# *AAM FILE FORMAT*

This document presents the structure of the AAM file format. The AAM format allows to describe the definition of a 3d scene, in particular its properties regarding: materials, geometry, animation, camera and lights. Actually camera and light properties are exported in a separate file (*.CAM) in order to keep distinct the objects properties from the general scene ones.
An AAM file is a plain text file containing tags and blocks of data. Tags are case-sensitive. Blocks are enclosed in braces and may be nested. Every line must end with a carriage return.

## PLAIN AAM MESHES

As an example, an AAM sample file with no animation data is presented in order to illustrate the basic materials/geometry definition.
Objects may be expressed in local or world coordinates, depending on how they were exported, but there is no chance to know this from the AAM file.

| AAM file line | Description |
|---|---|
| `AAM_MESH` | Header (optional: if present speeds up file loading) |
| `MATERIALS` | The <**`MATERIALS>`** tag indicates the beginning of the ***material list.*** |
| `MatCount: 2` | The <**`MatCount:`**> tag is followed by the ***total number*** of the materials actually used in the scene (i.e. associated to exported scene objects) |
| `Mat# 0` | The <**`Mat#`**> tag is followed by ***the progressive ID*** of the material. |
| `{` | In this position the <**`{`**> tag starts the block describing the material properties. |
| `Name: Box02_mtl [Box02]` | The <**`Name:`**> tag is followed by the ***name*** of the material (spaces are allowed) |
| `Class: Shell Material` | The <**`Class:`**> tag is followed by the ***class*** of the material (spaces are allowed, currently this tag is IGNORED if it is not **`Multi`**) |
| `Am: 0.5880 0.5880 0.5880` | The <**`Am:`**> tag is followed by three float numbers, separated with spaces, ranging in [0.0,1.0] representing the RGB values of the material ***ambient*** component. |
| `Di: 0.8941 0.6000 0.7216` | The <**`Di:`**> tag is followed by three float numbers, separated with spaces, ranging in [0.0,1.0] representing the RGB values of the material ***diffuse*** component. |
| `Sp: 0.1000 0.1000 0.1000` | The <**`Sp:`**> tag is followed by three float numbers, separated with spaces, ranging in [0.0,1.0] representing the RGB values of the material ***specular*** component. |
| `Tr: 0.0000` | The <**`Tr:`**> tag is followed by a float number, ranging in [0.0,1.0], representing the value of the material ***transparency*** (0.0 = opaque, 1.0 = transparent) |
| `Sh: 0.1000` | The <**`Sh:`**> tag is followed by a float number, ranging in [0.0,1.0], representing the value of the material ***shininess*** (0.0 = dark, 1.0 = bright) |
| `Tx: N` | The <**`Tx:`**> tag is followed by "N" (meaning there is no ***primary texture*** associated to the material) or "Y". In this last case a sub-list of texture properties starts, enclosed in braces {…} |
| `TS: Y` | The <**`TS:`**> tag is followed by "N" (meaning there is no ***secondary texture*** associated to the material) or "Y". In this last case a sub-list of texture properties starts, enclosed in braces {…} |

