Plan of implemnetation

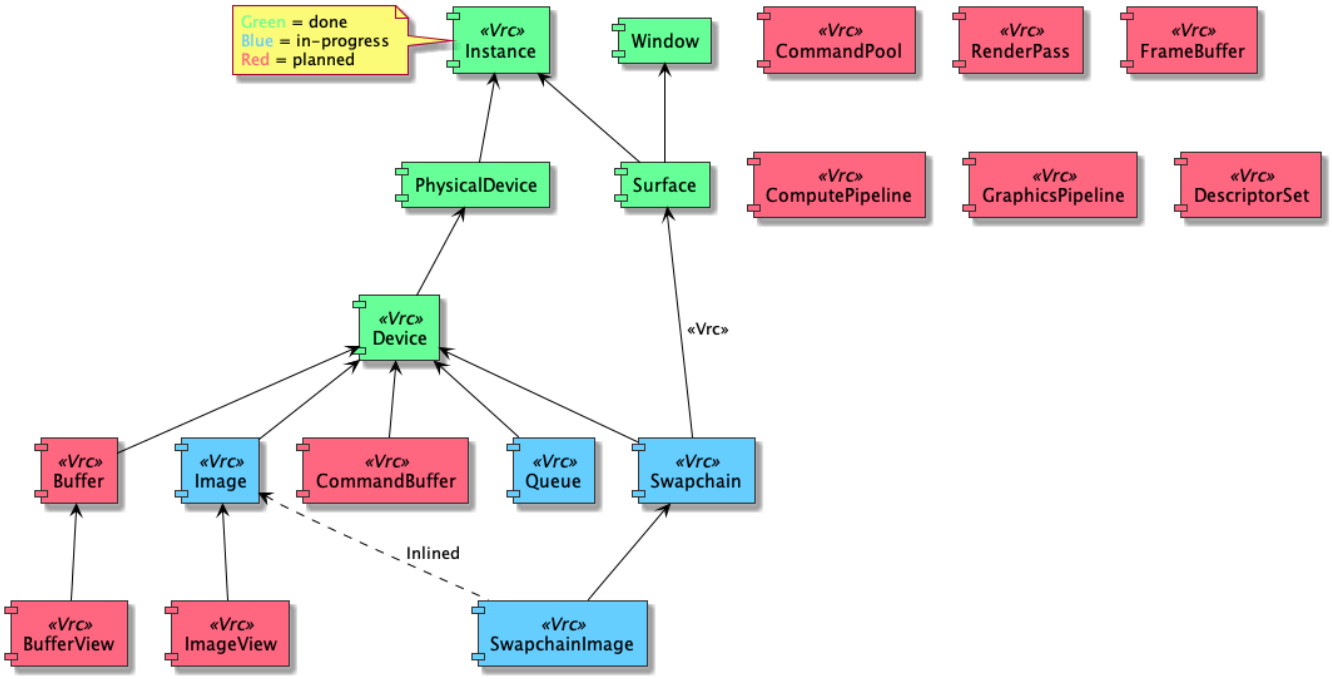

Figure 1: Object Dependency Graph

Validation

Validation of correct usage as dictated by the Vulkan specification.

Instance

Validations for `vkCreateInstance` and `VkInstanceCreateInfo`:

- ~~All required extensions for each extension in the `VkInstanceCreateInfo::ppEnabledExtensionNames` list must also be present in that list.~~
  - Left to validation layers

Device

Validations for `vkCreateDevice` and `VkDeviceCreateInfo`:

- ~~All required extensions for each extension in the `VkDeviceCreateInfo::ppEnabledExtensionNames` list must also be present in that list.~~
  - Left to validation layers
- ~~The queueFamilyIndex member of each element of `pQueueCreateInfos` must be unique within `pQueueCreateInfos`, except that two members can share the same queueFamilyIndex if one is a protected-capable queue and one is not a protected-capable queue.~~
  - Left to validation layers
- ~~If the `pNext` chain includes a `VkPhysicalDeviceFeatures2` [...]~~
  - `pNext` chain is not exposed through the API
- `ppEnabledExtensionNames` must not contain `VK_AMD_negative_viewport_height`
- `ppEnabledExtensionNames` must not contain both `VK_KHR_buffer_device_address` and

