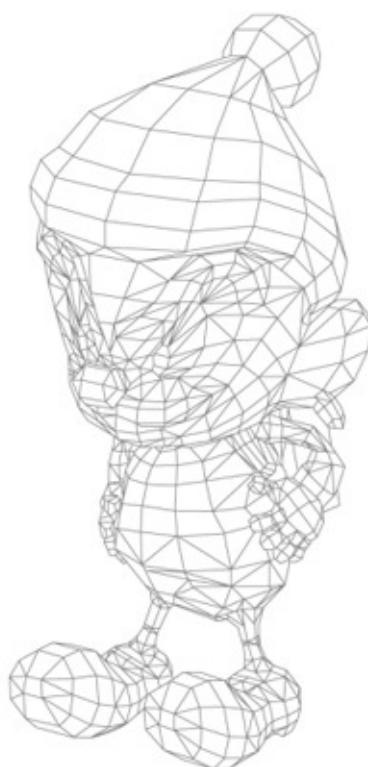

TRA File Ver.4.0 Data Format Specification

MascotCapsuleV3

日本語版



Ver.2.1

—更新履歴—

2003 年 10 月 8 日

- ・ Ver. 1.0 新規作成

2005 年 6 月 15 日

- ・ Ver. 1.0

更新内容

- 文章の一部およびレイアウトが修正

2007 年 3 月 20 日

- ・ Ver. 2.0

更新内容

- 文章およびレイアウトが修正
- サンプルコードを追加

2007 年 6 月 07 日

- ・ Ver. 2.1

更新内容

- 文章およびレイアウトが修正
- ボーンアニメーションのチャックについて補足を加えました。

目次

1. はじめに.....	1
1.1. TRA4 とは.....	1
2. マクロについて.....	2
2.1. マクロ詳細.....	2
2.1.1. [STRING].....	2
2.1.2. [A B].....	2
2.1.3. [INT].....	2
2.1.4. [FLOAT].....	2
2.1.5.	2
2.1.6. ; (セミコロン).....	2
3. ファイル構造.....	3
3.1. TRA4 ファイル構造.....	3
4. 各チャンクについて.....	6
4.1. (Head) チャンク.....	6
4.2. (Figure) チャンク.....	6
4.2.1. (name) チャンク.....	7
4.2.2. (totalFrame) チャンク.....	7
4.2.3. (bone) チャンク.....	8
3.2.3.1. (name [STRING]) チャンク.....	9
3.2.3.2. (translate) チャンク.....	9
3.2.3.3. (rotate) チャンク.....	10
3.2.3.4. (roll) チャンク.....	10
3.2.3.5. (scale) チャンク.....	10
3.2.3.6. (kf) チャンク.....	11
3.2.3.7. アニメーション定義補足.....	11
3.2.3.8. TRA におけるボーンアニメーションの補足.....	13
4.2.4. (DynamicPolygons) チャンク.....	14
3.2.4.1. (kgf [INT] [INT] [true false]) チャンク.....	14
5. サンプルコード.....	15
5.1. TRA4 サンプル 1.....	15
5.2. TRA4 サンプル 2.....	17
5.3. TRA4 サンプル 3.....	19

1. はじめに

本稿は **MascotCapsuleV2/V3** (以下、V2/V3 と記す) の中間ファイルである **TRA ファイル** (以下、TRA と記す) について解説しています。

TRA ファイルとは、V2/V3 で利用するモデル情報を持つデータに適用されたアニメーション情報を記述していきます。ファイルのバージョンは 3.0/4.0 (以下、TRA3/TRA4 と記す) があります。

TRA3 とは V2 で利用するモデルのアニメーション情報を持つデータファイルです。それに対して TRA4 は V3 で利用するモデルのアニメーション情報を持つデータファイルになります。

本稿ではこの TRA4 のデータフォーマットについて詳細を説明します。

1.1. TRA4 とは

TRA4 とはテキストフォーマット形式 (TRA3 はバイナリフォーマット形式) のファイルで、V3 で利用するモデルデータに適用されているアニメーション情報などを文字列で記述していきます。

