

シリコンスタジオ事例紹介

物理ベース時代のライトマップベイク奮闘記







本講演でお話すること

■ DCCツールでライトマップをベイクしたときの奮闘記

- 日頃はミドルウェアを作る1人のエンジニア
- High Dynamic Rangeなライトマップ
 - 物理ベースレンダリング

1人で扱えるワークフロー
 ちょっと扱いづらい





ライトマップを利用したMuseumデモ

物理ベースなリアルタイムレンダリングデモ GDC 2014, CEDEC 2014にて公開







動画



- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
- 4. クオリティアップ
- 5. まとめ







- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
- 4. クオリティアップ
- 5. まとめ





LightMaps







+AlbedoMaps (+Directional Light)







+Image-Based Lighting







+Local Reflection and Fog







ライトマップの効果

- Global Illumination
 - 屋内に差し込む直射日光
 - 時間をかけて計算した、正確な値
- Image Based LightingのCubemap撮影











- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
- 4. クオリティアップ
- 5. まとめ







■ 3ds Max + V-Rayでベイク

■ デザイナの作業環境が3ds Max

- V-Rayは物理ベースレンダリングに対応
- 描画エンジンの開発と並行して作業





簡単そう?

3ds MaxでライトマップのUV展開をして、ベイクしてるだけ パッと聞いた感じ簡単そう

そんなことはなかった





ワークフロー







ワークフローのポイント







Flatironの導入







複数オブジェクトを1枚にベイクしたい

- モデリングと ベイクの分業のため
 モデリング
 オブジェクトごとに 編集内容&履歴を管理したい
 ベイク
 - ・ 描画エンジンのために、
 解像度、枚数などを調整したい





3ds Maxの問題

複数オブジェクトを1枚にベイクできない オブジェクトごとにベイクされてしまう





■ UVの自動展開機能が扱いづらい

- パラメータの試行錯誤





Flatiron

- ベイク補助プラグイン
 複数オブジェクトを1枚にベイク
 簡易な設定によるUVの自動展開
 解像度
 - パディング

















オブジェクトを複数選択してUV展開

オブジェクトを複数選択して、 UV展開する作業を繰り返す

UVを確認して、
 各オブジェクトと
 ライトマップの対応を決定









UV展開の設定が保存されない



パラメータを指定し、 UV展開

再選択時に、 パラメータが表示されない

- ベイクする時に必要な情報が保存されない
 - 解像度、パディング
 - ライトマップとオブジェクトの対応
- 最新のFlatironでは改善された





名前付き選択セット

■ 3ds Maxの機能

ビューポート上で選択するオブジェクトの集合を
 名前のリストで管理







選択セットと共通パラメータによる管理



- 解像度、パディング
 - UV展開時のパラメータはなるべく共通化





fbxファイルとマテリアル変換







自動ベイク







まとめてベイクしたい

- ライトマップが45枚
 1枚ベイクに30分以上
- 22.5時間以上

自動化したい
 Flatironは自動ベイク機能なし







MAXScriptによる自動化

- Flatrionのプリセットファイル
 - UV展開のパラメータ
 - ライトマップのフォーマット
- csvファイル
 - 名前付き選択セット→ライトマップ



■ 完全な自動化には至らず

- ネットワークレンダリング
- ジョブを投げるダイアログ





ワークフローのポイント







ネットワークレンダリング

- Backburner
 - 3ds Maxに付属
 - ジョブ管理ツール
- サーバ3台

■ 8時間ぐらいで終わる - 一晩で終わる

am 9 Backburner Queue Monitor - is0477		
<u>M</u> anager Jobs <u>S</u> ervers <u>V</u> iew <u>H</u> elp		
🔻 🔻 🕻 🗙 🐐 🗏 🗏		
Show All		I
Job	🗆 Order 🛛 🔄 Priority 🗔 State 🔅	I,
test_rendering	1 50 Active	



Show All					
All Servers					
Ser	ver 🗆	Status 🗆	Current Job	Last Message	
8	is0304	Absent	None	N/A	
Ш,	is0477	Absent	None	N/A	
	pc13052	Absent	None	N/A	
	pc13053	Absent	None	N/A	
Ш,	pc13054	Busy	test_rendering	None	
.	pc13056	Idle	None	None	
8	pc13057	Idle	None	None	

