

Handle

HANDLE

NON_DISPATCHABLE_HAI

Handle

+ xxxCount: uint32_t

+ xxxIndex: uint32_t

Height: 26, 40, 56

+ xxxIndices: const uint32_t*

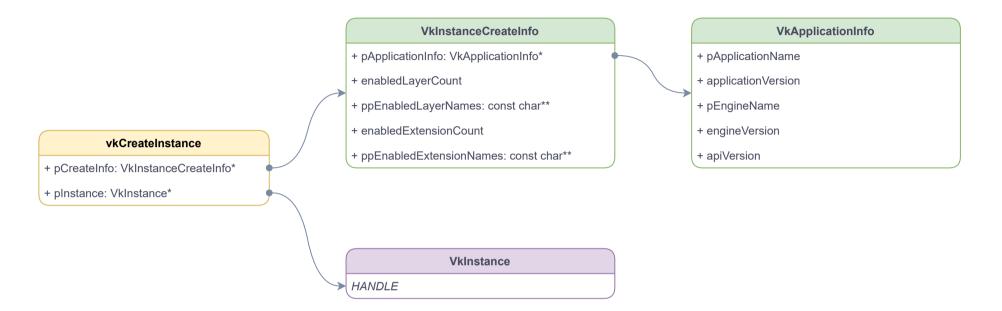
参考

Vulkan Specs 1.2