TRA4 において定義することができるアニメーションは、BAC6 ファイルにおいて定義したボーンに対する平行移動、回転、拡大縮小及びダイナミックポリゴンに対するパターンアニメーションです。モデルデータ情報に関しては BAC6 ファイル (別紙「BAC File Ver.6.0 Data Format Specification」を参照) に定義します。

TRA4 と BAC6 は V3 で利用する対となる中間ファイルです。TRA4 のファイル拡張子は「.tra」です。テキスト形式で扱う文字規約は「ASCII コード」の範囲でサポートしています。

2. マクロについて

ここでは TRA4 の説明及び記述の簡略化のためにマクロ表現を行うことがあります。そのため以下では各マクロについて説明します。

2.1. マクロ詳細

本稿を参照する上で TRA4 のパラメータ部を以下のマクロで定義しています。

2.1.1. [STRING]

[STRING]マクロはヌル終端処理された文字列を意味しています。

文字列はダブルクォーテーション(“”)で囲い、255 文字(byte)以内に収めてください。

2.1.2. [A | B]

[A|B]マクロは A あるいは B を意味しています。

A 又は B はヌル終端文字列で 255 文字(byte)以内に収めてください。

2.1.3. [INT]

[INT]マクロは整数値を表す文字列を意味します。

文字列（数値）の範囲は int 型のビット範囲に収めてください。

2.1.4. [FLOAT]

[FLOST]マクロは浮動小数値を表す文字列を意味します。

文字列（数値）の範囲は float 型のビット範囲に収めてください。

2.1.5. ...

...マクロは任意の繰り返しを意味します。

2.1.6. ; (セミコロン)

; マクロはセミコロンから行末までがコメント文字列であることを意味します。

3. ファイル構造

ここでは TRA4 の基本となる構造と規約について説明します。

3.1. TRA4 ファイル構造

TRA4 は、ファイル識別情報として先頭に定義する「;TRA」と、(Head) チャンクと (Figure) チャンクの2つのチャンクから構成されています。ヘッダーを意味する (Head) チャンクを定義した後に、BAC6 で定義するモデルデータに適用されたボーンアニメーション情報を定義する (Figure) チャンクも定義します。これら定義の順序は変更できません。

(Head) チャンクと (Figure) チャンクはそれぞれに属する子チャンクを持ち、その子チャンクはさらに子チャンクを複数持っています。

これらを踏まえた上で図 1「TRA4 ファイル構造」を参照してください。

【図】1「TRA4 ファイル構造」

```

;TRA

( Head
  ( traVersion [FLOAT] )
)

( Figure
  ( name [STRING] )

  ( totalFrame [INT] )

  ( bone
    ( name [STRING] )
    ( translate.x
      ( kf [INT] [FLOAT] )
      ( kf [INT] [FLOAT] )
      ...
    )
    ( translate.y
      ( kf [INT] [FLOAT] )
      ( kf [INT] [FLOAT] )
      ...
    )
    ( translate.z
      ( kf [INT] [FLOAT] )
      ( kf [INT] [FLOAT] )
      ...
    )
    ( scale.x
      ( kf [INT] [FLOAT] )
      ( kf [INT] [FLOAT] )
      ...
    )
    ( scale.y
      ( kf [INT] [FLOAT] )
      ( kf [INT] [FLOAT] )
      ...
    )
    ( scale.z
      ( kf [INT] [FLOAT] )
      ( kf [INT] [FLOAT] )
      ...
    )
  )
)

```



```

        ( rotate.x
            ( kf    [INT] [FLOAT] )
            ( kf    [INT] [FLOAT] )
            ...
        )
        ( rotate.y
            ( kf    [INT] [FLOAT] )
            ( kf    [INT] [FLOAT] )
            ...
        )
        ( rotate.z
            ( kf    [INT] [FLOAT] )
            ( kf    [INT] [FLOAT] )
            ...
        )
        ( roll
            ( kf    [INT] [FLOAT] )
            ( kf    [INT] [FLOAT] )
            ...
        )
    )

    ( DynamicPolygons
        ( kgf    [INT] [INT] [A | B] )
        ( kgf    [INT] [INT] [A | B] )
        ...
    )
)

