

MD3 キャラクタについて

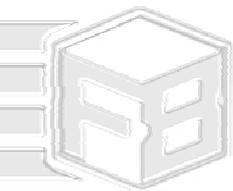
ファイル形式について

MD3 は元々Quake3 エンジンで使われていたモデルフォーマットです。それ以来 MD3 は Quake3 モデルエンジンの編集ツールだけでなく、他のソフトウェアパッケージの一部ともなりました。そのフォーマットはまずアニメーションキャラクター用に使われ、多くの動きのあるオブジェクトにも用いることができます。他のフォーマットとは違い、アニメーションは骨格に依存せず、頂点に依存し、MD3 を通じてメッシュに直接蓄積されます。

UC-win/Road は、人間や色々な動物のようなキャラクターモデルに MD3 フォーマットを使っています。ユーザーは色々な目的のためにそれらを作成することができます。MD3 コンポーネントやテクスチャ、アニメーションリファレンスファイル等のキャラクターファイルは'.PK3'ファイル形式にコンパイルされます。それは'.zip'ファイルのような単純な圧縮ファイルです。

説明の概要：

- MD3 キャラクタ作成によく使われるソフトウェア
- MD3 キャラクタのコンポーネント
- MD3 キャラクタ作成の基礎
- 色々なソフトウェアパッケージ間の作業
- 自作の MD3 モデルの UC-win/Road への取り込み
- MD3 モデルのテストとトラブルシューティング



MD3 キャラクタ作成によく使われるソフトウェア

Autodesk 「3D Studio Max」

3Ds Max は主要な 3D モデリングソフトウェアとして使われています。このソフトウェアを使って効果的にキャラクタをモデリングし、アニメーションを作るために、多くの情報が利用可能です。Max はモデルテクスチャを作成する際にも効果的なプログラムで、UVW Mapping やアンラッピングのような機能を使用できます。さらに、3Ds Max の現在のバージョンは、以下のものも含んでいます。

Autodesk 「キャラクタスタジオ」

キャラクタスタジオ (以下 CS) は強力なアニメーション作成ソフトで、複雑なアニメーションを作成することができます。モーショキャプチャの形式など、主なファイル形式で高度な互換性があります。

注記：キャラクタをモデリング、アニメーション化するソフトウェアプログラムは多くありますが、ここでは、3DsMax に関連する情報について記載しています。

chUmbaLum soft 「MilkShape 3D」

Milkshape は MD3 コンポーネントをエクスポートする時に使われるソフトウェアパッケージです。様々なファイル形式で作業できるよう、強力なインポート/エクスポート機能があります。また、MD3 コンポーネントに必要な構成ファイル作成にも使用します。この説明書は Max から Milkshape のインポートについても記載しています。

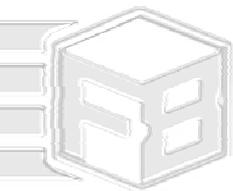
注記：Milkshape は Max、UC-win/Road 間のインポートに必ず必要なわけではありませんが、Max 最新版にはない様々な機能があります。

Adobe 「Photoshop」

3D モデルを使ったテクスチャを作成する標準的なソフトです。「Illustrator」のような他の Adobe ソフトウェアも一般的です。

ソフトウェアの圧縮 (WinRar)

UC-win/Road にキャラクタコンポーネントをインポートするには、圧縮ツールが必要です。



MD3 キャラクターのコンポーネント

一般に、MD3 はモデルタイプだと考えられていますが、MD3 ファイルはキャラクターパッケージの一部です。実際にキャラクターを作成するには3つのMD3ファイルが必要です。

MD3 ファイル

キャラクターの基本構成要素です。その形状とアニメーション情報などが入っています。

キャラクターは、3つのファイル頭・胴体・脚に分けられています。3つすべてのファイルが必要ですが、実際の情報は大抵の場合1つのMD3ファイルに保存されています。

アニメーション情報を保持しているメッシュオブジェクトを1つエクスポートし、MD3ファイルを作成します。キャラクター全体で、メッシュは3つまでに制限されています。

テクスチャファイル

1つのキャラクターに1つのテクスチャを使います。コンポーネントは全て1つのテクスチャに作成する必要があります。

キャラクターに様々な「スキン」をつけられるよう、テクスチャファイルを複数組み込むことができます。ファイル形式はJPEG (.jpg)あるいはBitmap (.bmp)のどちらかが推奨されますが、TARGA ファイル (.tga)でも動作します。

