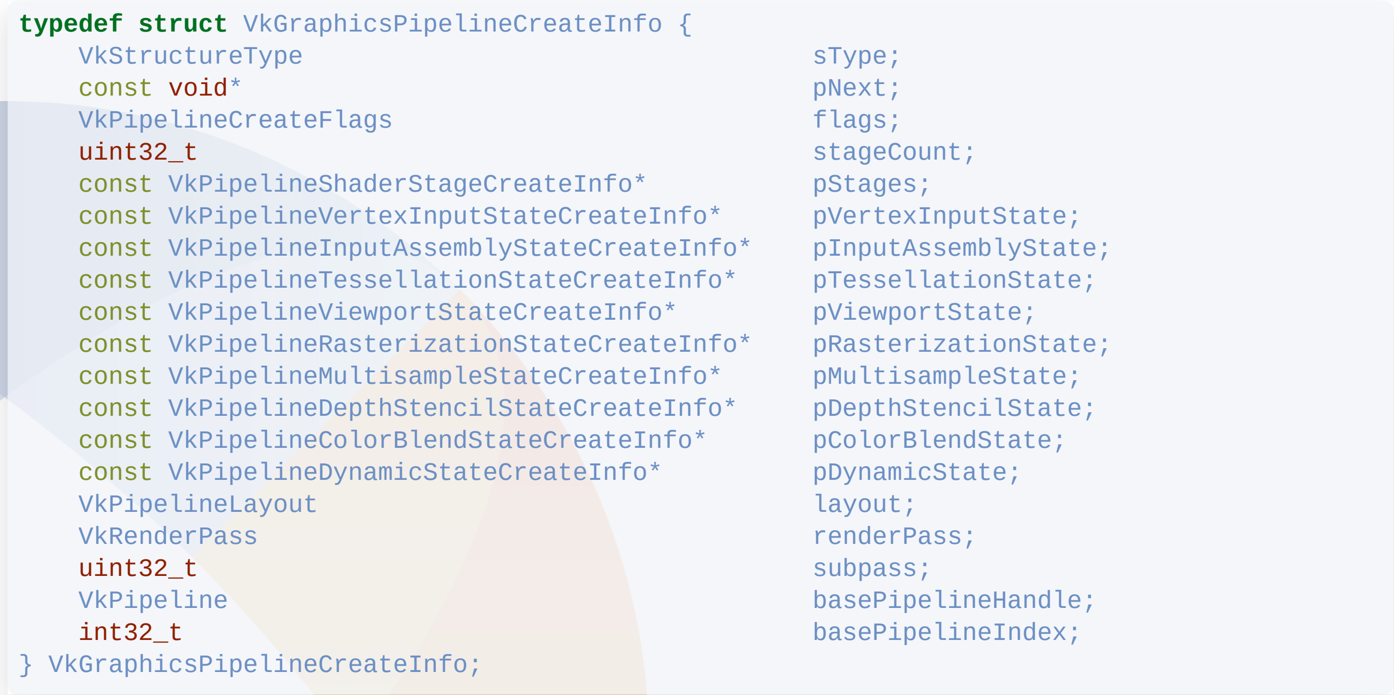


Diagram by Eric Lengyel released under the GNU FDL
`https://commons.wikimedia.org/wiki/File:The_OpenGL_-_DirectX_graphics_pipeline.png`

PIPELINE CREATION UNDER VULKAN



```
typedef struct VkGraphicsPipelineCreateInfo {
    VkStructureType                                  sType;
    const void*                                      pNext;
    VkPipelineCreateFlags                            flags;
    uint32_t                                         stageCount;
    const VkPipelineShaderStageCreateInfo*           pStages;
    const VkPipelineVertexInputStateCreateInfo*      pVertexInputState;
    const VkPipelineInputAssemblyStateCreateInfo*    pInputAssemblyState;
    const VkPipelineTessellationStateCreateInfo*     pTessellationState;
    const VkPipelineViewportStateCreateInfo*         pViewportState;
    const VkPipelineRasterizationStateCreateInfo*    pRasterizationState;
    const VkPipelineMultisampleStateCreateInfo*      pMultisampleState;
    const VkPipelineDepthStencilStateCreateInfo*     pDepthStencilState;
    const VkPipelineColorBlendStateCreateInfo*       pColorBlendState;
    const VkPipelineDynamicStateCreateInfo*          pDynamicState;
    VkPipelineLayout                                 layout;
    VkRenderPass                                     renderPass;
    uint32_t                                         subpass;
    VkPipeline                                       basePipelineHandle;
    int32_t                                          basePipelineIndex;
} VkGraphicsPipelineCreateInfo;
```

NEW DYNAMIC STATES

```
VK_DYNAMIC_STATE_CULL_MODE_EXT
VK_DYNAMIC_STATE_DEPTH_BOUNDS_TEST_ENABLE_EXT
VK_DYNAMIC_STATE_DEPTH_COMPARE_OP_EXT
VK_DYNAMIC_STATE_DEPTH_TEST_ENABLE_EXT
VK_DYNAMIC_STATE_DEPTH_WRITE_ENABLE_EXT
VK_DYNAMIC_STATE_FRONT_FACE_EXT
VK_DYNAMIC_STATE_PRIMITIVE_TOPOLOGY_EXT
VK_DYNAMIC_STATE_SCISSOR_WITH_COUNT_EXT
VK_DYNAMIC_STATE_STENCIL_OP_EXT
VK_DYNAMIC_STATE_STENCIL_TEST_ENABLE_EXT
VK_DYNAMIC_STATE_VERTEX_INPUT_BINDING_STRIDE_EXT
VK_DYNAMIC_STATE_VIEWPORT_WITH_COUNT_EXT
```

NEW COMMANDS

```
vkCmdBindVertexBuffers2EXT
vkCmdSetCullModeEXT
vkCmdSetDepthBoundsTestEnableEXT
vkCmdSetDepthCompareOpEXT
vkCmdSetDepthTestEnableEXT
vkCmdSetDepthWriteEnableEXT
vkCmdSetFrontFaceEXT
vkCmdSetPrimitiveTopologyEXT
vkCmdSetScissorWithCountEXT
vkCmdSetStencilOpEXT
vkCmdSetStencilTestEnableEXT
vkCmdSetViewportWithCountEXT
```

