

TOYSTUDIOによる頂点ウェイトの修正

以下の使用方法は、TOYSTUDIOの仕様には無い利用方法です。
このページの解説と、TOYSTUDIO作者様とは何も関係がありません。
使い方がわからない、エラーになる、などの理由でTOYSTUDIOのページで質問はしないように。

予想外のポリゴンの歪みが発見された場合、従来のtso2mqoによるMOD開発では、試行錯誤によって正しい参照TSOを発見していました。
tso2toyによって、tsoファイルをTOYSTUDIOにエクスポートすることにより、TOYSTUDIOの優れた頂点ウェイト編集機能を利用できます。

準備

以下のプログラムをダウンロードしてインストールします。

- TOYSTUDIO (内部バージョンが9になっている1.4.0.2 ~ 1.4.0.6まで)
- tso2toy a-0.1以上

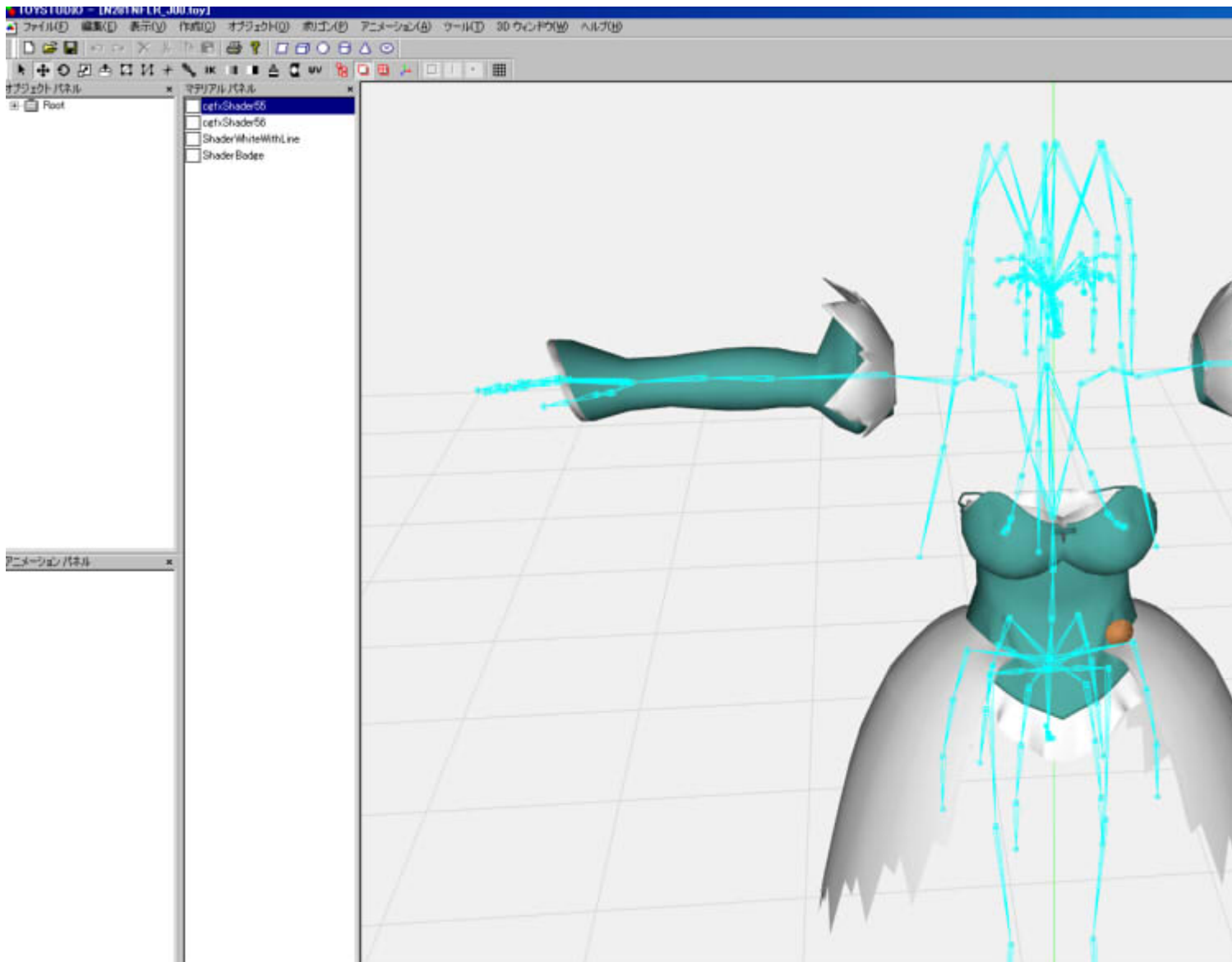
ウェイト問題の確認

以下のようなポリゴンの歪みを修正します。

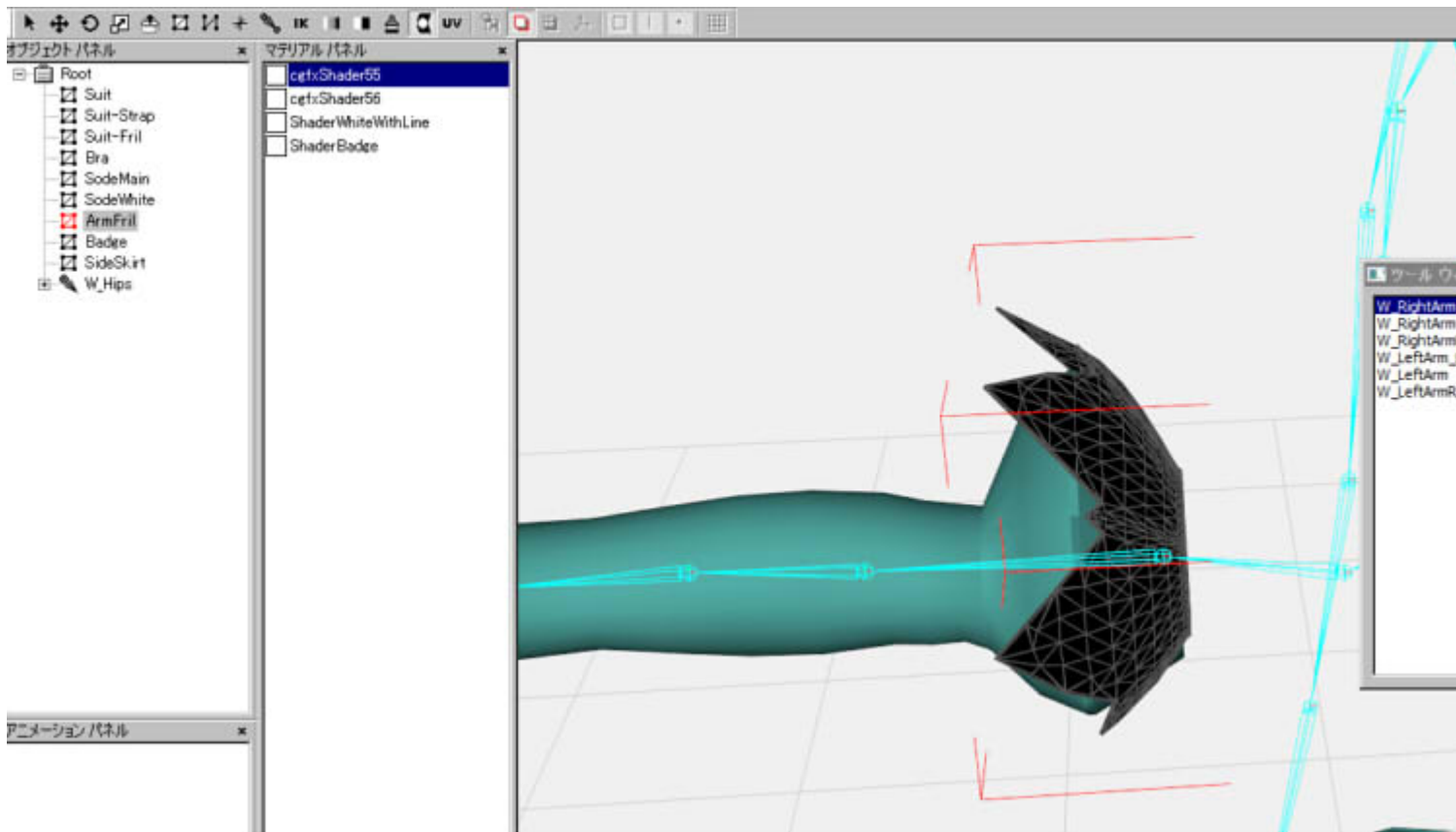


- まず問題になっているオブジェクトが含まれるtsoを特定します。

