

## Unityはじめるよ

#### ~オクルージョンカリングを試してみる~

統合開発環境を内蔵したゲームエンジン http://japan.unity3d.com/

※いろんな職業の方が見る資料なので説明を簡単にしてある部分があります。正確には本来の意味と違いますが上記理由のためです。ご了承ください。 この資料内の一部の画像、一部の文章はUnity公式サイトから引用しています。

本日の内容

・オクルージョンカリングとは

・オクルージョンカリングの使い方

© monolizm LLC

#### オクルージョンカリングとは

あるオブジェクトが他のオブジェクトの後ろに隠れて カメラから見えない場合は、どうせ見えないんだから そのオブジェクトの描画をやめて描画の負荷を軽くしよう、 というもの。

1人称視点や3人称視点などの

遠くまで見渡せるゲームの高速化に最適



負荷について

描画をしない分GPUの負荷を減らすことはできるが、 遮蔽判定をするためCPUの負荷が増える。

遮蔽判定をリアルタイムに行うのは負荷が大きすぎるため、 「セル」と呼ばれる立方体の範囲単位で事前に遮蔽判定の 計算をしておく。



青い立方体が「セル」

#### 遮蔽するものされるもの

#### 他のオブジェクトを遮蔽する可能性があるオブジェクトを Occuluder

他のオブジェクトから遮蔽される可能性があるオブジェクトを Occuludee

という。 インスペクタ右上から設定を行う。



#### どのオブジェクトにOcculuderとOcculudeeを セットすればいいの?

- ・プレイヤーなどの動くオブジェクト
   →OcculuderもOcculudeeも設定なし
- ・地形
   →OcculuderとOcculudeeの両方とも設定
- ・木や金網、標識、樽などの小さめのオブジェクト
   →Occuludeeのみ設定
- ・建物

→状況にもよるがOcculuderもOcculudeeとも設定

## オクルージョンカリングの使い方

オクルージョンカリングの使い方



#### 1.OcculuderとOcculudeeの設定 2.オクルージョンカリングの設定 3.Sceneビューで動作確認

<ol> <li>Inspector</li> </ol>	<u>코뷰</u> Lighting	🔀 Navigation	Occlusion	Services	=	
		Object Bak	e Visualization			
Set default para	meters					
The default parameters guarantee that any given scene computes fast and the occlusion culling results are good. As the parameters are always scene specific, better results will be achieved when fine tuning the parameters on a scene to scene basis. All the parameters are dependent on the unit scale of the scene and it is imperative that the unit scale parameter is set correctly before setting the default values.						
Smallest Occluder						
Smallest Hole	0.25					
Backface Thresho	d			01	100	

#### オクルージョンカリングウィンドウ。3つのタブで構成されている

### ■OculuderとOculudeeの設定

前説で説明した通り、 各オブジェクトに**Occuluder**と**Occuludee**の設定を行う。

**ポイント!** ShiftキーやCommant(Ctrl)キーを押しながらオブジェクトを 選択すると複数選択ができるので、複数選択していっぺんに 設定を行うとラク。 シーンに配置するオブジェクトをプレハブ化しておけば、 プレハブを変更するだけで良いのでもっとラク。

#### ■オクルージョンカリングの設定

#### 上部メニュー > Window > Occulusion Culling で、オクルージョンカリングの設定を行うウィンドウが表示 される。

<ol> <li>Inspector</li> </ol>	표는 Lighting	🔀 Navigation	Occlusion	Services	=
Set default para	imeters	Object Bak	e Visualization		
The default parameter are always scene spe are dependent on the values.	ers guarantee that any ecific, better results wi e unit scale of the sce	given scene compute II be achieved when f ne and it is imperative	s fast and the occlusio ine tuning the paramet that the unit scale par	n culling results are good. As the p iers on a scene to scene basis. All th rameter is set correctly before setting	arameters e parameters g the default
Smallest Occlude					
Smallest Hole	0.25				
Backface Thresho	ld				100

3つのタブで構成

Objectタブ

#### 選択中のオブジェクトのOcculuderとOcculudeeの 設定を行う。※インスペクタでも設定可能 フィルタでオブジェクトの絞り込みができる

Inspector	크는 Lighting	🔀 Navigation 🛛	Occlusion	Services	-=
		<b>Object</b> Bake	Visualization		
Scene Filter:					
🏹 All 💀 Rendere	ers 🕅 Occlusion Are				
🗟 Cube (4) (Mesh	Renderer)				
Occluder Static					
Occludee Static	<ul><li>✓</li></ul>				

Bakeタブ

オクルージョンカリングの使い方:手順2

#### オクルージョンカリング用のデータの生成用タブ。 Bakeボタンでオクルージョンカリング用のデータを生成。

<ol> <li>Inspector</li> </ol>	코는 Lightin	g 🛛	Navigation	Occlusion		*≡
Set default parar			Object Bake	e Visualization		
The default parameters guarantee that any given scene computes fast and the occlusion culling results are good. As the parameters are always scene specific, better results will be achieved when fine tuning the parameters on a scene to scene basis. All the parameters are dependent on the unit scale of the scene and it is imperative that the unit scale parameter is set correctly before setting the default values.						
	5					
Smallest Hole						
Backface Threshold						0 100
					Clear	Bake