| | |
|---|---|
| `{` | In this position the <**{**> tag starts the block indicating the texture properties. |
| **FN:** `Box02LightingMap.jpg` | The <**FN:**> tag is followed by the ***filename*** of the texture image file (spaces are NOT allowed). Although the extension may be any one, only ***png*** and ***jpg*** files are currently supported by XVR. |
| **Fi:** `PYRAMIDAL` | The <**Fi:**> tag is followed by the ***type of filtering*** used with this texture. Currently this tag is IGNORED. |
| **Ch:** `5` | The <**Ch:**> tag is followed by an integer value representing the ***secondary map channel***. The list of the available channels is:<br>`0 - ambient`<br>`1 - diffuse`<br>`2 - specular`<br>`3 - shininesNs`<br>`4 - shininess strength`<br>`5 - self-illumination`<br>`6 - opacity`<br>`7 - filter color`<br>`8 - bump`<br>`9 - reflection`<br>`10- refraction`<br>`11- displacement` |
| **In:** `1.0000` | The <**In:**> tag is followed by a float number, ranging in [0.0,1.0], representing the ***intensity*** of the texture. Currently the support to this parameter is under development. |
| **UA:** `0.0000` | The <**UA:**> tag is followed by a float number representing the ***U rotation angle*** of the texture. |
| **VA:** `0.0000` | The <**VA:**> tag is followed by a float number representing the ***V rotation angle*** of the texture. |
| **WA:** `0.0000` | The <**WA:**> tag is followed by a float number representing the ***W rotation angle*** of the texture. |
| **UO:** `0.0000` | The <**UO:**> tag is followed by a float number representing the ***U offset*** of the texture. |
| **VO:** `0.0000` | The <**VO:**> tag is followed by a float number representing the ***V offset*** of the texture. |
| **UT:** `1.0000` | The <**UT:**> tag is followed by a float number representing the ***U tiling*** of the texture. |
| **VT:** `1.0000` | The <**VT:**> tag is followed by a float number representing the ***V tiling*** of the texture. |
| `}` | End of the texture properties block. |
| `}` | End of the material properties block. |
| `Mat# 1` | |
| `{` | |
| `Name: 01 - Default [Box01]` | |
| `Class: Shell Material` | |
| `Am: 0.5882 0.5882 0.5882` | |
| `Di: 0.5882 0.5882 0.5882` | |
| `Sp: 0.1000 0.1000 0.1000` | |
| `Tr: 0.0000` | |
| `Sh: 0.1000` | |
| **Tx:** `Y` | The texture tags may be independently set to N or Y. |
| `{` | |
| `FN: opengl_logo.jpg` | |
| `Fi: PYRAMIDAL` | |
| `In: 1.0000` | |
| `UA: 0.0000` | |

```
        VA: 0.0000
        WA: 0.0000
        UO: 0.0000
        VO: 0.0000
        UT: 1.0000
        VT: 1.0000
    }
    TS: Y
    {
        FN: Box01LightingMap.jpg
        Fi: PYRAMIDAL
        Ch: 5
        In: 1.0000
        UA: 0.0000
        VA: 0.0000
        WA: 0.0000
        UO: 0.0000
        VO: 0.0000
        UT: 1.0000
        VT: 1.0000
    }
}
ENDMATERIALS

GEOMETRY SmGEnabled


NObj: 2

NFrames: 1

Animation_mode: None





Frame: 0


{

    Obj: 0 Box01

    Par: -1



    {
    MatID: 1

    V_List: 8
```

The <**ENDMATERIALS>** tag indicates the ending of the ***material list.***

The <**GEOMETRY>** tag indicates the start of the ***geometry list.*** It may be followed by the parameter <SmGEnabled>, which means that the smoothing groups are enabled.

The <**NObj:**> tag is followed by the ***total number*** of the actually exported objects present in the scene.

The <**NFrames:**> tag is followed by an integer value representing the number (if any) of ***animation frames***.

The <**Animation_mode:**> tag is followed by a string representing the ***animation type***, if present. The supported values are:

<Keyframe> – Animation keyframe based (only characters)

<Full> – Animation full-frames (only mesh)

<None> – No animation

The <**Frame:**> tag is followed by an integer representing the animation frame progressive ID. If no animation is present, this tag is unique and its value is 0.

In this position the <**{**> tag starts the ***list of objects*** present in the current frame.

The <**Obj:**> tag is followed by an integer, representing the obj unique ***ID***, and by a string representing the ***object name***.

The <**Par:**> tag is followed by an integer, representing the ID of the object's parent. If the object is direct descendant of the scene, the parent ID is set to -1. This parameter is significant only for AAM characters, whilst it is ignored in AAM meshes.

In this position the <**{**> tag starts the ***objects properties*** block.

The <**MatID:**> tag is followed by an integer representing the ***ID of the material*** associated to the current object.

