

レイトレ合宿6

Pentan(@c5h12)

_ershy

_ershy

(フラッターシャイ)

パストレーシングレンダラー



がちょっとレンダリングできる



- glTF標準の pbrMetallicRoughness に対応しているような気がする。
- Specification の Appendix B: BRDF Implementation に説明がある反射モデルをそのまま実装したつもり。
- 反射しか書いていないのと時間の兼ね合いでアルファによる透過の扱いは保留。
- アニメーションなどは未対応。
- 画像の読み書き(要はstb)が組み込めていないので画像テクスチャも未対応



- 公式のBlenderのエクスポートアドオンがあるので自前のデータエクスポーターを作る手間がいらない！😄
- ただそのBlenderのエクスポーターには少しバグがあるかもしれない
- カメラを書き出す時にY-upに直すオプションを入れるとカメラだけ直っていない。
- Z-upのまま書き出すとトランスフォームがおかしいっぽいのでBlender上で全部適用してから出すと安心？
- extraというフィールドには何を入れてもいいらしいので、今回は背景色の指定を入れている。（これは人力で後修正）



Swift

リベンジ

Swift

- クロスプラットフォーム (macOS, Linux)
- Dispatchによる並列処理
- Swift Package Managerによるパッケージ管理
- swift build で手軽にパッケージをビルド
- テストも付いている
- swift package -generate-xcodeproj でXcodeのプロジェクトも自動生成

Swift...

- Windowsは...
- Linuxの安定版にはDispatchの静的ライブラリが含まれていないので静的リンクするためには自分でビルドする必要がある
- パッケージの構成はディレクトリやファイルの名前ベースなのでちょっと融通がきかない。Sources直下はファイルとディレクトリを混在できないので1つディレクトリを作ってモジュール扱いにするとか
- パッケージの依存関係を記述する素敵な仕組みはあるのだがGitで管理しないといけないらしいとか

Swift...

- `for i in array { ... }` は便利だがイテレーター生成に結構時間がかかる処理ある？様子
- 苦し紛れに `for i in 0..array.count {...}` を多用しています。

まとめ

- レンダリングが終わっていたら嬉しい。
- 一般的なファイルをレンダリングできるのはシーン製作が
気楽でいい。

ありがとうございました