```

4. 各チャンクについて

ここでは TRA4 に記述する各チャンクの詳細を説明します。

4.1. (Head) チャンク

(traVersion [FLOAT]) チャンクには TRA のバージョンを記述し、ファイルヘッダーを定義します。
TRA4 はバージョン 4.0 なので、[FLOAT]には **4.0** と文字列で定義します。

【参照】図 2「Head チャンク定義」

```
( Head
    ( traVersion 4.0 )
)
```

図 2「Head チャンク定義」

4.2. (Figure) チャンク

(Figure) チャンクでは BAC6 で定義した各ボーンのボーン座標系からキーフレーム毎のボーンアニメーション及びダイナミックポリゴンのパターンアニメーション情報を定義します。

また (Figure) チャンクはボーンアニメーションやダイナミックポリゴンのアニメーション情報を定義するための子チャンクを幾つか持ちます。

【参照】図 3「Figure チャンク」

```
( Figure
    ( name ... )
    ( totalFrame ... )
    ( bone ... )
    ( DynamicPolygons ... )
)
```

図 3「Figure」チャンク


以下では (Figure) チャンク内で定義する各子チャンクについて説明します。4.2.1～4.2.4 を参照してください。

4.2.1. (name) チャンク

(Figure)チャンク内の(name)チャンクはアニメーション対象のボーン名を 255 文字(byte)以内の文字列で定義します。

ボーンの名前を必ずしも定義する必要はありませんが、定義する場合は(name "")のようにダブルクォーテーションで囲む必要があります。

【参照】図 4 「name チャンク定義」



```
( name "bone" )
```

図 4 「name チャンク定義」

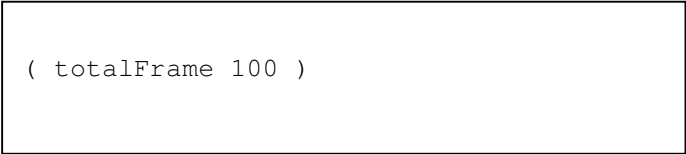
【関連】[4.2.3.1. \(name\) チャンク](#)

4.2.2. (totalFrame) チャンク

(totalFrame) チャンクは対象となるモデルデータに適用されているアニメーションのキーフレームの長さを定義します。

アニメーション対象となるボーンや、ダイナミックポリゴンで定義するキーフレーム値は、ここで定義するフレーム数未満である必要があります。

【参照】図 5 「totalFrame チャンク定義」



```
( totalFrame 100 )
```

図 5 「totalFrame チャンク定義」

キーフレームの有効範囲は符号無し整数値 1～32767 です。

4.2.3. (bone) チャンク

(bone) チャンクは BAC6 ファイル(別紙「BAC6 File Ver.6.0 Data Format Specification」を参照)で定義されているボーンのアニメーション情報を定義します。

(bone) チャンクは BAC6 において定義した対象となる (bone) チャンクの数だけ定義する必要があり、アニメーションの有無に関係なく静止状態を維持する場合においても全て定義する必要があります。定義の順序は BAC6 において定義したボーンの順序と必ず一致していなければなりません。定義の順序が BAC6 における (bone) チャンクと異なる場合、正しいアニメーションを行いません。

(bone) チャンク内では対象となるボーンに移動、拡大縮小、回転のアニメーション情報を定義することができます。変換の順序は

(scale) → (roll) → (rotate) → (translate)

となります。

アニメーション対象となるボーンのアニメーション値は、間に挟まれたキーフレーム値によって**線形補間**されます。

【参照】図 6「Bone チャンク定義」

```
( bone
  ( name "bone" )
  ( translate.x
    ( kf 0 0.000 )
    ...
  )
  ( translate.y
    ( kf 0 0.000 )
    ...
  )
  ( translate.z
    ( kf 0 0.000 )
    ...
  )
  ( scale.x
    ( kf 0 0.000 )
    ...
  )
  ( scale.y
    ( kf 0 0.000 )
    ...
  )
  ( scale.z
    ( kf 0 0.000 )
    ...
  )
  ( rotate.x
    ( kf 0 0.000 )
    ...
  )
)
```

```

        ( rotate.y
            ( kf 0 0.000 )
            ...
        )
        ( rotate.z
            ( kf 0 0.000 )
            ...
        )
        ( roll
            ( kf 0 0.000 )
            ...
        )
    )
    ...