スキンファイル

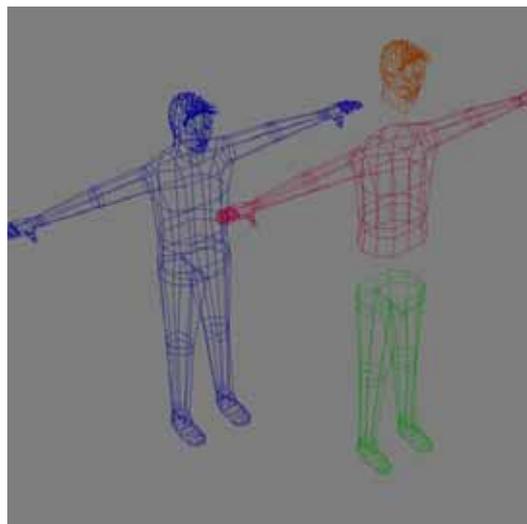
これらはファイル拡張子.skin の設定ファイルで、テクスチャにアクセスするMD3を基本的に示します。各テクスチャファイルのMD3パーツごとにスキンファイルが必要です。テクスチャが1つの場合、スキンファイルは3つ必要になります。テクスチャが2つの場合、6つ必要になります。

アニメーション設定ファイル

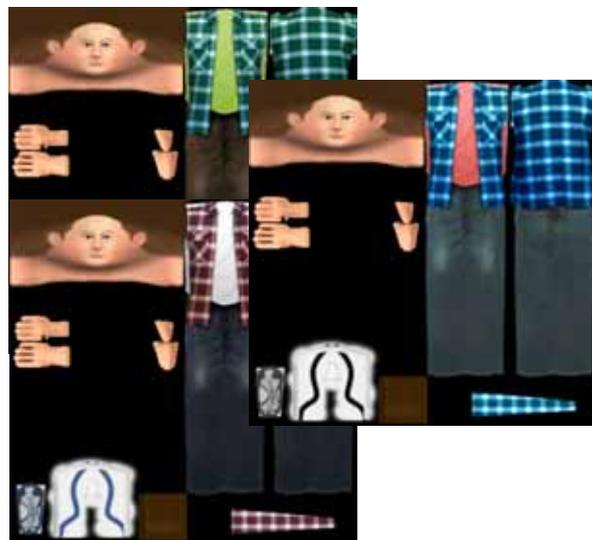
キャラクターのすべての動作は1つのアニメーションにあるので、どのアニメーションがどのキーフレームに連結するかということをソフトウェアに示す1つのファイルです。

アイコンファイル

64x64ピクセルの'tga'ファイル1つから成り立っています。ソフトウェアにアクセスし、キャラクターを表すアイコンとして動作します。



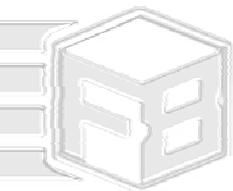
メッシュ (1つ、複数の場合)



テクスチャのサンプル



アイコンのサンプル



MD3 キャラクタ作成の基礎 モデル (3D Studio Max)

Max の作業終了後、処理を続ける前に留意しなくてはならない点がいくつかあります。このチュートリアルはキャラクタのモデリング、テクスチャリング、アニメーション化についてのものではありません。そういった作業については、オンラインに数え切れないほどのリソースがありますので、そちらを参照してください。

メッシュ

モデルは 3 パーツ以下でなければなりません。先ほど述べたとおり、MD3 は 3 パーツから成り立っていますが、各パーツは別々の MD3 ファイルになります。ほとんどの場合、1 つのメッシュが許容されますが形状/アニメーションは除外されます。

頭・胴体(+腕)・脚という区分が標準として推奨されます。

テクスチャ

モデルを最大限に細かくパーツ分けしたい場合でも、メッシュ全体で 1 つのテクスチャしか使うことができません。(アニメーション化のような)次の段階にすすむ前に、メッシュを 1 パーツにするということになります。全体をひとつにアンラップします。3DsMax2008 以降をご利用の場合は、複数のメッシュをアンラップできるので問題ありません。

