BasicHLSLFX11

<https://github.com/walbourn/directx-sdk-samples>

This is the DirectX SDK's Direct3D 11 sample updated to use Visual Studio 2012 and the Windows SDK 8.0 without any dependencies on legacy DirectX SDK content. This sample is a Win32 desktop DirectX 11.0 application for Windows 10, Windows 8.1, Windows 8, Windows 7, and Windows Vista Service Pack 2 with the DirectX 11.0 runtime.

**This is based on the legacy DirectX SDK (June 2010) Win32 desktop sample. This is not intended for use with Windows Store apps, Windows RT, or universal Windows apps.**

# Description



This sample loads a mesh, create vertex and pixel shaders from files, and then uses the shaders to render the mesh.

This is a Direct3D 11 version of the Direct3D 10 BasicHLSL10 sample. BasicHLSL11 is a version of the same sample that does not make use of Effects 11.

## How the sample works

First the sample checks for the currently supported feature level and compiles and creates vertex and pixel shaders from a file.

Next, the sample creates an input layout that matches the input layout of the mesh that will be loaded. This will be the same for all meshes loaded through CDXUTSDKMesh.

Next, the sample loads the geometry using the CDXUTSDKMesh class.

In OnD3D11FrameRender, the sample updates dynamic state contained in constant buffers such as the World\*View\*Projection matrix, then sets both static and dynamic state such as the current input layout and samplers to be used for rendering using the immediate context. The mesh is rendered using DrawIndexed calls called in a loop over the mesh subsets.

# Dependencies

DXUT-based samples typically make use of runtime HLSL compilation. Build-time compilation is recommended for all production Direct3D applications, but for experimentation and samples development runtime HLSL compilation is preferred. Effects11-based samples also need the D3DCompile APIs for reflection. Therefore, the D3DCompile\*.DLL must be available in the search path when these programs are executed.

* When using the Windows 8.x SDK and targeting Windows Vista or later, you can include the D3DCompile\_46 or D3DCompile\_47 DLL side-by-side with your application copying the file from the REDIST folder.

%ProgramFiles(x86)%\Windows kits\8.0\Redist\D3D\arm, x86 or x64

%ProgramFiles(x86)%\Windows kits\8.1\Redist\D3D\arm, x86 or x64

%ProgramFiles(x86)%\Windows kits\10\Redist\D3D\arm, x86 or x64

# More Information

[Where is the DirectX SDK (2015 Edition)?](https://walbourn.github.io/where-is-the-directx-sdk-2015-edition/)

[DXUT for Win32 Desktop Update](https://walbourn.github.io/dxut-for-win32-desktop-update/)

[Effects for Direct3D 11 Update](https://walbourn.github.io/effects-for-direct3d-11-update/)

[Games for Windows and DirectX SDK blog](https://walbourn.github.io/)