

Simple SamplerFeedback Sample

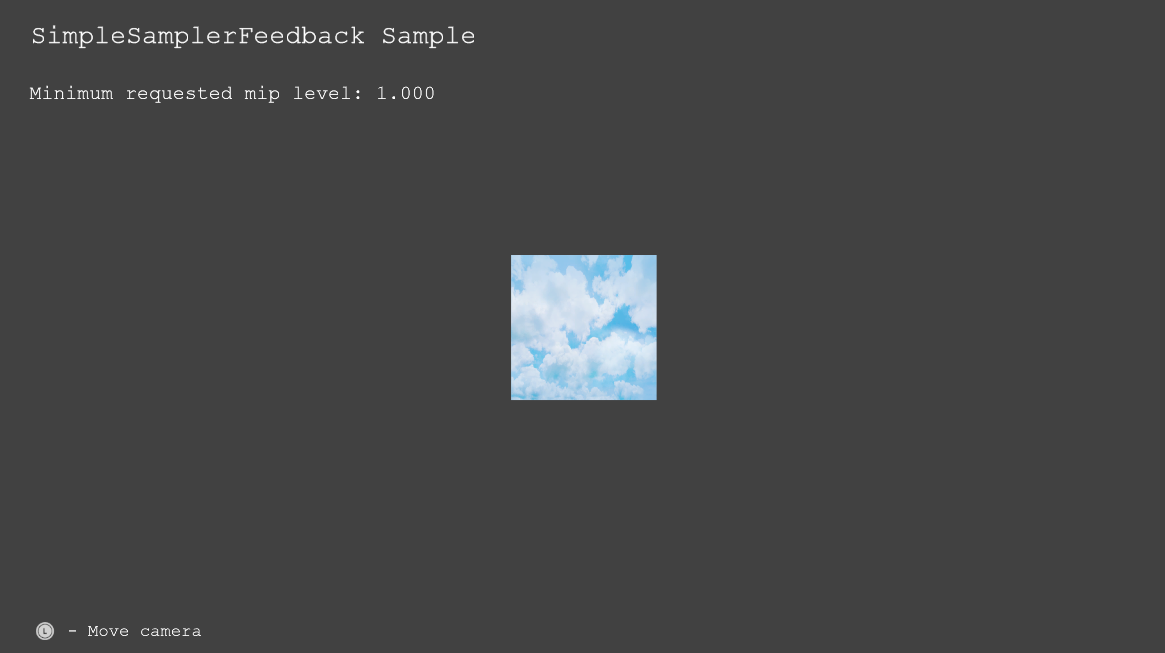
*This sample is compatible with the Microsoft Game Development Kit (June 2020)*

# Description

Sampler Feedback is a Direct3D feature for capturing and recording texture sampling information and locations. It can be used for things like texture-space shading and texture streaming. This sample only demonstrates a very simple implementation of sampler feedback.

The sample renders a textured quad with a camera that can move toward or away from the quad. As the camera moves closer to the quad, a higher detail mip, i.e. lower mip level, is used during rendering. Sampler feedback writes out this information to a MinMip feedback map.

Note: Sampler feedback is not supported on Xbox One, therefore this is a Scarlett only sample.



# Building the sample

This sample only supports Scarlett, so the active solution platform will be Gaming.Xbox.Scarlett.x64

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

# Using the sample

|  |  |
| --- | --- |
| Action | Gamepad |
| Move camera | Left thumb stick |
| Exit | View Button |

# Implementation notes

**Creation**

Two formats for feedback maps exist, MinMip and RegionUsed. This sample implements a MinMip feedback, i.e. it’s created using DXGI\_FORMAT\_SAMPLER\_FEEDBACK\_MIN\_MIP\_OPAQUE.

Sampler feedback is normally used with tiled resources. Therefore, the feedback map is usually sized as a fraction of the dimensions of its paired tiled texture, i.e. one texel per 64KB tile. In this very simple sample, we create a 1x1 feedback map, i.e. one feedback map value for the whole texture.

To bind the feedback map to a shader, and pair a regular texture to the feedback map, the API CreateSamplerFeedbackUnorderedAccessView is used.

**Scene rendering**

Before scene rendering, the feedback map has to be cleared. It cannot be cleared to the value zero, since that would mean that mip level 0 has been requested during scene rendering. Therefore, the sample clears the map to the value of -1, which indicates that no mip was requested.

Sampler feedback shader instructions are supported in shader model 6.5. This sample’s pixel shader uses the method WriteSamplerFeedback. The file pixelshader.hlsl also contains shader code to emulate sampler feedback, which might be useful on platforms that do not support sampler feedback.

**Read back**

To read the values on the CPU, the feedback map has to be transcoded using ResolveSubresourceRegion. This sample creates a readback texture which is used for the readback. On Scarlett, the values in the feedback map are 5.3 fixed point.

# Update history

12/05/2019 – Sample creation.

# 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/).