tso2toyで、問題のオブジェクトのtsoをTOYSTUDIOファイルに変換します。

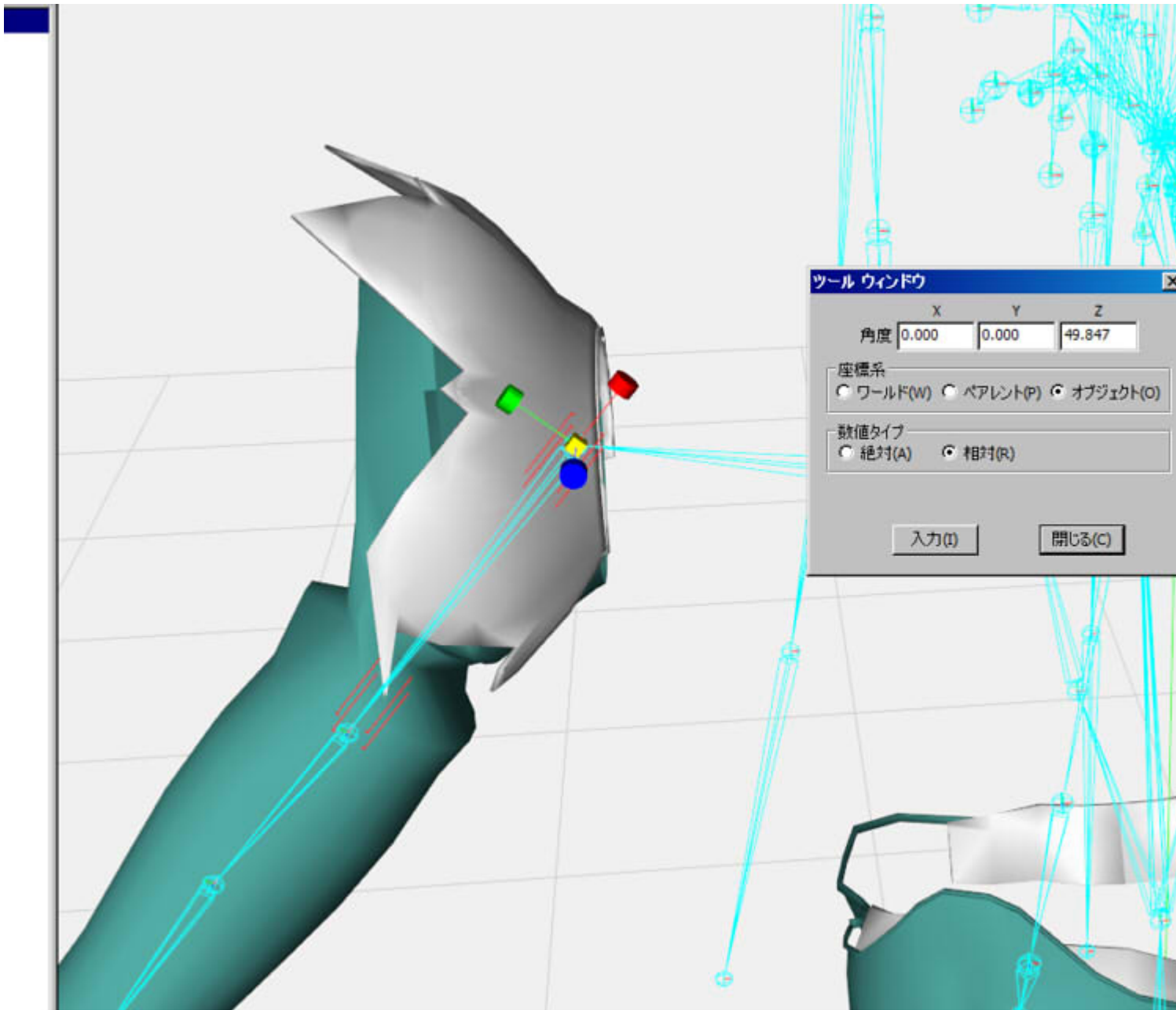


- オブジェクト単位で影響するボーンを確認します。

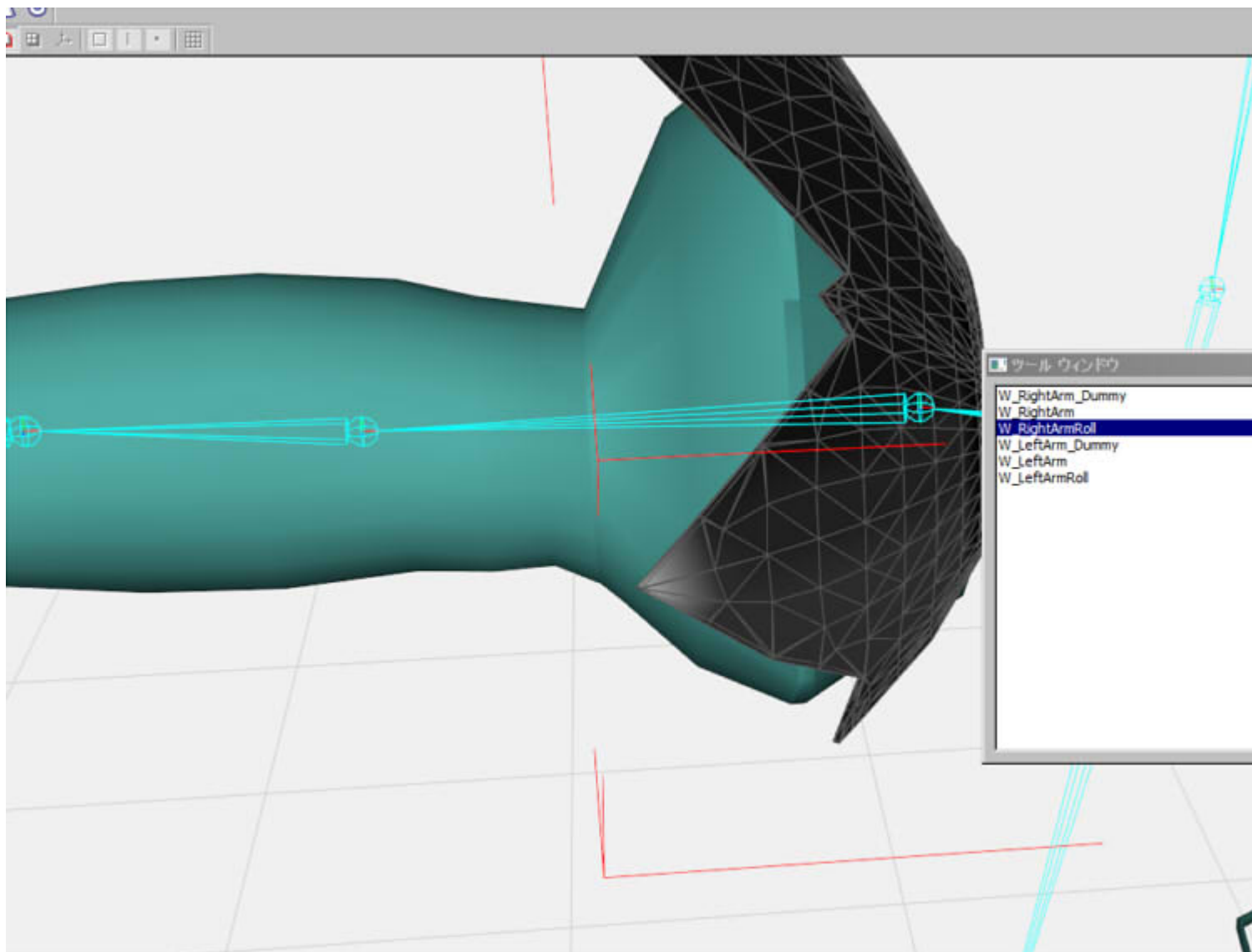


このケースではW_RightArm_Dummy (ダミージョイント), W_RightArmRoll (関節回転用ジョイント) がフリル形状に悪影響していることが推測できます。

