

## VkSubpassDescription for VkRenderPass

```
typedef enum VkPipelineBindPoint {
    VK_PIPELINE_BIND_POINT_GRAPHICS = 0,
    VK_PIPELINE_BIND_POINT_COMPUTE = 1,
    // Provided by VK_KHR_ray_tracing_pipeline
    VK_PIPELINE_BIND_POINT_RAY_TRACING_KHR = 1000165000,
    // Provided by VK_HUAWEI_subpass_shading
    VK_PIPELINE_BIND_POINT_SUBPASS_SHADING_HUAWEI = 1000369003,
    // Provided by VK_NV_ray_tracing
    VK_PIPELINE_BIND_POINT_RAY_TRACING_NV = VK_PIPELINE_BIND_POINT_RAY_TRACING_KHR,
} VkPipelineBindPoint;
```

index in `VkRenderPassCreateInfo::pAttachments`,  
or `VK_ATTACHMENT_UNUSED`

```
typedef struct VkSubpassDescription {
    VkSubpassDescriptionFlags    flags; // usually 0
    VkPipelineBindPoint          pipelineBindPoint;
    uint32_t                     inputAttachmentCount;
    const VkAttachmentReference* pInputAttachments;
    uint32_t                     colorAttachmentCount;
    const VkAttachmentReference* pColorAttachments;
    const VkAttachmentReference* pResolveAttachments;
    const VkAttachmentReference* pDepthStencilAttachment;
    uint32_t                     preserveAttachmentCount;
    const uint32_t*              pPreserveAttachments;
} VkSubpassDescription;
```

```
typedef enum VkImageLayout {  
    VK_IMAGE_LAYOUT_UNDEFINED = 0,  
    VK_IMAGE_LAYOUT_GENERAL = 1,  
    VK_IMAGE_LAYOUT_COLOR_ATTACHMENT_OPTIMAL = 2,  
    VK_IMAGE_LAYOUT_DEPTH_STENCIL_ATTACHMENT_OPTIMAL = 3,  
    VK_IMAGE_LAYOUT_DEPTH_STENCIL_READ_ONLY_OPTIMAL = 4,  
    VK_IMAGE_LAYOUT_SHADER_READ_ONLY_OPTIMAL = 5,  
    VK_IMAGE_LAYOUT_TRANSFER_SRC_OPTIMAL = 6,  
    VK_IMAGE_LAYOUT_TRANSFER_DST_OPTIMAL = 7,  
    VK_IMAGE_LAYOUT_PREINITIALIZED = 8,  
    ...  
} VkImageLayout;
```

```
typedef struct VkAttachmentReference {
    → uint32_t      attachment;
    → VkImageLayout layout;
} VkAttachmentReference;
```