```

図 6 「Bone チャンク定義」

これより以下では（bone）チャンク内で定義する各子チャンクについて説明します。

4.2.3.1. (name [STRING])チャンク

（bone）チャンク内の（name）チャンクにはアニメーション対象となるボーンの名前を定義します。

【参照】図 7 「name チャンク定義」

```

( bone
    ( name "bone" )
    ...
)

```

図 7 「name チャンク定義」

【関連】[4.2.1. \(name\) チャンク](#)

4.2.3.2. (translate)チャンク

（translate）チャンクは、BAC6 で定義した対象となるボーン座標系を原点位置として、ボーンのキーフレーム毎の平行移動成分値（差分）を定義します。

（translate）チャンクは（kf）チャンクを子チャンクとして持っており、（translate）チャンクにおいて定義するアニメーションのキーフレームにおける値は（kf）チャンクを使用します。

(translate.x)	BAC6 における対象ボーン座標系の原点位置 X 成分
(translate.y)	BAC6 における対象ボーン座標系の原点位置 Y 成分
(translate.z)	BAC6 における対象ボーン座標系の原点位置 Z 成分

【関連】[4.2.3.6 \(kf\) チャンク](#)

[4.2.3.7.1. \(translate\) チャンクにおける BAC6 ボーン座標系維持](#)

[4.2.3.8. TRA におけるボーンアニメーション補足](#)

4.2.3.3. (rotate)チャンク

(rotate) チャンクは、BAC6 で定義した対象となるボーン座標系における+Z 軸方向の方向ベクトルを定義します。

(rotate) チャンクは (kf) チャンクを子チャンクとして持っており、(rotate) チャンクにおいて定義するアニメーションのキーフレームにおける値は (kf) チャンクを使用します。

(rotate.x)	BAC6 における対象ボーン座標系の+Z 軸方向ベクトル X 成分
(rotate.y)	BAC6 における対象ボーン座標系の+Z 軸方向ベクトル Y 成分
(rotate.z)	BAC6 における対象ボーン座標系の+Z 軸方向ベクトル Z 成分

(rotate) チャンクによる回転は、BAC6 ファイルで定義したボーン座標系の Z 軸及び、この (rotate) チャンクで定義する新しい Z 軸による平面内での回転です。この平面内での回転軸は対象となるボーン座標系の原点を通り、平面に対して垂直になります。

【関連】 [4.2.3.6 \(kf\) チャンク](#)

[4.2.3.7.2. \(rotate\) チャンクにおける BAC6 ボーン座標系維持](#)

[4.2.3.8. TRA におけるボーンアニメーション補足](#)

4.2.3.4. (roll)チャンク

(roll) チャンクは、BAC6 で定義した対象となるボーン座標系における Z 軸を軸とした回転度数（スカラー）を定義します。

(roll) チャンクは (kf) チャンクを子チャンクとして持っており、(roll) チャンクにおいて定義するアニメーションのキーフレームにおける値は (kf) チャンクを使用します。

(roll)	BAC6 における対象ボーン座標系 Z 軸の回転角度
----------	----------------------------

【関連】 [4.2.3.6 \(kf\) チャンク](#)

[4.2.3.7.3. \(roll\) チャンクにおける BAC6 座標系維持](#)

[4.2.3.8. TRA におけるボーンアニメーション補足](#)

4.2.3.5. (scale)チャンク

(scale) チャンクは、BAC6 で定義した対象となるボーン座標系における原点位置から、各軸方向の拡大縮小成分値を定義します。

(scale) チャンクは (kf) チャンクを子チャンクとして持っており、(scale) チャンクにおいて定義するアニメーションのキーフレームにおける値は (kf) チャンクを使用します。

(scale.x)	BAC6 における対象ボーン座標系 X 軸方向の拡大縮小値
(scale.y)	BAC6 における対象ボーン座標系 Y 軸方向の拡大縮小値
(scale.z)	BAC6 における対象ボーン座標系 Z 軸方向の拡大縮小値

【関連】 [4.2.3.6 \(kf\) チャンク](#)

