Logical Devices

Vulkan: Overall Block Diagram

Vulkan: a More Typical (and Simplified) Block Diagram

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
};

// 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

```c
3 physical device layers enumerated:
0x00400038   1  'VK_LAYER_NV_optimus'  'NVIDIA Optimus layer'
0x00400033   1  'VK_LAYER_LUNARG_object_tracker'  'LunarG Validation Layer'
```

Vulkan: Specifying a Logical Device Queue

```c
float queuePriorities[1] =
{ 1.);
```

Vulkan: Creating a Logical Device

```c
VkDeviceCreateInfo vdci;
vkCreateLogicalDevice( PhysicalDevice, &vdci, PALLOCATOR, &LogicalDevice);
```
Vulkan: Creating the Logical Device’s Queue

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