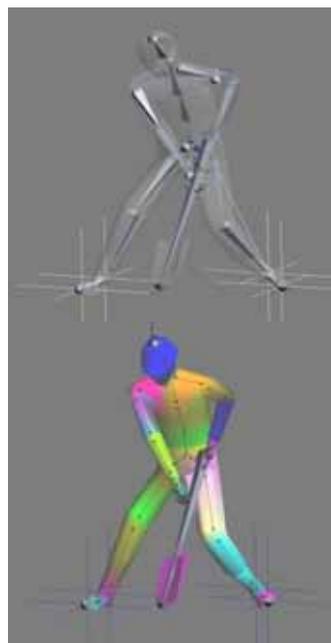
3D キャラクタは通常アンラップされ、テクスチャ化されています。

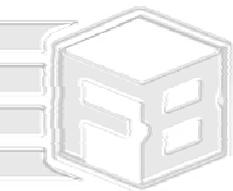
アニメーション

スキンモディファイヤや physique モディファイヤを使うと(アニメーションタイプによります)この段階で Max にアニメーションを作成する骨組を使うことができます。メッシュに駆け足/歩行/など、様々な動作にする場合も、1 つのアニメーション時間にしなければなりません(各コンポーネント)。

繰り返しのアニメーションにしたい場合(歩行や駆け足)繰り返す部分の頂点を操作しないことが大切です。最初のキーフレームと動作の最後のキーフレームが同じ動作(形状)でないといけません。歩行アニメーションの実際の移動は、UC-win/Road によって行われます。

1 つのアニメーションにしたい場合(キック/ジャンプなど)、つなぎ目を含むメッシュをアニメーション化することができ、UC-win/Road と対応して設定します。





MD3 キャラクタ作成の基礎 モデル (3D Studio Max、MilkShape)

必要不可欠というわけではありませんが、MilkShape はモデルを完全な MD3(.pk3 形式)に変換することのできる素晴らしい機能を備えています。

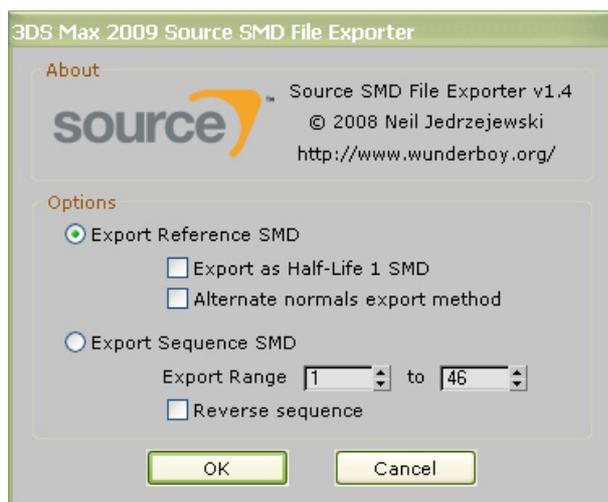
いずれの場合でも、MilkShape にエクスポートするには、3DsMax に SMD エクスポーターをダウンロードする必要があります (Half-life ファイル形式);wunderboy.org には異なるバージョンのファイルがあります。

エクスポートするには

1. 全てのコンポーネントを選択(骨、メッシュなど)
2. *.smd 形式を選んで「選択したものをエクスポート」

必要なパーツは2つあります。1つはアニメーション情報、もう1つは実際のメッシュ情報です(それらには拡散テクスチャが割り当てられていなければなりません)

3. 「Export Reference SMD」を選択(初期設定)。メッシュ情報だとわかるような名前をつけます。
4. 同じものを選択し、SMD をエクスポートを再び選択します。今回は、「Export Sequence SMD」を選択。初期設定のままにすることが推奨されます(必要なればエクスポート範囲は特に)。アニメーション情報だということがわかるような名前をつけます。



エクスポートリファレンス: メッシュコンポーネントに使用

エクスポートシーケンス: アニメーションに使用

MilkShape へのインポート

インポートするには

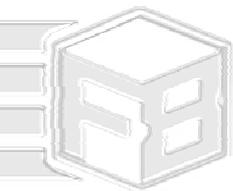
1. Milkshape を開き、ファイル→インポートを選択
→ Half-Life SMD
2. SMD でアニメーションを選択。「Triangles」のチェックを外す。
3. アニメーションを確認し、右下の「Anim.」ボタンを作動させ、骨の動作を確認できるようスライダをドラッグします。「Anim」の動作を無効にします。
4. ここで、メッシュ SMD をインポートし、「Triangles」のチェックを入れ「Skeleton」のチェックを外します。
5. メッシュが MilkShape にインポートされます。Anim ボタンを使ってスケルトンとのリンクを確認します。
6. メッシュに関連する名前の余分な骨格ができます。それを削除してください。