GRAPHICS PIPELINE

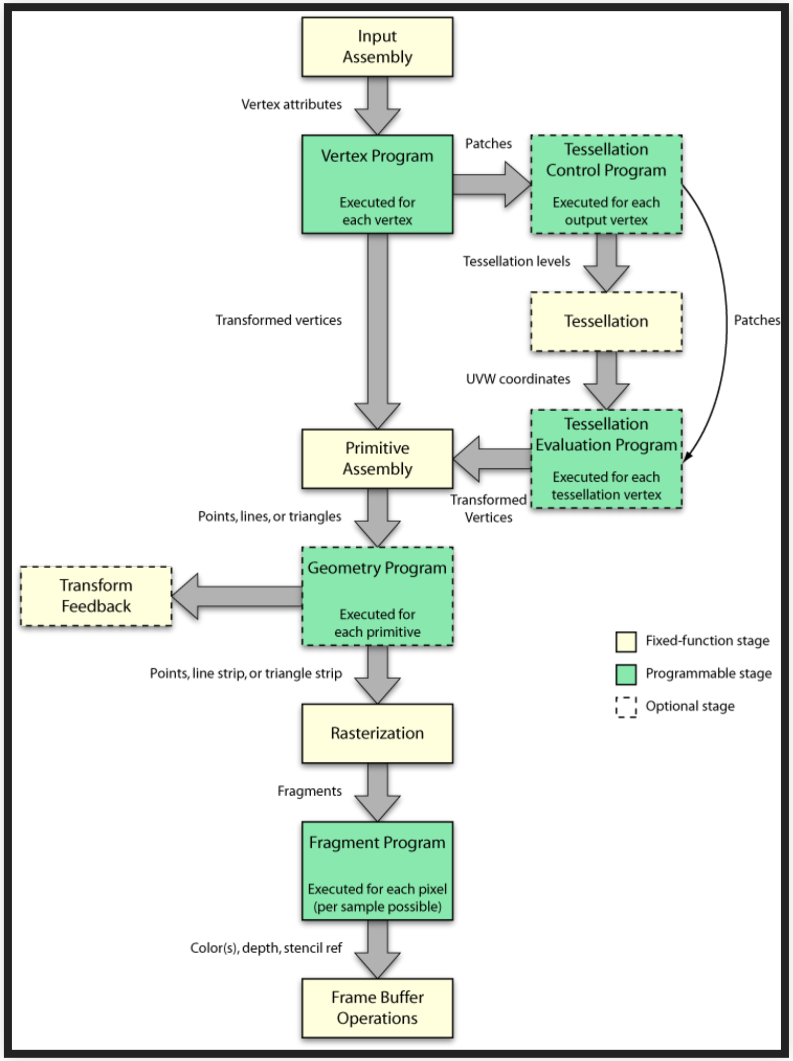
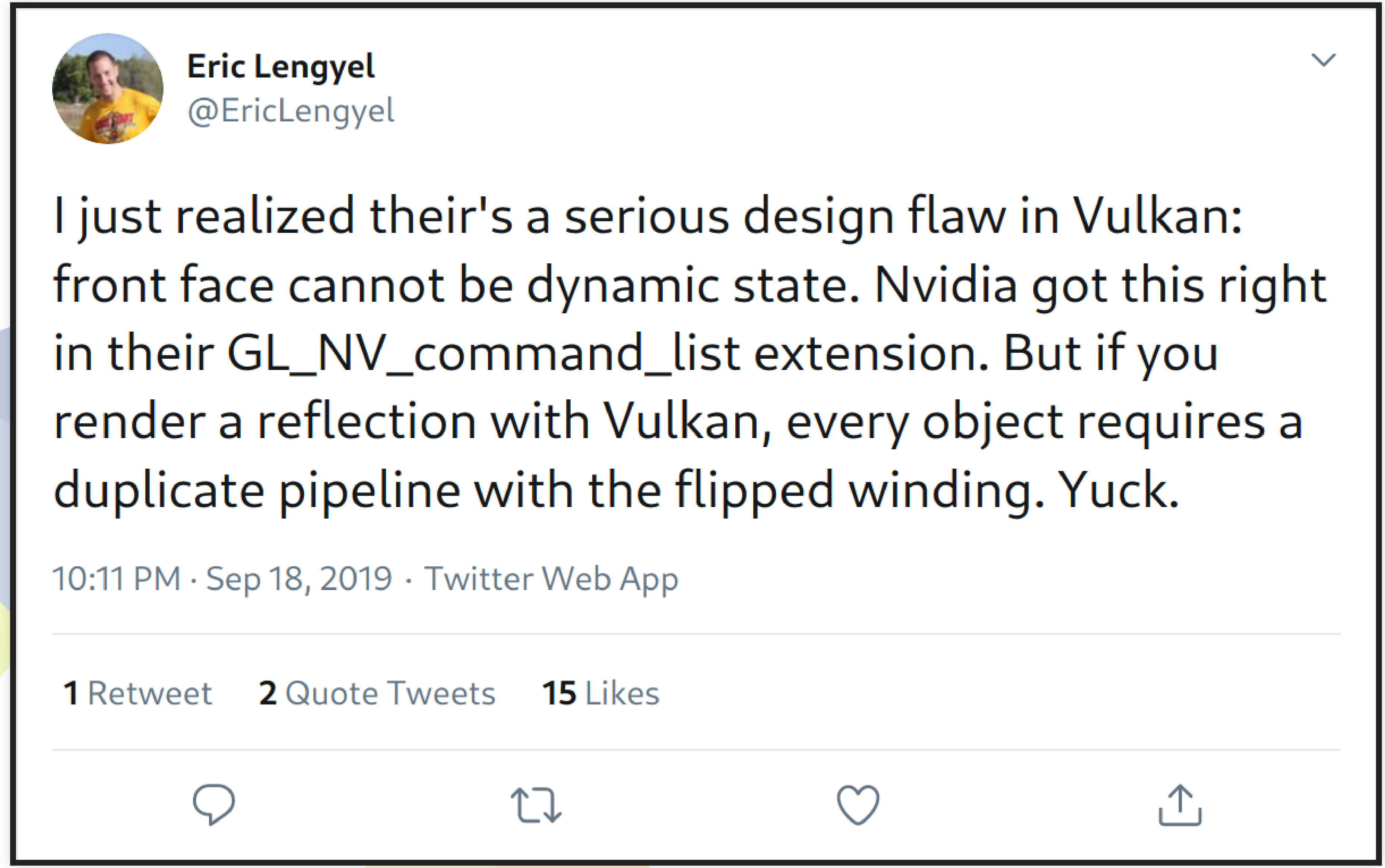


