Logical Devices

Application

Instance

Instance

Physical Device

Physical Device

Physical Device

Logical Device

Logical Device

Logical Device

Logical Device

Queue

Queue

Queue

Queue

Queue

Queue

Queue

Queue

Command Buffer

Command Buffer

Command Buffer

Command Buffer
Vulkan: a More Typical (and Simplified) Block Diagram

- **Application**
- **Instance**
- **Physical Device**
- **Logical Device**
- **Queue**
- **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
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

3 physical device layers enumerated:

0x00400038  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':
float queuePriorities[1] =
{
    1,
};

VkDeviceQueueCreateInfo vdqci =
{
    .sType = VK_STRUCTURE_TYPE_DEVICE_QUEUE_CREATE_INFO,
    .pNext = nullptr,
    .flags = 0,
    .queueFamilyIndex = 0,
    .queueCount = 1,
    .pQueueProperties = queuePriorities,
};

VkDeviceCreateInfo vdci =
{
    .sType = VK_STRUCTURE_TYPE_DEVICE_CREATE_INFO,
    .pNext = nullptr,
    .flags = 0,
    .queueCreateInfoCount = 1,  // # of device queues
    .pQueueCreateInfos = &vdqci,  // array of VkDeviceQueueCreateInfo's
    .enabledLayerCount = sizeof(myDeviceLayers) / sizeof(const char*),
    .ppEnabledLayerNames = myDeviceLayers,
    .enabledExtensionCount = 0,
    .ppEnabledExtensionNames = const_cast<const char**>(nullptr),  // no extensions
    .pEnabledFeatures = &PhysicalDeviceFeatures,
};

result = vkCreateLogicalDevice( PhysicalDevice, &vdci, PALLOCATOR, &LogicalDevice );
// get the queue for this logical device:
vkGetDeviceQueue( LogicalDevice, 0, 0, OUT &Queue );           // 0, 0 = queueFamilyIndex, queueIndex