[4.2.3.7.4. \(scale\) チャンクにおける BAC6 座標系維持](#)

[4.2.3.8. TRA におけるボーンアニメーション補足](#)

4.2.3.6. (kf)チャンク

(kf) チャンクは、対象となるボーンアニメーションにおけるキーフレーム値とそのフレーム値に対応するアニメーション値を定義します。

1 番目の[INT]パラメータには任意のキーフレーム値を定義します。キーフレーム値を定義する際は、開始キーフレーム値を必ず0 フレーム値とし定義してください。これによりキーフレーム値の定義は、以下の式で表すことができます。

$$\text{定義フレーム値} = \text{現在のフレーム値} - \text{開始フレーム値}$$

またキーフレーム値は (totalFrame) チャンクに定義するキーフレーム数未満でなければいけません。

2 番目の[FLOAT]パラメータには 1 番目の[INT]パラメータで定義したキーフレームに対応する各種アニメーション値（平行移動、回転、拡大縮小）を定義します。

【参照】図 8 「(kf) チャンク」

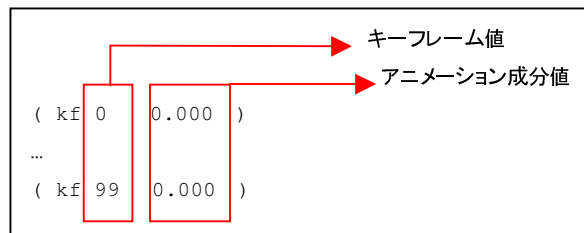


図 8 「(kf) チャンク」

【関連】 [4.2.3.2. \(translate\) チャンク](#) [4.2.3.3. \(rotate\) チャンク](#)
[4.2.3.4. \(roll\) チャンク](#) [4.2.3.5. \(scale\) チャンク](#)

4.2.3.7. アニメーション定義補足

(bone) チャンクにおける (translate)、(rotate)、(roll)、(scale) チャンクの定義は、BAC6 ファイルで定義したボーン座標系から静止状態を維持する場合であっても必ず定義する必要があります。

以下では、BAC6 ファイルで定義したボーンの姿勢を維持するための定義を各チャンク別に説明します。

4.2.3.7.1. (translate) チャンクにおける BAC6 ボーン座標系維持

BAC6 で定義されたボーン座標系から原点位置が平行移動しない場合は以下のように定義する必要があります。

```
( translate.x
    ( kf 0 0.000 )
    ( kf 99 0.000 )
)
( translate.y
    ( kf 0 0.000 )
    ( kf 99 0.000 )
)
( translate.z
    ( kf 0 0.000 )
    ( kf 99 0.000 )
)
```

【関連】 [4.2.3.2. \(translate\) チャンク](#)

4.2.3.7.2. (rotate) チャンクにおける BAC6 ボーン座標系維持

BAC6 で定義されたボーン座標系から (rotate) における回転がない場合は以下のように定義する必要があります。

```
( rotate.x
    ( kf 0 0.000 )
    ( kf 99 0.000 )
)
( rotate.y
    ( kf 0 0.000 )
    ( kf 99 0.000 )
)
( rotate.z
    ( kf 0 1.000 )
    ( kf 99 1.000 )
)
```

【関連】 [4.2.3.3. \(rotate\) チャンク](#)

4.2.3.7.3. (roll) チャンクにおける BAC6 ボーン座標系維持

BAC6 で定義されたボーン座標系から Z 軸を軸にした回転がない場合は以下のように定義する必要があります。

```
( roll
    ( kf 0 0.000 )
    ( kf 99 0.000 )
)
```

【関連】 [4.2.3.4. \(roll\) チャンク](#)

4.2.3.7.4. (scale) チャンクにおける BAC6 ボーン座標系維持

BAC6 で定義されたボーン座標系から拡大縮小しない場合は以下のように定義する必要があります。等倍を XYZ (100,100,100) とします。

```
( scale.x
    ( kf 0 100.000 )
    ( kf 99 100.000 )
)
( scale.y
    ( kf 0 100.000 )
    ( kf 99 100.000 )
)
( scale.z
    ( kf 0 100.000 )
    ( kf 99 100.000 )
)
```