The <**V_List:**> tag is followed by an integer ***N***, representing the total number of the current object ***vertices***, and by N series of three floats, where the i-th series represents the XYZ coordinates of the i-th vertex.

```
    -0.5000 -0.5000 0.0000
    0.5000 -0.5000 0.0000
    -0.5000 0.5000 0.0000
    0.5000 0.5000 0.0000
    -0.5000 -0.5000 1.0000
    0.5000 -0.5000 1.0000
    -0.5000 0.5000 1.0000
    0.5000 0.5000 1.0000
TV_List: 36
```

The <**TV_List:**> tag is followed by an integer ***M***, representing the total number of the current object ***texture vertices***, and by M series of two floats, where the i-th series represents the UV texture coordinates of the i-th vertex.

```
   0.0000 0.0000
   1.0000 0.0000
   0.0000 1.0000
   1.0000 1.0000
   0.0000 0.0000
   1.0000 0.0000
   0.0000 1.0000
   1.0000 1.0000
   0.0000 0.0000
   1.0000 0.0000
   0.0000 1.0000
   1.0000 1.0000
   0.3506 0.6494
   0.6494 0.6494
   0.3506 0.3506
   0.6494 0.3506
   0.0300 0.6713
   0.3287 0.6713
   0.0300 0.9700
   0.3287 0.9700
   0.6713 0.0300
   0.9700 0.0300
   0.6713 0.3287
   0.9700 0.3287
   0.3287 0.0300
   0.0300 0.0300
   0.3287 0.3287
   0.0300 0.3287
   0.3287 0.3506
   0.0300 0.3506
   0.3287 0.6494
   0.0300 0.6494
   0.3506 0.0300
   0.6494 0.0300
   0.3506 0.3287
   0.6494 0.3287
I_List: 12 1
```

The <**I_List:**> tag is followed by an integer ***I***, representing the total number of the ***current object indices***, and by another integer representing the number of mesh groups. If the material linked to the object is not single-material, it is possible to have more than one group, otherwise there will be just one group.

```
   NEWGROUP: 1
```

The <**NEWGROUP:**> tag indicates the start of a ***group description***. A group (or ***subset***) is basically a set of triangles, belonging to the current object, which share the same material. This tag is followed by an integer representing the ID of the group, unique inside the current object.

```
I: 0 2 3 2
```

The <**I:**> tag is followed by four integers. The first three define the current triangle and represent the ***indices*** for accessing the vertex list to compose the triangle. The last one represent the smoothing group. The edges of triangles belonging to the same smoothing group are smoothed when rendered. If smoothing groups are disabled, the last number is not significant.

After an <I:> definition, it may follow:

another **<I:>** line, meaning the triangle is not texturized

a single **<TI:>** line, meaning that the triangle is texturized with one texture map layer

two **<TI:>** line, meaning that the triangle is texturized with two texture map layers

```
TI: 9 11 10
```

The **<TI:>** tag is followed by three integers. They define the ***texture indices*** for accessing the texture coordinates list, used to map the texture onto the current triangle. The first **<TI:>** tag is related to the main texture layer.

```
TI: 12 14 15
```

If a further <**TI:**> is encountered, it specifies the texture coordinates for the secondary texture map. The modalities are the same of any other <TI:>.

```
I: 3 1 0 2
TI: 10 8 9
TI: 15 13 12
I: 4 5 7 4
TI: 8 9 11
TI: 16 17 19
I: 7 6 4 4
TI: 11 10 8
TI: 19 18 16
I: 0 1 5 8
TI: 4 5 7
TI: 32 33 35
I: 5 4 0 8
TI: 7 6 4
TI: 35 34 32
I: 1 3 7 16
TI: 0 1 3
TI: 20 21 23
I: 7 5 1 16
TI: 3 2 0
TI: 23 22 20
I: 3 2 6 32
TI: 4 5 7
TI: 29 28 30
I: 6 7 3 32
TI: 7 6 4
TI: 30 31 29
I: 2 0 4 64
TI: 0 1 3
TI: 25 24 26
I: 4 6 2 64
TI: 3 2 0
TI: 26 27 25
ENDGROUP
```