## VkSubpassDependency for VkRenderPass

```
typedef enum VkPipelineStageFlagsBits {
    VK_PIPELINE_STAGE_TOP_OF_PIPE_BIT = 0x00000001,
    VK_PIPELINE_STAGE_DRAW_INDIRECT_BIT = 0x00000002,
    VK_PIPELINE_STAGE_VERTEX_INPUT_BIT = 0x00000004,
    VK_PIPELINE_STAGE_VERTEX_SHADER_BIT = 0x00000008,
    VK_PIPELINE_STAGE_TESSELLATION_CONTROL_SHADER_BIT = 0x00000010,
    VK_PIPELINE_STAGE_TESSELLATION_EVALUATION_SHADER_BIT = 0x00000020,
    VK_PIPELINE_STAGE_GEOMETRY_SHADER_BIT = 0x00000040,
    VK_PIPELINE_STAGE_FRAGMENT_SHADER_BIT = 0x00000080,
    VK_PIPELINE_STAGE_EARLY_FRAGMENT_TESTS_BIT = 0x00000100,
    VK_PIPELINE_STAGE_LATE_FRAGMENT_TESTS_BIT = 0x00000200,
    VK_PIPELINE_STAGE_COLOR_ATTACHMENT_OUTPUT_BIT = 0x00000400,
    VK_PIPELINE_STAGE_COMPUTE_SHADER_BIT = 0x00000800,
    VK_PIPELINE_STAGE_TRANSFER_BIT = 0x00001000,
    VK_PIPELINE_STAGE_BOTTOM_OF_PIPE_BIT = 0x00002000,
    VK_PIPELINE_STAGE_HOST_BIT = 0x00004000,
    VK_PIPELINE_STAGE_ALL_GRAPHICS_BIT = 0x00008000,
    VK_PIPELINE_STAGE_ALL_COMMANDS_BIT = 0x00010000,

    /* Provided by VK_VERSION_1_3 */
    VK_PIPELINE_STAGE_MESH = 9,
    /* Provided by VK_EXT_transform_feedback */
    VK_PIPELINE_STAGE_TRANSFORM_FEEDBACK_BIT_EXT = 0x01000000,
    /* Provided by VK_EXT_conditional_rendering */
    VK_PIPELINE_STAGE_CONDITIONAL_RENDERING_BIT_EXT = 0x00040000,
    /* Provided by VK_KHR_acceleration_structure */
    VK_PIPELINE_STAGE_ACCELERATION_STRUCTURE_BUILD_BIT_KHR = 0x00200000,
    VK_PIPELINE_STAGE_RAY_TRACING_SHADER_BIT_KHR = 0x00020000,
    /* Provided by VK_EXT_fragment_density_map */
    VK_PIPELINE_STAGE_FRAGMENT_DENSITY_PROCESS_BIT_EXT = 0x00000000,
    /* Provided by VK_KHR_fragment_shading_rate */
    VK_PIPELINE_STAGE_FRAGMENT_SHADING_RATE_ATTACHMENT_BIT_KHR = 0x00000000,
    /* Provided by VK_NV_optical_flow_motion_vectors */
    VK_PIPELINE_STAGE_COMMAND_PREPROCESS_BIT_NV = 0x00000000,
    /* Provided by VK_EXT_mesh_shader */
    VK_PIPELINE_STAGE_MESH_SHADER_BIT_EXT = 0x00000000,
    /* Provided by VK_EXT_mesh_shader */
    VK_PIPELINE_STAGE_MESH_SHADER_BIT_EXT = 0x00000000,
    /* Provided by VK_NV_shader_core_properties */
    VK_PIPELINE_STAGE_FRAGMENT_SHADING_RATE_ATTACHMENT_BIT_KHR,
    /* Provided by VK_NV_ray_tracing */
    VK_PIPELINE_STAGE_ACCELERATION_STRUCTURE_BUILD_BIT_KHR,
    /* Provided by VK_NV_ray_tracing */
    VK_PIPELINE_STAGE_ACCELERATION_STRUCTURE_BUILD_BIT_KHR,
    /* Provided by VK_NV_mesh_shader */
    VK_PIPELINE_STAGE_MESH_SHADER_BIT_KHR,
    /* Provided by VK_PIPELINE_STAGE_MESH_SHADER_BIT_KHR */
    VK_PIPELINE_STAGE_MESH_SHADER_BIT_KHR,
    /* Provided by VK_KHR_synchronization2 */
    VK_PIPELINE_STAGE_MESH_SHADER_BIT_KHR = VK_PIPELINE_STAGE_MESH_SHADER_BIT_EXT,
} VkPipelineStageFlagsBits;
```

```

typedef enum VkAccessFlagBits {
    VK_ACCESS_INDIRECT_COMMAND_READ_BIT = 0x00000001,
    VK_ACCESS_INDEX_READ_BIT = 0x00000002,
    VK_ACCESS_VERTEX_ATTRIBUTE_READ_BIT = 0x00000004,
    VK_ACCESS_UNIFORM_READ_BIT = 0x00000008,
    VK_ACCESS_INPUT_ATTACHMENT_READ_BIT = 0x00000010,
    VK_ACCESS_SHADER_READ_BIT = 0x00000020,
    VK_ACCESS_SHADER_WRITE_BIT = 0x00000040,
