

ゼロから始める ベイク生活

「ベイク」への理解を深めるための
アーティストフレンドリーな講座

Clemens Berger・Florian Gantzert



トピックス

1. ベイクって？
2. 描画プリミティブっておいしい？
3. スペースって？
4. リアルタイムライトにおける描画？
5. ノーマルマップの仕組みって？
6. ノーマルマップを使っているいろいろ
7. そして、まとめ

ベイクって？



ベイクって？

ベイクは破壊的な操作で
情報を都合のいい別の情報に焼きこむ
プロセス

何が都合がいいのか見極める必要がある

ベイクって？

- なぜ人はベイクするのか
- ワークフローの最適化のため
- コンテンツの最適化のため

ベイクって？

- なぜ人はベイクするのか
 - ワークフロー最適化

**ワークステップがいい感じにつながるように
情報を変換することが目標**

- よりシンプルなデータの渡し
- 前処理されているデータの渡し

ベイクって？

- なぜ人はベイクするのか
 - ワークフロー最適化

アニメーション・シミュレーションベイク

- <http://www.alembic.io/>
- <https://www.sidefx.com/community/fx-adventures-in-uncharted-4-a-thiefs-end/>

ベイクって？

- なぜ人はベイクするのか
 - ワークフロー最適化
 - コンテンツ最適化
- **処理時間を短縮する**
- **メモリー量を減らす**

の二つがよくある目的

リアルタイムコンテンツにおいて
メモリーの許す範囲、
処理時間短縮を優先する場合が多い

ベイクって？

- なぜ人はベイクするのか
 - ワークフロー最適化
 - コンテンツ最適化
 - トレードオフ

最適化はトレードオフ

ベイクって？

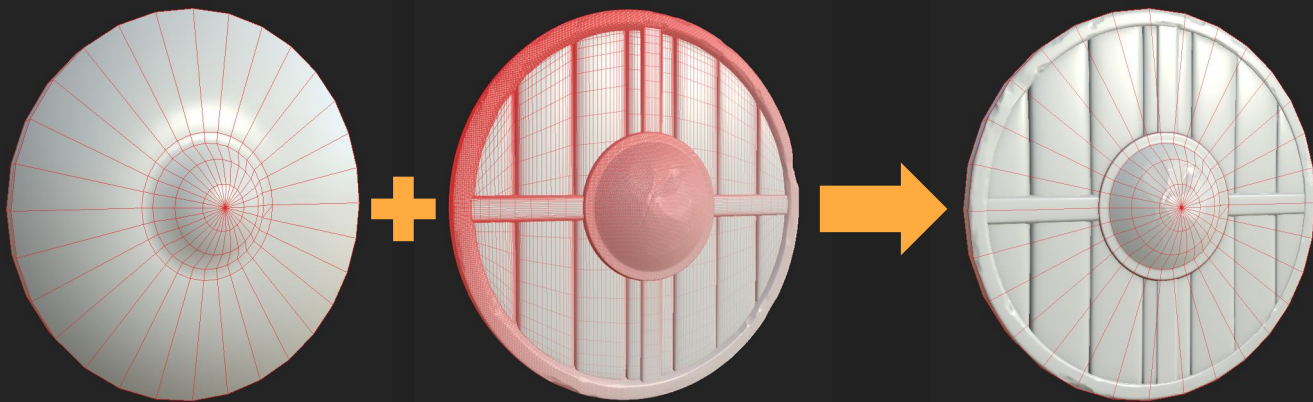
- なぜ人はベイクするのか
 - ワークフロー最適化
 - コンテンツ最適化
 - トレードオフ
- ボトルネックを洗い出したうえで適切な対応方法を見つける必要がある
- ボトルネックが時代とともに変わったりする
- ベイクステップが増える
- データも増える
- 自分の都合を洗い出す必要がある

ベイクって？

- なぜ人はベイクするのか
 - ワークフロー最適化
 - コンテンツ最適化
 - トレードオフ
 - テクスチャベイク

- テクスチャベイク

リアルタイムコンテンツでよく使用され、
このセミナーのメインテーマである！



ベイクって？

- なぜ人はベイクするのか
 - ワークフロー最適化
 - コンテンツ最適化
 - トレードオフ
 - テクスチャベイク

テクスチャベイクは
かならず**空間**に対して行う

空間というのは意外とややこしいもので
テクスチャベイクの説明に入る前に
空間について考えよう

ベイクって？

- なぜ人はベイクするのか
 - ワークフロー最適化
 - コンテンツ最適化
 - トレードオフ
 - テクスチャベイク

その前に
少しでも**描画**に触れよう...

描画プリミティブって おいしい？

- 描画プリミティブ

(GPUによる)描画されるものを
ここで**描画プリミティブ**と呼ぼう



描画プリミティブって おいしい？

- 描画プリミティブ

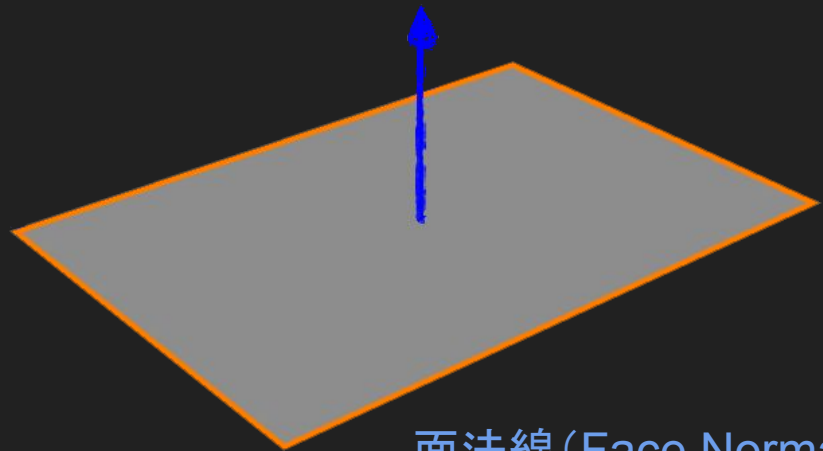
描画するのに、平坦な面が使われることが多い

- 特に最近のリアルタイムコンテンツでは
- ボクセルは一旦忘れよう
- あと、曲面も

描画プリミティブって おいしい？

- 描画プリミティブ

表面の特徴



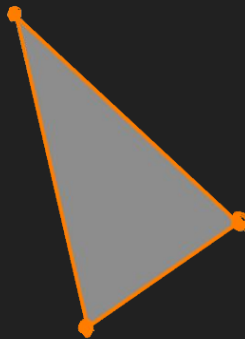
面法線 (Face Normal)