- 実際に関節を動かしてみて問題を再現します。



3Dカスタム少女の実行環境で見た歪みと同様の歪みが確認できました。
頂点ウェイトを確認すると、ゆがむ部分にはW_RightArmRollが作用していることがわかります。



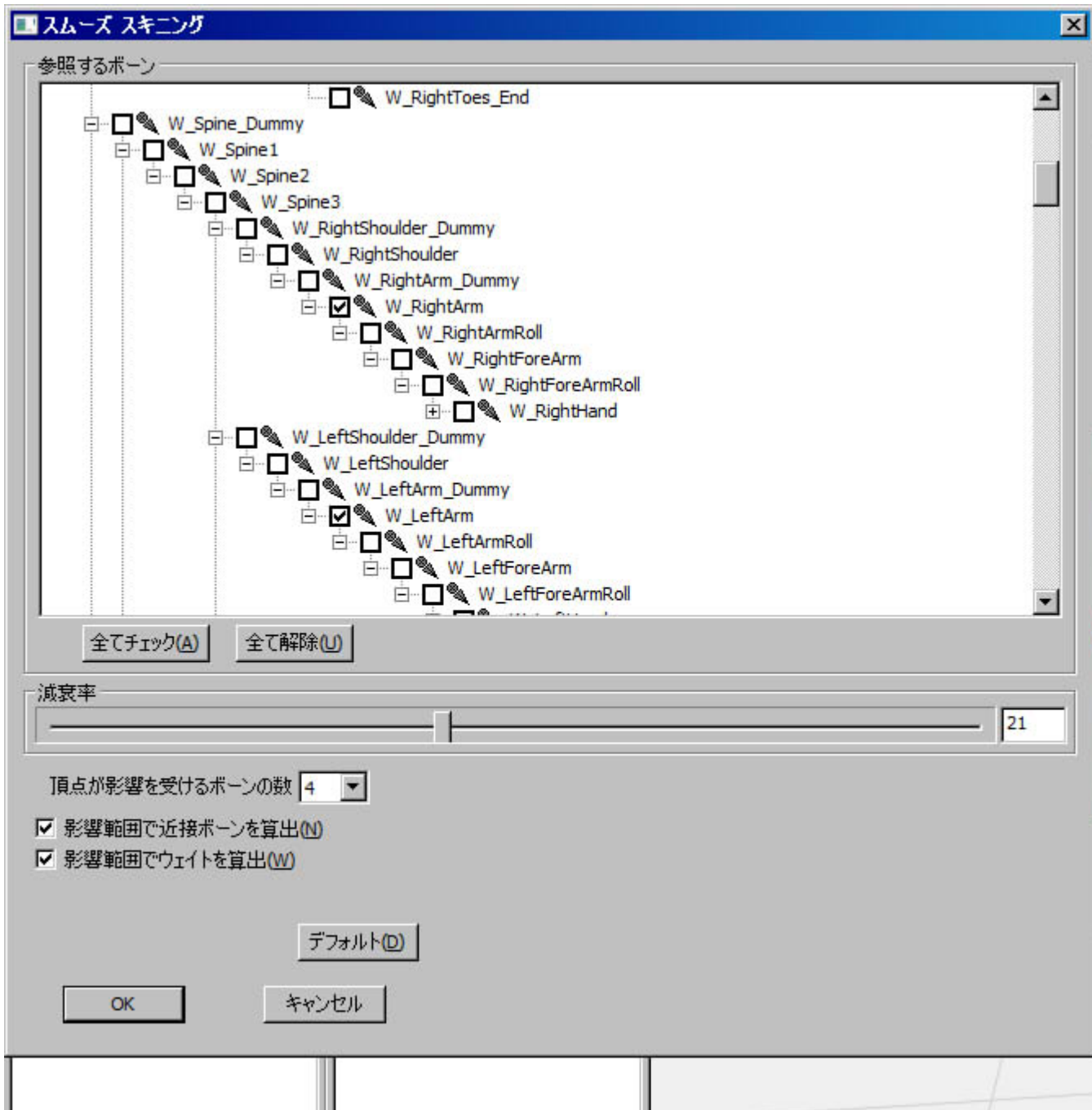
このフリルに対する頂点ウェイトを修正すればよいわけです。

頂点ウェイトの修正

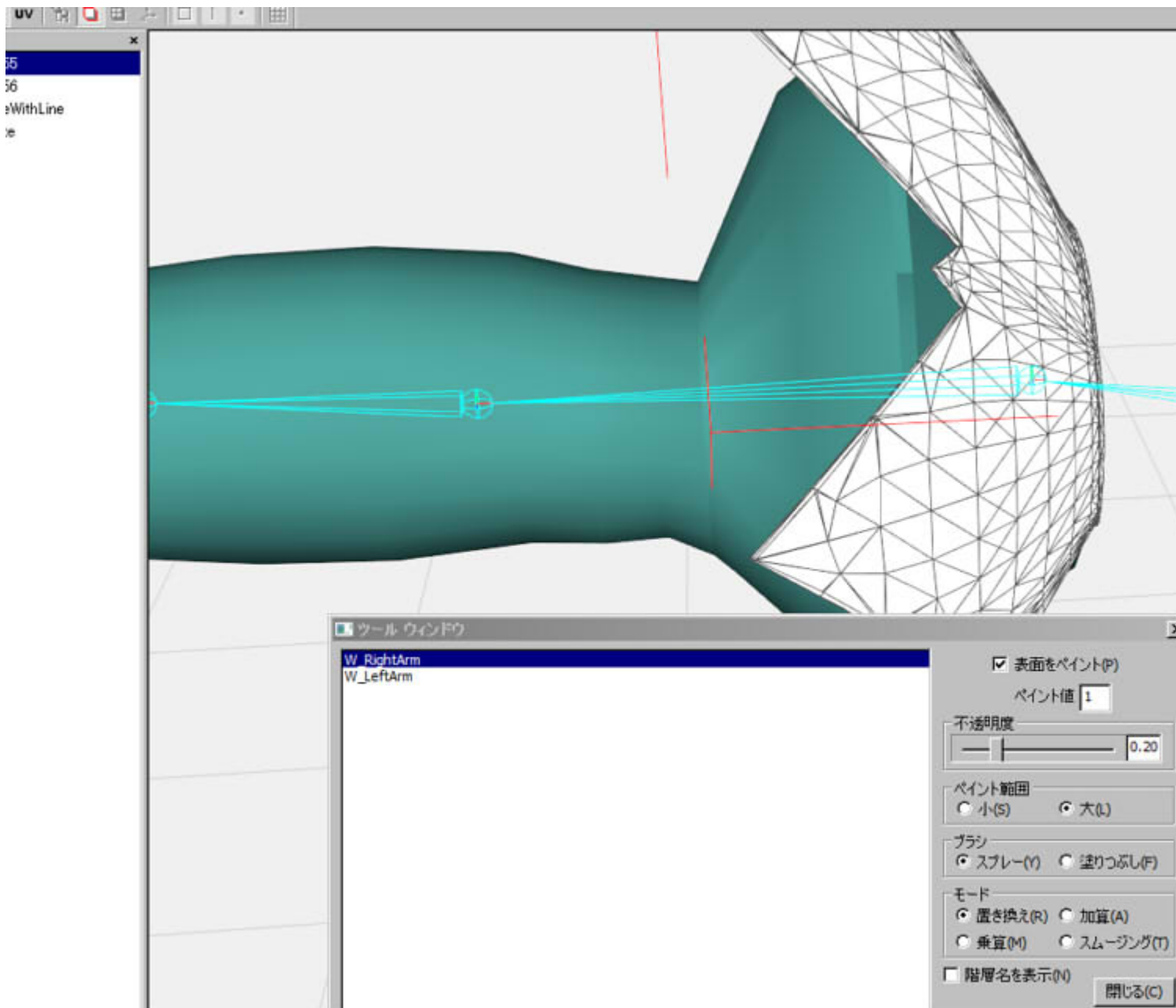