アニメーションファイルの SMD オプション



メッシュファイルの SMD オプション





MD3 キャラクタ作成の基礎

モデル (MilkShape)

タグ付け

MD3 パーツをどのように組み立てるかは、タグからエンジン (この場合 UC-win/Road) に指示します。メッシュキャラクタが 1 つの場合と複数の場合で方法が異なります。

メッシュが 1 つの場合

メッシュが 1 つの場合、メッシュ内にボックスを 2 つ作成します。腹部部分に上下に重ねて配置するのが一般的です。小さい方を上に置きます。実際のサイズでなくてかまいません。アニメーションで、メッシュの外側にはみだしていないかを確認して下さい。必要であればそれを骨に設定します。

メッシュが複数 (3 つ) の場合

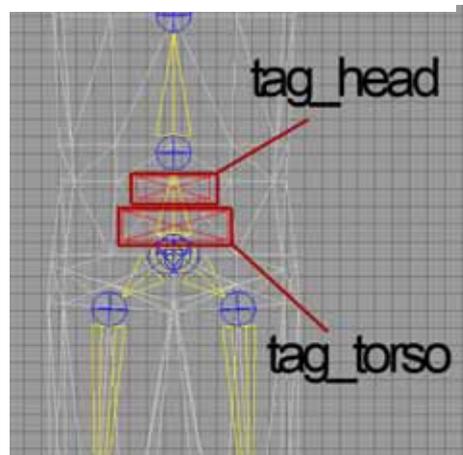
メッシュが複数の場合、タグ付けは少し複雑になります。三角形を作成し、メッシュのつなぎ目 (通常首や腰) に配置します。どの位置でも、短い辺が肩方向を、長い辺が正面方向を差します。オブジェクトの中心近くに 90 度の角度を作り、三角形になります。この三角形はモデルの片側のみになります (どちら側にするかは何度かやり直して修正して下さい) 。

エクスポートする時に、少し問題が生じます。何もしないと、UC-win/Road でメッシュの組み立てがうまくいかないようになっていきます。MilkShape のマッピングに従ってコンポーネントとタグを動かして、UC-win/Road で正しい組み立て方になるよう、回転させます。

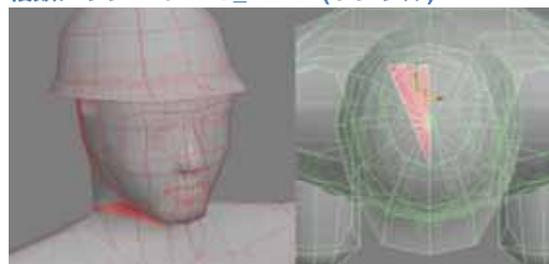
タグには一般的な名前がつけられています。頭/胴体はまとめて tag head、胴体/脚のタグは tag torso といいます。

この時に、メッシュコンポーネントの名前を確認します。頭は h head、胴体は u torso、脚は l legs と名前が付けられます。コンポーネントが 1 つしかない場合、l legs と名前を付けます。

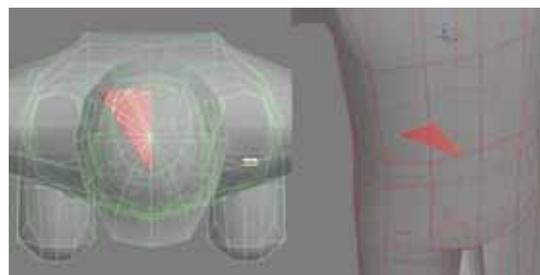
メッシュが 1 つのタグ付け

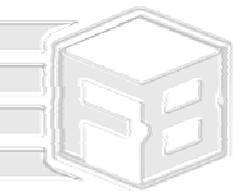


複数メッシュの TAG_HEAD (サンプル)



複数メッシュの TAG_TORSO (サンプル)





MD3 キャラクタ作成の基礎

モデル (MilkShape)

エクスポートの準備

タグの場所、パーツの名前、アニメーションが正しいことを確認すると、エクスポートの準備は完了です。

MD3 の頭から作業をはじめます。

メッシュが 1 つの場合:

- 1) tag_head を選択。
- 2) Tools → Quake III Arena → Generate Control File を選択します。

Head.qc と名前を付け、それを開きます (自動的に開かない場合)

表示されたとおりに編集します (編集する箇所は少ししかありません)

```
// Quake III Arena MD3 control file, generated by MilkShape 3D
//
$model "models/players/EXAMPLE/head.md3"
// reference frame
//$frames -1 -1
// frame 1-100
$frames 1 100
$flags 0
$numskins 0
```

```
// you can have one or no parent tag
//$parenttag "tag_head"
```

```
// tags
$tag "tag_head"
```

```
// meshes (surfaces)
```

IMPORTANT: in the folder structure "example" this will be your actual character's name as read by UC-win/Road. Further, you have removed any references to a mesh or material as well as information about the torso tag.