Vulkan Tutorial

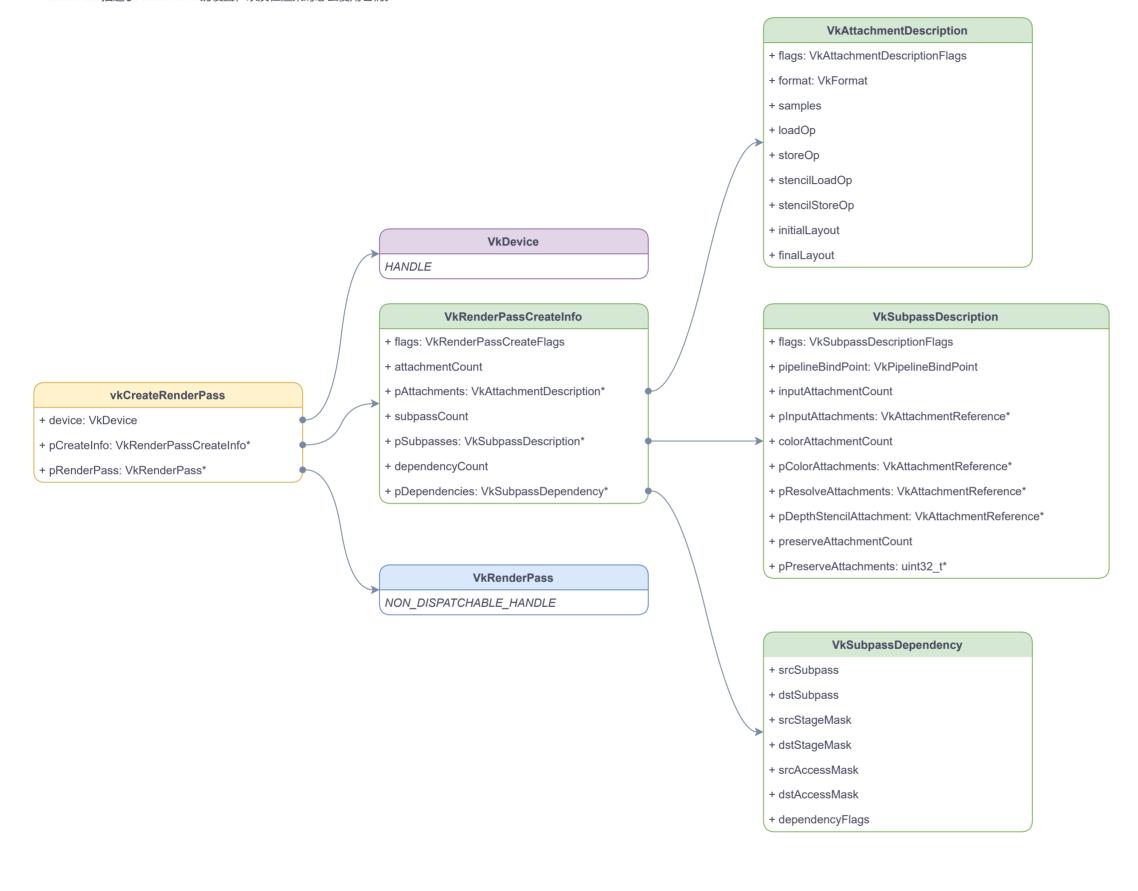
Instance

通过Instance使用Vulkan API。



RenderPass

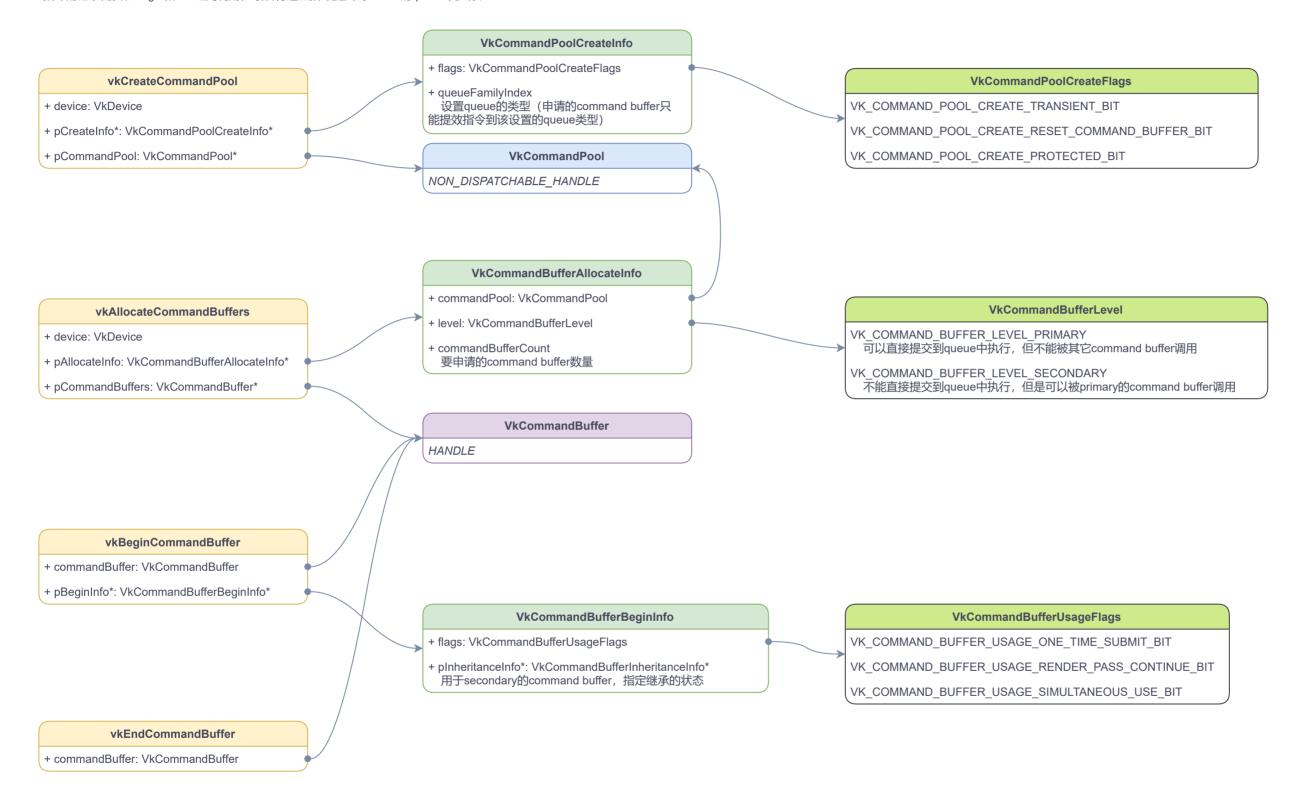
RenderPass描述了Attachments的设置,以及在渲染时怎么使用它们。