TOYSTUDIOによる頂点ウェイト設定には、大きくわけて以下の三種類があります。

- スムーズスキニング
- ウェイトペイント機能による指定
- 影響範囲指定オブジェクトによる指定

このケースでは、一番簡単なスムーズスキニングによる修正を試みます。



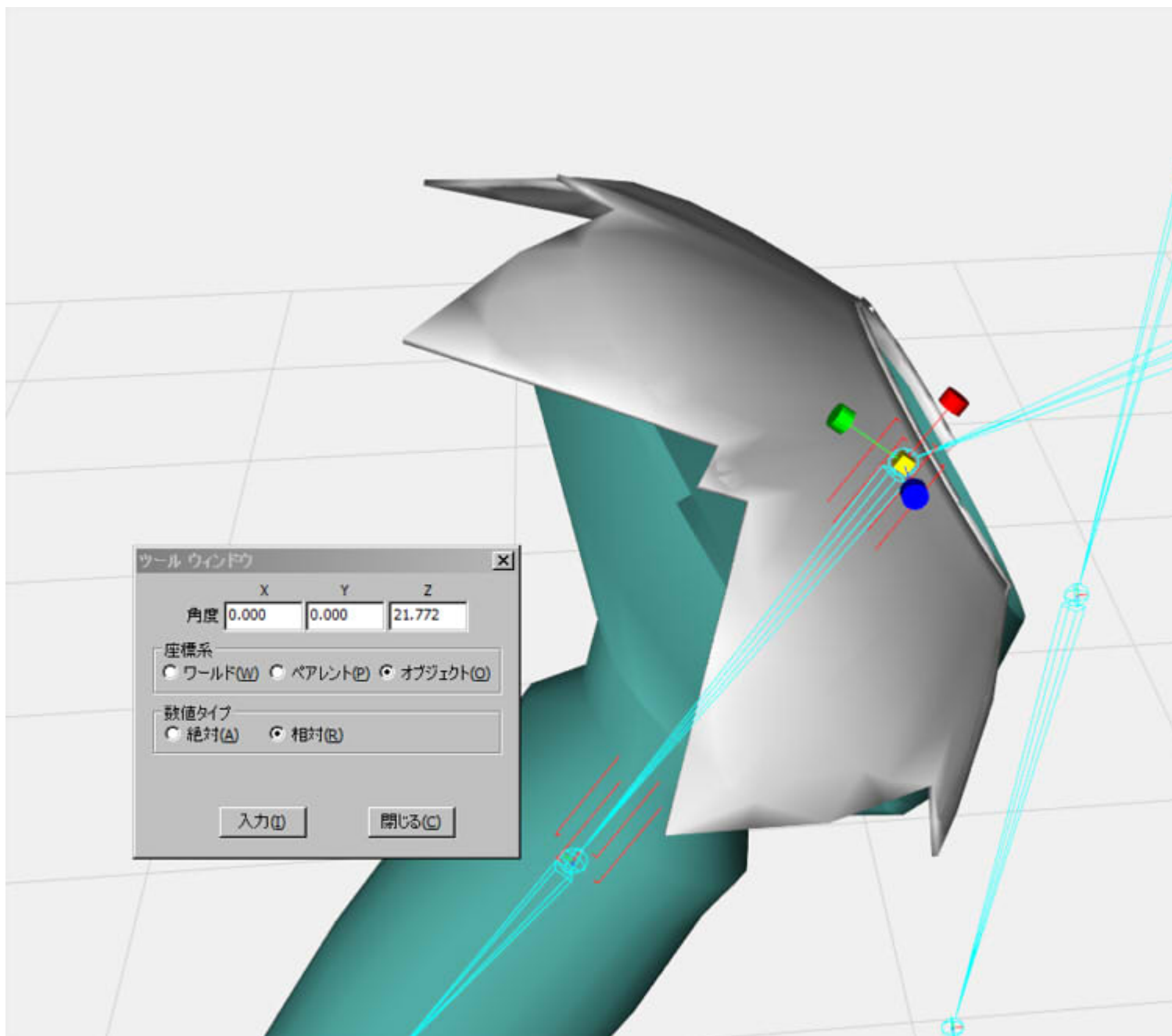
スムーズスキニングモードで、オブジェクトArmFrillに対して、W_RightArm,W_LeftArmの2つのボーンだけを設定してスキニングし直します。



