

SimplePBR サンプル

# *このサンプルは、Windows 10 2018 年 10 月アップデートSDK（17763）と互換性があります。*

# */ 2016 年 8 月の Xbox One XDK と互換性があります。*

# 概要

このサンプルでは、DirectX 12 を使用した UWP および Xbox One XDK での物理ベースのレンダリング (PBR) を紹介します。このサンプルでは、次のパラメーターを使用して、フォワードレンダリングされた Disney スタイルの PBR をスタンドアロン シェーダーとして実装します：

1. アルベド：照明なしのベースの RGB カラー
2. 法線マップ： 非圧縮 3 チャンネル法線マップ (Y ポジティブ) と、

以下を指定する RMA マップ：

1. 粗さ: [0..1]、反射ハイライトのサイズと形状を表す正規分布。粗さは、Disney ペーパーに従って拡大縮小されます。
2. メタリック: (通常 0 または 1、ブレンド可能)、アルベドの屈折率、反射、拡散分布を制御します。
3. アンビエント オクルージョン：反射と拡散のコントリビューションを拡大縮小する値 [0..1]。

パラメーターは、定数のみまたはテクスチャのみ (ただし混在しない) として表すことができます。シェーダーでは、イメージ ベースの照明 (事前に計算された拡散マップと反射マップを使用) と指向性ライトがサポートされています。

PBR の詳細については、このドキュメントの最後にある「実装/参照」セクションを参照してください。



# サンプルをビルドする

SimplePBRUWP12 または SimplePBRXbox12 のソリューションをビルドして実行します。すべてのアセットが「Assets」サブディレクトリで共有され、共有コード (シェーダー ソースを含む) は「Shared」サブディレクトリにあります。シェーダーは Visual Studio 2015 プロジェクトによってコンパイルされているため、外部ビルドの手順は必要ありません。

# サンプルの使用

UWP では、マウス (マウス ホイール、右クリック) とキーボード (W、A、S、D の各キー) を使用して、カメラをシーン内で移動します。

ゲームパッドのカメラ コントロールは、UWP と Xbox One の両方で利用できます。カメラの向きと半径を変更するには、左右のスティックを使用します。

# 実装上の注意

PBREffect クラスはシェーダーの実装をラップします。シェーダーには、定数とテクスチャの 2 つの構成があります。定数構成は主にデバッグ用です。テクスチャ構成では、入力パラメーター (アルベド、粗さ、メタリック、AO) がテクスチャとして指定されます。

テクスチャ シェーダーを作成するには、EffectFlags 列挙型を使用します。

m\_effect = std::make\_unique<ATG::PBREffect>(device, EffectFlags::Texture, pipelineState);

テクスチャ パラメーターを設定するには、各テクスチャとサンプラーの記述子を渡すだけです。

m\_effect->SetSurfaceTextures(m\_descriptors->GetGpuHandle(AlbedoIndex),

m\_descriptors->GetGpuHandle(NormalIndex),

m\_descriptors->GetGpuHandle(RoughnessMetallicAOIndex),

commonStates->AnisotropicWrap());

シェーダーは Visual Studio プロジェクトの一部としてコンパイルされ、3 つのファイルに分割されます。

1. PBREffect\_VSConstant - 共有頂点シェーダー
2. PBREffect\_PSConstant - 定数パラメーター ピクセル シェーダー
3. PBREffect\_PSTextured - テクスチャ パラメーター ピクセル シェーダー

以下が含まれる 2 つの HLSL もあります。

1. PBREffect\_Math - BRDF などのための共有数学関数。
2. PBREffect\_Common - ルート署名、定数、および共有照明関数「PBR\_LightSurface」。

## 照明

PBREffect では、指向性照明とイメージ ベースの照明の両方がサポートされます。呼び出し元は、事前に計算された放射照度テクスチャ (拡散環境での照明用) と放射輝度テクスチャ (反射環境での照明用) を提供する必要があります。 テクスチャは HDR 形式の DDS キューブ マップにする必要があります。