ベイク結果のセルが青い線で表示される

プロパティー	機能	簡単に言うと	
Smallest Occluder	オクルージョンカリングを行う際に、他のオブジェクトを非表示にするために使用されるオブジェクトの最小サ イズです。このサイズよりも小さなオブジェクトは、それらによってオクルージョンされたオブジェクトをカリング することはありません。たとえば、値が5の場合では、5メートルより高いまたは広いすべてのオブジェクトは、 その背後に隠れたオブジェクトがカリングされることになります(レンダリングされず、レンダリング時間を節約 します)。このプロパティーのよい値を選ぶことによって、オクルージョンデータのためにオクルージョンの精度 とストレージのサイズのパランスをとることができます。	ここで指定したサイズより小さ いオブジェクトは遮蔽を行わな い。 小さくすればするほど計算が 多くなり負荷となる。	
Smallest Hole	この値は、カメラから見ることのできるジオメトリ間のもっとも小さな隙間を表します。値は、穴を通り抜けるこ とができたオブジェクトの直径です。シーンでは、カメラが見ることができる非常に小さな隙間がある場合、最 小の穴の値は隙間より小さくなければなりません。	オブジェクト同士の隙間からカ メラで覗き込んだ場合、隙間 の幅がここで指定した値より 小さければ、奥にあるオブジェ クトの描画は行わない。	
Backface Threshold	Unityのオクルージョンは、裏面をテストすることで不必要なディティールを削減し、データサイズを最適化しています。デフォルト値の100は、堅牢で、データセットから裏面を削除することはありません。値を5にした場合は、目に見える裏面の位置に基づいて、積極的にデータを省きます。一般的に、カメラ位置からはほとんどの裏面、例えば、地形モデルの裏側や、たどり着けない位置にある固体オブジェクトの内側などは、見えないためです。閾値が100より小さい場合、Unityは、データセットから完全にこれらの領域を削除します。これにより、オクルージョンのためのデータサイズが削減されます。	100より小さい値を設定すると、 裏面の描画を省略していく。 事前計算データが小さくなる。	

オクルージョンカリングの使い方:手順3

#### ■Sceneビューでの動作確認 Visualizationタブ

シーンビューでカリング状態の確認を行うための設定。



#### ハマりかけたこと

・カメラがセルの範囲に入っていないと
 オクルージョンカリングが機能しない。



オクルージョンカリングのテスト用に 上記のようにオブジェクトを配置したが これだとオクルージョンカリングされない。



地面を配置してカメラが「セル」の中に収まるようにしたら、ちゃんと機能した。 が、地面を作らなくても解決方法あり。↓

#### オクルージョンエリアで解決



ObjectタブのOcculusionAreasのCreateNewから オクルージョンカリングのセルを作成する範囲を 指定するオブジェクトを生成できる



薄緑色のエリアがオクルージョンエリア

#### ハマりかけたこと

・カメラ側の設定もお忘れなく!

🔻 🖆 🔽 Camera		💽 🌣,
Clear Flags	Skybox	\$
Background		1
Culling Mask	Everything	\$
Projection	Perspective	\$
Field of View	••	60
Clipping Planes	Near 0.3	
	Far 1000	
Viewport Rect	X 0 Y 0	
	W 1 H 1	
Depth	-1	
Rendering Path	Use Graphics Settings	+
Target Texture	None (Render Texture)	0
Occlusion Culling	$\checkmark$	
Allow HDR	✓	
Allow MSAA	$\checkmark$	
Allow Dynamic Resolution		
Target Display	Display 1	÷

デフォルトでチェックが入っているので大丈夫なはず

#### オクルージョンエリアについて

前ページで説明したオクルージョンエリア。 これはコンポーネントなので、普通にインスペクタの AddComponentボタンからも追加可能。

ちなみに、このコンポーネントを使うことで、 指定した範囲は「セル」が細かく、それ以外は大まかに してくれるので、うまく使えば処理速度を稼げる。



中心のCubeが多い部分にオクルージョンエリアを指定。 中心はセルが細かいが、周りは大まかになっている。

オクルージョンポータルについて

オクルージョンポータルは、 動的にオクルージョンカリングのオンオフを切り替える機能。 ドアが閉まっている屋内に入る・出るシーンで有効。

今回は取り上げないけど、 ゲームによっては利用シーンが多そう。

テラシュールブログさんで詳細を解説されていました。 <u>http://tsubakit1.hateblo.jp/entry/2018/01/07/021622</u>

#### 体験してみよう!

#### まとめ

基本的な使い方は簡単。 1人称視点や3人称視点のような遠くを見渡せる ゲームを作る時は、オクルージョンカリングと LODをうまく使って、いかに描画を減らすかが鍵。

地形や建物を作る際に、なるべく遮蔽しやすい・さ れやすいマップにする工夫も必要。

また、モバイルの場合CPU側の負荷も考慮しない といけないので、物理演算を多用するようなゲーム の場合は、検証が必要。

# ご清聴ありがとうございました

© monolizm LLC