Diagram by Eric Lengyel released under the GNU FDL
`https://commons.wikimedia.org/wiki/File:The_OpenGL_-_DirectX_graphics_pipeline.png`

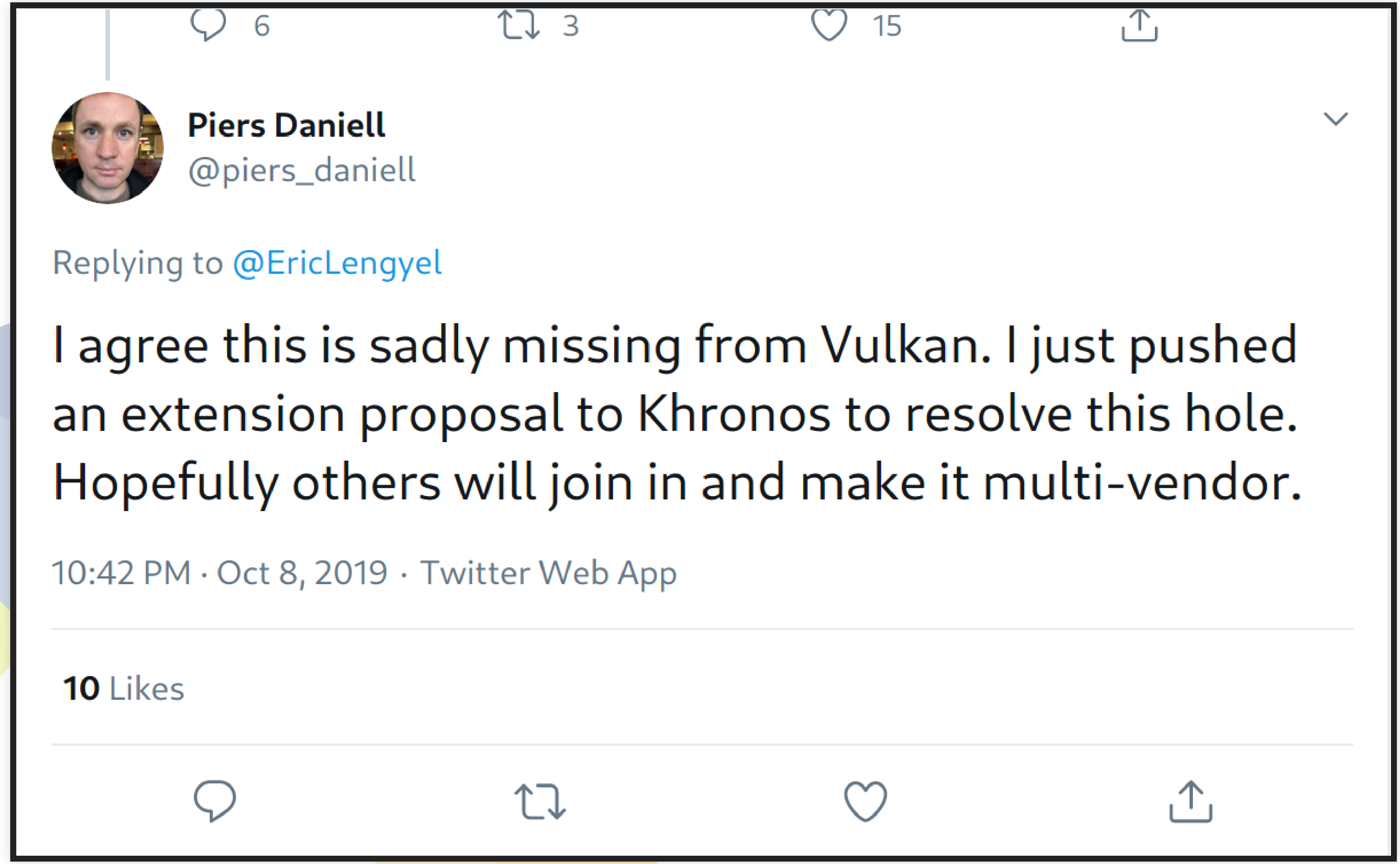


Eric Lengyel
@EricLengyel

I just realized their's a serious design flaw in Vulkan: front face cannot be dynamic state. Nvidia got this right in their GL_NV_command_list extension. But if you render a reflection with Vulkan, every object requires a duplicate pipeline with the flipped winding. Yuck.