スキニングが完了しました。

修正結果の確認

ポーンを動かしてみて、修正結果が予想通りであることを確認します。



tsoの更新

tso2toyを使用して、ウェイト情報を更新した新しいtsoを作成します。
TAH Decripterなどを使用して、TAHに梱包し、実行環境に配備します。

実行環境での確認

3Dカスタム少女を実際に起動して、修正結果を確認します。



この例では、フリルの歪みが無くなりました。

tso2toyユーティリティ

tso2toyユーティリティは以下の機能を提供します

- tsoファイルからTOYSTUDIOファイルの生成

tsoファイルから直接、TOYSTUDIOの内部バージョン9のファイルを作成します。
テクスチャファイルは生成しないため、tso2mqoなどで事前にtsoからエクスポートしておく必要があります。

- TOYSTUDIOファイルからtsoファイルへの頂点ウェイトの書き戻し

参照TSOの内容にTOYSTUDIOで編集された頂点ウェイトの情報を上書きし、新しいtsoファイルを作成します。

コマンド

tso2toy x 読み込みtsoファイル 出力toyファイル
tsoファイルからTOYSTUDIOファイルを生成します。

tso2toy c 参照tsoファイル 参照toyファイル 出力tsoファイル
新しいtsoファイルを作成します。

tso2toy cs 参照tsoファイル 参照toyファイル 出力tsoファイル
新しいtsoファイルを作成します。
cコマンドとの差は、サブメッシュ再構成を行うため、1オブジェクト16ボーンの制限が無いことです。(試験機能)

実行環境

tso2toyの実行には、.NET Framework (バージョンいくつだ(笑)。調査します...) が必要です。
ログの出力には Apache Foundationのlog4netを使用しています。

更新履歴

2008/09/27 a0.1 初回リリース版
2008/09/28 a0.11 p20080927-001修正
2008/10/04 a-r3 csコマンドの追加

障害記録

#	報告日	ステータス	対応v	内容	原因	対処
p20080927-001	2008/09/27	対応済み	a0.11	ポリゴンを一つも持たないオブジェクトがあったばあい、tsoファイル作成時に例外が発生	TOYSTUDIOファイルはポリゴンを持たないオブジェクトにはBoneWeightsタグを生成しないが、tso2toyが対応していなかった	プログラム修正
p20081001-001	2008/10/01	対応	a-r3	出力TOYSTUDIOファイルのFecesタグCount属性値が3倍の数値になっている。	プログラムバグ	プログラム修正

問題識別の基礎資料

tso2toyが異常終了した場合などの問題識別に使用する資料は以下の三種類です。
作者が暇だったら追跡するかもしれませんが。

- ログ

tso2toy.exeをインストールしたディレクトリに、tso2toy.yyyymmdd.log.txtとして、テキスト形式で記録されます。
このログはteo2toyの使用時には常に書き込まれており、巨大化する傾向があるので、時々削除してください。