VK_EXT_buffer_device_address
- ~~If the pNext chain includes a VkPhysicalDeviceVulkan11Features [...]~~
  - pNext chain is not exposed through the API
- ~~If the pNext chain includes a VkPhysicalDeviceVulkan11Features [...]~~
  - pNext chain is not exposed through the API
- ~~If ppEnabledExtensions contains VK_KHR_draw_indirect_count and the pNext chain includes a VkPhysicalDeviceVulkan12Features structure, then VkPhysicalDeviceVulkan12Features::drawIndirectCount must be VK_TRUE~~
  - pNext chain is not exposed through the API
- ~~If ppEnabledExtensions contains VK_KHR_sampler_mirror_clamp_to_edge and the pNext chain includes a VkPhysicalDeviceVulkan12Features structure, then VkPhysicalDeviceVulkan12Features::samplerMirrorClampToEdge must be VK_TRUE~~
  - pNext chain is not exposed through the API
- ~~If ppEnabledExtensions contains VK_EXT_descriptor_indexing and the pNext chain includes a VkPhysicalDeviceVulkan12Features structure, then VkPhysicalDeviceVulkan12Features::descriptorIndexing must be VK_TRUE~~
  - pNext chain is not exposed through the API
- ~~If ppEnabledExtensions contains VK_EXT_sampler_filter_minmax and the pNext chain includes a VkPhysicalDeviceVulkan12Features structure, then VkPhysicalDeviceVulkan12Features::samplerFilterMinmax must be VK_TRUE~~
  - pNext chain is not exposed through the API
- ~~If ppEnabledExtensions contains VK_EXT_shader_viewport_index_layer and the pNext chain includes a VkPhysicalDeviceVulkan12Features structure, then VkPhysicalDeviceVulkan12Features::shaderOutputViewportIndex and VkPhysicalDeviceVulkan12Features::shaderOutputLayer must both be VK_TRUE~~
  - pNext chain is not exposed through the API

Queue

Validations for VkDeviceQueueCreateInfo:

- queueFamilyIndex must be less than pQueueFamilyPropertyCount returned by vkGetPhysicalDeviceQueueFamilyProperties
- queueCount must be less than or equal to the queueCount member of the VkQueueFamilyProperties structure, as returned by vkGetPhysicalDeviceQueueFamilyProperties in the pQueueFamilyProperties[queueFamilyIndex]
- Each element of pQueuePriorities must be between 0.0 and 1.0 inclusive
- ~~If the protected memory feature is not enabled, the VK_DEVICE_QUEUE_CREATE_PROTECTED_BIT bit of flags must not be set.~~
  - Flags are not exposed through the API

Swapchain

Validations for VkSwapchainCreateInfoKHR:

- surface must be a surface that is supported by the device as determined using vkGetPhysicalDeviceSurfaceSupportKHR
- ~~minImageCount must be less than or equal to the value returned in the maxImageCount member of the VkSurfaceCapabilitiesKHR structure returned by vkGetPhysicalDeviceSurfaceCapabilitiesKHR for the surface if the returned maxImageCount is not zero~~
  - Done
- ~~If presentMode is not VK_PRESENT_MODE_SHARED_DEMAND_REFRESH_KHR nor~~

