

エンターテインメントの未来がここにある
Compile -Future Entertainment-

CEDEC

CEEA Developers Conference

2010

3DCGツール間データコンバートのお話。

株式会社セガ CSR&D推進部 緒方 修

Copyright (C) 2010 SEGA Corp. CS R&D Support Dept.

CSR&D推進部の組織

グラフィクスサポートセクション

DCCツール
プログラマー

社内ツール
プログラマー

テクニカル
アーティスト

システムサポート
セクション

ITサポート
セクション

モーションキャプチャ
セクション

話の内容

データコンバートは何故必要なのか？	(5分)
データコンバートの作業手順	(5分)
データコンバートに、必要な知識	(25分)
データコンバートに、必要な道具	(15分)



データコンバートは何故必要なのか？ (約5分)

データコンバートは何故必要なのか？

セガでは概ね3パターンに分類される

- チーム内で、ツールを使い分けている—
- チーム間で、違うツールを使っている—
- 過去の資産を流用して、タイトルの開発を行う事になった—

データコンバートは何故必要なのか？

—チーム内で、ツールを使い分けている—

適材適所でツールを使い分ける事が多い

コンバート前提のデータ作成が可能なのでなんとかなる可能性高

難易度：低

データコンバートは何故必要なのか？

ーチーム間で、違うツールを使っているー

外注さんと共同作業するが違うツールを使用している
内作だったのを、外注さんに依頼したがツールが違う
その逆パターン

現在進行形、或いは、最近のプロジェクトの場合、引継が出来ればなんとかなる可能性高
難易度：中

データコンバートは何故必要なのか？

—過去の資産を流用して、開発を行う事になった—

過去のリソースを利用しようとした、が、開発ツールが違う
時間がたった場合、記憶が風化

作成ルールが徹底されていない場合、自動で処理出来る割合が凄く低い

難易度：高

データコンバートは何故必要なのか？

—補足—

「コンバートできた！！ = スタート」

「ゲームデータとして利用する事 = ゴール」

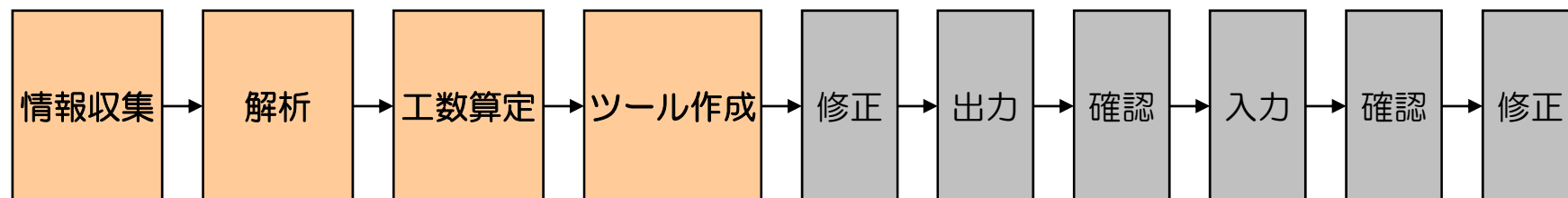


データコンバートの作業手順 (約5分)

データコンバートの作業手順

—全体の流れ—

まずは2、3個のサンプルから工程を確認し、パスを通す
良好ならバッチ処理



データコンバートの作業手順

—情報収集—

どのプラットフォーム向けに作られたのか？
使用していたミドルウェアはなんなのか？
ツール上でどのようにシーンが構築されていたのか？
どのような仕様で作成されたのか？