呼び出し元は、放射輝度テクスチャ内の MIP レベルの数も指定する必要があります。PBR 用の事前に計算されたマップの生成に関する詳細については、[AMD Cubemapgen for physically based rendering](https://seblagarde.wordpress.com/2012/06/10/amd-cubemapgen-for-physically-based-rendering/)を参照してください。

m\_effect->SetIBLTextures(

m\_descriptors->GetGpuHandle(m\_radTexDescIndex),

m\_radianceTexture->*GetDesc*().*MipLevels*,

m\_descriptors->GetGpuHandle(m\_irrTexDescIndex),

m\_commonStates->AnisotropicClamp());

また、呼び出し元が SetLight\* メソッドを使用して指向性照明を指定することもできます。シェーダーは、指向性照明とイメージ照明をブレンドします。

## 参照

<https://www.allegorithmic.com/system/files/software/download/build/PBR_Guide_Vol.1.pdf>

<https://disney-animation.s3.amazonaws.com/library/s2012_pbs_disney_brdf_notes_v2.pdf>

<http://blog.selfshadow.com/publications/s2015-shading-course/>

<http://graphicrants.blogspot.com/2013/08/specular-brdf-reference.html>

<https://github.com/dariomanesku/cmftStudio>

## Xbox の UWP

Xbox One で DirectX 12 をサポートするには、UWP アプリのアプリ タイプを「ゲーム」に設定する必要があります。そうしないと、開発者コンソールでソフトウェア デバイス (WARP12) しか利用できません。これは、市販のコンソールではサポートされていません。開発中は、アプリ タイプを DevHome を使用して設定できます。このサンプルでは、既定ではこれを達成するための機能が制限されている [extendedResources](https://msdn.microsoft.com/ja-jp/library/windows/desktop/mt808808.aspx) を使用しますが、パッケージのアプリ タイプを「ゲーム」に設定した後に削除できます。この制限された機能を使用している場合、Windows ストアに送信されたアプリの検証が失敗します。

# 既知の問題点

* このサンプルのシェーダーではアルファ チャネルがサポートされません。

注: このサンプルの PBREffect とシェーダーは、アルベド テクスチャおよびオプションの放射テクスチャにあるアルファ チャネルのサポートを含めて、[DX11](https://github.com/Microsoft/DirectXTK/wiki/PBREffect) / [DX12](https://github.com/Microsoft/DirectXTK12/wiki/PBREffect) 用の DirectX Tool Kit に統合されています。

# 更新履歴

2016 年 10 月の初回リリース。[DirectX Tool Kit tone-mapper](https://github.com/Microsoft/DirectXTK12/wiki/PostProcess) を利用するために 2017 年 6 月に更新されました。

Xbox One S / Xbox One X での HDR 出力、および HDR10 出力が構成された Windows 10 Creators Update PC をサポートするために 2017 年 10 月に更新されました。

2018 年 2 月に、[glTF2](https://github.com/KhronosGroup/glTF) の仕様と一致させるため、RMA テクスチャのチャンネル順序が *metalness* は青、*roughness* は緑、*ambient occlusion* は赤に変更されました。

2019 年 1 月に、法線テクスチャの BC5 圧縮の使用をサポートするために、.z コンポーネントを再構築するように法線テクスチャのシェーダーが更新されました。

# プライバシーステートメント

サンプルをコンパイルして実行すると、サンプルの実行可能ファイルの名前がMicrosoftに送信され、サンプルの使用状況の追跡に役立ちます。このデータ収集を無効にするには、Main.cppの「Sample Usage Telemetry」というラベルの付いたコードブロックを削除します。

マイクロソフトのプライバシーポリシー全般に関する詳細については、 [Microsoftのプライバシーステートメント](https://privacy.microsoft.com/ja-jp/privacystatement/)をご参照ください。