- 表面には向きがある
- 表面の向き先は面法線と呼ぶ
- 表面に別情報を写像することができる

描画プリミティブって おいしい？

- 描画プリミティブ

表面を三角形として描画することが多い



(三角形はかならずぺったんこだから)

描画プリミティブって おいしい？

- 描画プリミティブ

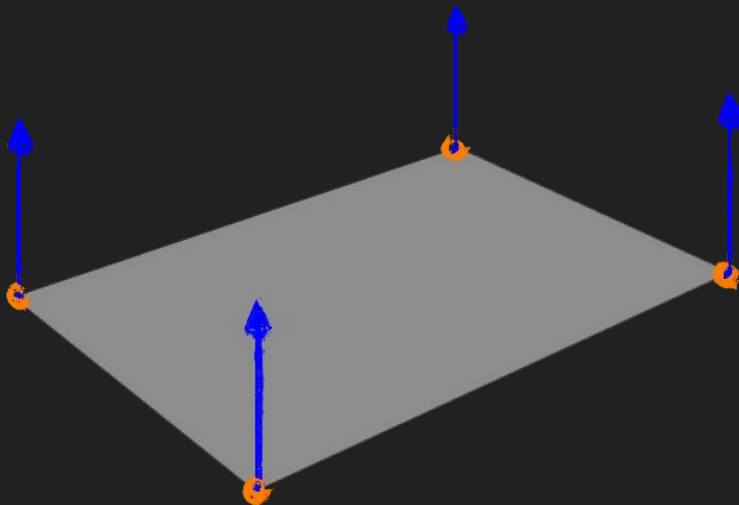
ここまではいいのだが...

GPUに投げるものは
表面そのものではなく
その**表面を構築する点**

描画プリミティブって おいしい？

- 描画プリミティブ

それによって、**頂点法線**が誕生する



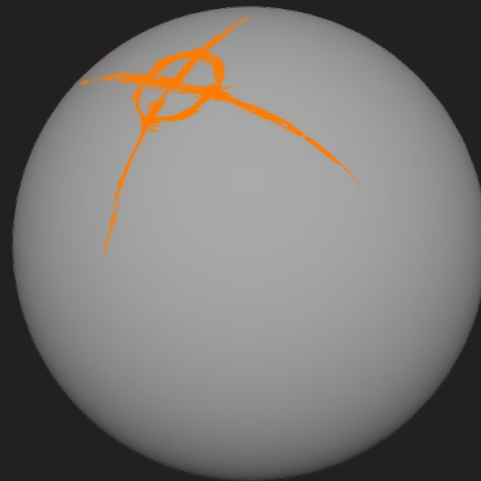
頂点法線 (Vertex Normal)

(そして、描画に関して、面法線が闇に消える)