データコンバートの作業手順

—解析—

作成ルールは徹底されているのか？

他のツールで、再現不可能なデータになっていないか？

アニメーションの有無

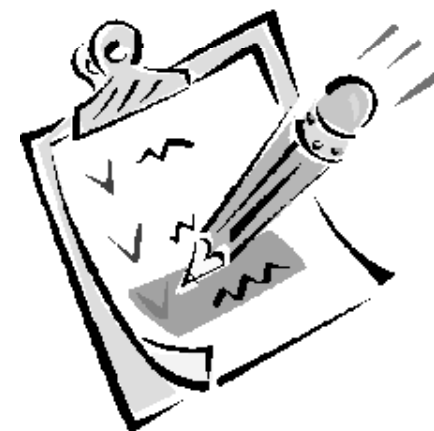
ユーザーアトリビュートの有無



データコンバートの作業手順

ー修正項目のリスト化ー

問題がある項目の修正工程洗い出し
自動処理が出来る事、出来ない事の切り分け
作成するツールの仕様策定
ツールの実装



データコンバートの作業手順

出力

この段階で、インポート側で問題が起きるようなデータになってはダメ
出力された段階で、ほとんど結果は決まっている

中間データの確認

正しくデータが出力されているか中間ファイルを確認

入力

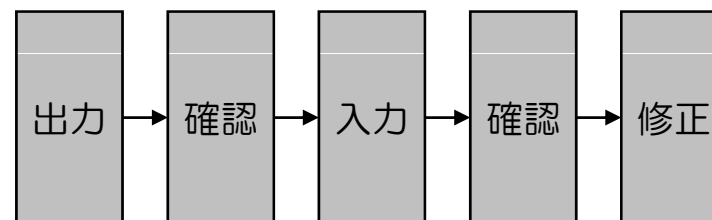
使用するツールへ入力

確認

正しくデータが入力されているか確認

修正

入力後でないと出来ない修正作業を確認
出来るだけ自動処理





データコンバートに、必要な知識 (約25分)

データコンバートの為に必要な知識

3DCGツールの仕様

今日の本題、実はすごく違う所がある

データエクスポート、インポートの仕様

どんな処理をどこで行ってるか把握しないと都合の良いデータは出来ない

データフォーマットの仕様

データフォーマットに入らない情報は絶対にコンバートできない

システム、ミドルウェアの仕様

コンバート後、すぐにゲームデータとするなら必要


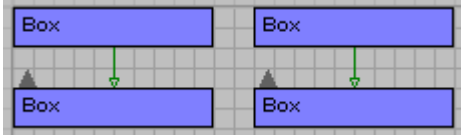
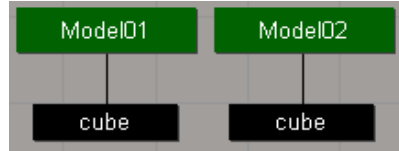
座標系

maya	max	softimage
Yアップ/ Zアップ切替 (標準Yアップ)	Zアップのみ	Yアップのみ

文字制限

maya	max	softimage
<p>英数、'-','_'</p> <p>無効な文字は自動でアンダースコアに変換</p> <p>先頭は数字不可（自動削除）</p> <p>” 1cube” → “cube”</p>	<p>何でもOK</p> <p>マテリアルは標準で先頭が数字</p> <p>例：“01 - Default”</p> <p>テクスチャは標準が2byte文字</p> <p>例：“マップ #1”</p>	<p>英数、'-','_'</p> <p>無効な文字は自動でアンダースコアに変換</p> <p>先頭は数字不可（アンダースコア自動追加）</p> <p>“cube” → “_1cube”</p>

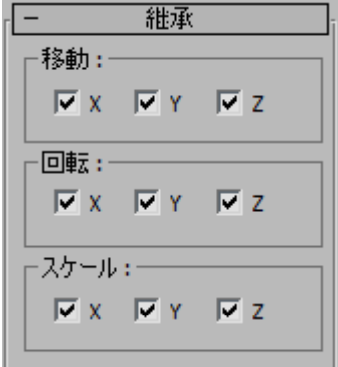

命名規則

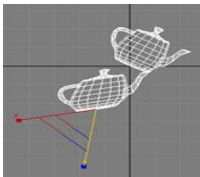
maya	max	softimage
		
<ul style="list-style-type: none"> ■ モデル名 階層が違えば同名可 ■ マテリアル名 同名不可 ■ テクスチャ名 同名不可 <p>※マテリアルとテクスチャは同名不可</p>	<ul style="list-style-type: none"> ■ モデル名 同名可 ■ マテリアル名 同名可 ■ テクスチャ名 同名可 	<ul style="list-style-type: none"> ■ モデル名 同名不可 ■ マテリアル名 同名不可 ■ テクスチャ名 マテリアルライブラリが異なれば同名可 <p><small>※softimage 但し、モデルノードの配下に階層構造を設定した場合、同名可 「Surface_node」と「image_node」は同名不可</small></p>

基点 (ピボット)

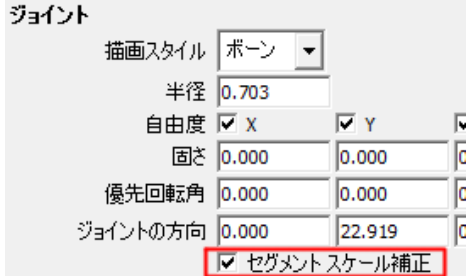
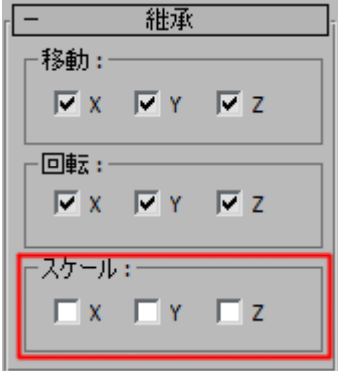
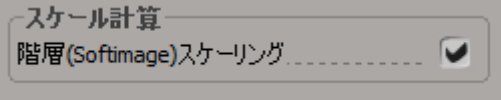
maya	max	softimage
TRS個別	TRSで共通	TRSで共通