10:11 PM · Sep 18, 2019 · Twitter Web App

1 Retweet **2** Quote Tweets **15** Likes

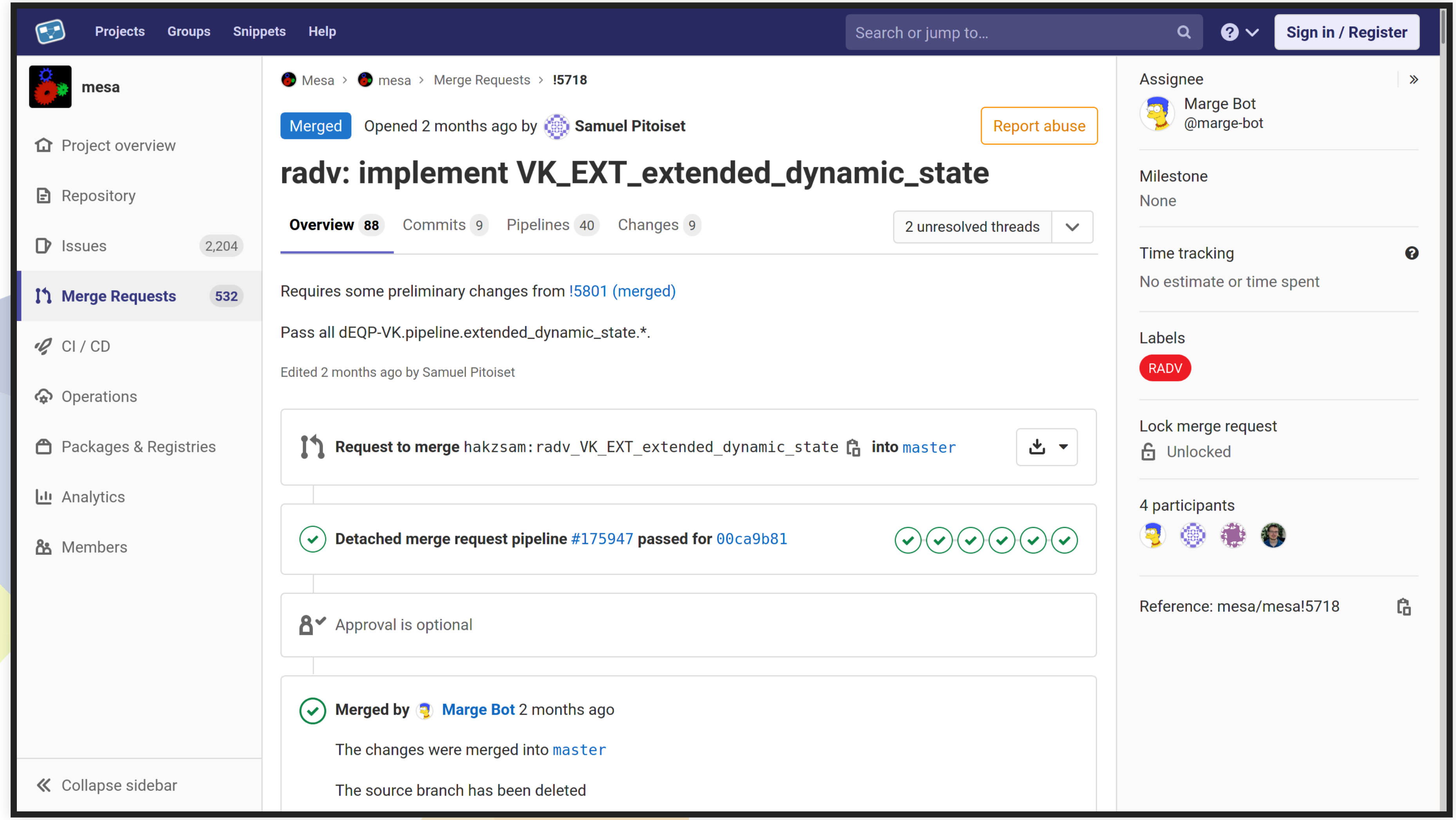


Piers Daniell
@piers_daniell

Replying to @EricLengyel

I agree this is sadly missing from Vulkan. I just pushed an extension proposal to Khronos to resolve this hole. Hopefully others will join in and make it multi-vendor.

10:42 PM · Oct 8, 2019 · Twitter Web App



Merged Opened 2 months ago by Samuel Pitoiset

radv: implement VK_EXT_extended_dynamic_state

Requires some preliminary changes from !5801 (merged)

Pass all dEQP-VK.pipeline.extended_dynamic_state.*.