空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間
- タンジェント空間

タンジェント空間



描画プリミティブって おいしい？

- 描画プリミティブ

じゃ、表面法線って

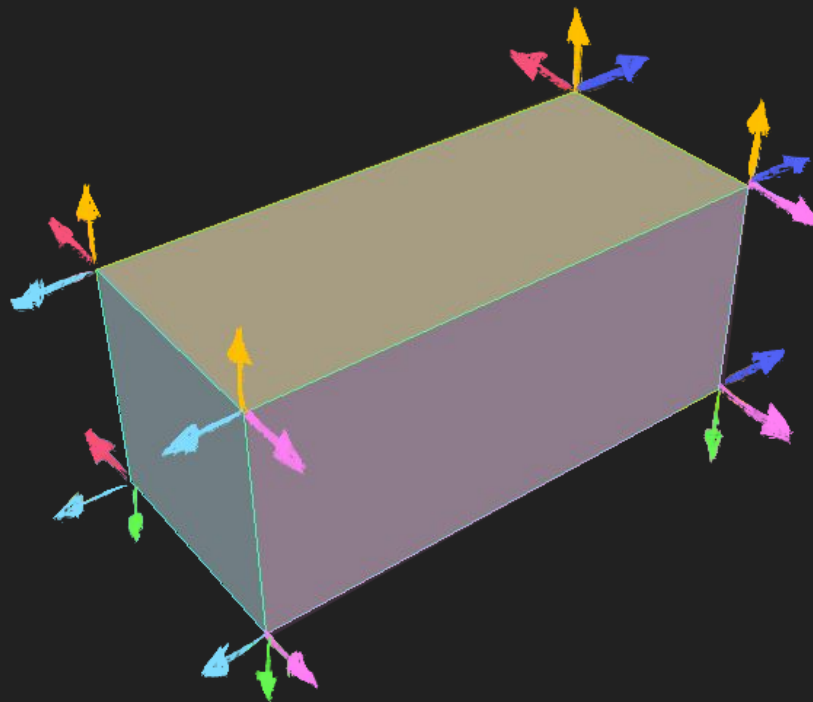
どうやって頂点法線になるんだ？



描画プリミティブって おいしい？

- 描画プリミティブ

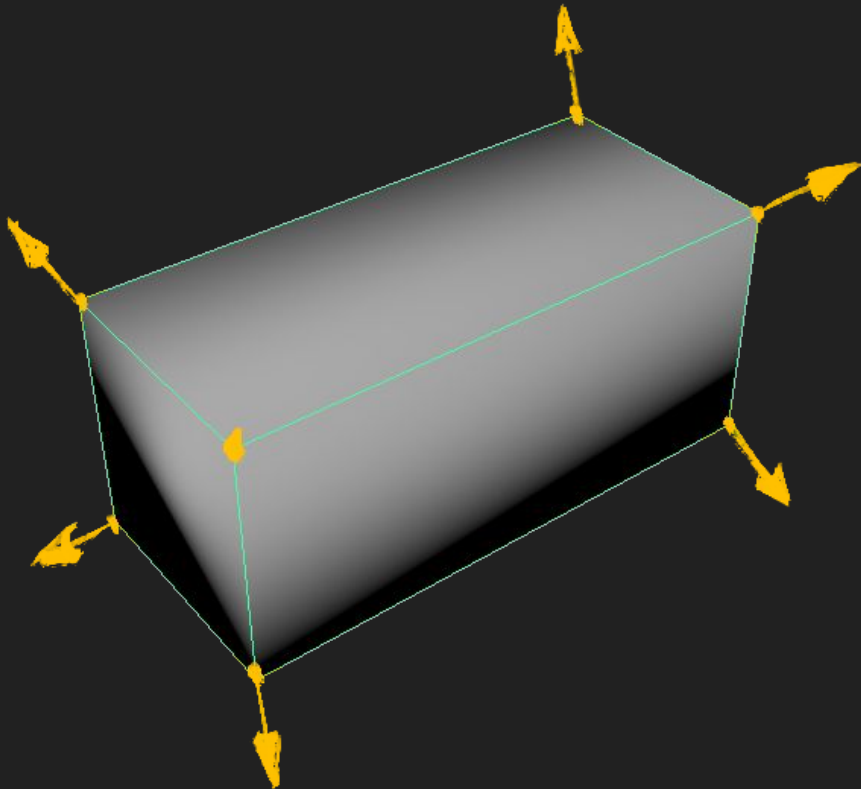
Hard Edge



描画プリミティブって おいしい？

- 描画プリミティブ

Soft Edge



描画プリミティブって おいしい？

- 描画プリミティブ

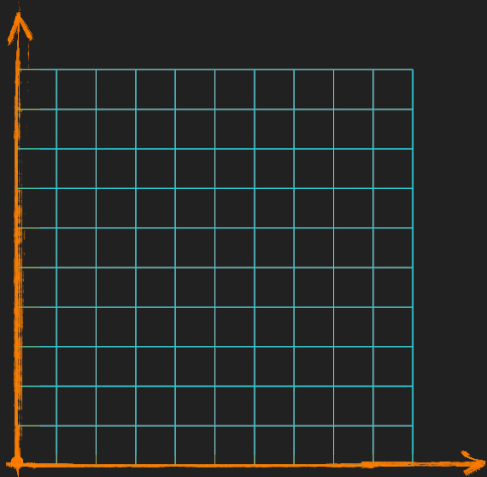
だが、問題はそこだけではない
特殊な空間も生まれてしまう

空間って？

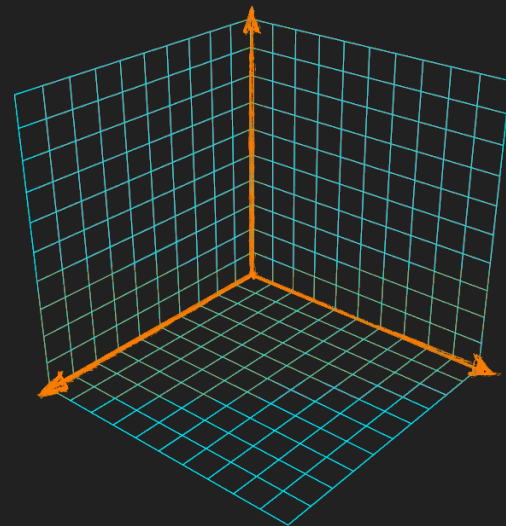
- 位置、向きと空間
- 空間は位置と向きに意味を与えるもの
- 言い換えると、あるもの**位置と向き**は
かならずある**空間の中でしか意味を持たない**
- 2D・3Dアーティストの作業は
ユークリッド空間で行われることが多い
 - これから出てくる空間はすべてユークリッド空間

