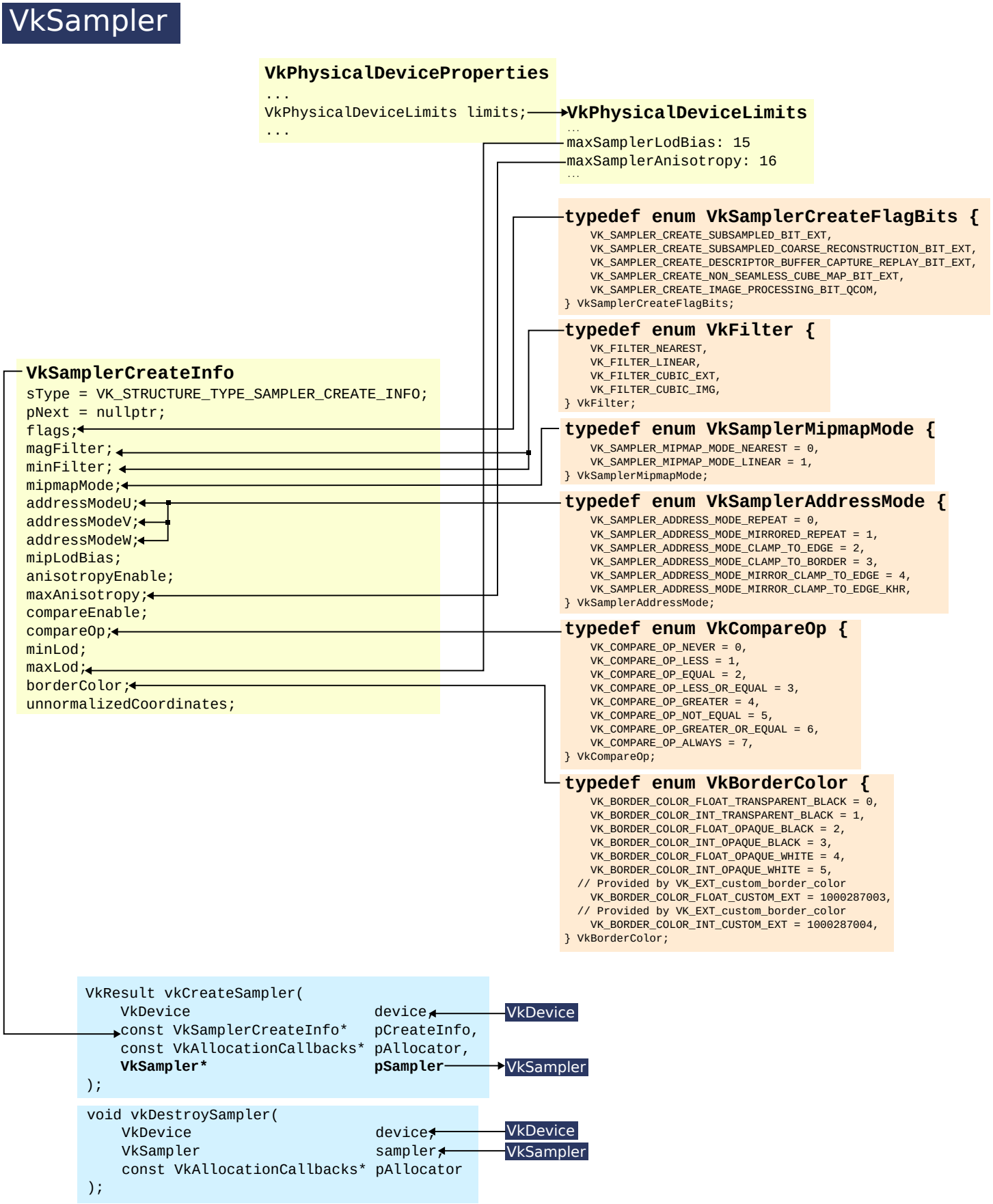
VkSampler


VkPhysicalDeviceProperties
...
VkPhysicalDeviceLimits limits;
...