Edited 2 months ago by Samuel Pitoiset

Request to merge `hakzsam:radv_VK_EXT_extended_dynamic_state` **into** `master`

Detached merge request pipeline #175947 **passed for** `00ca9b81`

Approval is optional

Merged by Marge Bot 2 months ago

The changes were merged into `master`

The source branch has been deleted

Assignee
Marge Bot
@marge-bot

Milestone
None

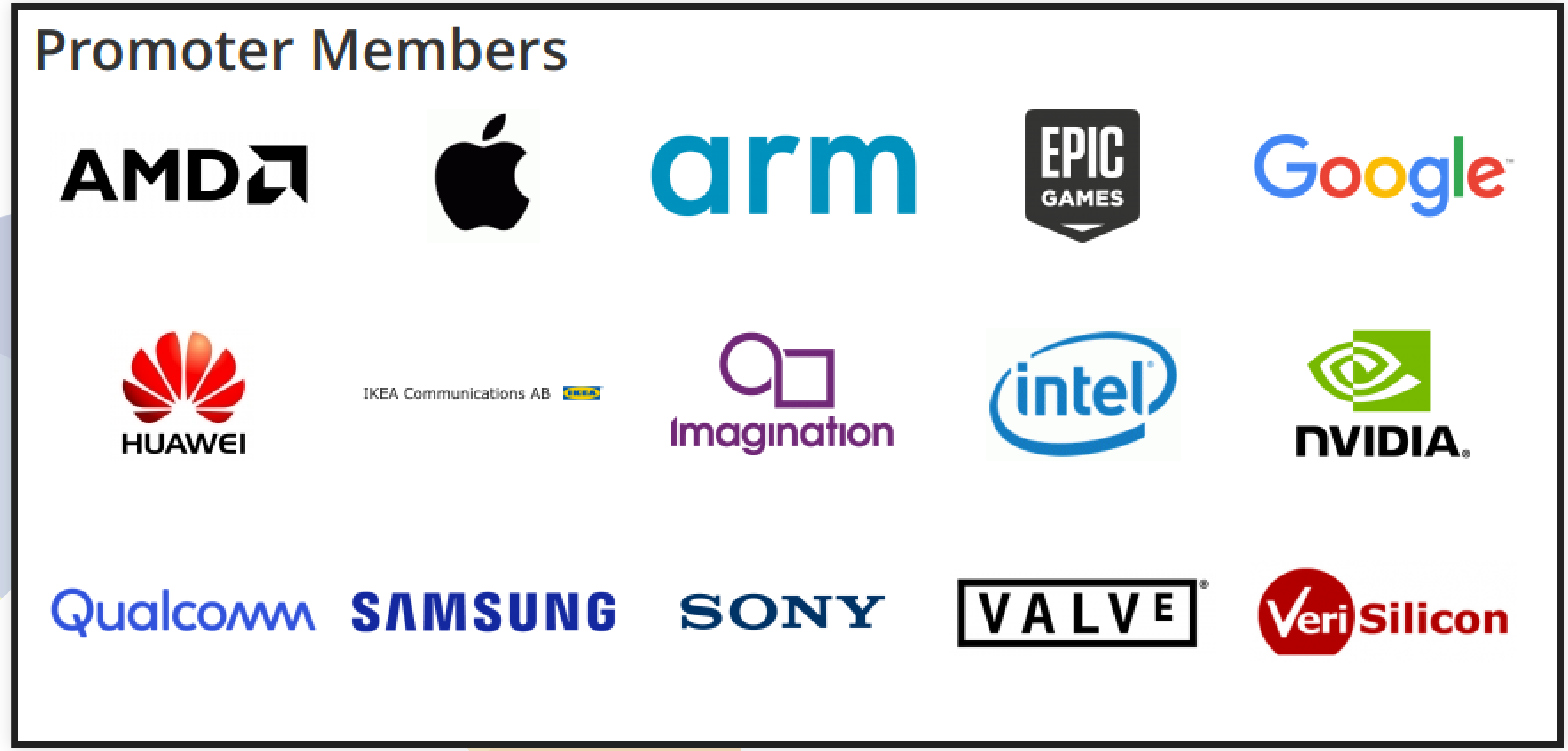
Time tracking
No estimate or time spent

Labels
RADV

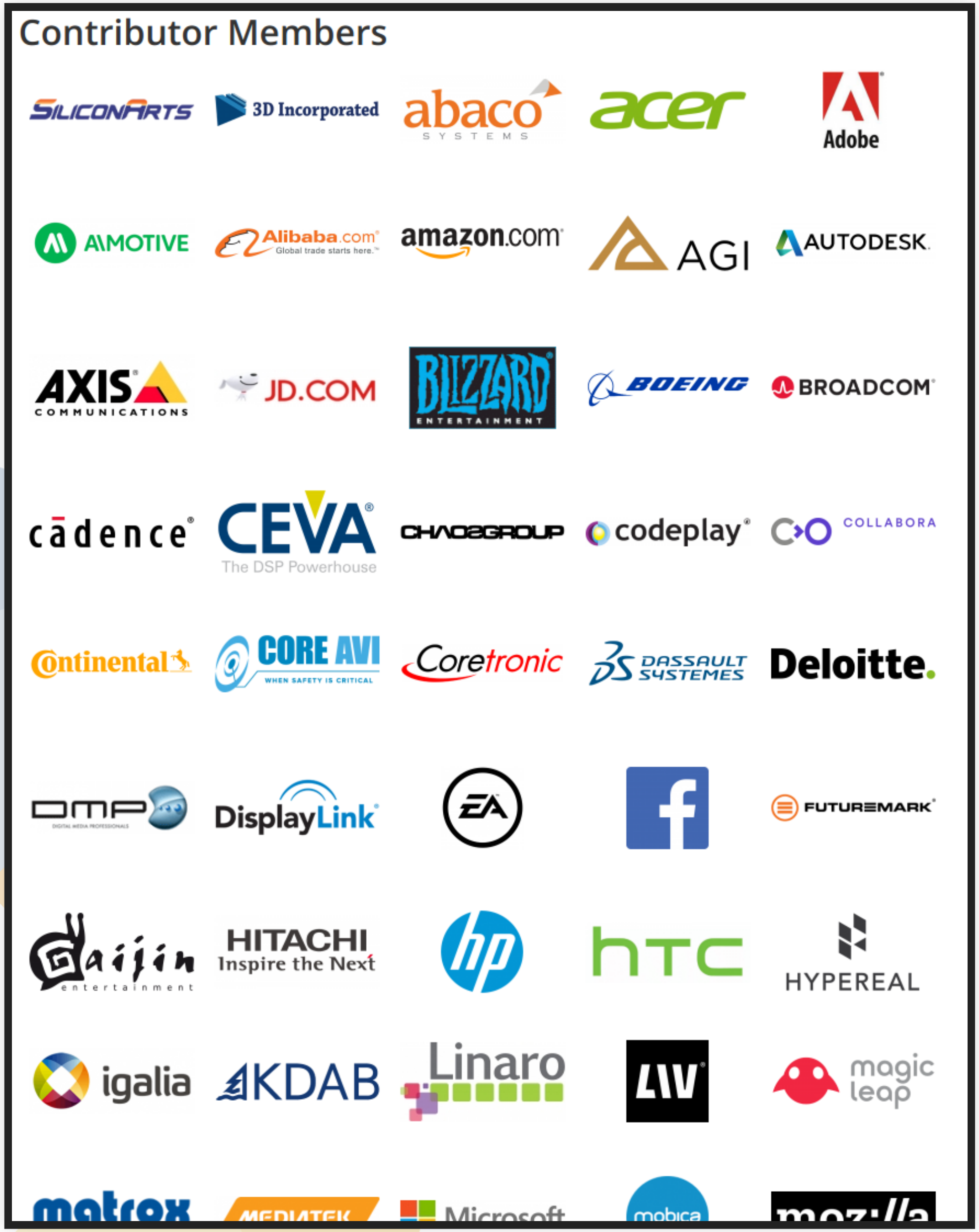
Lock merge request
Unlocked

4 participants

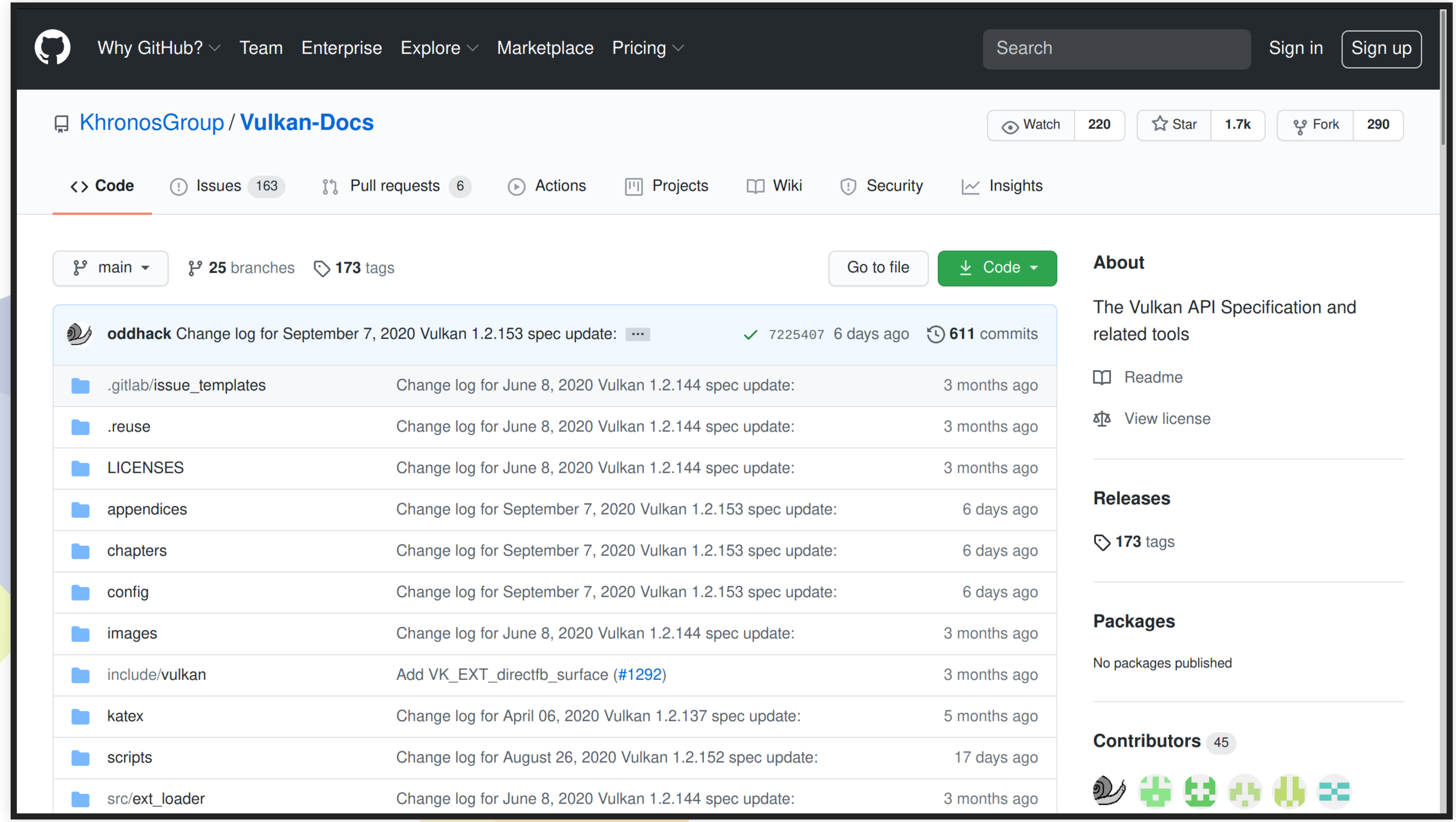
Reference: mesa/mesa!5718



Promoter Members



Contributor Members



KhronosGroup / **Vulkan-Docs**

oddhack Change log for September 7, 2020 Vulkan 1.2.153 spec update: ··· ✓ 7225407 6 days ago **611** commits

.gitlab/issue_templates	Change log for June 8, 2020 Vulkan 1.2.144 spec update:	3 months ago
.reuse	Change log for June 8, 2020 Vulkan 1.2.144 spec update:	3 months ago
LICENSES	Change log for June 8, 2020 Vulkan 1.2.144 spec update:	3 months ago
appendices	Change log for September 7, 2020 Vulkan 1.2.153 spec update:	6 days ago
chapters	Change log for September 7, 2020 Vulkan 1.2.153 spec update:	6 days ago
config	Change log for September 7, 2020 Vulkan 1.2.153 spec update:	6 days ago
images	Change log for June 8, 2020 Vulkan 1.2.144 spec update:	3 months ago
include/vulkan	Add VK_EXT_directfb_surface (#1292)	3 months ago
katex	Change log for April 06, 2020 Vulkan 1.2.137 spec update:	5 months ago
scripts	Change log for August 26, 2020 Vulkan 1.2.152 spec update:	17 days ago
src/ext_loader	Change log for June 8, 2020 Vulkan 1.2.144 spec update:	3 months ago