空間って？

- 位置、向きと空間
 - ユークリッド空間
- ユークリッド空間には
原点と回転軸がかならず存在する
- すべての位置と向きはそれをもとに述べる



2D空間

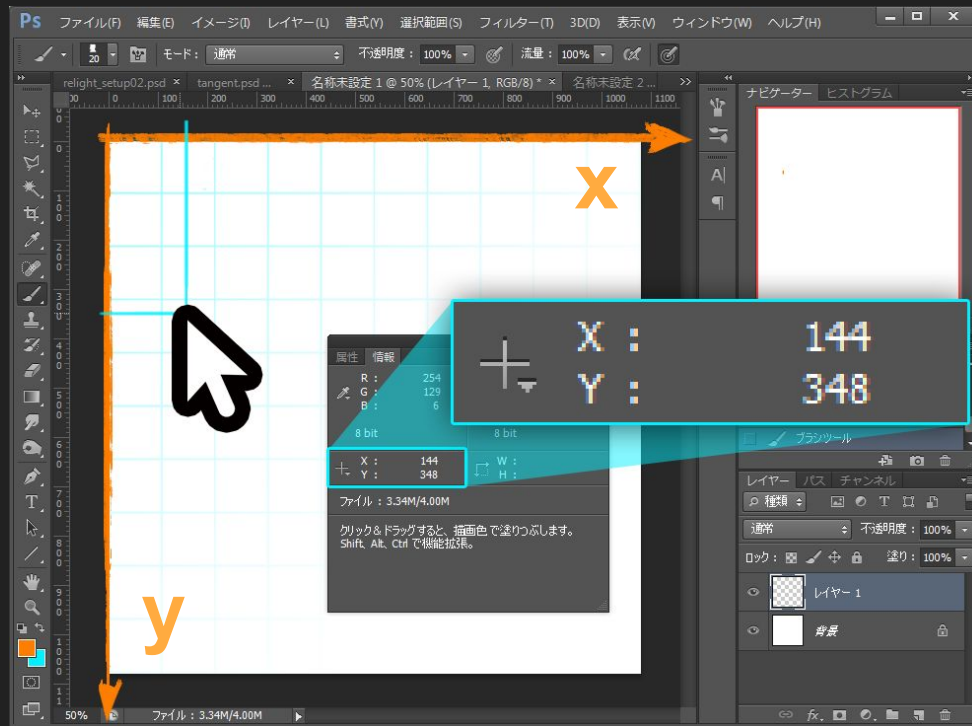


3D空間

空間って？

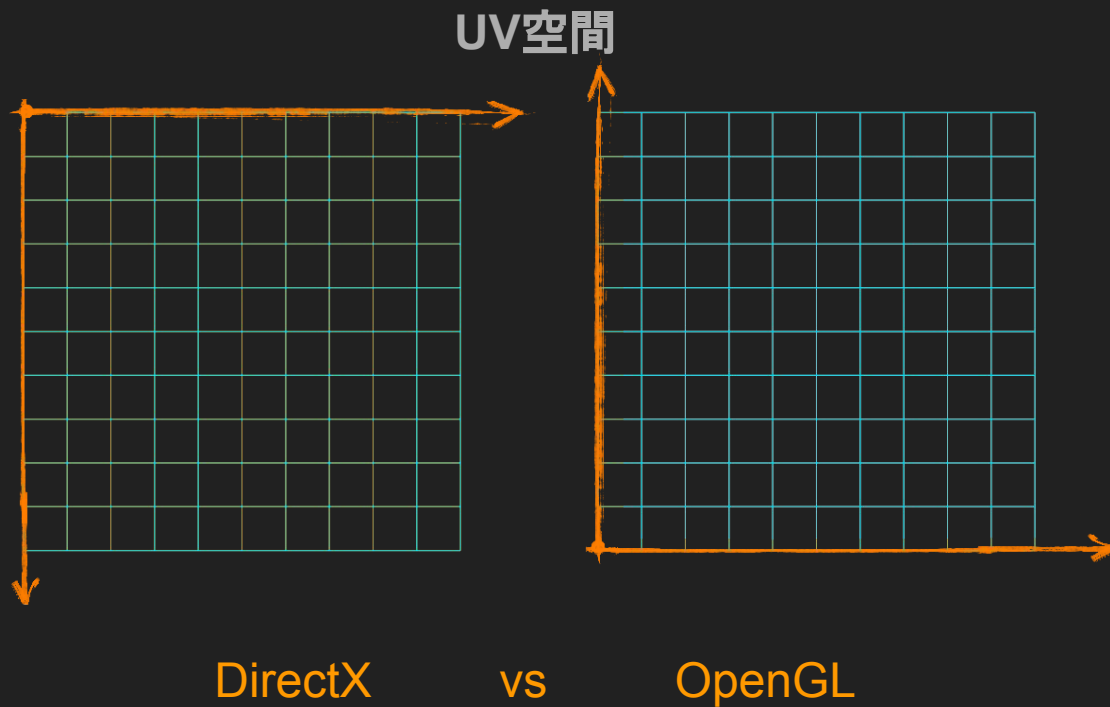
「フォトショップ空間」

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」



空間って？

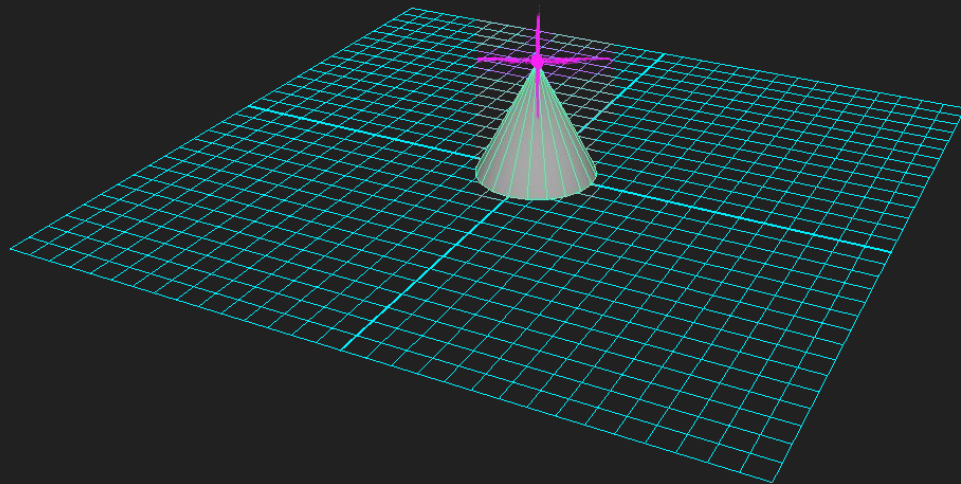
- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間