重要：フォルダ構造「example」の場合、UC-win/Road に読み込まれる実際のキャラクタ名になります。

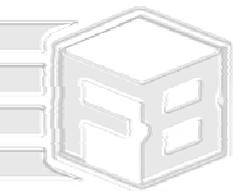
- 3) ファイルを保存します。選択した同じオブジェクトで、File → Export → Quake II Arena MD3 を選択します。QC ファイルと同じ場所に head.md3 をエクスポートします。

メッシュが複数の場合:

メッシュが 1 つの場合との違いはあまりありません。コントロールファイルと MD3 を作成するとき、選択した tag head と h head(メッシュ)の両方あるかたちになります。

重要：フォルダ構造「example」の場合、UC-win/Road に読み込まれる実際のキャラクタ名になります。また、torso tag の情報と同じようにメッシュや

また、コントロールファイルは少し異なっているようにみえます。これを最後に加えます。



```
// meshes (surfaces)
$mesh "h_head"
$skin "models/players/EXAMPLE/EXAMPLE.tga"
$flags 0
```

「EXAMPLE」をキャラクタ名と実際のテクスチャ（適切な拡張子で）に置き換えます。

MD3 キャラクタ作成の基礎

次に胴体の作業を行います。

メッシュが1つの場合

- 1) tag_head と tag_torso を選択。
- 2) Tools → Quake III Arena → Generate Control File を選択します。

upper.qc と名前を付け、それを開きます(自動的に開かない場合)

表示されたとおりに編集します（編集する箇所は少ししかありません）

```
// Quake III Arena MD3 control file, generated by MilkShape 3D
//
$model "models/players/EXAMPLE/upper.md3"
// reference frame
//$frames -1 -1
// frame 1-100
$frames 1 100
$flags 0
$numskins 0
```

```
// you can have one or no parent tag
//$parenttag "tag_torso"
```

```
// tags
$tag "tag_head"
$tag "tag_torso"
```

```
// meshes (surfaces)
```

- 3) ファイルを保存します。選択した同じオブジェクトで、File → Export → Quake II Arena MD3 を選びます。QC ファイルと同じ場所に upper.md3 をエクスポートします。

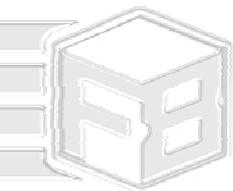
メッシュが複数の場合

メッシュが1つの場合との違いはあまりありません。コントロールファイルと MD3 を作成するとき、選択した tag head と tag torso、u torso(メッシュ)すべてがあるかたちになります。

また、コントロールファイルは少し異なっているように見えます。これを最後に加えます。

```
// meshes (surfaces)
$mesh "u_torso"
$skin "models/players/EXAMPLE/EXAMPLE.tga"
$flags 0
```

「EXAMPLE」をキャラクタ名と実際のテクスチャ（適切な拡張子で）に置き換えます。



MD3 キャラクタ作成の基礎

最後に脚の作業を行います。

メッシュが1つの場合

1) tag_torso と l_legs を選択。

これはメッシュデータを保存するファイルになります。

2) Tools → Quake III Arena → Generate Control File を選択します。

lower.qc と名前を付け、それを開きます(自動的に開かない場合)

表示されたように編集します(編集する箇所は少ししかありません)

```
// Quake III Arena MD3 control file, generated by MilkShape 3D
//
$model "models/players/EXAMPLE/lower.md3"
// reference frame
//$frames -1 -1
// frame 1-100
$frames 1 100
$flags 0
$numskins 0

// you can have one or no parent tag
//$parenttag "tag_torso"

// tags
$tag "tag_torso"

// meshes (surfaces)
$mesh "l_legs"
$skin "models/players/EXAMPLE/EXAMPLE.tga"
$flags 0
```

