



Logical Devices

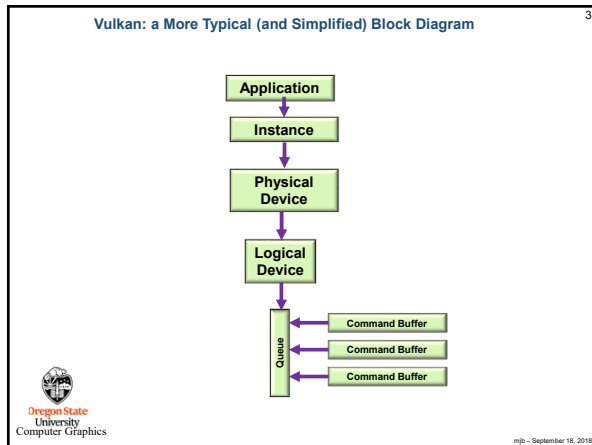
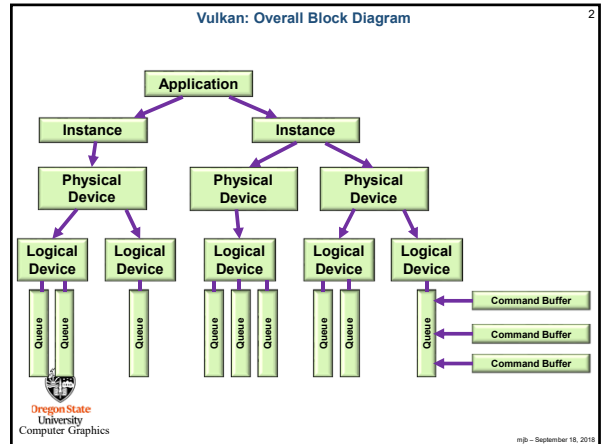


Oregon State University
Mike Bailey
mb@cs.oregonstate.edu

This work is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nc-nd/4.0/).



LogicalDevices.pdf mjb - September 18, 2018



Looking to See What Device Layers are Available

```

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);
  
```

Oregon State University Computer Graphics mjb - September 18, 2018

Looking to See What Device Extensions are Available

```

// see what device extensions are available:
uint32_t extensionCount;
vkEnumerateDeviceExtensionProperties(PhysicalDevice, deviceLayers[0].layerName,
&extensionCount, (VkExtensionProperties *)nullptr);
VkExtensionProperties * deviceExtensions = new VkExtensionProperties[extensionCount];
result = vkEnumerateDeviceExtensionProperties(PhysicalDevice, deviceLayers[0].layerName,
&extensionCount, deviceExtensions);
  
```

Oregon State University Computer Graphics mjb - September 18, 2018

What Device Layers and Extensions are Available

```

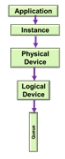
3 physical device layers enumerated:
0x00400039 1 "VK_LAYER_NV_optimus" "NVIDIA Optimus layer"
0 device extensions enumerated for "VK_LAYER_NV_optimus":

0x00400033 1 "VK_LAYER_LUNARG_object_tracker" "LunarG Validation Layer"
0 device extensions enumerated for "VK_LAYER_LUNARG_object_tracker":

0x00400033 1 "VK_LAYER_LUNARG_parameter_validation" "LunarG Validation Layer"
0 device extensions enumerated for "VK_LAYER_LUNARG_parameter_validation":
  
```


Oregon State University Computer Graphics mjb - September 18, 2018

Vulkan: Specifying a Logical Device Queue



```
float queuePriorities[] =
{
    1.
};

VkDeviceQueueCreateInfo vddqi;
vddqi.sType = VK_STRUCTURE_TYPE_DEVICE_QUEUE_CREATE_INFO;
vddqi.pNext = nullptr;
vddqi.flags = 0;
vddqi.queueFamilyIndex = 0;
vddqi.queueCount = 1;
vddqi.pQueueProperties = queuePriorities;
```




Oregon State University
Computer Graphics

mjb - September 18, 2018

Vulkan: Creating a Logical Device

```
VkDeviceCreateInfo vdc;
vdc.sType = VK_STRUCTURE_TYPE_DEVICE_CREATE_INFO;
vdc.pNext = nullptr;
vdc.flags = 0;
vdc.queueCreateInfoCount = 1; // # of device queues
vdc.pQueueCreateInfos = IN vddqi; // array of VkDeviceQueueCreateInfo's
vdc.enabledLayerCount = sizeof(myDeviceLayers) / sizeof(char *);
vdc.enabledLayerCount = 0;
vdc.ppEnabledLayerNames = myDeviceLayers;
vdc.enabledExtensionCount = 0;
vdc.ppEnabledExtensionNames = (const char **)nullptr; // no extensions
vdc.enabledExtensionCount = sizeof(myDeviceExtensions) / sizeof(char *);
vdc.ppEnabledExtensionNames = myDeviceExtensions;
vdc.pEnabledFeatures = IN &PhysicalDeviceFeatures;
```

```
result = vkCreateLogicalDevice( PhysicalDevice, IN &vdc, PALLOCATOR, OUT &LogicalDevice );
```




Oregon State University
Computer Graphics

mjb - September 18, 2018

Vulkan: Creating the Logical Device's Queue

```
// get the queue for this logical device:
vkGetDeviceQueue( LogicalDevice, 0, 0, OUT &Queue ); // 0, 0 = queueFamilyIndex, queueIndex
```



Oregon State University
Computer Graphics

mjb - September 18, 2018