空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間

オブジェクトとワールド空間



オブジェクト空間

ワールド空間

$x\ 0\ y\ 1\ z\ 0$

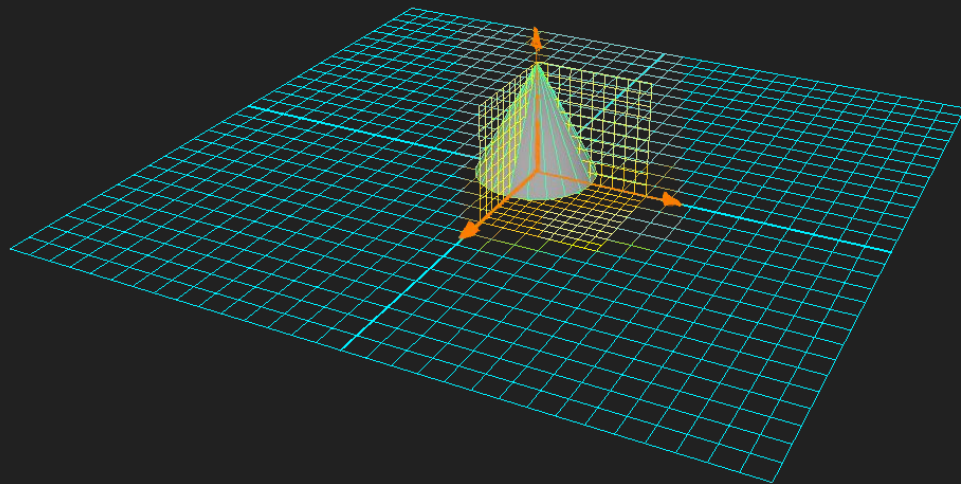


空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間

オブジェクトとワールド空間

空間が位置している



オブジェクト空間

$x\ 0\ y\ 1\ z\ 0$

ワールド空間

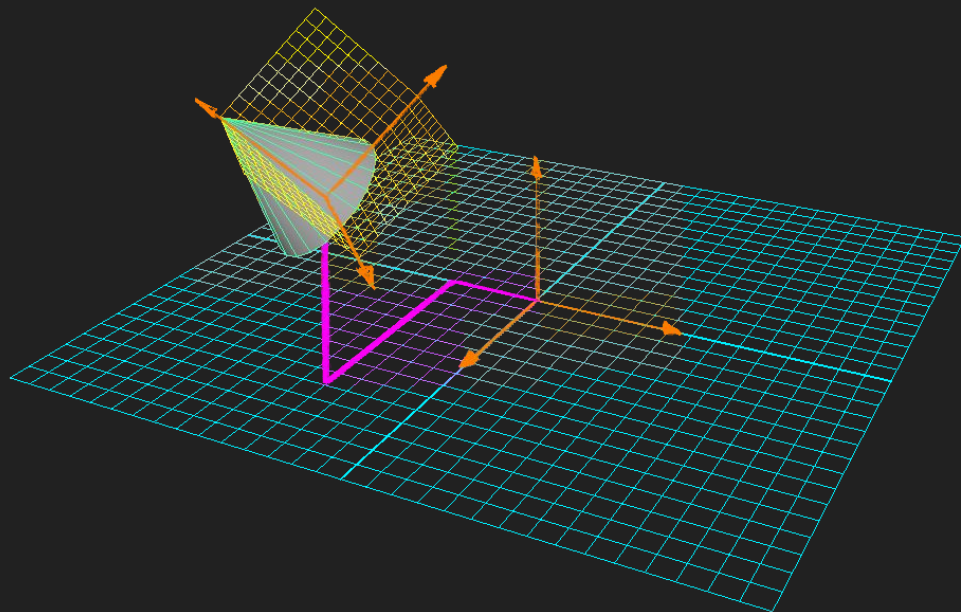
$x\ 0\ y\ 1\ z\ 0$



空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間

オブジェクトとワールド空間



オブジェクト空間

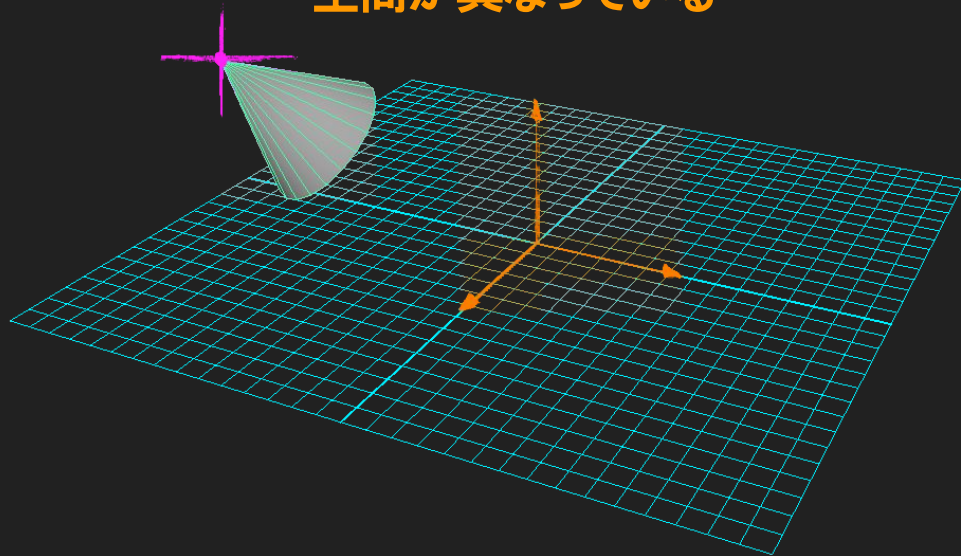
ワールド空間

空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間

オブジェクトとワールド空間

空間が異なっている



オブジェクト空間

$x\ 0\ y\ 1\ z\ 0$

ワールド空間

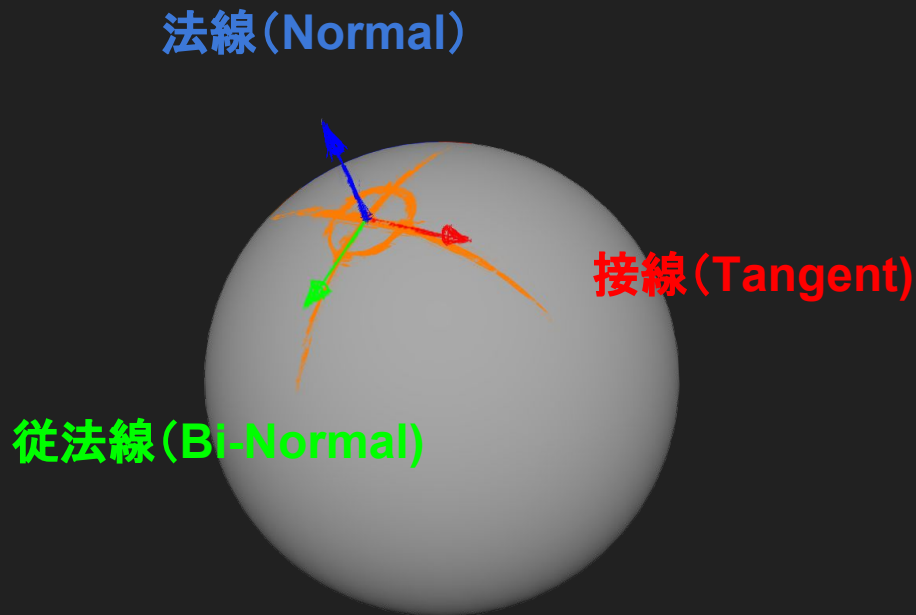
$x\ -1.7\ y\ 2.1\ z\ -3.8$



空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間
- タンジェント空間

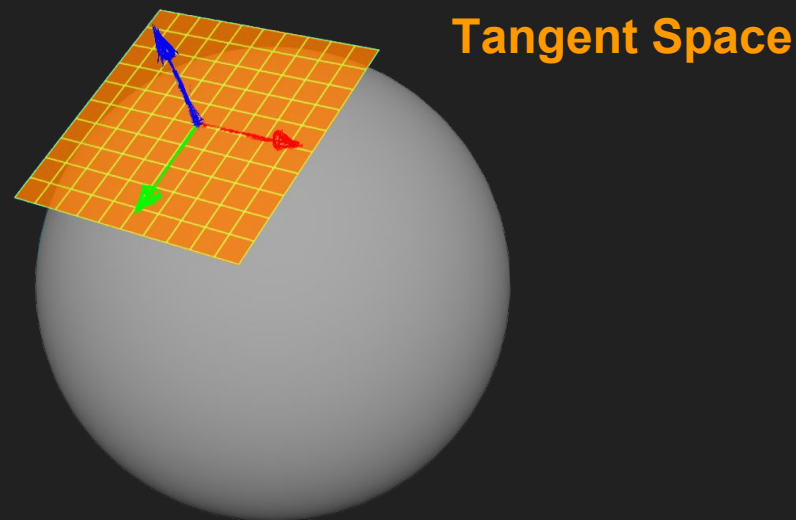
タンジェント空間



空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間
- タンジェント空間

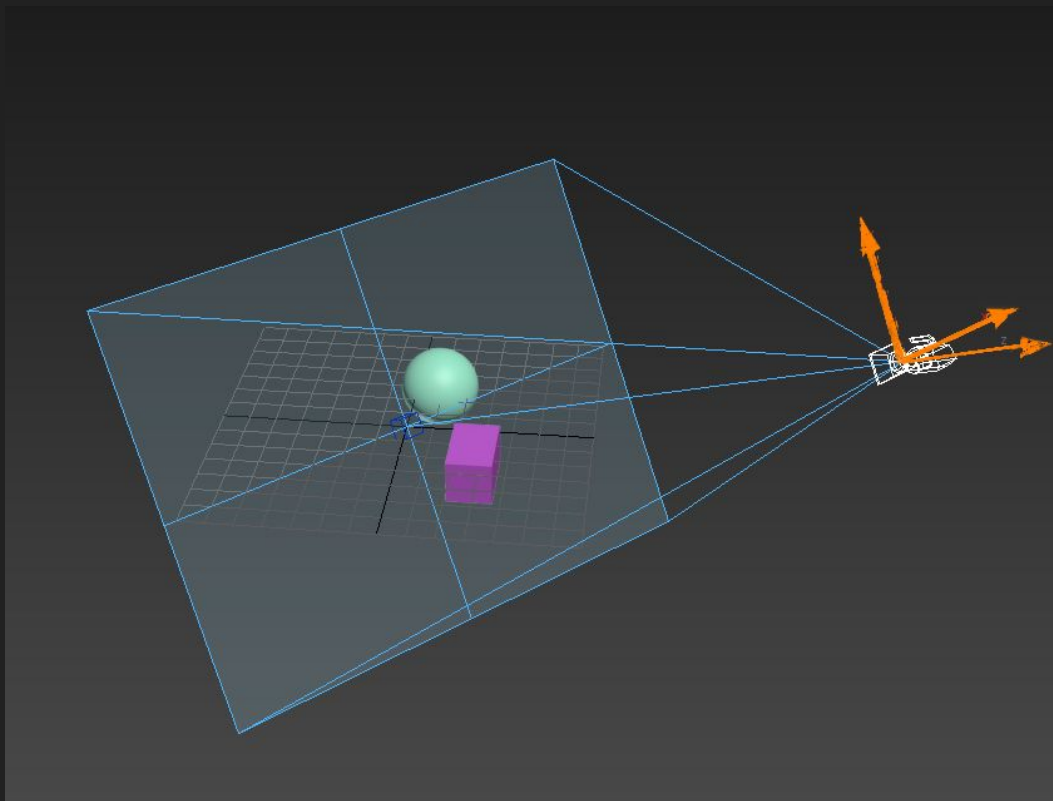
タンジェント空間



空間って？

ビュー空間・カメラ空間

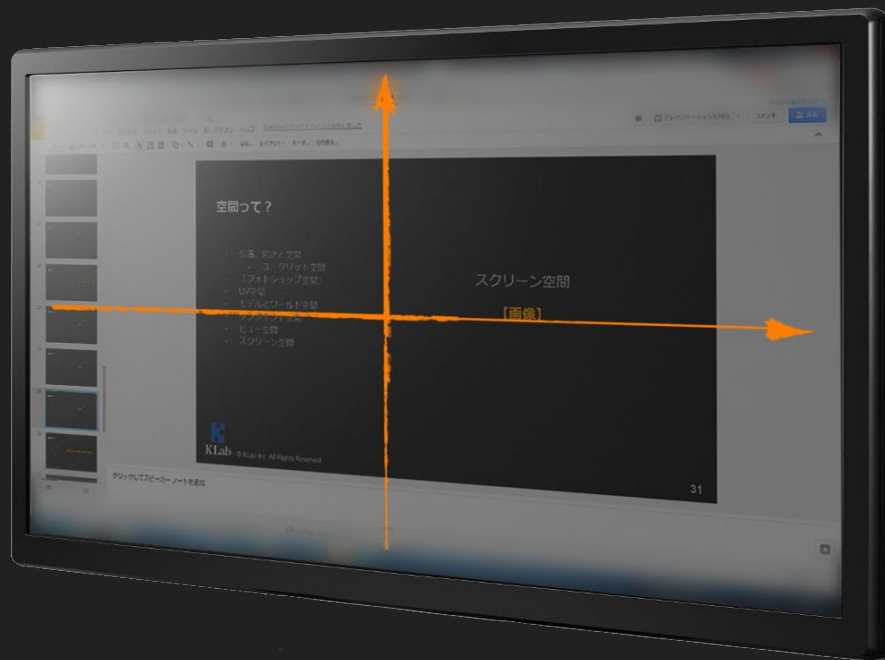
- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間
- タンジェント空間
- ビュー空間・カメラ空間



空間って？

- 位置、向きと空間
 - ユークリッド空間
- 「フォトショップ空間」
- UV空間
- モデルとワールド空間
- タンジェント空間
- ビュー空間・カメラ空間
- スクリーン空間

スクリーン空間



リアルタイムライト における描画

- 表現の幅が増える

フォトリアル



© Clemens Berger



リアルタイムライト における描画

- 表現の幅が増える

非写實的レンダリング



© Mizuno Saori / Clemens Berger



リアルタイムライト における描画

- 表現の幅が増える
- 法線があればあるほど綺麗なシェーディングが可能

法線があればあるほど綺麗なシェーディングが可能

光の当たり方を計算するのに面の向きに対する情報は適切出ないと魅力的な絵にはならない。

シェーディングのディテールを増やすのに頂点を増やしてしまうと...

リアルタイムライト における描画

- 表現の幅が増える
- 法線があればあるほど綺麗なシェーディングが可能
- 負担がかかる



GPUの限界が確実に来る...

処理時間を短縮する必要がある！



法線をテクスチャにベイクすることで！



ノーマルマップの仕組み

現世代では
色の情報を持つテクスチャ以外に
色じゃない情報を色として持つテクスチャが
よく使われる

ノーマルマップ (Normal Map) は
そのようなテクスチャの一種

ノーマルマップの仕組み

ノーマルマップでは

三次元の**面法線**(x, y, z)を
ピクセルごと**色情報**(r, g, b)として
テクスチャに書き込む

法線の色への変換方法は
様々...

ノーマルマップの仕組み

Software	Red	Green	Blue
Maya	X+	Y+	Z+
3ds Max	X+	Y-	Z+
Blender	X+	Y+	Z+
Unity	X+	Y+	Z+
Unreal Engine	X+	Y-	Z+
CryEngine	X+	Y-	Z+