TRSの継承

maya	max	softimage
<p><input checked="" type="checkbox"/> トランスフォームの継承</p> <p>TRS個別設定は不可 マトリクス全体の継承設定のみ可</p>	 <p>TRS個別に設定可 XYZ個別に設定可</p>	 <p>TRS個別に設定可 XYZ個別は設定不可</p>



不均等スケール時のせん断（シア、シヤー）

maya	max	softimage
 <p>せん断する シア値の設定可 jointの場合はスケール補正をONにして回避できる スケール継承しない</p>	 <p>せん断する 設定変更可 スケール継承しない</p>	 <p>せん断しない 設定変更可 ローカル座標でスケール継承を行う</p>

頂点（カラー）

maya	max	softimage
<p>RGBA4要素 段数無制限 浮動小数 1.0以上の値が入る</p>	<p>RGB3要素 2段まで (UVWを代用すれば 増やせる) 浮動小数 1.0以上の値が入る</p>	<p>RGBA4要素 段数無制限 浮動小数 0.0~1.0までの値 (拡張可能)</p>

頂点（ユーザー法線）

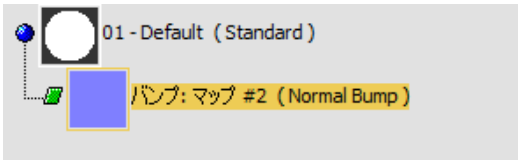
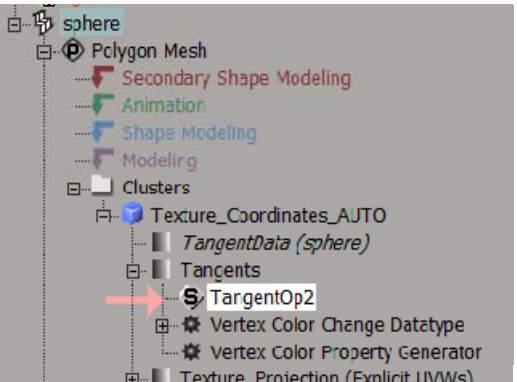
maya	max	softimage
段数無制限	モディファイア にスタックする 方法でなら段数 無制限	データとしては 段数無制限

頂点 (UV)

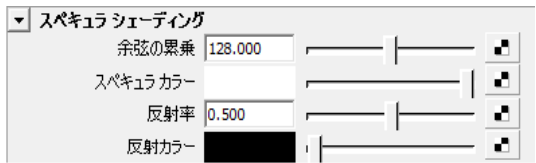
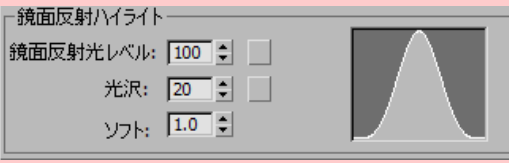
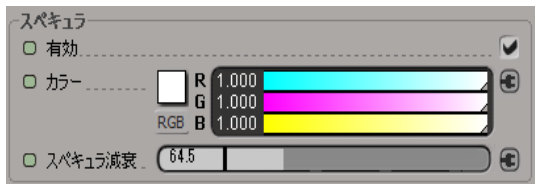
maya	max	softimage
UV 2要素 段数無制限 浮動小数 名前で管理	UVW 3要素 99段まで 浮動小数 IDで管理 名前も設定可	UVW 3要素 段数無制限 浮動小数 名前で管理 ※Wに注意

※softimage
 但しWはエディタでは見えない＝編集出来ない
 maxからコンバートされたデータにはWが入っているので要注意

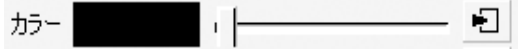
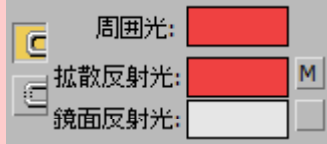

頂点（接空間）

maya	max	Softimage
<p>SDKで取得可能（法線、UVから生成） UVの数だけ取得可能 専用のデータ格納領域は無い</p>	 <p>マテリアル→バンプに「法線バンプ」設定時、SDKから取得可能 専用のデータ格納領域は無い</p>	 <p>カスタムオペレータで「TangentOp2」設定時、SDKから取得可能（頂点カラーとして確認可）</p>

