Creating a Pipeline with Dynamically Changeable State Variables

The graphics pipeline data structure is full of state information, and as previously discussed, is largely immutable, that is, the information contained inside it is fixed and can only be changed by creating a new graphics pipeline data structure with new information.

That isn't quite true. To a certain extent, Vulkan allows you to declare parts of the pipeline state changeable. This allows you to alter pipeline state information on the fly.

This is useful for managing state information that needs to change frequently. This also creates possible optimization opportunities for the Vulkan driver.

Creating a Pipeline

Which Pipeline State Variables can be Changed Dynamically

The possible dynamic variables are shown in the VkDynamicState enum:

- VK_DYNAMIC_STATE_VIEWPORT
- VK_DYNAMIC_STATE_SCISSOR
- VK_DYNAMIC_STATE_LINE_WIDTH
- VK_DYNAMIC_STATE_DEPTH_BIAS
- VK_DYNAMIC_STATE_BLEND_CONSTANTS
- VK_DYNAMIC_STATE_DEPTH_BOUNDS
- VK_DYNAMIC_STATE_STENCIL_COMPARE_MASK
- VK_DYNAMIC_STATE_STENCIL_WRITE_MASK
- VK_DYNAMIC_STATE_STENCIL_REFERENCE

Creating a Pipeline

Filling the Dynamic State Variables in the Command Buffer

First call:

```
vkCmdBindPipeline( … );
```

Then, the command buffer bound function calls to set these dynamic states are:

- `vkCmdSetViewport(commandBuffer, firstViewport, viewportCount, pViewports);`
- `vkCmdSetScissor(commandBuffer, firstScissor, scissorCount, pScissors);`
- `vkCmdSetLineWidth(commandBuffer, linewidth);`
- `vkCmdSetDepthBias(commandBuffer, depthBiasConstantFactor, depthBiasClamp, depthBiasSlopeFactor);`
- `vkCmdSetBlendConstants(commandBuffer, blendConstants[4]);`
- `vkCmdSetDepthBounds(commandBuffer, minDepthBounds, maxDepthBounds);`
- `vkCmdSetStencilCompareMask(commandBuffer, faceMask, compareMask);`
- `vkCmdSetStencilWriteMask(commandBuffer, faceMask, writeMask);`
- `vkCmdSetStencilReference(commandBuffer, faceMask, reference);`
This is from one of the Vulkan .h Files

Does this mean more Dynamic States are in the Works?

```
VK_DYNAMIC_STATE_VIEWPORT = 0,
VK_DYNAMIC_STATE_SCISSOR = 1,
VK_DYNAMIC_STATE_LINE_WIDTH = 2,
VK_DYNAMIC_STATE_BLEND_CONSTANTS = 4,
VK_DYNAMIC_STATE_DEPTH_BIAS = 3,
VK_DYNAMIC_STATE_DEPTH_BOUNDS = 5,
VK_DYNAMIC_STATE_STENCIL_COMPARE_MASK = 6,
VK_DYNAMIC_STATE_STENCIL_WRITE_MASK = 7,
VK_DYNAMIC_STATE_STENCIL_REFERENCE = 8,
VK_DYNAMIC_STATE_CULL_MODE = 1000267000,
VK_DYNAMIC_STATE_FRONT_FACE = 1000267001,
VK_DYNAMIC_STATE_PRIMITIVE_TOPOLOGY = 1000267002,
VK_DYNAMIC_STATE_VIEWPORT_WITH_COUNT = 1000267003,
VK_DYNAMIC_STATE_SCISSOR_WITH_COUNT = 1000267004,
VK_DYNAMIC_STATE_VERTEX_INPUT_BINDING_STRIDE = 1000267005,
VK_DYNAMIC_STATE_DEPTH_TEST_ENABLE = 1000267006,
VK_DYNAMIC_STATE_DEPTH_WRITE_ENABLE = 1000267007,
VK_DYNAMIC_STATE_DEPTH_COMPARE_OP = 1000267008,
VK_DYNAMIC_STATE_DEPTH_BOUNDS_TEST_ENABLE = 1000267009,
VK_DYNAMIC_STATE_STENCIL_TEST_ENABLE = 1000267010,
VK_DYNAMIC_STATE_STENCIL_OP = 1000267011.
```