Metallic

L8

Normal

RGB16F

Normal mixing

Combine

Ambient occlusion mixing

Multiply

Additional maps

Bake textures

TextureSet List



Solo

All

Settings



VikingShield

Main shader

DEMO 01

normal_manual_baker.mb

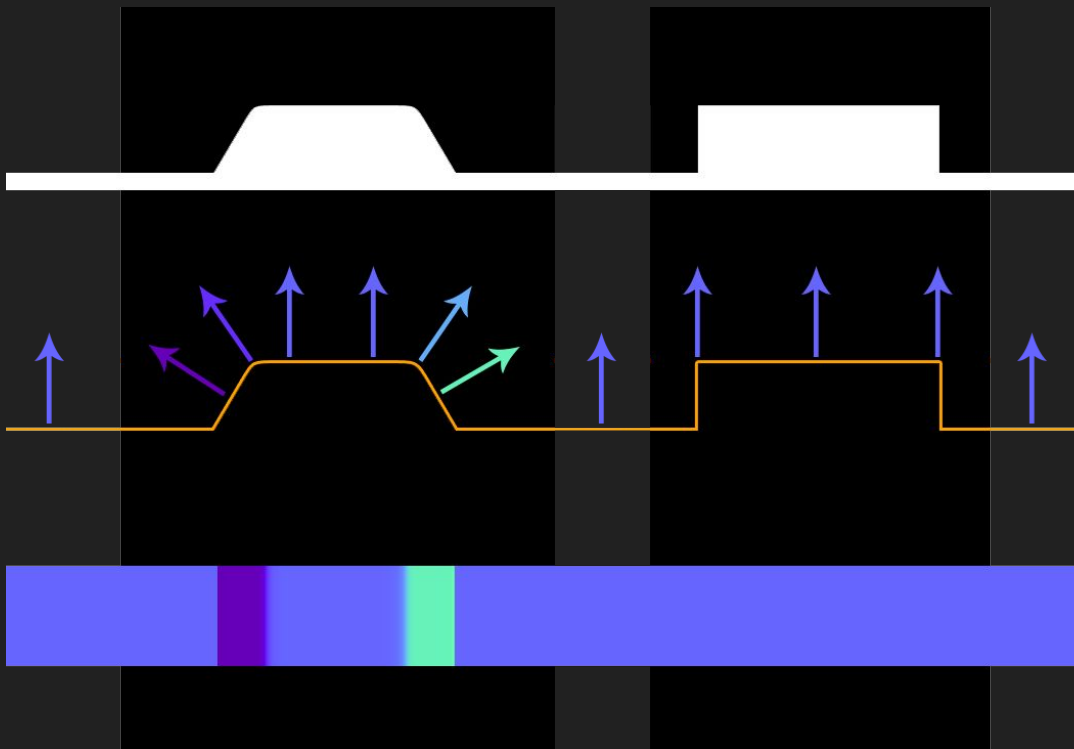
https://github.com/TheClemensBerger/KCS_baking



オープンの中の世界

- バイクボタンを押すと実際に何が起きているのか？
- 平面

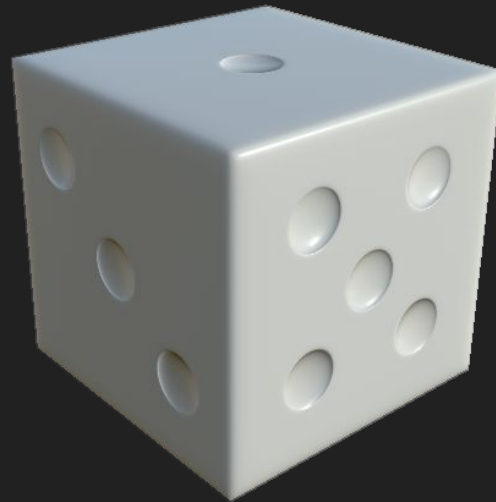
上下の差は法線と関係ない！



オープンの中の世界

立体はどうなる？

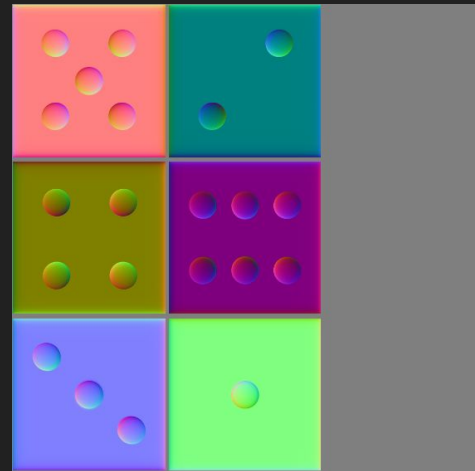
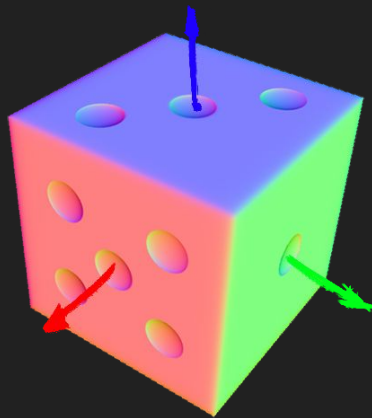
- ベイクボタンを押すと実際に何が起きているのか？
- 平面
- 立体



オープンの中の世界

- バイクボタンを押すと実際に何が起きているのか？
- 平面
- 立体

オブジェクトスペースノーマルマップ

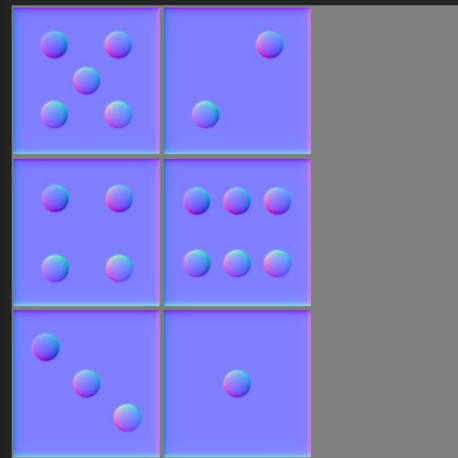
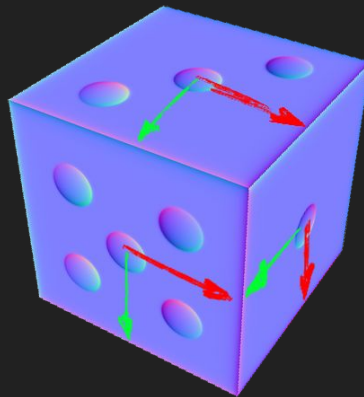


- + 計算量すくない
- + エッジ処理はしやすい
- デフォメーションするメッシュに対応していない

オープンの中の世界

- ベイクボタンを押すと実際に何が起きているのか？
- 平面
- 立体
- オブジェクトノーマル OR タンジェントノーマル？

タンジェントスペースノーマルマップ

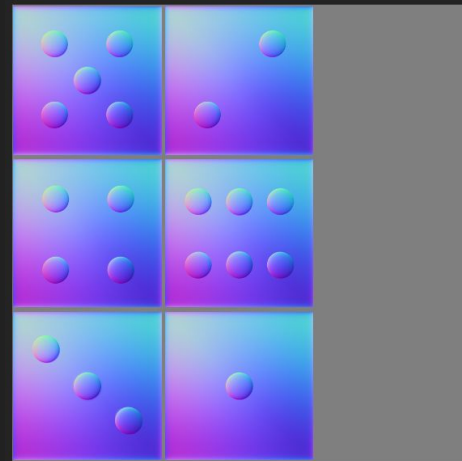
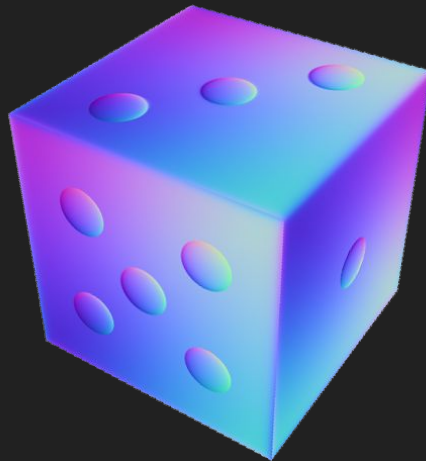