サーバー覧



Silicon Studio ©Silicon Studio Corp., all rights reserved

外部参照ファイルでのUV展開&ベイク

 OKボタンをクリックする度に アセットがzip圧縮&保存
 アセットの総量が1.16GB

 外部参照ファイルを ネットワーク共有
 UV展開&ベイク可能







描画エンジンへの入力







描画エンジンへの入力ファイル

■ fbxファイル

- オブジェクトごとのマテリアル分割
- ディフューズテクスチャ
- ジオメトリ、ライト
- ライトマップ
 - 45枚のddsファイル
- csvファイル
 - 各オブジェクトとライトマップの対応



4	Wagon_002	chairs_and_ladders.dds
5	Wagon_001	chairs_and_ladders.dds
6	Desk_A_006	desks.dds
7	Desk_B_003	desks.dds
8	Desk_A_007	desks.dds
9	Desk_A_008	desks.dds
10	Desk_A_009	desks.dds
11	Desk_A_0010	desks.dds
4.0		1 1 11



各オブジェクトとライトマップの対応

- ベイク時に、Flatironがシェルマテリアルを付加
 - 元マテリアルを保持しつつ、ライトマップをオブジェクトへ追加
- MAXScriptで抽出







ベイク手順

- 1. ベイクモデルを3ds Maxでロード
- 2.
 UV展開
 1.
 複数オブジェクトを選択セットに登録

 2.
 各選択セットをFlatironでUV展開
 - パラメータはあとから確認できないのでメモ
 - . UVを確認してダメなら2に戻る
- 3. UV展開後のモデルをfbxでエクスポート
- 4. モデルをV-Rayマテリアルに変換するMAXScriptを実行
- 5. 自動ベイク
 1. ベイクするMAXScript用設定ファイルを用意
 UV展開のパラメータ、ライトマップのフォーマット、
 選択セットごとの出カライトマップファイル名

 2. V-Rayのパラメータを設定
 3. ベイクするMAXScriptを実行
 4. ネットワークレンダリングのためのダイアログを、ライトマップの枚数分クリック
 5. 8時間待つ
 - 6. ライトマップを確認して、ダメなら8に戻る
- 6. 描画エンジンへの入力 1. 2.

ベイク済みモデルからオブジェクトとライトマップの対応表を抽出するMAXScriptを実行 モデルファイル(.fbx)、ライトマップ(.dds)、対応表(.csv)を描画エンジンに入力

7. 描画エンジンの結果を確認





ワークフローのまとめ

∎ つぎはぎ

- DCCツールやプラグインの合わせ技
- MAXScriptでなるべく自動化
- 1人でも扱えるが・・
 - 面倒な作業
 - 1枚1枚チェックしないと心配
 - 繰り返すのはつらい






- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
- 4. クオリティアップ
- 5. まとめ





物理的に正しいライトマップとは?

実世界に則した値がベイクされたもの

- V-Rayは物理ベースレンダリング
 - 実世界に則した計算
 - 描画エンジンも

■実世界に則した計算を行うには、 物理的に正しいライトマップが必要







- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
 - i. ベイク環境の確認
 - ii. 実世界の光源環境を再現するパラメータ
- 4. クオリティアップ
- 5. まとめ







- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
 - i. ベイク環境の確認
- ii. 実世界の光源環境を再現するパラメータ
- 4. クオリティアップ
- 5. まとめ





Museumデモの光源設定

- directional light × 1
 - 太陽の直接光を再現
 - パラメータ
 - ・ マルチプライヤ (強さのスカラー値)
 - カラー
- sky light
 - 天空光を再現
 - 3ds Maxにおける環境マップ
 - HDRスカイマップ









ベイク環境で確認すべきこと

- ライトマップにベイクされている値の確認
 - マニュアルには詳しく書かれていない
- 光源の入力と出力の関係を確認
 - directional light: マルチプライヤ、カラー
 - sky light: スカイマップ
- 光源の設定を行うために確認する必要がある





ライトマップにベイクされるべき値

Global Illumination

- 間接光による値









"間接光"はどんな値なのか

 照度であると考えられる ■ 間接光による照度が **Global Illumination**



■ V-Rayで間接光の照度は出力可能か?