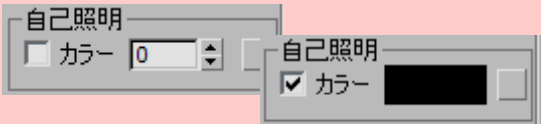
マテリアル（スペキュラレベル）

maya	max	softimage
 <p>無い カラーのみ</p>	 <p>ある スペキュラカラーと レベルが掛け算され た結果 = スペキ ュラカラーとなる</p>	 <p>無い カラーのみ</p>

マテリアル（テクスチャ設定時の動作）

maya	max	softimage
 <p>カラー情報は失われる 編集は不可能 テクスチャ削除時、カラーは元の値に戻らない カラー指定はテクスチャのカラーゲイン等に設定する</p>	 <p>カラー情報は失われ ない 編集も可能</p>	 <p>カラー情報は失われる 編集は不可能 但し、imagenodeを削除した場合、カラーは元の値に戻る カラー指定はマテリアルにレイヤを追加し、入力カラーに設定等する</p>

マテリアル（白熱、自己照明）

maya	max	softimage
<p>RGB 浮動小数 アンビエント、ディ フューズに対する足 し算</p>	 <p>オプションで切替 整数0~100 または RGB 浮動小数 アンビエントに対す る足し算</p>	<p>RGB 浮動小数 アンビエント、ディ フューズに対する足 し算</p>

マテリアル（透明）

maya	max	softimage
透明度 浮動小数 3チャンネル	不透明度 浮動小数 1チャンネル (UIでは0~100)	透明度 浮動小数 4チャンネル

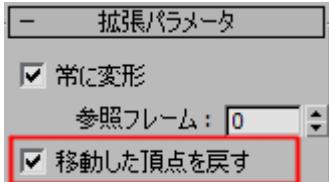
初期姿勢

maya	max	softimage
 <p>bindposeから取得する</p>	 <p>2種類ある スキンモディファイアに登録されている骨の場合、登録時のTRS情報をスキンモディファイアから取得できる スキンポーズに登録していれば、それを初期姿勢にする事も出来る その場合、登録したノードから取得できる</p>	<p>骨として登録されたノードから登録時のTRS情報を取得 TRS情報は骨にある</p>

ヌルボーン（ロケータ、ヘルパ）

maya	max	softimage
ルートノードがjoint であれば、あらゆる ノードを骨として設 定出来る	どのようなノードで も骨として設定可	どのようなノードで も骨として設定可

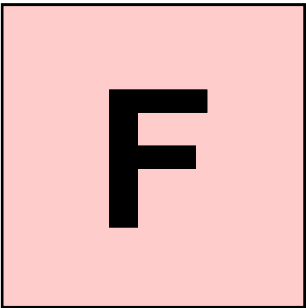
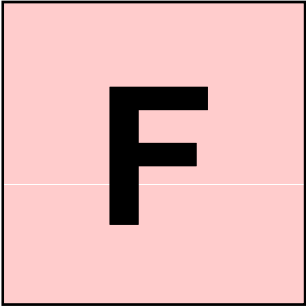
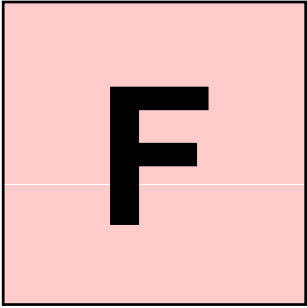
エンベロープの動作

maya	max	softimage
<p>骨の階層途中にエンベロープノードがある場合、TRSの影響+エンベロープの影響になる</p>	 <p>骨の階層途中にエンベロープノードがある場合、スキンモディファイアの設定で選択が可能</p>	<p>骨の階層途中にエンベロープノードがある場合、エンベロープの影響のみ受ける</p>

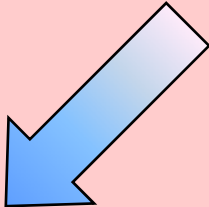
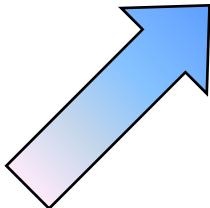
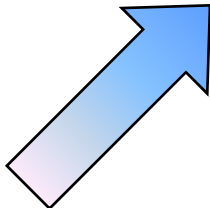
モーフ、シェイプ

maya	max	softimage
ブレンドシェイプ ターゲットが削除された場合キャッシュの情報を使用 ターゲットに変更があれば自動更新	モーファー 常時キャッシュの情報を使用 ターゲットに変更があれば自動更新 (手動更新も可)	デフォーム⇒シェイプ ターゲットが削除された場合キャッシュの情報を使用 ターゲットのリロードは不可、再登録する必要有り