CommandBuffer

CommandBuffer用来存储vulkan相关操作指令,例如draw指令。

指令的记录需要在begin和end之间调用;最后将这些指令提交到device的queue中执行。



Buffer

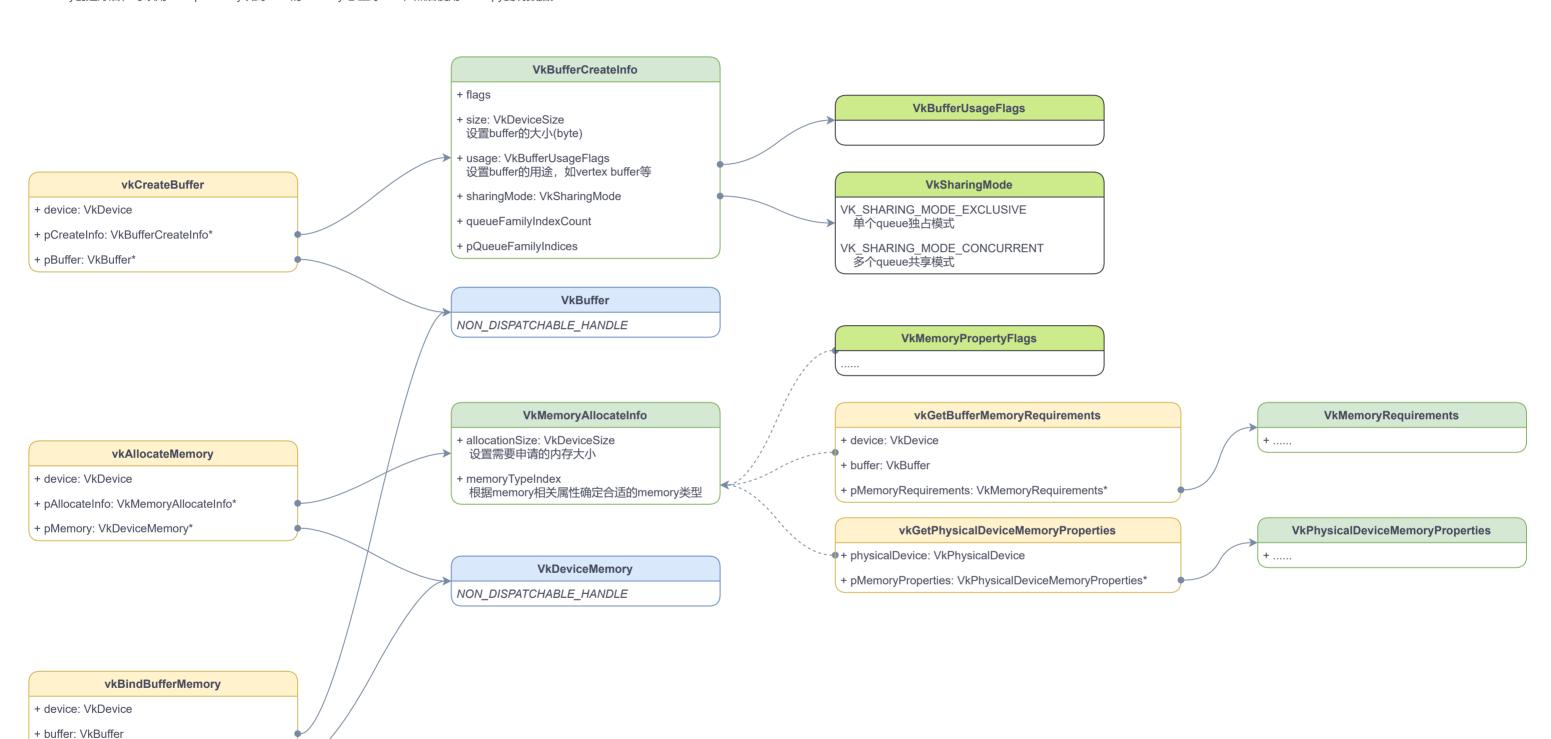
+ memory: VkDeviceMemory+ memoryOffset: VkDeviceSize

Buffer可以用来存储任意类型的用于GPU使用的数据。

Buffer创建好后,需要为其申请memory,并将buffer和memory绑定。

Memory是GPU使用的内存,其类型需要根据buffer和memory需要求确定。

Memory创建好后,可以用vkMapMemory映到GPU的memory地址到CPU,然后使用memcpy复制数据。



Descriptor

Descriptor用于shader获取uniform、image等资源。

- * 在pipeline中设置descriptor set layout;
- * 从descriptor pool中申请descriptor set;
- * 绑定descriptor set到render pass;

VkDescriptorSetLayoutBinding binding: uint32 t 设置shader中的binding值 descriptorType: VkDescriptorType 设置descriptor类型,例如uniform buffer descriptorCount 设置descriptor数量,用于数组类型 - stageFlags: VkShaderStageFlags 设置descriptor的作用域,例如vertex shader plmmutableSamplers: VkSampler* 用于设置纹理资源的采样器 vkCreateDescriptorSetLayout + device: VkDevice **VkDescriptorSetLayoutCreateInfo** + pCreateInfo: VkDescriptorSetLayoutCreateInfo* + flags: VkDescriptorSetLayoutCreateFlags + pSetLayout: VkDescriptorSetLayout* + bindingCount - pBindings: VkDescriptorSetLayoutBinding* VkDescriptorSetLayout NON DISPATCHABLE HANDLE **VkWriteDescriptorSet** dstSet: VkDescriptorSet vkUpdateDescriptorSets 设置需要更新的descriptor set + device: VkDevice dstBinding: uint32 t 设置sahder中对应descriptor set的binding + descriptorWriteCount dstArrayElement: uint32_t + pDescriptorWrites: VkWriteDescriptorSet* - descriptorCount + descriptorCopyCount + descriptorType: VkDescriptorType + pDescriptorCopies: VkCopyDescriptorSet* plmagelnfo*: VkDescriptorImageInfo*

- pBufferInfo*: VkDescriptorBufferInfo*

- pTexelBufferView: VkBufferView*

VkDescriptorPoolSize + type: VkDescriptorType + descriptorCount vkCreateDescriptorPool VkDescriptorPoolCreateInfo + device: VkDevice + flags: VkDescriptorPoolCreateFlags + pCreateInfo: VkDescriptorPoolCreateInfo* maxSets: uint32 t + pDescriptorPool*: VkDescriptorPool* 设置descriptor set的数量 - poolSizeCount + pPoolSize: VkDescriptorPoolSize* 设置pool中的descriptor类型和容量 VkDescriptorPool NON_DISPATCHABLE_HANDLE **VkDescriptorSetAllocateInfo** vkAllocateDescriptorSets + device: VkDevice + descriptorPool: VkDescriptorPool + pAllocateInfo: VkDescriptorSetAllocateInfo* + descriptorSetCount + pDescriptorSet: VkDescriptorSet* + pSetLayout: VkDescriptorSetLayout* VkDescriptorSet NON_DISPATCHABLE_HANDLE