

SimpleDirectStorageCombo Sample

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

# Description

This sample shows several different ways to use DirectStorage on both the Console as well as the Desktop.

* SimpleLoad – Minimum interface to initialize DirectStorage, open a file, enqueue a request, and wait for completion.
* StatusBatch – Demonstrates how to create a batch of requests using a status array for notification.
* StatusFence – Demonstrates how to create a batch of requests using an ID3DFence for notification.
* MultipleQueues – Demonstrates how create multiple queues using different priority levels.
* Cancellation – Demonstrates how to cancel pending requests.
* RecommendedPattern – Demonstrates the recommended pattern for using DirectStorage to achieve maximum performance.
* Xbox Hardware Decompression – Demonstrates how to use the hardware zlib decompression when running on an Xbox Series X|S console.
* Xbox In Memory Hardware Decompression – Demonstrates how to use the hardware zlib decompression available on an Xbox Series X|S console to decompress data already in memory.
* Xbox Software Decompression – Demonstrates how to use software zlib decompression when running on an Xbox One family console.
* Desktop CPU Decompression – Demonstrates how to use the title supplied CPU decompression codec support with DirectStorage on Desktop.

# Building the sample

This sample supports the following platforms

* Gaming.Desktop.x64
  + Using the DirectStorage on PC API set.
* Gaming.Scarlett.xbox.x64
  + Using the Xbox DirectStorage implementation available on Xbox Series X|S consoles
* Gaming.XboxOne.xbox.x64
  + Using a provided software emulation layer that provides the functionality of the Xbox DirectStorage implementation but using the Win32 API set internally.

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

# Using the sample

The sample will automatically create a data file and then execute each of the mentioned sub-pieces.

# Implementation notes

All the implementations are contained in the SampleImplementations folder. They are heavily documented with details on each step taken.

For an example on how to use BCPack compression see the TextureCompression sample.

The zlib library (version 1.2.11) is subject to this license: <http://zlib.net/zlib_license.html>

# Update history

Initial release February 2022

Updated October 2022 to add Desktop CPU decompression

# 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/).