

SimpleCompute のサンプル

*このサンプルは、Microsoft ゲーム開発キットのプレビュー (2019 年 11 月) に対応しています。*

# 説明



SimpleCompute では、DirectX 12 で DirectCompute™ (Direct3D の計算シェーダー) を使用する方法を説明します。このサンプルでは、両方のグラフィック コマンド リストへ演算処理を渡す方法、D3D12\_COMMAND\_LIST\_TYPE\_COMPUTE インターフェイスを使用して非同期計算シェーダーの処理を渡す方法を実演します。計算シェーダーでマンデルブロット セットを計算することにより、テクスチャを更新します。

# サンプルのビルド

Xbox One の devkit を使用している場合は、アクティブなソリューション プラットフォームを Gaming.Xbox.XboxOne.x64 に設定します。

Project Scarlett を使用している場合は、アクティブなソリューション プラットフォームを Gaming.Xbox.Scarlett.x64 に設定します。

*詳細については、GDK ドキュメント*の「サンプルの実行」を*参照してください*。

# サンプルの使用

|  |  |
| --- | --- |
| 動作 | コントローラー |
| 非同期計算を切り替える | A ボタン |
| 視点を既定位置に戻す | Y ボタン |
| 視点をパンする | 左スティック |
| 視点を拡大する | 右スティック |
| 拡大速度を上げる | 右トリガー |
| 終了 | ビュー ボタン |
| メニュー | ヘルプを表示/非表示にする |

# 実装に関する注意事項

このサンプルは主に、簡単な計算シェーダーの作成、使用に慣れていただくためのものです。

* **CreateDeviceDependentResources:**コンパイルした計算シェーダーをロードし、各種 Direct3D レンダリング リソースを作成します。シェーダーは Visual Studio によってコンパイルされます。
* **Render**:サンプルが非同期計算を使用していない場合、計算シェーダーを実行した後で、この計算シェーダーの計算結果を必要とするドローコールを送信します。これにより、テクスチャを 1 フレームごとに更新します。
* **AsyncComputeProc**:サンプルが非同期計算を使用している場合、処理を開始するようスレッドに命令した時点で、計算シェーダーがそのスレッドから実行されます。レンダーは、非同期タスクの完了が通知されてから、そのタスクに依存するドローコールを実行します。

# プライバシーに関する声明

サンプルをコンパイルして実行すると、サンプルの使用状況を追跡するため、サンプル実行可能ファイルのファイル名が Microsoft に送信されます。このデータ収集を無効にするには、「Sample Usage Telemetry」とラベル付けされた Main.cpp 内のコードのブロックを削除します。

Microsoft のプライバシー方針の詳細については、「[Microsoft プライバシーに関する声明](https://privacy.microsoft.com/en-us/privacystatement/)」を参照してください。