DrawIndirect

传统情况下实现多个物体同一个DrawCall绘制的办法就是GPU Instance，通过指定同一套材质属性，同一套管线状态和同一个模型，在屏幕上同时绘制多次，**然而GPU Instance限制非常大顶点数量必须一致。**所以大多数情况下都是用于同样模型，同样的材质的情况下的提交(注意的是Transform不用相同),如，在instancing 例子中：

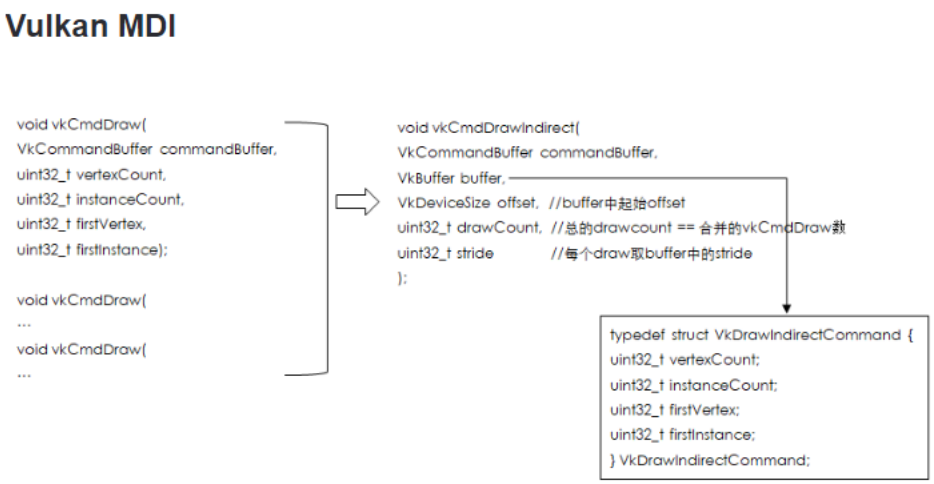
```
// Instanced rocks
      vkCmdBindDescriptorSets(drawCmdBuffers[i], VK_PIPELINE_BIND_POINT_GRAPHICS, pipelineLayout, 0, 1, &descriptorSets.instancedRocks, 0,
      vkCmdBindPipeline(drawCmdBuffers[i], VK_PIPELINE_BIND_POINT_GRAPHICS, pipelines.instancedRocks);
      // Binding point 0 : Mesh vertex buffer
      vkCmdBindVertexBuffers(drawCmdBuffers[i], VERTEX_BUFFER_BIND_ID, 1, &models.rock.vertices.buffer, offsets);
      // Binding point 1 : Instance data buffer
      vkCmdBindVertexBuffers(drawCmdBuffers[i], INSTANCE_BUFFER_BIND_ID, 1, &instanceBuffer.buffer, offsets);
      // Bind index buffer
      vkCmdBindIndexBuffer(drawCmdBuffers[i], models.rock.indices.buffer, 0, VK_INDEX_TYPE_UINT32);

      // Render instances
      vkCmdDrawIndexed(drawCmdBuffers[i], models.rock.indices.count, INSTANCE_COUNT, 0, 0, 0);
```

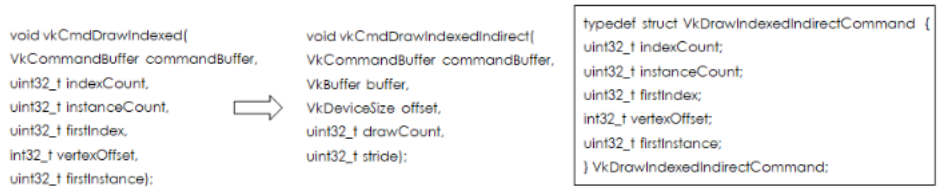
会将需要提交的都放在同一个Buffer中让其进行提交。

Nvidia在2012年提出AZDO（For Approaching Zero Driver Overhead）的思想后，也推出了Multi-draw-indirect的思想，并提出了GL_ARB_multi_draw_indirect的extension，里面也提到了如果当你想要绘制的场景中，存在着很多类似的model，但是却需要通过大量的drawcall来绘制的时候，可以考虑采用MultiDrawIndirect的技术来降低API的调用，从而减少CPU侧的开销。在介绍中，nvidia也给出了一个demo，来展示使用GL_ARB_multi_draw_indirect来绘制很多类似的物体时候，CPU侧开销非常低，同时帧率也保持较高。

Vulkan中目前也已经有MultiDrawIndirect的API：分别是vkCmdDrawIndirect和vkCmdDrawIndexedIndirect。可以具体查看官方的文档。可以说MDI的提出，是对CPU侧的进一步释放。



可以看到，当存在多个vkCmdDraw，并且其中并未发生pipeline的切换时，可以通过vkCmdDrawIndirect将这些vkCmdDraw打包成一个API。在vkCmdDrawIndirect中，会有一个指向VkDrawIndirectCommand的buffer，它记载的是每个draw的vertexCount，instanceCount等的信息。同样的，多个vkCmdDrawIndexed也是可以组合成vkCmdDrawIndexedIndirect。




```
// If the multi draw feature is supported:
      // One draw call for an arbitrary number of objects
      // Index offsets and instance count are taken from the indirect buffer
      if (vulkanDevice->features.multiDrawIndirect)
      {
        vkCmdDrawIndexedIndirect(drawCmdBuffers[i], indirectCommandsBuffer.buffer, 0, indirectDrawCount, sizeof(VkDrawIndexedIndirectComman
      }
      else
      {
        // If multi draw is not available, we must issue separate draw commands
        for (auto j = 0; j < indirectCommands.size(); j++)
        {
          vkCmdDrawIndexedIndirect(drawCmdBuffers[i], indirectCommandsBuffer.buffer, j * sizeof(VkDrawIndexedIndirectCommand), 1, sizeof(Vk
        }
      }
```

https://developer.huawei.com/consumer/cn/forum/topic/0203331652326770210?fid=0103325401414330531&postId=0303331652326770845