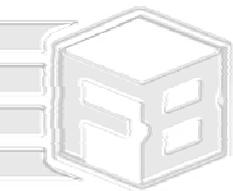
「EXAMPLE」をキャラクタ名と実際のテクスチャ(適切な拡張子で)に置き換えます。

3) ファイルを保存します。選択した同じオブジェクトで、File → Export → Quake II Arena MD3 を選びます。QC ファイルと同じ場所に lower.md3 をエクスポートします。

メッシュが複数の場合

メッシュが1つの方法にはこの MD3 にあるメッシュが含まれているので、メッシュが複数の場合も同様になります。

この段階で、(QC のある)MD3 がすべて作成され、フォルダ構造に配置されていることになります。Model/players/EXAMPLE



MD3 キャラクタ作成の基礎 スキンファイル

スキンファイルは使用マップごとに計3つあります。また、キャラクターは1つのテクスチャにまとめてマップする必要がありますが、選択したテクスチャ（単数/複数）が可能です。例はビジネススーツとカジュアルな服装のテクスチャです。また、UVW マッピングも残ります。

処理は一般的に各パーツで同じですが、（特定の MD3 と参照テクスチャに依存する）特定の MD3 の内容とファイル名が違います。

スキンファイルを作成するには：

- 1) Notebook を開きます
- 2) 下記を入力します

頭コンポーネント（メッシュが1つの場合）：

models/players/EXAMPLE/EXAMPLE.tga
tag_head,

上半身コンポーネント（メッシュが1つの場合）

models/players/EXAMPLE/EXAMPLE.tga
tag_torso,

脚コンポーネント（メッシュが1つ/複数の場合）

l_legs.models/players/EXAMPLE/EXAMPLE.tga
tag_torso,

- 3) 保存。頭のスキンは head default.skin、上半身コンポーネントは upper default.skin、脚は legs default.skin として保存します。

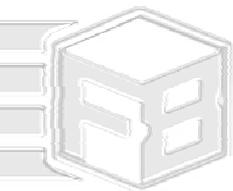
メッシュが複数の場合：

メッシュの名前（h head や u torso）とカンマをファイルに追加します。h head,models/...か u torso.models/...になります。
理解するには脚コンポーネントスキンファイルを参照して下さい。

テクスチャが複数の場合

変更点は2つのみです。まず、各ファイルのテクスチャ名を他のテクスチャ情報に変更します。例：/EXAMPLE-2.tgs

次に、各コンポーネントファイルの固定部分（コンポーネント名）を維持しながら違う名前でも保存する必要があります。Head NEW.skin のようになります。
‘NEW’という言葉は、どれにでも適切に使われます。UC-win/Road では、スキンファイル名で‘_’のあとに使われた言葉は、テクスチャドロップに表れます。



MD3 キャラクタ作成の基礎 アニメーション設定ファイル

冒頭で述べたとおり、これは頂点が変形したアニメーションをソフトウェアがどのように処理するのかを説明するファイルです。すべてのアニメーションが1つの情報になり、その情報は頂点に保存されるので、これが必要になります。

アニメーションの基本設定ファイルは下記のとおりです (NOTEPAD):

```
// animation config file

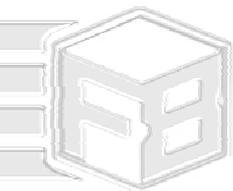
sex      m
headoffset 0 0 0
footsteps normal

// first frame, num frames, looping frames, frames per second

0      0      0      0      // BOTH_DEATH1
0      0      0      0      // BOTH_DEAD1
0      0      0      0      // BOTH_DEATH2
0      0      0      0      // BOTH_DEAD2
0      0      0      0      // BOTH_DEATH3
0      0      0      0      // BOTH_DEAD3
0      0      0      0      // TORSO_GESTURE
0      0      0      0      // TORSO_ATTACK1      (MUST NOT
CHANGE -- hand animation is synced to this)
0      0      0      0      // TORSO_ATTACK2      (MUST NOT
CHANGE -- hand animation is synced to this)
0      0      0      0      // TORSO_DROP      (MUST NOT
CHANGE -- hand animation is synced to this)
0      0      0      0      // TORSO_RAISE (MUST NOT CHANGE --
hand animation is synced to this)
0      0      0      0      // TORSO_STAND
0      0      0      0      // TORSO_STAND2
0      0      0      0      // LEGS_WALKCR
0      0      0      0      // LEGS_WALK
0      0      0      0      // LEGS_RUN
0      0      0      0      // LEGS_BACK
0      0      0      0      // LEGS_SWIM
0      0      0      0      // LEGS_JUMP
0      0      0      0      // LEGS_LAND
0      0      0      0      // LEGS_JUMPB
0      0      0      0      // LEGS_LANDB
0      0      0      0      // LEGS_IDLE
0      0      0      0      // LEGS_IDLECR
0      0      0      0      // LEGS_TURN
```