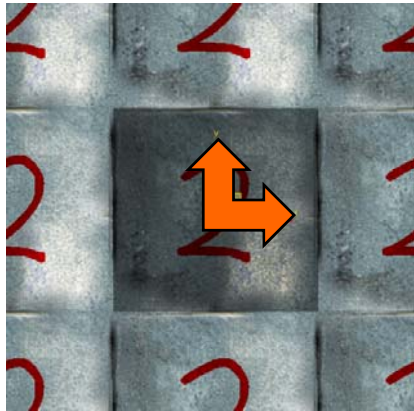

UV座標

maya	Max	Softimage
 <p>(0, 0) (1, 1)</p>	 <p>(0, 0) (1, 1)</p>	 <p>(0, 0) (1, 1)</p>

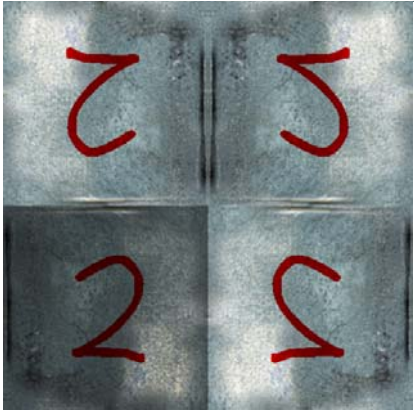
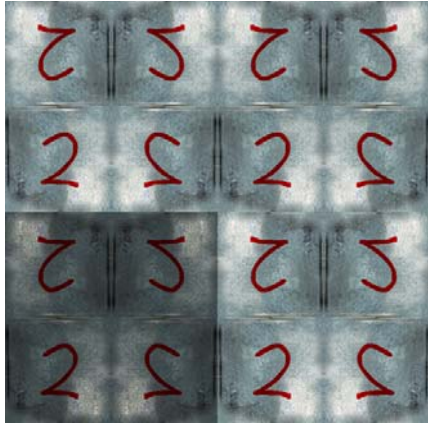
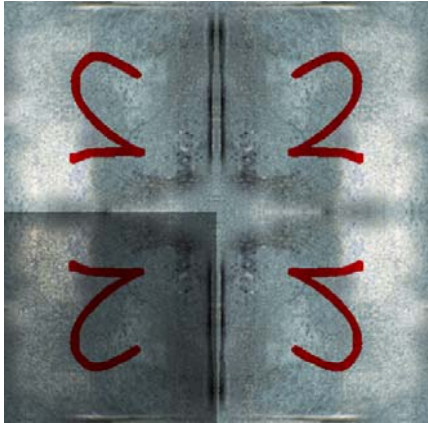
UVオフセット

maya	max	softimage
 <p>プラス値で左、下に スクロール</p>	 <p>プラス値で右、上に スクロール</p>	 <p>プラス値で右、上に スクロール</p>

テクスチャ（回転中心）




maya	max	softimage
		
<p>真ん中 (0.5, 0.5) リpeatした場合、1 つ目の中心で回転</p>	<p>真ん中 (0.5, 0.5)</p>	<p>左下 (0, 0) リpeatした場合 でもUV座標(0, 0) で回転</p>

テクスチャ（ミラー）

maya	max	softimage
		
<p>リピート = 2, 2 ミラーU、ミラーV</p>	<p>リピート = 2, 2 ミラーU、ミラーV</p>	<p>リピート = 2, 2 ミラーU、ミラーV</p>

※siはリピート1でもミラーするので注意！！

テクスチャ (タイル)

maya	max	softimage
 <p data-bbox="309 1157 801 1289"> リピート = 2, 2 スケール中心 = 左下 </p>	 <p data-bbox="875 1157 1368 1289"> リピート = 2, 2 スケール中心 = 中央 </p>	 <p data-bbox="1417 1093 1944 1289"> リピート = 2, 2 or スケール = 0.5, 0.5 スケール中心 = 左下 </p>

レイヤテクスチャ

maya	max	softimage
<ul style="list-style-type: none"> なし オーバー イン アウト 加算 減算 乗算 差 明るく 暗く 彩度の増加 彩度の減少 照らす 	<ul style="list-style-type: none"> 通常 平均値 加算 減算 比較(暗) 乗算 焼き込みカラー 焼き込み線形 比較(明) スクリーン 覆い焼きカラー 覆い焼き線形 スポットライト スポットライト ブレンド オーバーレイ ソフトライト ハードライト ピンライト ハードミックス 違い 除外 色相 彩度 カラー 数値 	<ul style="list-style-type: none"> オーバー イン アウト プラス 上限付き加算 表示/非表示(乗算) 上限付き 表示/非表示(乗算) 差 比較(暗) 比較(明) ハードライト ソフトライト スクリーン オーバーレイ ブレンド