【関連】 [4.2.3.5. \(scale\) チャンク](#)

4.2.3.8. TRAにおけるボーンアニメーションの補足

下記の図 9「ボーン座標系関係図」では TRA4 及び BAC6 におけるボーン座標系で定義するアニメーションに必要な (translate)、(rotate)、(roll)、(scale) チャンクの関係及び、BAC6 におけるボーン座標系との関連性を表したものです。この図 9「ボーン座標系関係図」のように、TRA4 では (rotate) チャンクと (roll) チャンクによりボーンの回転後の位置を定義します。また TRA4 におけるボーンアニメーションは、BAC6 で定義するボーン座標系からのキーフレーム毎の差分として定義します。

図 10「計算順序」ではボーン座標を定義する行列の計算順序を表しています。
(rotate) チャンクと (roll) チャンクの行列はどちらを先に掛け合わせても問題はありません。

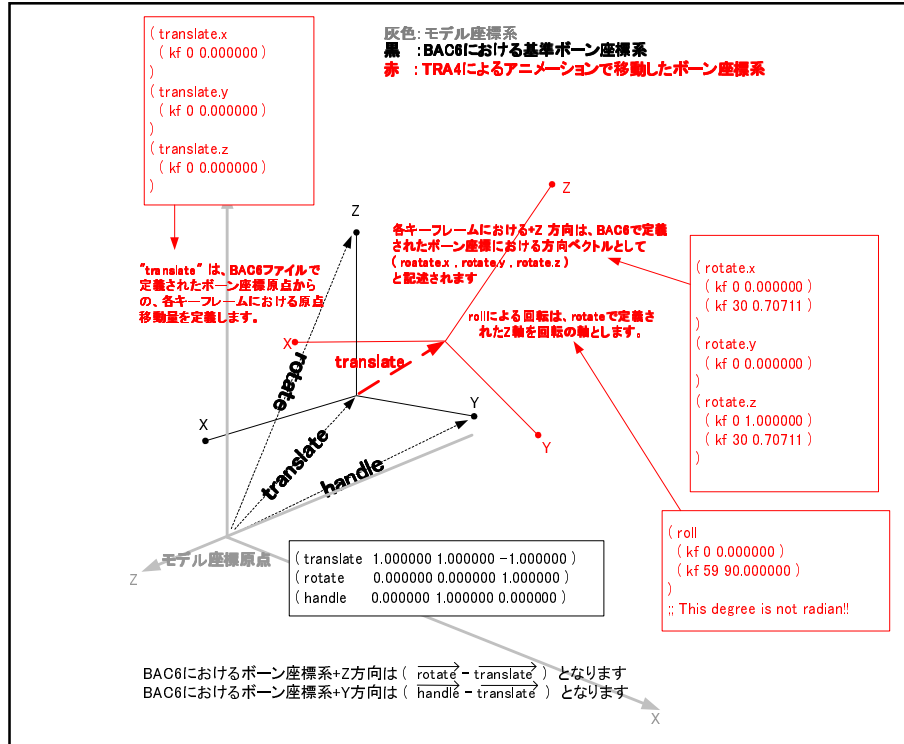


図 9「ボーン座標系関係図」

$$\vec{T}(\text{rotate}) \cdot \vec{T}(\text{roll}) \cdot \vec{T}(\text{scale}) \cdot \vec{P}$$

図 10「計算順序」

4.2.4. (DynamicPolygons)チャンク

(DynamicPolygons)チャンクでは BAC6 において定義したダイナミックポリゴンのパターン情報に対するアニメーションを定義します。

(DynamicPolygons) チャンクに定義するアニメーションのキーフレームの値は (kgf) チャンクにより定義します。

BAC6 にダイナミックポリゴンのパターンが定義されていない場合は (Figure) チャンク内にこのチャンクを定義する必要はありません。

4.2.4.1. (kgf [INT] [INT] [true|false])チャンク

(kgf)チャンクはダイナミックポリゴンにおけるパターンアニメーションの表示/非表示の切り替えを任意のキーフレーム単位に定義します。

1 番目の[INT]/パラメータはダイナミックポリゴンの表示/非表示を切り替えるためのキーフレーム値を定義します。アニメーション開始フレーム値は 0 フレームとして定義する必要があり、キーフレームの定義範囲は (totalFrame) チャンクにおいて定義したフレーム値未満でなければいけません。