注記：次の行にはみ出るセクションはドキュメントの同じ行になるようにします ('MUST NOT' ...部分)

このファイルは：animation.cfg として保存されます。



MD3 キャラクタ作成の基礎 アニメーション設定ファイル（続き）

設定ファイルの右側に、動作コマンドがあります。MD3の基本で、これらパーツにマップされる時間ラインが必ず必要です。が、どのように動作するかを理解することが大切です。

4つの0はアニメーションがどのように作成されたかに基づいて設定が必要な4つの数値を表します。

First Frame:アニメーションサイクルの最初のフレームの数です。(歩行、駆け足など)

Num Frames:アニメーションが続くフレームの数。60フレームの歩行サイクルの場合、60を入力します。

Looping Frame:1サイクルで繰り返すフレームの数です。これにより、サイクル開始部分の前にアニメーションの開始部分が可能です。60秒のアニメーションの場合、30のフレームの繰り返しのみが必要で、30を入力します。

Frame per Second:アニメーション一秒ごとのフレームの数。大体30くらいになります。いろいろな速度になるようこの数で操作します。(アニメーションを速くしたり、遅くしたりできます)

固定動作について:

「*BOTH*」:複数メッシュのキャラクタに使う動作。同じアニメーション情報を使うよう、これは胴体と脚の両方のメッシュを操作します(上記情報)。動作がメッシュ全体に組み込まれたときに役立ちます。

「*TORSO*」:複数のメッシュキャラクタに使われる動作。これにより胴体の動作のみを操作します。MD3が分かれている方がよいという例です。

「*LEGS*」:メッシュが1つでも複数でも使われる動作。複数メッシュのキャラクタは脚にのみ影響しますが、メッシュが1つの場合全身に影響します。

EXAMPLE:

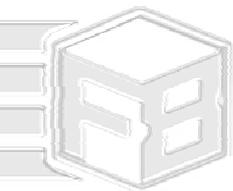
歩行アニメーションサイクルの設定

基本的な歩行サイクルがわかると(フレーム、フレーム番号、フレームのループ、秒ごとのフレーム)構成ファイルを方向付けることができます。

90フレームのアニメーションで、開始フレームが30フレーム、繰り返しフレームが20フレーム(繰り返しフレーム数70) 毎秒25フレームの場合。'// LEGS_WALK'の行で、4つの0に次の値を入力します(スペースはそのまま)

30 90 70 25

キャラクタを UC-win/Road にインポートする時、Legs Walk コマンドを選択します。キャラクタはフレーム形状と同じ方向に歩きます。UC-win/Road の構成ファイルも確認します。



ファイルの完成

アイコンファイル：

必要な最新コンポーネントはアイコンファイルです。冒頭部分で説明したことを繰り返すようですが、アイコンは64x64ピクセル(24ビット)の.tgaファイル1つです。UC-win/Roadメニューで表示される画像になります。

ファイルのサムネイルを使うことが推奨されています(頭を中心にして)

icon_default.tga としてファイルを保存します。

ファイルとフォルダ構造

キャラクターに必要なフォルダ構造は下記のとおりです。

models/players/EXAMPLE/

UC-win/Roadでキャラクター名として「EXAMPLE」という言葉が現れます。このフォルダではスペースや数は使わないよう推奨されます。

また、エクスポートする前にこれを適当な位置に置きます。多くのファイルにこの情報が入っていますが、後で変更するよりエクスポート前に作業の方が簡単です。

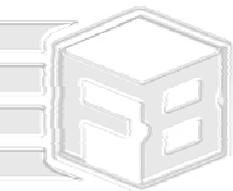
フォルダにある最新ファイル

下記のファイルがあることを確認して下さい。

- Head.md3
- Upper.md3
- Lower.md3
- Head.qc
- Upper.qc
- Lower.qc
- Head_default.skin
- Upper_default.skin
- Lower_default.skin
- Animation.cfg
- Icon_default.tga
- EXAMPLE.tga (ファイルで使用したテクスチャ)