The <**ENDGROUP**> tag indicates the ***end of the current group description***.

```
}

Obj: 1 Box02
Par: -1
{
MatID: 0
V_List: 8
    -2.5000 -2.5607 0.0000
    2.5000 -2.5607 0.0000
    -2.5000 2.4393 0.0000
    2.5000 2.4393 0.0000
    -2.5000 -2.5607 -0.1000
    2.5000 -2.5607 -0.1000
    -2.5000 2.4393 -0.1000
    2.5000 2.4393 -0.1000
TV_List: 24
   0.0300 0.0300
   0.4900 0.0300
   0.0300 0.4900
   0.4900 0.4900
   0.0300 0.9700
   0.4900 0.9700
   0.0300 0.5100
   0.4900 0.5100
   0.6570 0.0300
   0.6570 0.4900
   0.6662 0.0300
   0.6662 0.4900
   0.6080 0.4900
   0.6080 0.0300
   0.6172 0.4900
   0.6172 0.0300
   0.5590 0.4900
   0.5590 0.0300
   0.5682 0.4900
   0.5682 0.0300
   0.5100 0.0300
   0.5100 0.4900
   0.5192 0.0300
   0.5192 0.4900
I_List: 12 1
   NEWGROUP: 0
   I: 2 0 3 2
   TI: 2 0 3

   I: 1 3 0 2
   TI: 1 3 0
   I: 5 4 7 4
   TI: 5 4 7
   I: 6 7 4 4
   TI: 6 7 4
   I: 1 0 5 8
   TI: 21 20 23
   I: 4 5 0 8
   TI: 22 23 20
   I: 3 1 7 16
```

For this object, as there is just a layer of textures in the associated materials, only a **<TI:>** tag follows each **<I:>** tag.

| | |
|---|---|
| TI: 9 8 11<br>I: 5 7 1 16<br>TI: 10 11 8<br>I: 2 3 6 32<br>TI: 16 17 18<br>I: 7 6 3 32<br>TI: 19 18 17<br>I: 0 2 4 64<br>TI: 12 13 14<br>I: 6 4 2 64<br>TI: 15 14 13<br>ENDGROUP<br>}<br>} | |
| ENDGEOMETRY | The <**ENDGEOMETRY>** tag indicates the ending of the ***geometry list.*** |

**MULTIFRAME AAM MESHES**

Another example will illustrate the definition of multi-frame AAM meshes. In this case animation data are expressed specifying, for the first frame, all the data like in the normal case and, for each following frame, only the new vertex coordinates. This method of animation-handling may produce huge files, but in certain cases it is the only way to deal with animations which can not be expressed in terms of interpolation of keyframes.

| AAM file line | Description |
|---|---|
| AAM_MESH_MULTIFRAME<br>MATERIALS<br>MatCount: 1<br>Mat# 0<br>{<br>Name: Color0<br>Class: Color<br>Am: 0.3412 0.8784 0.5608<br>Di: 0.3412 0.8784 0.5608<br>Sp: 0.0 0.0 0.0<br>Tr: 0.0<br>Sh: 0.0<br>Tx: N<br>TS: N<br>}<br>ENDMATERIALS<br>GEOMETRY SmGEnabled<br>NObj: 1 | Header (optional: if present speeds up file loading) |
| **NFrames:** 11<br>**Animation**_mode: Full | In this case we have 11 animation frames.<br>The <**Animation_mode:**> is “Full”, which means that we will have 11 <Frame:> blocks; the first one will present the full description (vertex coordinates, texture coordinates, indices etc.), the other ones only the vertex coordinates relative to the frame. |
| **Frame:** 0<br>{<br>Obj: 0 Box01 | Frame 0 : full description |

```
   Par: -1
   {
   MatID: 0
   V_List: 8
       -32.0756 -40.8935 0.0000
       27.0309 -40.8935 0.0000
       -32.0756 20.2749 0.0000
       27.0309 20.2749 0.0000
       -32.0756 -40.8935 49.4845
       27.0309 -40.8935 49.4845
       -32.0756 20.2749 49.4845
       27.0309 20.2749 49.4845
   TV_List: 0
   I_List: 12 1
      NEWGROUP: 0
      I: 0 2 3 2
      I: 3 1 0 2
      I: 4 5 7 4
      I: 7 6 4 4
      I: 0 1 5 8
      I: 5 4 0 8
      I: 1 3 7 16
      I: 7 5 1 16
      I: 3 2 6 32
      I: 6 7 3 32
      I: 2 0 4 64
      I: 4 6 2 64
      ENDGROUP
   }
}
```

Frame 1: partial description

```
Frame: 1
{
   Obj: 0 Box01
   Par: -1
   {
   MatID: 0
```