Fカーブ

maya	max	softimage
スプライン リニア クランプ ステップ ステップ ネクスト フラット 固定 プラトー	スプライン リニア コンスタント	スプライン リニア コンスタント内挿

ユーザーデータUI

説明	関連付ける変数の型	Maya	max	XSI
ラベルです。変更できない文字列として表示されます。	変数とは関連付けられません。	未対応	テストLabel0	
文字列を入力するテキストボックスです。	string	TestTextField0 TestText	テストTextField0 TestText	テストTextF0 TestText
複数行にわたったテキストボックスで、メモ帳のようなイメージです。	string	TestTextArea0 TestText	テストTextA0 TestText	テストTextA0 TestText
整数を入力するためのテキストボックスです。	int, uint	TestIntField0 256	テストIntField0 256	テストIntField0 256
整数を操作するためのスライダです。	int, uint	TestSlider0 0	テストSlider0	テストSlider0 0
IntField とIntSliderを組み合わせたものです。	int, uint	TestSliderG0 0	テストSliderG0	テストSliderG0 8
符号無し整数を入力するためのテキストボックスです。	uint	TestIntField0 256	テストIntField0 256	テストIntField0 256
符号無し整数を16進数入力をするためのテキストボックスです。	uint	TestHexField0 128 HEX { 0x***** } / DEC { ***** }	テストHexField0 0x80	テストHexField0 0x80
浮動小数点数を入力するためのテキストボックスです。	float,double	TestFloatField0 1.000000	テストFloatField0 1.0	テストFloatField0 1.0
浮動小数点数を操作するためのスライダです。	float,double	TestFSlider0 0.000000	テストFSlider0	テストFSlider0 0
FloatText とFloatSliderを組み合わせたものです。	float,double	TestSliderG0 32.000000	テストSliderG0 32.0	テストSliderG0 32
Color 値をカラーパレットにより選択します。	color_rgb	TestColorPaletteRGB0	テストColorPaletteRGB0	テストColorPaletteRGB0 R 1.000 G 1.000 B 1.000
Color 値をカラーパレットにより、α付きで選択します。α付のカラーパレットが無いものに関しては、カラーをカラーパレット、α値をfloatSliderで指定します。	color_rgb	テストColorPaletteRGBA0(RGB) 1.000000 (Alpha)	テストColorPaletteRGBA0	テストColorPaletteRGBA0 R 0.000 G 0.000 B 0.000 A 1.000
精度を、α値をカラーパレット、或いはFloatSliderで指定します。	float, double	TestColPalA0 1.000000	テストColPalA0 1.0	テストColPalA0 1
bool 値を操作するためのチェックボックスです。	bool	<input checked="" type="checkbox"/> TestCheckBox0	<input checked="" type="checkbox"/> TestCheckBox0	テストCheck-Box0 <input checked="" type="checkbox"/>
有限個の決まった文字列の中から1つを選択するためのコンボボックスです。	enum_string	TestCnmbxRxn0 TestItem01 (Enable Radio0)	テストComboBox0 TestItem01 (Radio0有効)	テストComboBox0 テストItem01 (Radio0有効)
有限個の決まった文字列の中から1つを選択するためのラジオボタンです。	enum_string	TestRadio0 TestItem00 TestItem01	テストRadio0 TestItem00 TestItem01 TestItem02	テストRadio0 TestItem00 TestItem01 TestItem02
ボタンです。押されたことのみを検出することができ、変数の値を変更することはありません。	int (値は変更されませんが、押されたことを検出するために関連付けを行うこと)	未対応	未対応	未対応
File Browser Dialogを起動するボタン、及び関連付けられた文字列を入力するテキストボックスで、パスのファイル名を指定します。	string	未対応	テストFileName0	テストFileName0
File Browser Dialogを起動するボタン、及び関連付けられた文字列を入力するテキストボックスで、パスのみを指定します。	string	未対応	テストFileName0	テストPathName0 c:\w



データコンバートに、必要な道具 (約15分)

