Simple Dynamic Resources Sample

*This sample is compatible with the Microsoft Game Development Kit (April 2021)*

# Description

This sample demonstrates how to use HLSL Dynamic Resources in HLSL Shader Model 6.6. It is functionally identical to SimpleTexture, except resources are accessed directly through the heap using ResourceDescriptorHeap[] and SamplerDescriptorHeap[] in HLSL.



# Building the sample

If using an Xbox One devkit, set the active solution platform to Gaming.Xbox.XboxOne.x64.

If using Xbox Series X|S, set the active solution platform to Gaming.Xbox.Scarlett.x64.

Building for PC (Gaming.Desktop.x64) requires the [DirectX Agility SDK](https://devblogs.microsoft.com/directx/gettingstarted-dx12agility/) due to the usage of HLSL SM 6.6 features. The Agility SDK is included as a NuGet package in the sample. It also makes use of Microsoft.Windows.SDK.CPP NuGet packages to get the latest Windows SDK (22000) version of the DXC.exe compiler. Developers can also use the latest DXC directly from [Github](https://github.com/microsoft/DirectXShaderCompiler/releases).

*For more information, see* Running samples*, in the GDK documentation.*

# Using the sample

The sample has no controls other than exiting.

# Implementation notes

This sample borrows nearly all the code from SimpleTexture. The only difference is in the access of resources.

This sample removes the bound resources from the root signature, replacing them with ResourceDescriptorHeap[] and SamplerDescriptorHeap[] accesses in the HLSL shader code. This requires ensuring SetDescriptorHeaps() is called before SetGraphicsRootSignature(), and adding the flags CBV\_SRV\_UAV\_HEAP\_DIRECTLY\_INDEXED and SAMPLER\_HEAP\_DIRECTLY\_INDEXED to the root signature.

For more information about HLSL 6.6 Dynamic Resources, see [HLSL SM 6.6 Dynamic Resources](https://microsoft.github.io/DirectX-Specs/d3d/HLSL_SM_6_6_DynamicResources.html).

For more advanced usage of Dynamic Resources, see the Graphics\VisibilityBuffer sample.

# Privacy Statement

When compiling and running a sample, the file name of the sample executable will be sent to Microsoft to help track sample usage. To opt-out of this data collection, you can remove the block of code in Main.cpp labeled “Sample Usage Telemetry”.

For more information about Microsoft’s privacy policies in general, see the [Microsoft Privacy Statement](https://privacy.microsoft.com/en-us/privacystatement/).