2 番目の[INT]/パラメータは、BAC6 における (DynamicPolygons) チャンク内の (group) チャンクに自動的に割り当てられる 0 から始まるインデックス ID を定義します。

3 番目の[true|false]/パラメータは、ダイナミックポリゴンパターンに対して表示/非表示をフラグメント処理として定義します。TRUE が定義された場合、2 番目に定義されたインデックス ID のパターンは表示する定義とされ、FALSE の場合は非表示とする定義になります。

【参照】図 11 「(DynamicPolygons) チャンク」

```
( DynamicPolygons
    ( kgf    0    0    true )
    ( kgf   10    1    true )
    ( kgf   10    0   false )
    ( kgf   15    1   false )
)
```

図 11 「(DynamicPolygons) チャンク」

【関連】別紙「BAC6 Data Format Specification」4.2.9.1. (group) チャンク

5. サンプルコード

5.1. TRA4 サンプル 1

ここでは Z 軸まわりに回転するアニメーションを出力したコードを記述しています。

【Sample01.tra】

```
;TRA

( Head
  ( traVersion 4.0 )
)
( Figure
  ( totalFrame 11 )

  ( bone
    ( name "sample01" )
    ( translate.x
      ( kf 0 0.000000 )
      ( kf 10 0.000000 )
    )
    ( translate.y
      ( kf 0 0.000000 )
      ( kf 10 0.000000 )
    )
    ( translate.z
      ( kf 0 0.000000 )
      ( kf 10 0.000000 )
    )
    ( scale.x
      ( kf 0 100.000000 )
      ( kf 10 100.000000 )
    )
    ( scale.y
      ( kf 0 100.000000 )
      ( kf 10 100.000000 )
    )
    ( scale.z
      ( kf 0 100.000000 )
      ( kf 10 100.000000 )
    )
    ( rotate.x
      ( kf 0 0.000000 )
      ( kf 10 0.000000 )
    )
  )
)
```

```

        ( rotate.y
          ( kf 0 0.000000 )
          ( kf 10 0.000000 )
        )
        ( rotate.z
          ( kf 0 1.000000 )
          ( kf 10 1.000000 )
        )
        ( roll
          ( kf 0 0.000000 )
          ( kf 1 6.138396 )
          ( kf 2 22.589470 )
          ( kf 3 46.406628 )
          ( kf 6 132.590363 )
          ( kf 7 156.407547 )
          ( kf 8 172.858566 )
          ( kf 9 178.997116 )
          ( kf 10 0.000000 )
        )
      )
    )
  )