    VK_ACCESS_COLOR_ATTACHMENT_READ_BIT = 0x00000080,
    VK_ACCESS_COLOR_ATTACHMENT_WRITE_BIT = 0x00000100,
    VK_ACCESS_DEPTH_STENCIL_ATTACHMENT_READ_BIT = 0x00000200,
    VK_ACCESS_DEPTH_STENCIL_ATTACHMENT_WRITE_BIT = 0x00000400,
    VK_ACCESS_TRANSFER_READ_BIT = 0x00000800,
    VK_ACCESS_TRANSFER_WRITE_BIT = 0x00001000,
    VK_ACCESS_HOST_READ_BIT = 0x00002000,
    VK_ACCESS_HOST_WRITE_BIT = 0x00004000,
    VK_ACCESS_MEMORY_READ_BIT = 0x00008000,
    VK_ACCESS_MEMORY_WRITE_BIT = 0x00010000,

    // Provided by VK_KHR_samplerless_sampler_image
    VK_ACCESS_NONE = 0,

    // Provided by VK_EXT_transform_feedback
    VK_ACCESS_TRANSFORM_FEEDBACK_WRITE_BIT_EXT = 0x02000000,

    // Provided by VK_EXT_transform_feedback
    VK_ACCESS_TRANSFORM_FEEDBACK_COUNTER_READ_BIT_EXT = 0x04000000,

    // Provided by VK_EXT_transform_feedback
    VK_ACCESS_TRANSFORM_FEEDBACK_COUNTER_WRITE_BIT_EXT = 0x08000000,

    // Provided by VK_EXT_conditional_rendering
    VK_ACCESS_CONDITIONAL_RENDERING_READ_BIT_EXT = 0x00100000,

    // Provided by VK_EXT_image_operation_advanced
    VK_ACCESS_COLOR_ATTACHMENT_READ_NONRESOLVABLE_BIT_EXT = 0x00000000,

    // Provided by VK_KHR_acceleration_structure
    VK_ACCESS_ACCELERATION_STRUCTURE_READ_BIT_KHR = 0x00000000,

    // Provided by VK_KHR_acceleration_structure
    VK_ACCESS_ACCELERATION_STRUCTURE_WRITE_BIT_KHR = 0x00000000,

    // Provided by VK_KHR_fragment_density_map
    VK_ACCESS_FRAGMENT_DENSITY_MAP_READ_BIT_EXT = 0x01000000,

    // Provided by VK_KHR_fragment_shading_rate
    VK_ACCESS_FRAGMENT_SHADING_RATE_ATTACHMENT_READ_BIT_KHR = 0x00000000,

    // Provided by VK_KHR_device_generated_commands
    VK_ACCESS_COMMAND_PREPROCESS_READ_BIT_NV = 0x00000000,

    // Provided by VK_KHR_device_generated_commands
    VK_ACCESS_COMMAND_PREPROCESS_WRITE_BIT_NV = 0x00000000,

    // Provided by VK_KHR_shading_rate_image
    VK_ACCESS_SHADING_RATE_IMAGE_READ_BIT_NV = VK_ACCESS_FRAGMENT_SHADING_RATE_ATTACHMENT_READ_BIT_KHR,

    // Provided by VK_KHR_ray_tracing
    VK_ACCESS_ACCELERATION_STRUCTURE_READ_BIT_NV = VK_ACCESS_ACCELERATION_STRUCTURE_READ_BIT_KHR,

    // Provided by VK_KHR_ray_tracing
    VK_ACCESS_ACCELERATION_STRUCTURE_WRITE_BIT_NV = VK_ACCESS_ACCELERATION_STRUCTURE_WRITE_BIT_KHR,

    // Provided by VK_KHR_synchronization2
    VK_ACCESS_NONE_KHR = VK_ACCESS_NONE,
} VkAccessFlagBits;

```

## VkSubpassDependency

```
srcSubpass; // index
dstSubpass; // index
srcStageMask;
dstStageMask;
srcAccessMask;
dstAccessMask;
dependencyFlags;
```

```
-typedef enum VkDependencyFlagBits {  
    VK_DEPENDENCY_BY_REGION_BIT = 0x00000001,  
    // Provided by VK_VERSION_1_1  
    VK_DEPENDENCY_DEVICE_GROUP_BIT = 0x00000004,  
    // Provided by VK_VERSION_1_1  
    VK_DEPENDENCY_VIEW_LOCAL_BIT = 0x00000002,  
    // Provided by VK_EXT_attachment_feedback_loop_layout  
    VK_DEPENDENCY_FEEDBACK_LOOP_BIT_EXT = 0x00000008,  
    // Provided by VK_KHR_multiview  
    VK_DEPENDENCY_VIEW_LOCAL_BIT_KHR = VK_DEPENDENCY_VIEW_LOCAL_BIT,  
    // Provided by VK_KHR_device_group  
    VK_DEPENDENCY_DEVICE_GROUP_BIT_KHR = VK_DEPENDENCY_DEVICE_GROUP_BIT,  
} VkDependencyFlagBits;
```