VkPhysicalDeviceLimits
...
maxSamplerLodBias: 15
maxSamplerAnisotropy: 16
...
typedef enum VkSamplerCreateFlagBits {
VK_SAMPLER_CREATE_SUBSAMPLED_BIT_EXT,
VK_SAMPLER_CREATE_SUBSAMPLED_COARSE_RECONSTRUCTION_BIT_EXT,
VK_SAMPLER_CREATE_DESCRIPTOR_BUFFER_CAPTURE_REPLAY_BIT_EXT,
VK_SAMPLER_CREATE_NON_SEAMLESS_CUBE_MAP_BIT_EXT,
VK_SAMPLER_CREATE_IMAGE_PROCESSING_BIT_QCOM,
} VkSamplerCreateFlagBits;
typedef enum VkFilter {
VK_FILTER_NEAREST,
VK_FILTER_LINEAR,
VK_FILTER_CUBIC_EXT,
VK_FILTER_CUBIC_IMG,
} VkFilter;
VkSamplerCreateInfo
sType = VK_STRUCTURE_TYPE_SAMPLER_CREATE_INFO;
pNext = nullptr;
flags;
magFilter;
minFilter;
mipmapMode;
addressModeU;
addressModeV;
addressModeW;
mipLodBias;
anisotropyEnable;
maxAnisotropy;
compareEnable;
compareOp;
minLod;
maxLod;
borderColor;
unnormalizedCoordinates;
typedef enum VkSamplerMipmapMode {
VK_SAMPLER_MIPMAP_MODE_NEAREST = 0,
VK_SAMPLER_MIPMAP_MODE_LINEAR = 1,
} VkSamplerMipmapMode;
typedef enum VkSamplerAddressMode {
VK_SAMPLER_ADDRESS_MODE_REPEAT = 0,
VK_SAMPLER_ADDRESS_MODE_MIRRORED_REPEAT = 1,
VK_SAMPLER_ADDRESS_MODE_CLAMP_TO_EDGE = 2,
VK_SAMPLER_ADDRESS_MODE_CLAMP_TO_BORDER = 3,
VK_SAMPLER_ADDRESS_MODE_MIRROR_CLAMP_TO_EDGE = 4,
VK_SAMPLER_ADDRESS_MODE_MIRROR_CLAMP_TO_EDGE_KHR,
} VkSamplerAddressMode;
typedef enum VkCompareOp {
VK_COMPARE_OP_NEVER = 0,
VK_COMPARE_OP_LESS = 1,
VK_COMPARE_OP_EQUAL = 2,
VK_COMPARE_OP_LESS_OR_EQUAL = 3,
VK_COMPARE_OP_GREATER = 4,
VK_COMPARE_OP_NOT_EQUAL = 5,
VK_COMPARE_OP_GREATER_OR_EQUAL = 6,
VK_COMPARE_OP_ALWAYS = 7,
} VkCompareOp;
typedef enum VkBorderColor {
VK_BORDER_COLOR_FLOAT_TRANSPARENT_BLACK = 0,
VK_BORDER_COLOR_INT_TRANSPARENT_BLACK = 1,
VK_BORDER_COLOR_FLOAT_OPAQUE_BLACK = 2,
VK_BORDER_COLOR_INT_OPAQUE_BLACK = 3,
VK_BORDER_COLOR_FLOAT_OPAQUE_WHITE = 4,
VK_BORDER_COLOR_INT_OPAQUE_WHITE = 5,
// Provided by VK_EXT_custom_border_color
VK_BORDER_COLOR_FLOAT_CUSTOM_EXT = 1000287003,
// Provided by VK_EXT_custom_border_color
VK_BORDER_COLOR_INT_CUSTOM_EXT = 1000287004,
} VkBorderColor;
VkResult vkCreateSampler(
VkDevice device,
const VkSamplerCreateInfo* pCreateInfo,
const VkAllocationCallbacks* pAllocator,
VkSampler* pSampler
);
VkDevice
VkSampler
void vkDestroySampler(
VkDevice device,
VkSampler sampler,
const VkAllocationCallbacks* pAllocator
);
VkDevice
VkSampler