Logical Devices

Mike Bailey
mjb@cs.oregonstate.edu

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License
Vulkan: a More Typical (and Simplified) Block Diagram

- Application
  - Instance
    - Physical Device
      - Logical Device
        - Queue
          - Command Buffer
          - Command Buffer
          - Command Buffer
Looking to See What Device Layers are Available

```c
const char * myDeviceLayers[ ] =
{
    // "VK_LAYER_LUNARG_api_dump",
    // "VK_LAYER_LUNARG_core_validation",
    // "VK_LAYER_LUNARG_image",
    "VK_LAYER_LUNARG_object_tracker",
    "VK_LAYER_LUNARG_parameter_validation",
    // "VK_LAYER_NV_optimus"
};

const char * myDeviceExtensions[ ] =
{
    "VK_KHR_surface",
    "VK_KHR_win32_surface",
    "VK_EXT_debug_report"
    // "VK_KHR_swapchains"
};

// see what device layers are available:

uint32_t layerCount;
vkEnumerateDeviceLayerProperties(PhysicalDevice, &layerCount, (VkLayerProperties *)nullptr);

VkLayerProperties * deviceLayers = new VkLayerProperties[layerCount];

result = vkEnumerateDeviceLayerProperties( PhysicalDevice, &layerCount, deviceLayers);
```
Looking to See What Device Extensions are Available

```c
// see what device extensions are available:

uint32_t extensionCount;
vkEnumerateDeviceExtensionProperties(PhysicalDevice, deviceLayers[i].layerName,
    &extensionCount, (VkExtensionProperties *)nullptr);

VkExtensionProperties * deviceExtensions = new VkExtensionProperties[extensionCount];

result = vkEnumerateDeviceExtensionProperties(PhysicalDevice, deviceLayers[i].layerName,
    &extensionCount, deviceExtensions);
```
What Device Layers and Extensions are Available

4 physical device layers enumerated:

0x00401063  1 'VK_LAYER_NV_optimus' 'NVIDIA Optimus layer'
            0 device extensions enumerated for 'VK_LAYER_NV_optimus':

0x00401072  1 'VK_LAYER_LUNARG_core_validation' 'LunarG Validation Layer'
            2 device extensions enumerated for 'VK_LAYER_LUNARG_core_validation':
                0x00000001 'VK_EXT_validation_cache'
                0x00000004 'VK_EXT_debug_marker'

0x00401072  1 'VK_LAYER_LUNARG_object_tracker' 'LunarG Validation Layer'
            2 device extensions enumerated for 'VK_LAYER_LUNARG_object_tracker':
                0x00000001 'VK_EXT_validation_cache'
                0x00000004 'VK_EXT_debug_marker'

0x00401072  1 'VK_LAYER_LUNARG_parameter_validation' 'LunarG Validation Layer'
            2 device extensions enumerated for 'VK_LAYER_LUNARG_parameter_validation':
                0x00000001 'VK_EXT_validation_cache'
                0x00000004 'VK_EXT_debug_marker'
float queuePriorities[1] =
{
    1.
};
VkDeviceQueueCreateInfo vdqci;
    vdqci.sType = VK_STRUCTURE_TYPE_DEVICE_QUEUE_CREATE_INFO;
    vdqci.pNext = nullptr;
    vdqci.flags = 0;
    vdqci.queueFamilyIndex = 0;
    vdqci.queueCount = 1;
    vdqci.pQueueProperties = queuePriorities;

result = vkCreateLogicalDevice( PhysicalDevice, IN &vdci, PALLOCATOR, OUT &LogicalDevice );
// get the queue for this logical device:

vkGetDeviceQueue( LogicalDevice, 0, 0, OUT &Queue ); // 0, 0 = queueFamilyIndex, queueIndex