About

The Vulkan API Specification and related tools

Readme

View license

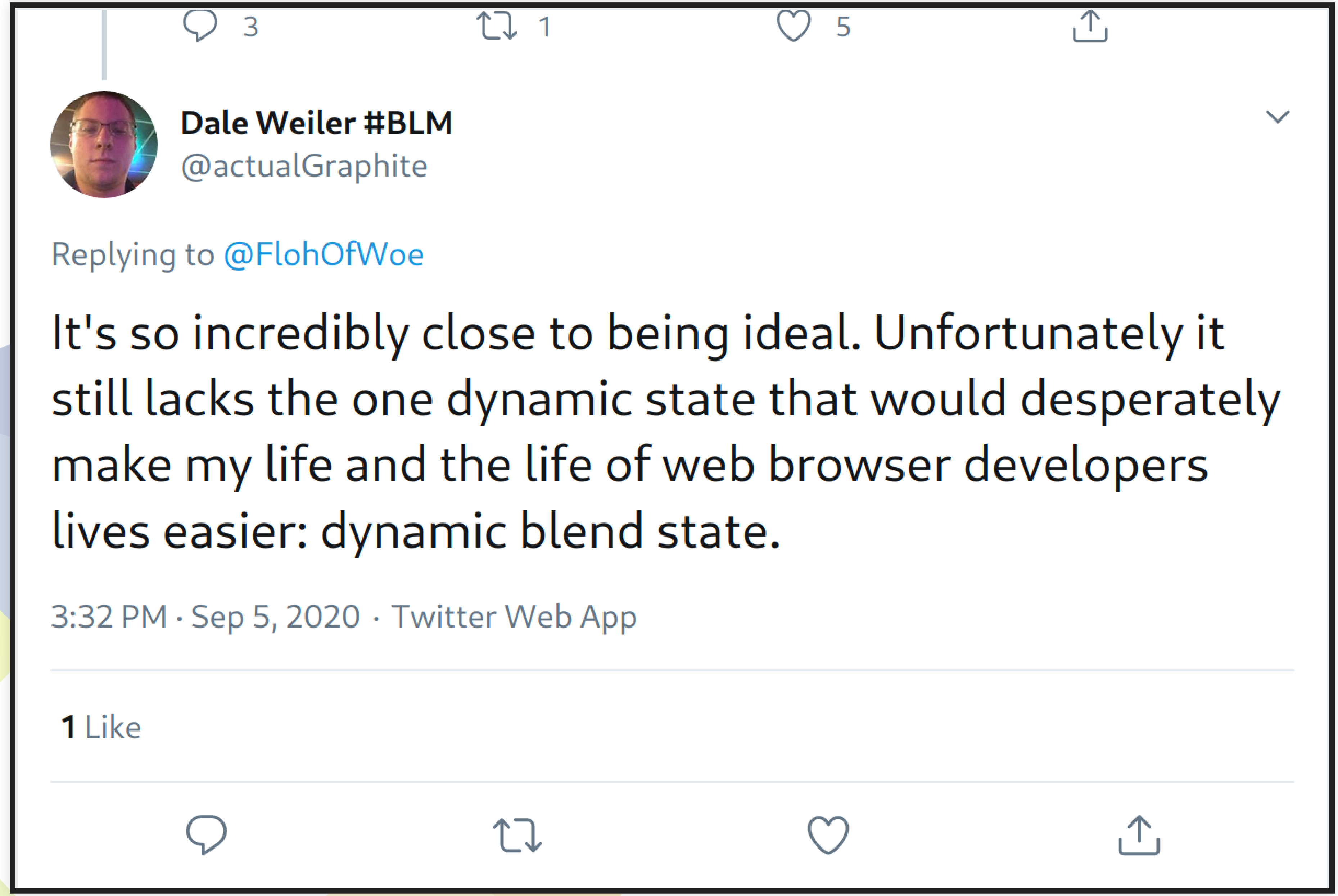
Releases

173 tags

Packages

No packages published

Contributors 45



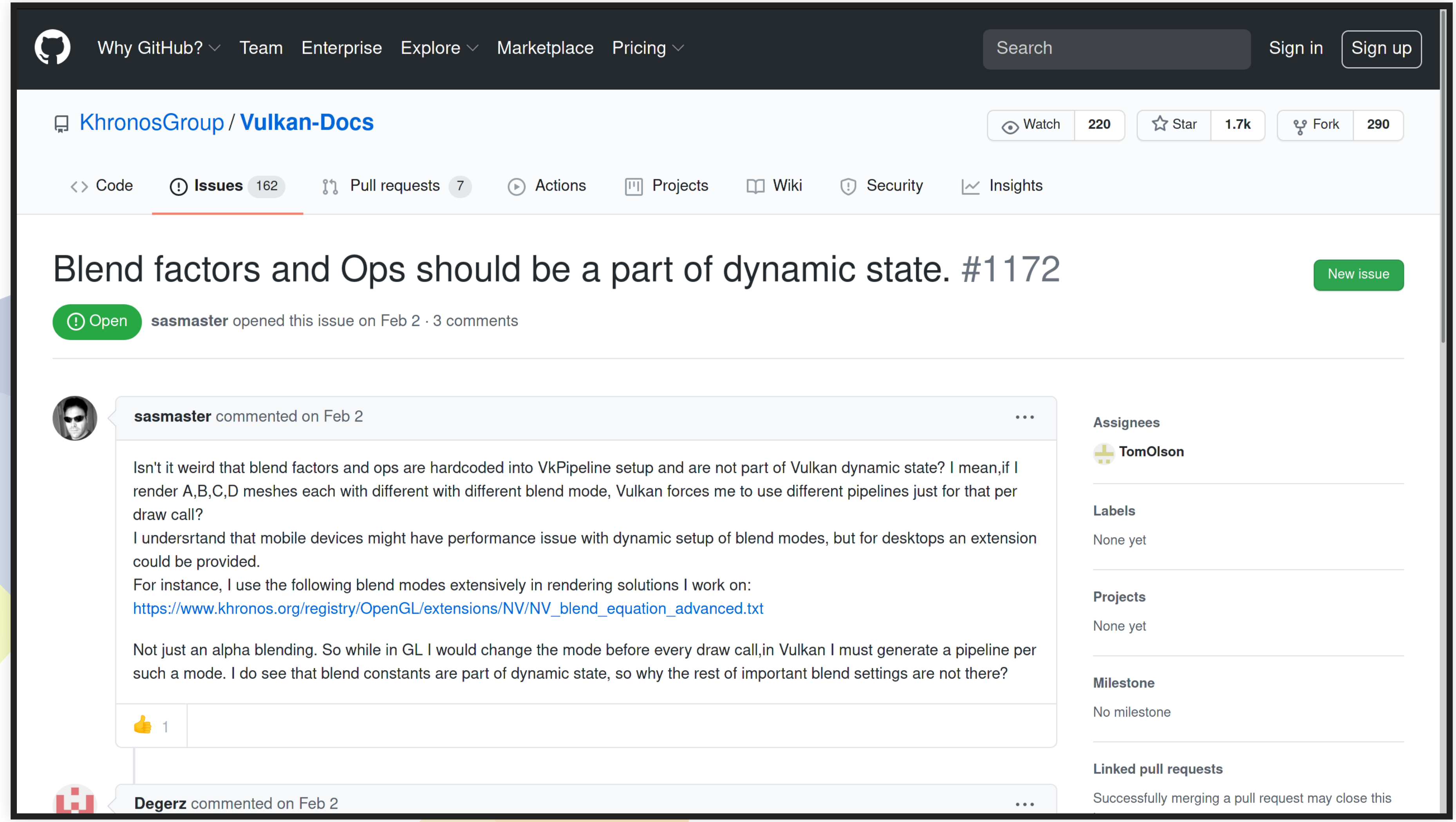
Dale Weiler #BLM
@actualGraphite

Replying to @FlohOfWoe

It's so incredibly close to being ideal. Unfortunately it still lacks the one dynamic state that would desperately make my life and the life of web browser developers lives easier: dynamic blend state.