TOYファイル、TSOファイル

一般に変換ミスが会った場合に比較用の資料になります。

• MQOファイル

補助資料になる場合があります。

無保証

tso2toyには動作に関して一切の保証はありません。

当ユーティリティを動作させたことにより受けた損害に対して、一切の保証は無いものとします。

謝辞

tso2mqoユーティリティ作者様に、tso2mqoがなければこのモジュールもありませんでした。

フェースのストリップなどにtso2mqoのソースコードを使用しています。

TOYSTUDIO作者ピエール様に、秀逸なモーション統合環境が、これだけの軽い環境で動作しているのは驚き以外にありません。

tso2toyユーティリティの制限事項とTIPS

• TOYSTUDIOでの編集制限

TOYSTUDIOでは頂点ウェイト以外の編集を行ってはならない

• サブメッシュ内の影響ボーン数の制限("c"コマンド)

tso2toyはcコマンドでtso作成時、参照tsoのサブメッシュをそのまま使用し、サブメッシュ内で頂点に対する影響度の総和が大きい16個だけを採用する。

省かれたボーンがある頂点では、残った影響の総和が1になるように正規化されるが、ボーン影響が無くなったことに代わりはないため、このようなケースではTOYSTUDIOでの表示と必ずしも同じ動作をしない。

tso2toyがいくつかのボーンを無視した場合、ログ、またはコンソールに以下のような警告が出力されている。

```
2008-09-27 05:33:27,490 WARN : AMI_TEBUKURO_C.submesh(0) で参照ボーン数が16個を超えました(21個)。以下のボーン影響を無視します。
```

```
2008-09-27 05:33:27,492 WARN : ignore W_RightHandIndex4(50) 影響度総和:12.86225
```

この制限への対策は、以下の通り。

csコマンドでtsoを作成する、(a-r3追加機能)

スキニング時に影響させたいボーンを16個以内にして選択する。

メタセコイアでオブジェクトをあらかじめ分割しておく。

メタセコイアで、1) オブジェクトが別、2) マテリアルが別、の条件のいずれかを満たせば、そのポリゴンは必ず別サブメッシュに配置される。

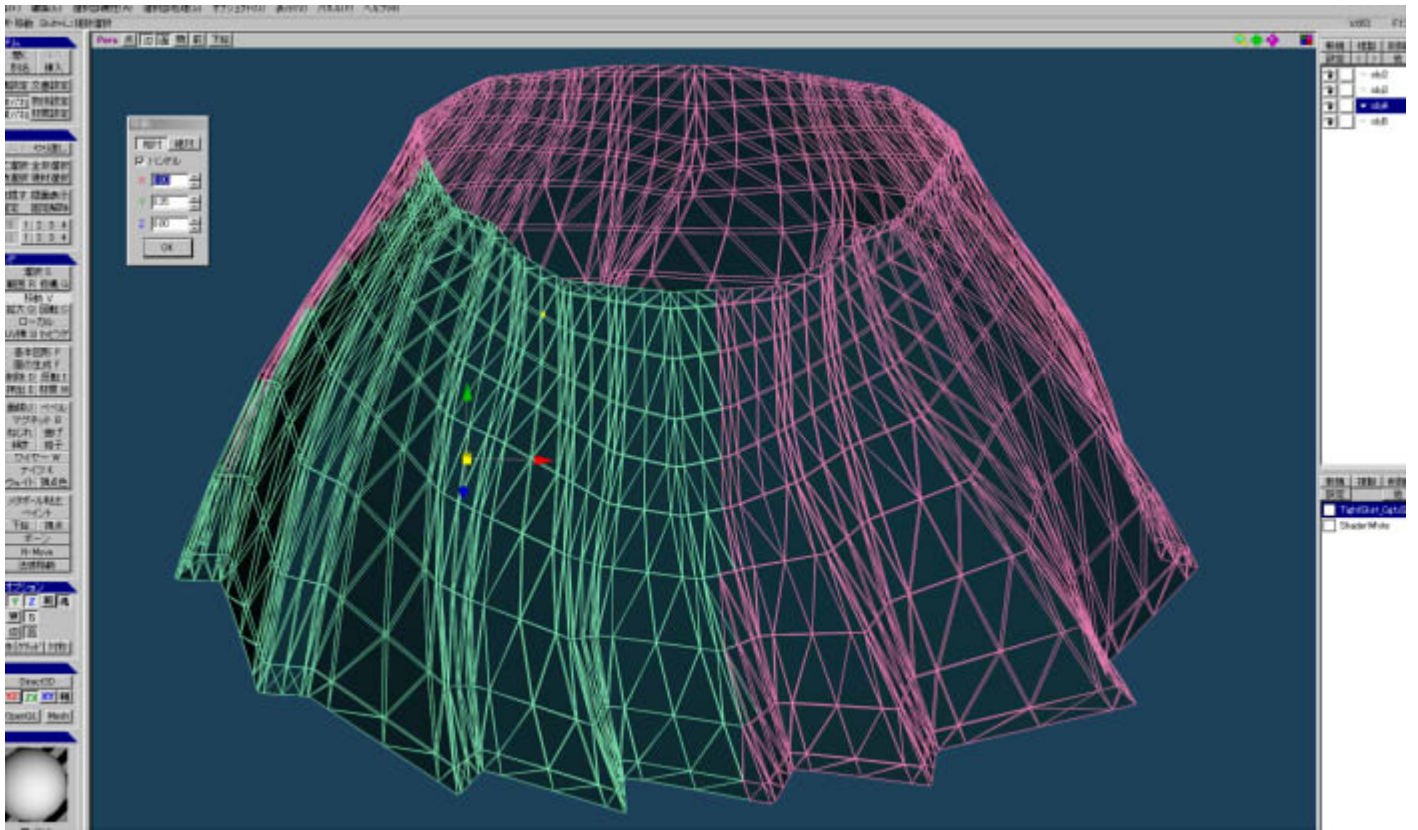
この特性を理解して、全身を覆うようなオブジェクトは細かく分割してそれぞれに対してスキニングを行って16ボーンの制限を守る。

