

简单实例化示例

此示例与 Microsoft 游戏开发工具包预览版（2019 年 11 月）兼容

# 说明

本示例演示如何使用 Direct3D 12 API 进行实例化。

# 构建示例

如果使用 Xbox One 开发工具包，请将活动解决方案平台设置为 Gaming.Xbox.XboxOne.x64。

如果使用 Project Scarlett，请将活动解决方案平台设置为 Gaming.Xbox.Scarlett.x64。

有关详细信息，请参阅 GDK 文档中的“运行示例”。

# 使用示例



|  |  |
| --- | --- |
| 操作 | 游戏手柄 |
| 旋转相机 | 左操纵杆 |
| 更改实例计数 | LB / RB |
| 重置模拟 | A |
| 退出 | “视图”按钮 |

# 实现说明

在呈现实例化几何图形时，需要考虑四个方面：

1. 几何数据

本示例中的集合数据包括一个容纳顶点和索引（用于描述立方体的表面）的顶点和索引缓冲区。它还包括呈现出此立方体所需的管道状态对象。所有这些组件的设置和操作方式都与非实例化几何图形相同。（请参阅 SimpleInstancing.cpp 中的 *CreateDeviceDependentResources*）

1. 实例数据

对于标准 D3D12 实例化呈现，每个实例数据通过一个或多个顶点缓冲区提供。将按照与任何其他顶点缓冲区相同的方式创建这些顶点缓冲区。本示例使用两个顶点缓冲区。一个是静态的，包含每个实例的颜色数据（在本示例的生命周期内不会改变）。另一个是动态的，包含每个实例的位置和方向信息（在每一帧中都发生变化）。

1. 实例化布局

为了呈现实例化的集合图形，D3D 需要了解已提供顶点数据的解释方式。这是通过 D3D12\_INPUT\_ELEMENT\_DESC结构的数组实现的，与标准呈现的方法基本相同。但向此结构中添加了额外的元素。几何数据像往常一样用 D3D12\_INPUT\_CLASSIFICATION\_PER\_VERTEX\_DATA 值来标记 InputSlotClass 元素，但每实例数据使用 D3D12\_INPUT\_CLASSIFICATION\_PER\_INSTANCE\_DATA 值。“InputSlot”元素还可以表示从中提取每个数据段的顶点流。

顶点着色器使用一个顶点结构，该顶点结构被定义为好像几何数据和每实例数据都被集合在一起一样（这反映了 D3D12\_INPUT\_ELEMENT\_DESC 数组中描述的布局）。

注意：本示例使用 AlignedByteOffset 元素的 D3D12\_APPEND\_ALIGNED\_ELEMENT 常量自动在输入布局中正确对齐数据。只有当所述顶点缓冲区的结构中包含正确对齐的数据时，这才有效。如果要跳过（或忽略）顶点数据中的元素，需要精确的对齐偏移量。

1. 呈现

一旦确定了先前的要点，呈现实例化数据就很简单。使用 ID3D12GraphicsCommandList::IASetVertexBuffers API 设置用作输入的顶点缓冲区（本示例中为输入缓冲区），使用 ID3D12GraphicsCommandList::DrawIndexedInstanced API 来呈现内容。其余呈现设置的执行方式与标准非实例化呈现相同。

# 隐私声明

在编译和运行示例时，示例可执行文件的文件名将发送给 Microsoft，用于帮助跟踪示例使用情况。要选择退出此数据收集，你可以删除 Main.cpp 中标记为“示例使用遥测”的代码块。

有关 Microsoft 的一般隐私策略的详细信息，请参阅《[Microsoft 隐私声明](https://privacy.microsoft.com/en-us/privacystatement/)》。