3:32 PM · Sep 5, 2020 · Twitter Web App

1 Like



Blend factors and Ops should be a part of dynamic state. #1172

Open **sasmaster** opened this issue on Feb 2 · 3 comments

sasmaster commented on Feb 2

Isn't it weird that blend factors and ops are hardcoded into VkPipeline setup and are not part of Vulkan dynamic state? I mean,if I render A,B,C,D meshes each with different with different blend mode, Vulkan forces me to use different pipelines just for that per draw call?
I undersrtand that mobile devices might have performance issue with dynamic setup of blend modes, but for desktops an extension could be provided.
For instance, I use the following blend modes extensively in rendering solutions I work on:
https://www.khronos.org/registry/OpenGL/extensions/NV/NV_blend_equation_advanced.txt

Not just an alpha blending. So while in GL I would change the mode before every draw call,in Vulkan I must generate a pipeline per such a mode. I do see that blend constants are part of dynamic state, so why the rest of important blend settings are not there?

👍 1

Degerz commented on Feb 2

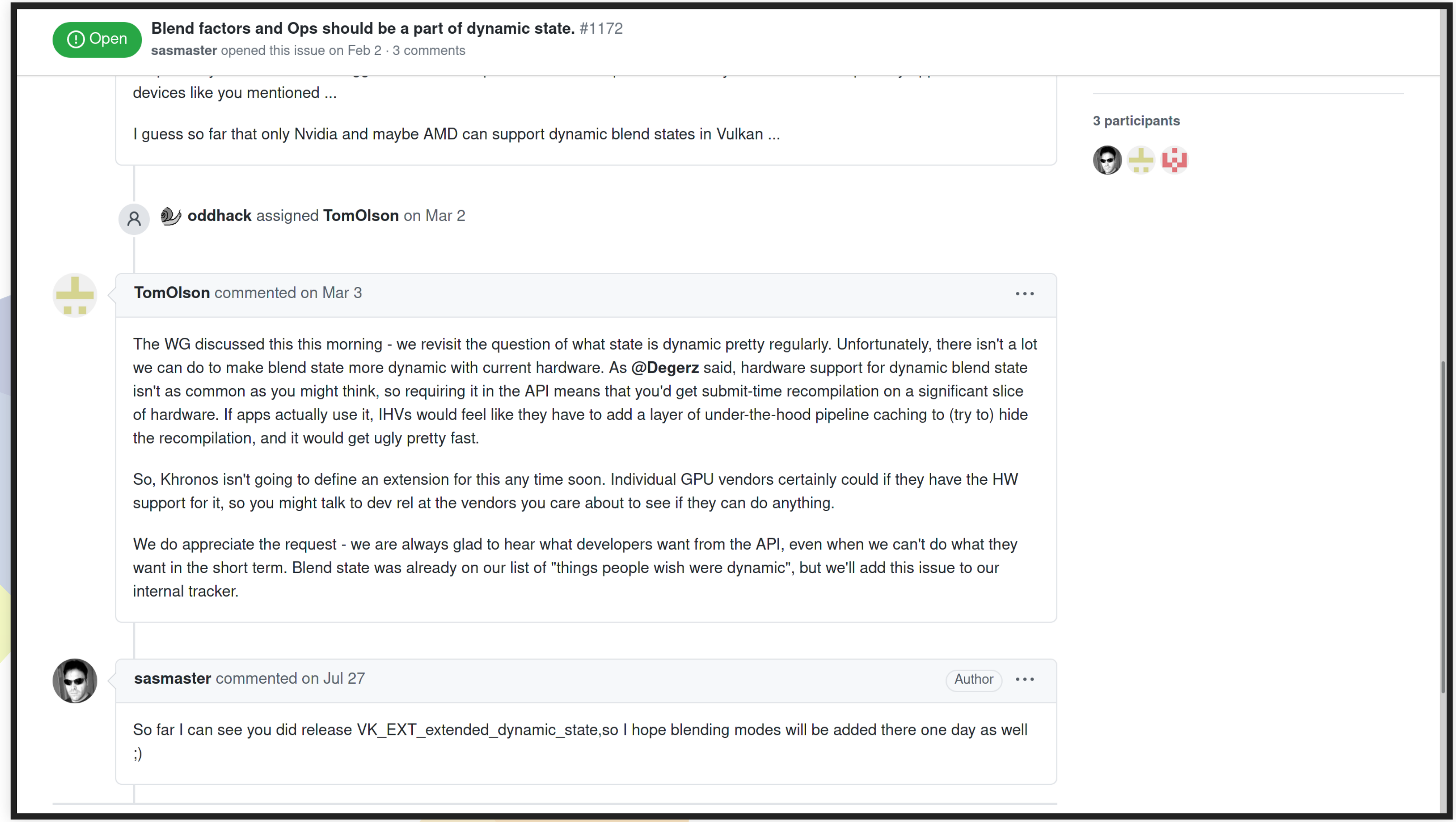
Assignees
TomOlson

Labels
None yet

Projects
None yet

Milestone
No milestone

Linked pull requests
Successfully merging a pull request may close this



Blend factors and Ops should be a part of dynamic state. #1172

sasmaster opened this issue on Feb 2 · 3 comments

devices like you mentioned ...

I guess so far that only Nvidia and maybe AMD can support dynamic blend states in Vulkan ...

oddhack assigned **TomOlson** on Mar 2

TomOlson commented on Mar 3

The WG discussed this this morning - we revisit the question of what state is dynamic pretty regularly. Unfortunately, there isn't a lot we can do to make blend state more dynamic with current hardware. As **@Degerz** said, hardware support for dynamic blend state isn't as common as you might think, so requiring it in the API means that you'd get submit-time recompilation on a significant slice of hardware. If apps actually use it, IHVs would feel like they have to add a layer of under-the-hood pipeline caching to (try to) hide the recompilation, and it would get ugly pretty fast.

So, Khronos isn't going to define an extension for this any time soon. Individual GPU vendors certainly could if they have the HW support for it, so you might talk to dev rel at the vendors you care about to see if they can do anything.

We do appreciate the request - we are always glad to hear what developers want from the API, even when we can't do what they want in the short term. Blend state was already on our list of "things people wish were dynamic", but we'll add this issue to our internal tracker.

sasmaster commented on Jul 27

So far I can see you did release VK_EXT_extended_dynamic_state,so I hope blending modes will be added there one day as well ;)

THANKS!
QUESTIONS?