データコンバートの為に必要な道具

信頼できるエクスポート、インポート

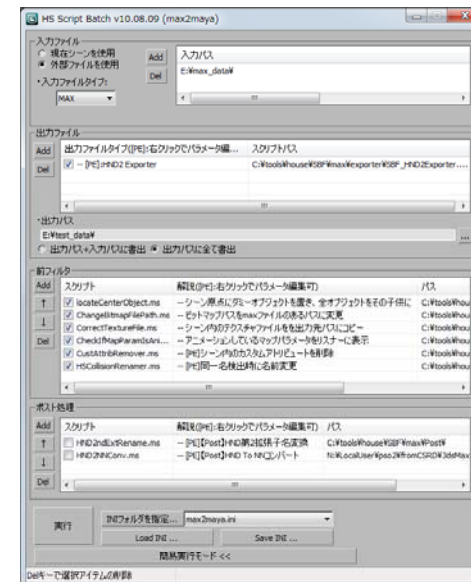
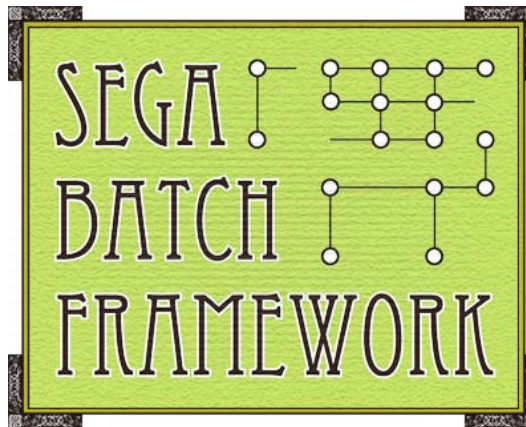
バッチ処理しても、へこたれないタフなモノが欲しい
問題がおきた時にすぐ対応できないと駄目



データコンバートの為に必要な道具

バッチ処理システム

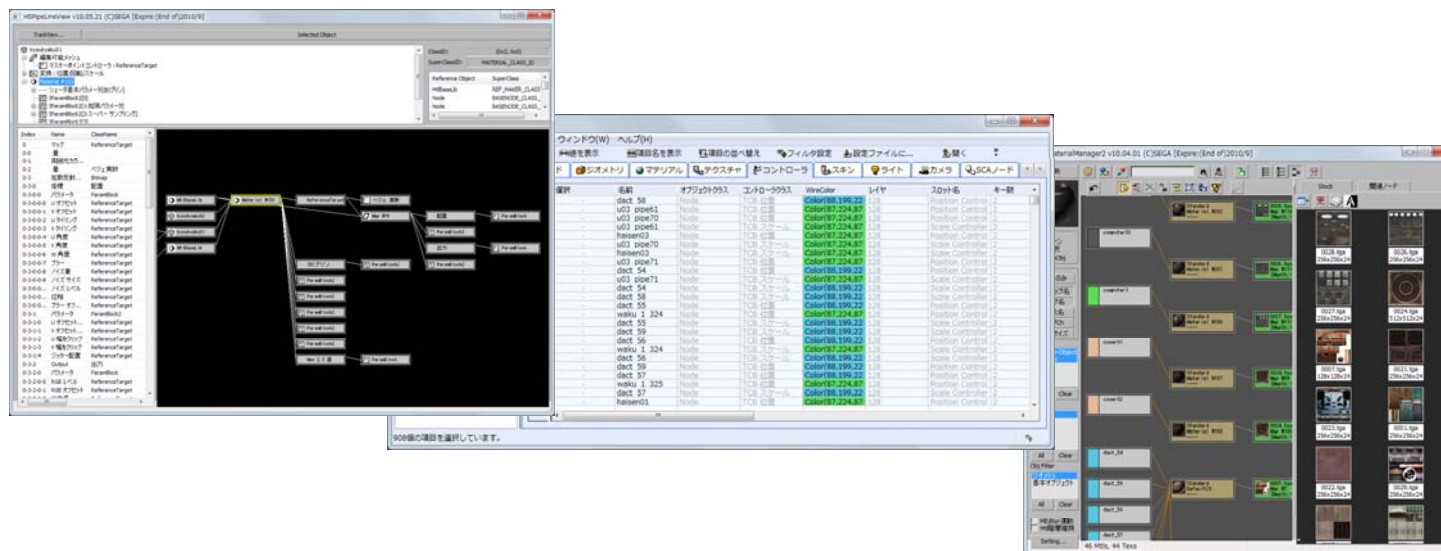
処理をカスタマイズ出来れば修正も自動化が可能
インポータ、エクスポータはデータを入出力する事に専念させる