V-Rayがライトマップに出力する値

- レンダリング設定で選択可能
 - VRayGlobalIllumination
 - VRayRawLighting
 - VRayRawGlobalIllumination
 - …などなど
- VRayRawGlobalIlluminationのようだが…
 - マニュアルでは"Global Illuminationの値"
- 本当にVRayRawGlobalIlluminationで正解なのか?





簡易なシーンでレンダリング



- VRayRawGlobalIlluminationが間接光のようだが
 - 照度かどうかはわからない
- もう少し詳しく調べる必要がある





VRayRawGlobalIlluminationはどんな値か?

- 間接光を含むV-Ray向けシーンをベイクして、 実際に値を確認
 - 簡単な数式で、照度を推定できるようにする
 - ベイク結果の確認と比較を簡単にするため

■ 間接光を発するsky lightによる確認シーンを用意





sky lightのV-Rayシーン

- スカイマップは一様に(1.0, 1.0, 1.0)
- 照度
 - 一様な輝度の半球積分

 $\int_{\Omega} L_i(x,\vec{\omega})(\vec{\omega}\cdot\vec{n})d\vec{\omega}$



 $= (スカイマップのRGB) * \int_{\Omega} (\vec{\omega} \cdot \vec{n}) d\vec{\omega}$ $= (スカイマップのRGB) * \pi$ $= (\pi, \pi, \pi)$







sky lightのシーンをベイクした結果

- ライトマップを確認
 - ベイクされた値は(1.0, 1.0, 1.0)



■ スカイマップの値を変更しても、^{照度}がベイクされる - (0.5, 0.5, 0.5), (2.0, 0.0, 0.0)など

■ sky lightによって、
$$\frac{照度}{\pi}$$
が出力されることを確認





ライトマップにベイクされる値

- VRayRawGlobalIllumination
 - V-Rayの出力
 - 間接光による値

- sky lightによって、 $\frac{照度}{\pi}$ が出力される

ライトマップにベイクされている値の確認ができた





各光源の入力と出力の関係は?

- sky light(は確認済み
- directional lightに関しても確認する必要がある
 - 簡単な数式で、照度を推定できるようにする
 - V-Rayの出力結果の確認を簡単にするため





directional lightのV-Rayシーン

- マルチプライヤ: 1.0
- カラー: (1.0, 1.0, 1.0)
- 照度
 - punctual lightなのでπ乗算
 - $-\pi * マルチプライヤ * カラー = (π, π, π)$

■ 予想

- sky lightと同じ値が出力されるはず

$$- \frac{照 g}{\pi} = (1.0, 1.0, 1.0)$$

Silicon Studio ©Silicon Studio Corp., all rights reserved.



directional light



directional lightのシーンの出力をした結果

- V-Rayの出力を確認
 - 出力された値は(1.0, 1.0, 1.0)
 - 想定通り

■ マルチプライヤとカラーの値を変えても、 $\frac{照度}{\pi}$ が出力された

■ directional lightの入力と出力の関係が確認できた





全光源の入力と出力の関係を確認した

- 各光源の入力と出力の関係を確認できた
 - sky light
 - directional light
- 光源の設定を行う準備ができた







- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
 - i. ベイク環境の確認
 - ii. 実世界の光源環境を再現するパラメータ
- 4. クオリティアップ
- 5. まとめ





スカイマップの修正

- スカイマップの太陽 は取り除く
 - センサのレンジ外
 - 不正確な値



■ 太陽は直接光をdirectional lightで再現





実世界の光源環境の再現





directional light 太陽の直接光を再現

sky light 天空光を再現

実世界での相対強度を再現する





実世界を再現する光源のパラメータ

- 太陽の直接光と天空光による照度の比を再現
- directional lightのマルチプライヤを設定
 - デザイナが選んだスカイマップから







現実的な値域内の、マルチプライヤ

- スカイマップから、正確に実世界を再現するのは困難
 - 撮影日時、場所、大気の状態は不明
 - 太陽の直接光の強さも変化するはず
- きっちり計算するのは難しい

現実的な値域を計算し、
 それを満たすようにマルチプライヤを調整する
 実世界の昭度の比









- 太陽の直接光による法線照度
 - 32000 \sim 130000 lux



- 天空光による照度
 - 10000 \sim 25000 lux

※各照度は[Hiscocks2011]より参照





Silicon Studio ©Silicon Studio Corp., all rights reserved.