```

5.2. TRA4 サンプル2

ここでは Y 軸まわり 90 度回転、Z 軸まわり 90 度回転を組合わせたアニメーションを出力したコードを記述しています。

【Sample02.tra】

```
;TRA

( Head
  ( traVersion 4.0 )
) ; Head

( Figure
  ( totalFrame 11 )

  ( bone
    ( name "sample02" )
    ( translate.x
      ( kf 0 0.000000 )
      ( kf 10 0.000000 )
    )
    ( translate.y
      ( kf 0 0.000000 )
      ( kf 10 0.000000 )
    )
    ( translate.z
      ( kf 0 0.000000 )
      ( kf 10 0.000000 )
    )
    ( scale.x
      ( kf 0 100.000000 )
      ( kf 10 100.000000 )
    )
    ( scale.y
      ( kf 0 100.000000 )
      ( kf 10 100.000000 )
    )
    ( scale.z
      ( kf 0 100.000000 )
      ( kf 10 100.000000 )
    )
    ( rotate.x
      ( kf 0 0.000000 )
      ( kf 1 0.043925 )
      ( kf 2 0.160469 )
      ( kf 3 0.313841 )
      ...
    )
  )
)
```

```

        ( kf 4 0.446917 )
        ( kf 5 0.500000 )
        ( kf 6 0.446928 )
        ( kf 7 0.313862 )
        ( kf 8 0.160494 )
        ( kf 9 0.043951 )
        ( kf 10 0.000026 )
    )
    ( rotate.y
        ( kf 0 0.000000 )
        ( kf 1 0.001933 )
        ( kf 2 0.026450 )
        ( kf 3 0.110765 )
        ( kf 4 0.275800 )
        ( kf 6 0.724176 )
        ( kf 7 0.889218 )
        ( kf 8 0.973542 )
        ( kf 9 0.998065 )
        ( kf 10 1.000000 )
    )
    ( rotate.z
        ( kf 0 1.000000 )
        ( kf 1 0.999033 )
        ( kf 2 0.986686 )
        ( kf 3 0.942992 )
        ( kf 4 0.850999 )
        ( kf 5 0.707116 )
        ( kf 6 0.525189 )
        ( kf 7 0.332839 )
        ( kf 8 0.162661 )
        ( kf 9 0.043994 )
        ( kf 10 0.000026 )
    )
    ( roll
        ( kf 0 0.000000 )
        ( kf 1 2.519793 )
        ( kf 2 9.359774 )
        ( kf 3 19.439661 )
        ( kf 4 31.679462 )
        ( kf 6 58.319023 )
        ( kf 7 70.558815 )
        ( kf 8 80.638649 )
        ( kf 9 87.478539 )
        ( kf 10 89.998497 )
    )
)
)

```

5.3. TRA4 サンプル 3

ここではダイナミックポリゴンが適用されているアニメーションを出力したコードを記述します。

【Sample03.tra】

```
;TRA

( Head
  ( traVersion 4.0 )
)
( Figure
  ( totalFrame 45 )

  ( bone
    ( name "sample03" )
    ( translate.x
      ( kf 0 0.000000 )
      ( kf 44 0.000000 )
    )
    ( translate.y
      ( kf 0 0.000000 )
      ( kf 44 0.000000 )
    )
    ( translate.z
      ( kf 0 0.000000 )
      ( kf 44 0.000000 )
    )
    ( scale.x
      ( kf 0 100.000000 )
      ( kf 44 100.000000 )
    )
    ( scale.y
      ( kf 0 100.000000 )
      ( kf 44 100.000000 )
    )
    ( scale.z
      ( kf 0 100.000000 )
      ( kf 44 100.000000 )
    )
    ( rotate.x
      ( kf 0 0.000000 )
      ( kf 44 0.000000 )
    )
    ( rotate.y
      ( kf 0 0.000000 )
      ( kf 44 0.000000 )
    )
    ...
  )
)
```

```
        ( rotate.z
          ( kf 0 1.000000 )
          ( kf 44 1.000000 )
        )
        ( roll
          ( kf 0 0.000000 )
          ( kf 44 0.000000 )
        )
      )
    ( DynamicPolygons
      ( kgf 0 0 true )
      ( kgf 0 1 false )
      ( kgf 0 2 false )
      ( kgf 0 3 false )
      ( kgf 9 0 false )
      ( kgf 9 1 true )
      ( kgf 9 2 false )
      ( kgf 9 3 false )
      ( kgf 19 1 false )
      ( kgf 19 2 true )
      ( kgf 19 3 false )
      ( kgf 29 2 false )
      ( kgf 29 3 true )
      ( kgf 39 3 false )
    )
  )
```


TRA File Ver.4.0 Data Format Specification MascotCapsuleV3

日本語版

バージョン 2.1
発行日 2007 年 6 月 7 日
発行者 株式会社 エイチアイ
〒153-0043 東京都目黒区東山 1-4-4 目黒東山ビル 5F
<http://www.hicorp.co.jp/>

－著作権・商標・免責事項について－

・本書は著作権法上の保護を受けています。本書の一部または全部について(ソフトウェアおよびプログラムを含む)、株式会社エイチアイから書面による承諾を得ずに、いかなる方法においても無断で複写、複製、転載することは禁じられています。

・MascotCapsule(R)は、株式会社エイチアイの日本国における登録商標です。本書に記載されているその他の製品名は、各社の商標、または登録商標です。

・本書に掲載されている情報を利用することで発生するトラブルや損失・損害に対して、当社は一切責任を負いません。

・本書の内容に関しては訂正・改善のため、将来予告なしに変更することがあります。