データコンバートの為に必要な道具

3DCGツール上の解析ツール

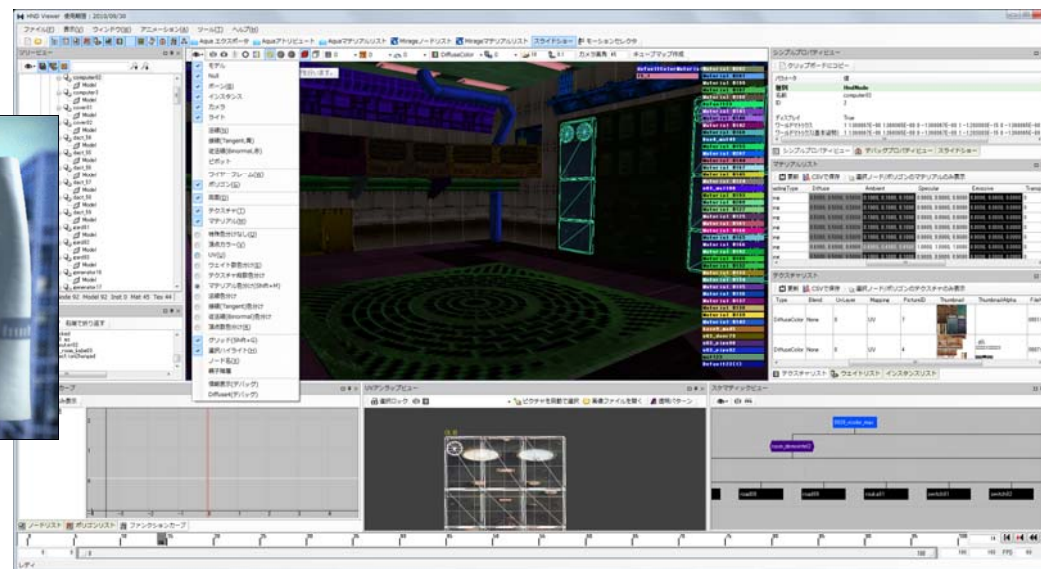
標準機能では、解析出来ない事が多い



データコンバートの為に必要な道具

中間ファイルの解析ツール

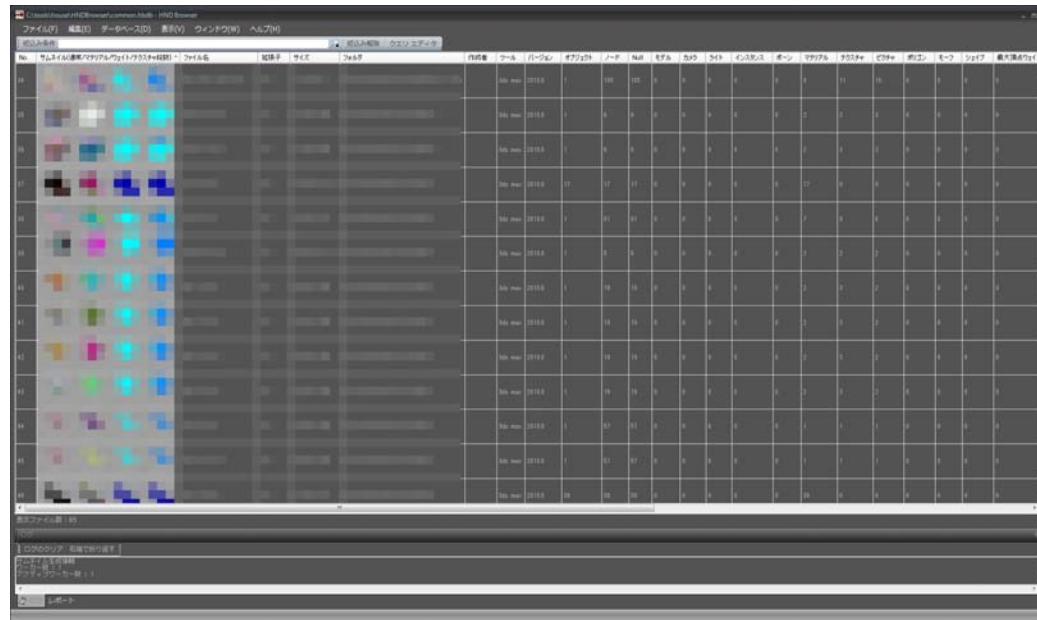
出力がおかしいのか、入力がおかしいのかを切り分ける



データコンバートの為に必要な道具

中間ファイルのデータベース

全体像の把握に必要



No.	ファイル名	拡張子	サイズ	作成日	更新日	バージョン	ステータス	コメント	...
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20

まとめ

安くない

環境を作るだけでもかなりお金がかかる。
調査にもそれなりにお金がかかる。

解析は命

ゲームデータのパイプライン構築にも必要。

AUTODESKさんに期待する

お願いなのでFBXのデバグは用意して下さい。



ご静聴ありがとうございました。