(追加テクスチャや付属スキンファイルを加える。)

.QC ファイルを移動します。必要に応じて他の場所に保存することもできます(3つのフォルダ設定以外に)。これらは最終的なコンパイルに必要ありません。



拡張子.PK3 にファイルを圧縮 Winrar でインストール

- 1) 端の方にある「モデル」フォルダを右クリックします。
- 2) 「アーカイブ」を選択 - Winrar で利用できる機能
- 3) ダイアログボックスで、アーカイブ形式を ZIP に選択
- 4) 「アーカイブ名」フィールドに適切な名前をつけます。圧縮方法は「通常」のままにし、OK をクリックします。

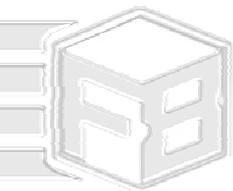
ファイルの圧縮 (zip 形式) を完了する時 (エラーが起きないように少し時間を置きます) zip ファイルを右クリックし、拡張子を .zip から .pk3 に変更します。

フォルダオプションの「hide file extensions for known file types」機能を無効にします。

変更を確認します。

.PK3 ファイルが完成し、UC-win/Road にインポートすることができます。

ファイルを UC-win/Road に取り込みます。



自作の MD3 モデルの UC-win/Road への取り込み

- 1) デフォルトファイル (あるいは他のファイル) を開いたまま、UC-win/Road を開く
- 2) 「MD3 キャラクタの読み込み」アイコンを選択：
- 3) 「読み込み」をクリックし、作成した.pk3 ファイルを選択

MD3 ウィンドウが下記のように表示されます。



エラーが生じた場合は最終ページのトラブルシューティングを参照して下さい。

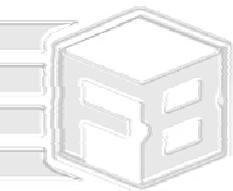
テスト：

1 つあるいは複数のメッシュキャラクターを使うかどうかによって、アニメーション.cfg ファイルでそれをどのようにするかに基づいて動作を選択します。1 つのメッシュオブジェクトに関しては、「下半身の動作」のみがキャラクターに影響します。設定によって利用可能なスキンから選択できます。

使用：

自作の MD3 キャラクターを使うには、「OK」をクリックしてください (設定完了後)。飛行パスの準備をします。「MD3 キャラクターの登録」で、「移動体に追加」を選択。設定を入力して確定します。実際に使えるようにするには、「環境・キャラクターの表示」アイコンを選択。：





MD3 モデルのテストとトラブルシューティング

UC-win/Road ヘインポート時にエラーが生じた場合
エラーが生じる原因は様々ですが、ほとんどの場合簡単に解決できます。

エラーかなと思ったら

- 各 MD3 に正しいパーツがあることを確認して下さい。(メッシュが複数の場合は各メッシュに1パーツずつ、メッシュが1つの場合は1パーツのみ)
- フォルダ名に同じ数字を使わないようにして下さい。エラーの原因となります。
- .zip ファイルを.PK3 ファイルに変換するのが速すぎる時にエラーが生じることがあります。変換する ZIP ファイルをコンパイルした後、少し時間をおきます。
- ZIP フォルダの構成が正しくない場合があります。ファイルは、models/players/EXAMPLE/[files]です。- zip ファイルを解凍するときはその構成になるように気をつけてください。
- パーツがバラバラになってしまうことがあります - 複数メッシュのキャラクタでよく生じるエラーです。MilkShape でタグやオブジェクトが間違っ配置されていることが原因です。前述の通り、この作業は、テストとエラーを繰り返し修正していくものですので、そのエラーを修正していけば問題ありません。

UC-win/Road ヘインポートする時にエラーが生じた場合

このエラーの原因は2つあります。

- モデルが適切にアンラップされ、MilkShape/Max で適切にテクスチャ化されているかを確認して下さい。これはインポート初期段階で生じるエラーです。
- .skin ファイルが正しく、適切にメッシュ・フォルダ・テクスチャを参照しているかを確認して下さい。

人間だけではありません！

人間だけでなく、いろんな動物を UC-win/Road に取り込むことができます。人間以外のモデルを作成するときも人間と同じ方法で行うことができます。

理解を深めるには

FORUM8 の.PK3 コンピレーション (Winrar 使用) を試してみてください。ファイルを開いて中に何があるかをご覧くださいと、アニメーション設定ファイルやスキンファイルに関する説明もありますので参照して下さい。