たとえば、スカートなどでは無理矢理右半分と左半分にオブジェクトを分割するだけで、参照するボーン数がそれぞれ半分近くに減ることになる。

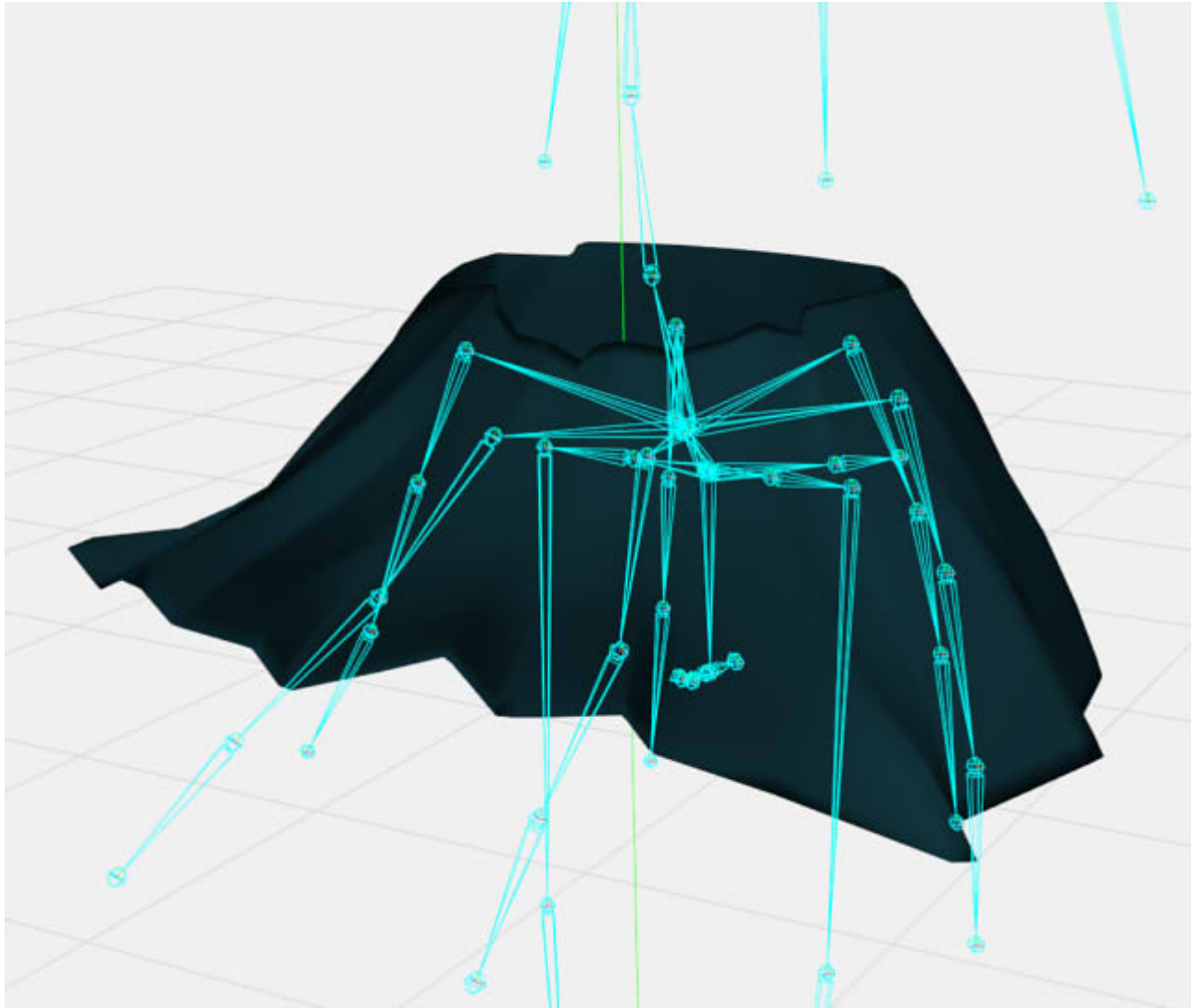
以下の例のスカートでは、スカート全体を左右前後で4オブジェクトに分割してから、**それぞれのオブジェクトに対して、スカートの全てのボーンを使って**スムーズスキニングを行った。

たとえばスカートの右前部分は、スカート左後ろのボーンの影響範囲から外れているため、スキニング時に指定しても全ての頂点に対して影響度が0になる。

オブジェクト内の全ての頂点への影響度が0のボーン影響なら、無視してもよいわけだ。



スキニングが正しく行われていれば、別々のオブジェクトであってもスムーズにつながって変形する。



tso2toyのサブメッシュ再構成機能

•

a-r3で追加されました。厳密な試験を行っていませんのであくまで試験的機能ってことでひとつ。

• a-r3 リリースノート

確認されている障害と対処について

【現象】

a0.11で作成したTOYFILEを読み込むと、「Faces に設定されたパラメータ数がCount属性からの逆算値と一致しません。」致命的エラーになる

【原因】

障害番号 #20081001-001による影響です。

tso2toy a0.11で作成したTOYSTUDIOファイルはフェース数の計算にバグがあり、Count属性の数値が3倍になっていることがあります。

【対処】

下記のいずれかの対処を取ってください。

- 1 . TOYSTUDIOファイルを当バージョンで生成し直す
- 2 . エラー報告にあったメッシュ名をもつMeshタグの下にあるFacesタグのCount属性値を手作業で3で割った数値にする