~~VK_PRESENT_MODE_SHARED_CONTINUOUS_REFRESH_KHR~~, ~~then~~ ~~minImageCount~~ ~~must be greater than or equal to the value returned in the~~ ~~minImageCount~~ ~~member of the~~ ~~VkSurfaceCapabilitiesKHR~~ ~~structure returned by~~ ~~vkGetPhysicalDeviceSurfaceCapabilitiesKHR~~ ~~for the surface~~
  - Done
- ~~minImageCount~~ ~~must be~~ ~~1~~ ~~if~~ ~~presentMode~~ ~~is either~~ ~~VK_PRESENT_MODE_SHARED_DEMAND_REFRESH_KHR~~ ~~or~~ ~~VK_PRESENT_MODE_SHARED_CONTINUOUS_REFRESH_KHR~~
  - Done
- ~~imageFormat~~ ~~and~~ ~~imageColorSpace~~ ~~must match the~~ ~~format~~ ~~and~~ ~~colorSpace~~ ~~members, respectively, of one of the~~ ~~VkSurfaceFormatKHR~~ ~~structures returned by~~ ~~vkGetPhysicalDeviceSurfaceFormatsKHR~~ ~~for the surface~~
  - Done
- ~~imageExtent~~ ~~must be between~~ ~~minImageExtent~~ ~~and~~ ~~maxImageExtent~~, ~~inclusive, where~~ ~~minImageExtent~~ ~~and~~ ~~maxImageExtent~~ ~~are members of the~~ ~~VkSurfaceCapabilitiesKHR~~ ~~structure returned by~~ ~~vkGetPhysicalDeviceSurfaceCapabilitiesKHR~~ ~~for the surface~~
  - Done
- ~~mageExtent~~ ~~members~~ ~~width~~ ~~and~~ ~~height~~ ~~must both be non-zero~~
  - Guaranteed by the type system
- ~~imageArrayLayers~~ ~~must be greater than~~ ~~0~~ ~~and less than or equal to the~~ ~~maxImageArrayLayers~~ ~~member of the~~ ~~VkSurfaceCapabilitiesKHR~~ ~~structure returned by~~ ~~vkGetPhysicalDeviceSurfaceCapabilitiesKHR~~ ~~for the surface~~
  - Done, lower bound guaranteed by the type system
- ~~If~~ ~~presentMode~~ ~~is~~ ~~VK_PRESENT_MODE_IMMEDIATE_KHR~~, ~~VK_PRESENT_MODE_MAILBOX_KHR~~, ~~VK_PRESENT_MODE_FIFO_KHR~~ ~~or~~ ~~VK_PRESENT_MODE_FIFO_RELAXED_KHR~~, ~~imageUsage~~ ~~must be a subset of the supported usage flags present in the~~ ~~supportedUsageFlags~~ ~~member of the~~ ~~VkSurfaceCapabilitiesKHR~~ ~~structure returned by~~ ~~vkGetPhysicalDeviceSurfaceCapabilitiesKHR~~ ~~for surface~~
  - Done
- If presentMode is VK_PRESENT_MODE_SHARED_DEMAND_REFRESH_KHR or VK_PRESENT_MODE_SHARED_CONTINUOUS_REFRESH_KHR, imageUsage must be a subset of the supported usage flags present in the sharedPresentSupportedUsageFlags member of the VkSharedPresentSurfaceCapabilitiesKHR structure returned by vkGetPhysicalDeviceSurfaceCapabilities2KHR for surface
- ~~If~~ ~~imageSharingMode~~ ~~is~~ ~~VK_SHARING_MODE_CONCURRENT~~, ~~pQueueFamilyIndices~~ ~~must be a valid pointer to an array of~~ ~~queueFamilyIndexCount~~ ~~uint32_t~~ ~~values~~
  - Guaranteed by the type system
- ~~If~~ ~~imageSharingMode~~ ~~is~~ ~~VK_SHARING_MODE_CONCURRENT~~, ~~queueFamilyIndexCount~~ ~~must be greater than~~ ~~1~~
  - Guaranteed by the type system
- If imageSharingMode is VK_SHARING_MODE_CONCURRENT, each element of pQueueFamilyIndices must be unique and must be less than pQueueFamilyPropertyCount returned by either vkGetPhysicalDeviceQueueFamilyProperties
- preTransform must be one of the bits present in the supportedTransforms member of the VkSurfaceCapabilitiesKHR structure returned by vkGetPhysicalDeviceSurfaceCapabilitiesKHR for the surface
- compositeAlpha must be one of the bits present in the supportedCompositeAlpha member of the VkSurfaceCapabilitiesKHR structure returned by vkGetPhysicalDeviceSurfaceCapabilitiesKHR for the surface
- presentMode must be one of the VkPresentModeKHR values returned by vkGetPhysicalDeviceSurfacePresentModesKHR for the surface
- ~~If the logical device was created with~~ ~~VkDeviceGroupDeviceCreateInfo::physicalDeviceCount~~ ~~equal to~~ ~~1~~, ~~flags~~ ~~must not contain~~ ~~VK_SWAPCHAIN_CREATE_SPLIT_INSTANCE_BIND_REGIONS_BIT_KHR~~
  - flags are not exposed through the API
- ~~If~~ ~~oldSwapchain~~ ~~is not~~ ~~VK_NULL_HANDLE~~, ~~oldSwapchain~~ ~~must be a non-retired swapchain associated with native window referred to by surface~~

  - Done, recreation is handled specially
- The implied image creation parameters of the swapchain must be supported as reported by vkGetPhysicalDeviceImageFormatProperties
- ~~If flags contains VK_SWAPCHAIN_CREATE_MUTABLE_FORMAT_BIT_KHR then the pNext chain must include a VkImageFormatListCreateInfo structure with a viewFormatCount greater than zero and pViewFormats must have an element equal to imageFormat~~
  - flags are not exposed through the API
- ~~If flags contains VK_SWAPCHAIN_CREATE_PROTECTED_BIT_KHR, then VkSurfaceProtectedCapabilitiesKHR::supportsProtected must be VK_TRUE in the VkSurfaceProtectedCapabilitiesKHR structure returned by vkGetPhysicalDeviceSurfaceCapabilities2KHR for surface~~
  - flags are not exposed through the API
- ~~If the pNext chain includes a VkSurfaceFullScreenExclusiveInfoEXT structure with its fullScreenExclusive member set to VK_FULL_SCREEN_EXCLUSIVE_APPLICATION_CONTROLLED_EXT, and surface was created using vkCreateWin32SurfaceKHR, a VkSurfaceFullScreenExclusiveWin32InfoEXT structure must be included in the pNext chain~~
  - pNext chain is not exposed through the API

vkCreateCommandPool and VkCommandPoolCreateInfo:

- pCreateInfo→queueFamilyIndex must be the index of a queue family available in the logical device device.
- ~~If the protected memory feature is not enabled, the VK_COMMAND_POOL_CREATE_PROTECTED_BIT bit of flags must not be set.~~
  - flags are not exposed through the API

CommandBuffer

Validations for VkCommandBufferAllocateInfo:

- ~~commandBufferCount must be greater than 0~~
  - Guaranteed by the type system

RenderPass

Validations for VkRenderPassCreateInfo:

- If the attachment member of any element of pInputAttachments, pColorAttachments, pResolveAttachments or pDepthStencilAttachment, or any element of pPreserveAttachments in any element of pSubpasses is not VK_ATTACHMENT_UNUSED, it must be less than attachmentCount
- For any member of pAttachments with a loadOp equal to VK_ATTACHMENT_LOAD_OP_CLEAR, the first use of that attachment must not specify a layout equal to VK_IMAGE_LAYOUT_SHADER_READ_ONLY_OPTIMAL or VK_IMAGE_LAYOUT_DEPTH_STENCIL_READ_ONLY_OPTIMAL.
- For any member of pAttachments with a stencilLoadOp equal to VK_ATTACHMENT_LOAD_OP_CLEAR, the first use of that attachment must not specify a layout equal to VK_IMAGE_LAYOUT_SHADER_READ_ONLY_OPTIMAL or VK_IMAGE_LAYOUT_DEPTH_STENCIL_READ_ONLY_OPTIMAL.
- For any member of pAttachments with a loadOp equal to VK_ATTACHMENT_LOAD_OP_CLEAR,

the first use of that attachment must not specify a layout equal to VK_IMAGE_LAYOUT_DEPTH_READ_ONLY_STENCIL_ATTACHMENT_OPTIMAL.
- For any member of pAttachments with a stencilLoadOp equal to VK_ATTACHMENT_LOAD_OP_CLEAR, the first use of that attachment must not specify a layout equal to VK_IMAGE_LAYOUT_DEPTH_ATTACHMENT_STENCIL_READ_ONLY_OPTIMAL.
- If the pNext chain includes a VkRenderPassInputAttachmentAspectCreateInfo structure, the subpass member of each element of its pAspectReferences member must be less than subpassCount
- If the pNext chain includes a VkRenderPassInputAttachmentAspectCreateInfo structure, the inputAttachmentIndex member of each element of its pAspectReferences member must be less than the value of inputAttachmentCount in the member of pSubpasses identified by its subpass member
- If the pNext chain includes a VkRenderPassInputAttachmentAspectCreateInfo structure, for any element of the pInputAttachments member of any element of pSubpasses where the attachment member is not VK_ATTACHMENT_UNUSED, the aspectMask member of the corresponding element of VkRenderPassInputAttachmentAspectCreateInfo::pAspectReferences must only include aspects that are present in images of the format specified by the element of pAttachments at attachment
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, and its subpassCount member is not zero, that member must be equal to the value of subpassCount
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, if its dependencyCount member is not zero, it must be equal to dependencyCount
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, for each non-zero element of pViewOffsets, the srcSubpass and dstSubpass members of pDependencies at the same index must not be equal
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, for any element of pDependencies with a dependencyFlags member that does not include VK_DEPENDENCY_VIEW_LOCAL_BIT, the corresponding element of the pViewOffsets member of that VkRenderPassMultiviewCreateInfo instance must be 0
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, elements of its pViewMasks member must either all be 0, or all not be 0
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, and each element of its pViewMasks member is 0, the dependencyFlags member of each element of pDependencies must not include VK_DEPENDENCY_VIEW_LOCAL_BIT
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, and each element of its pViewMasks member is 0, correlatedViewMaskCount must be 0
- If the pNext chain includes a VkRenderPassMultiviewCreateInfo structure, each element of its pViewMask member must not have a bit set at an index greater than or equal to VkPhysicalDeviceLimits::maxFramebufferLayers
- For any element of pDependencies, if the srcSubpass is not VK_SUBPASS_EXTERNAL, all stage flags included in the srcStageMask member of that dependency must be a pipeline stage supported by the pipeline identified by the pipelineBindPoint member of the source subpass
- For any element of pDependencies, if the dstSubpass is not VK_SUBPASS_EXTERNAL, all stage flags included in the dstStageMask member of that dependency must be a pipeline stage supported by the pipeline identified by the pipelineBindPoint member of the destination subpass
- The srcSubpass member of each element of pDependencies must be less than subpassCount
- The dstSubpass member of each element of pDependencies must be less than subpassCount

Framebuffer

ComputePipeline

GraphicsPipeline

Image

ImageView

Buffer

BufferView