+ デフォメーションするメッシュに対応
- 計算量が増える

オープンの中の世界

- ベイクボタンを押すと実際に何が起きているのか？
- 平面
- 立体
- オブジェクトノーマル OR タンジェントノーマル？
- ハードエッジ OR ソフトエッジ？

タンジェントスペースノーマルマップ(ソフト)



- + 作成しやすい
- + 頂点数が減る
- + 場合によって有利
- 計算量が増える



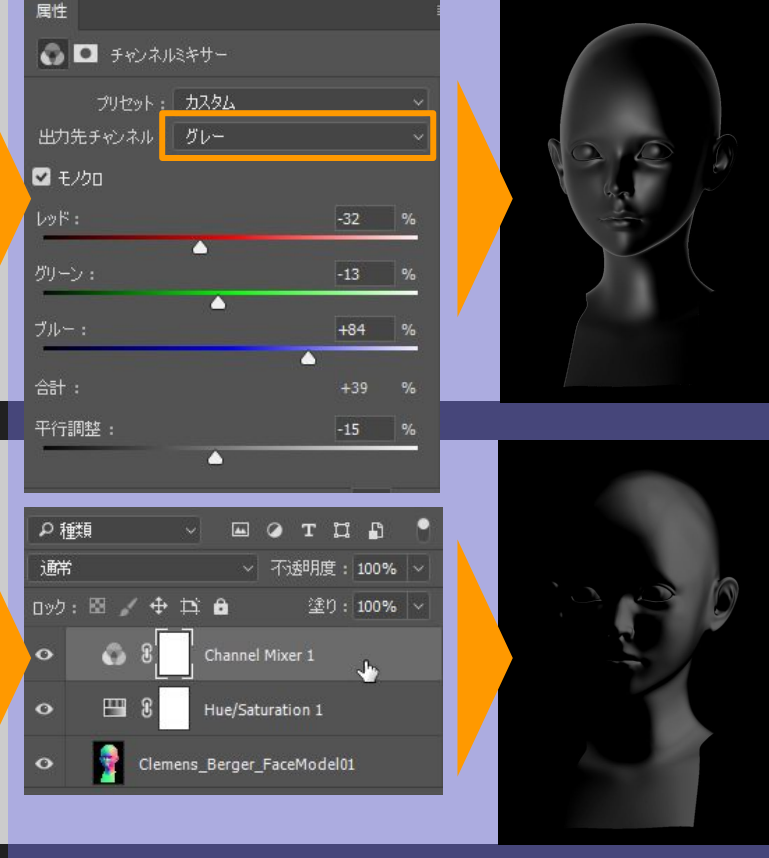
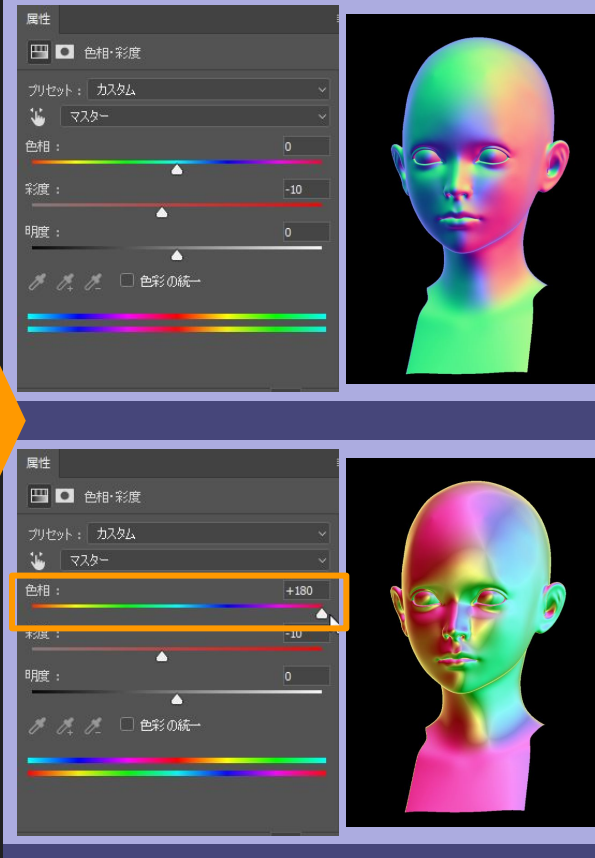
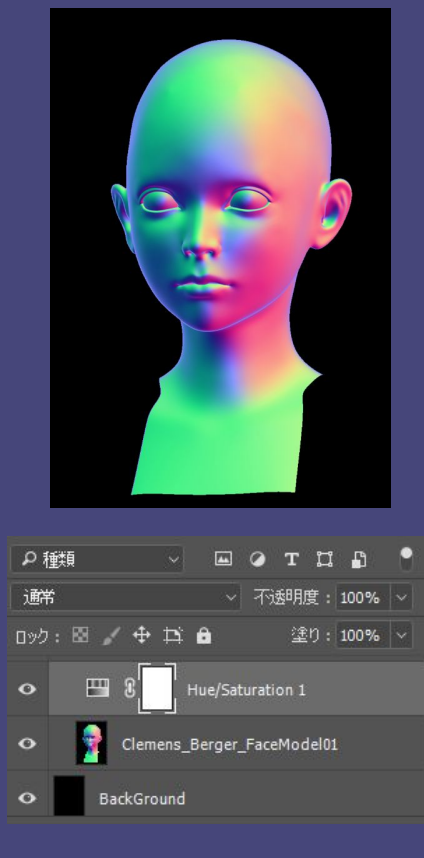
DEMO 02

フォトショップで法線をリライト

relight_setup.psd

https://github.com/TheClemensBerger/KCS_baking





①色相調整レイヤを作成。
色相を変えることで光源方向を変更。

②チャンネルミキサーレイヤを重ねて作成。
出力先チャンネルを「グレー」に設定。

まとめ

- バイク

バイクは
最適化を目的に
破壊的な操作で
情報をデータに焼きこむプロセス

まとめ

- ベイク
- 描画プリミティブ

現世代グラフィックスでは
描画するのに**三角形**はよく使用され、

面法線の代わり頂点法線が
利用される

まとめ

- ベイク
- 描画プリミティブ
- ノーマルマップ

リアルタイムコンテンツにおいて
ノーマルマップ
を使うことで
実現できる表現が大幅に増える