実世界の照度の比

- 直接光と天空光の照度の比の値域
 - 直接光の強さによって、両照度は変化
 - 最小値同士、最大値同士で照度の比を計算

■実世界の照度の比の値域を満たすように、 マルチプライヤを決める







- sky light
 - sky lightだけベイク
 - π * ライトマップの値
 - ライトマップは $\frac{照度}{\pi}$









マルチプライヤの決定

■ 実世界の照度の比の値域内で、マルチプライヤを調整

 $3.2 \leq \frac{\text{directional lightの照度}}{\text{sky lightの照度}} \leq 5.2$

■ 描画エンジンの結果を見つつ決定





正しいライトマップのまとめ

- ベイク環境の確認
 - ライトマップにベイクされている値
 - 光源の入力と出力の関係
- 実世界を光源環境を再現するパラメータ
 - 光源間の相対強度を、実世界に即すように

物理ベースレンダリングを なるべく正しく行うための確認と設定







- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
- 4. クオリティアップ
- 5. まとめ





ノイズを減らしたい

デフォルト設定でGlobal Illuminationをベイクすると、
 結構ノイズがのる

- V-Rayのパラメータを変更し、いろいろ試したが
 - ノイズの閾値を下げるのが1番効果的
 - 基本的にV-Rayのマニュアル通り





サンプラの設定

- Min Samples: 16
 - 最低サンプリング量
- Noise threshold: 0.001
 - ノイズを減らすために最小値に
 - 全ベイクで8時間で済んだ
- Global subdivs multiplier: 8.0
 - シーン全体のサンプリング数に影響





Global Illuminationの設定

- Primary bounces
 - Irradiance map
 特定箇所で計算し、補間
 - プリセット: High
 - サンプリング数 →
 - 補間サンプル数
- Secondary bounces ---->
 - Brute force

モンテカルロ・パストレーシング





Irradiance Mapのサンプリング数









- 間接光だけで照らされている箇所
- サンプリング数はベイク時間に直結





Irradiance Mapのサンプリング数





50

200

- ノイズがのりやすいライトマップだけ上げる
 - 間接光だけで照らされている箇所
 - サンプリング数はベイク時間に直結





Irradiance Mapの補間サンプル数



20

200

■ ノイズをぼやかして、ごまかしたい時に使う

- ベイク時間にあまり影響しない





Irradiance Mapの補間サンプル数





20



■ ノイズをぼやかして、ごまかしたい時に使う

- ベイク時間にあまり影響しない




補間サンプル数を上げ過ぎると





■ のっぺりする ■ 接地感がなくなる





Silicon Studio ©Silicon Studio Corp., all rights reserved.



補間サンプル数を上げ過ぎると

 ポリゴンの境目が 目立つ



■ のっぺりする
■ 接地感がなくなる



多用しないこと





よくやっていたベイク手順

- 同じ設定で全部ベイク
 - サンプラ, Global Illumination

ノイズが目立つライトマップがあったら

- Irradiance mapの補間サンプルを上げてぼやかす
- Irradiance mapのサンプリング数を上げる





きれいなライトマップの完成









- 1. ライトマップの効果
- 2. ワークフローの構築
- 3. 正しいライトマップを目指して
- 4. クオリティアップ
- 5. まとめ







- 1. ライトマップの効果
 - 高品質なGlobal Illumination
- 2. ワークフローの構築
 - 3ds MaxとV-Rayによるベイク、FlatironによるUV展開
- 3. 正しいライトマップを目指して
 - 物理ベースレンダリングをなるべく正しく行うための確認と設定
- 4. クオリティアップ

– ノイズ低減







■ 改善点

- さらなる自動化
- 不安定

単純なライトモデルなら、細かい確認が可能 最終的なクオリティは高い

ご静聴ありがとうございました







[Hiscocks2011] Peter D.Hiscocks, "Measuring Light" http://www.ee.ryerson.ca/~phiscock/astronomy/light-pollution/photometry.pdf