Only the V_List is described

```
   V_List: 8
       -29.3581 -40.8935 0.0000
       29.7484 -40.8935 0.0000
       -29.3581 20.2749 0.0000
       29.7484 20.2749 0.0000
       -29.3581 -40.8935 49.4845
       29.7484 -40.8935 49.4845
       -29.3581 20.2749 49.4845
       29.7484 20.2749 49.4845
   }
}

…
<Frames 2 to 9 are described in
the same way>
```

Frame 10: partial description

```
Frame: 10
{
   Obj: 0 Box01
   Par: -1
   {
   MatID: 0
```

| | |
|---|---|
| **V_List:** 8<br>64.9769 -40.8935 0.0000<br>124.0834 -40.8935 0.0000<br>64.9769 20.2749 0.0000<br>124.0834 20.2749 0.0000<br>64.9769 -40.8935 49.4845<br>124.0834 -40.8935 49.4845<br>64.9769 20.2749 49.4845<br>124.0834 20.2749 49.4845<br>}<br>}<br>ENDGEOMETRY | Only the V_List is described |

### AAM CHARACTERS

AAM characters description is slightly different from the AAM mesh, as it is conceived to support hierarchy of interconnected objects, rather than plain geometry. However, even if a hierarchy it is not actually present, it is useful to export characters, instead of meshes, as they offer a more extended support to animations. Each object is expressed in local coordinates but additional tags are present giving information on its global placement.

| AAM file line | Description |
|---|---|
| AAM_CHARACTER | Header (optional: if present speeds up file loading) |
| MATERIALS<br>MatCount: 1<br>Mat# 0<br>{<br>Name: Color0<br>Class: Color<br>Am: 0.3412 0.8784 0.5608<br>Di: 0.3412 0.8784 0.5608<br>Sp: 0.0 0.0 0.0<br>Tr: 0.0<br>Sh: 0.0<br>Tx: N<br>TS: N<br>}<br>ENDMATERIALS<br>GEOMETRY SmGEnabled | |
| **NObj:** 1 | The <**NObj:**> tag is followed by the ***total number*** of the actually exported objects present in the scene which will be exported as character components. |
| **NFrames:** 1 | The <**NFrames:**> tag is followed by an integer value representing the number (if any) of ***animation frames*** |
| **Animation_mode:** Keyframe | The <**Animation_mode:**> tag for an AAM character is ALWAYS set to <Keyframe> |
| **Frame:** 0 | The <**Frame:**> tag for an AAM character is ALWAYS followed by 0, even if there are animation frames (the value is simply ignored) |
| {<br>Obj: 0 Box01<br>**Par:** -1 | The <**Par:**> tag is followed by the ID of the current object's parent. If the object has no parent, this value equals -1. |
| { | The characters components, being expressed in local coordinates, need the specification of more information (Pivot, Transformation etc.). All the values are expressed in local coordinates (i.e. referred to the parent reference frame) |

| | |
|---|---|
| **Piv:** 0.0000 0.0000 0.0000<br>0.0000 0.0000 0.0000 0.0000 | The <**Piv:**> tag is followed by seven floats, three to represent the Pivot position and four for the Orientation (expressed as a quaternion) of the Pivot Reference Frame. Usually the orientation values equal 0.0. |
| **Mir:** 0 | The <**Mir:**> tag is followed by 1, if the object is the result of a mirror operation, or 0 otherwise. The default value is 0 |
| **TM:** 1.0000 0.0000 0.0000 0.0000<br>1.0000 0.0000 0.0000 0.0000<br>1.0000 2.5223 -10.3093 0.0000<br>1.0000 1.0000 1.0000 | The <**TM:**> tag is followed by fifteen floats, corresponding to the transformation: the first nine values correspond to the rotation matrix, the next three values to the position, the last three values to the scaling along the three main axes. |
| MatID: 0<br>V_List: 8<br>-29.5533 -30.5842 0.0000<br>29.5533 -30.5842 0.0000<br>-29.5533 30.5842 0.0000<br>29.5533 30.5842 0.0000<br>-29.5533 -30.5842 49.4845<br>29.5533 -30.5842 49.4845<br>-29.5533 30.5842 49.4845<br>29.5533 30.5842 49.4845<br>TV_List: 0<br>I_List: 12 1<br>NEWGROUP: 0<br>I: 0 2 3 2<br>I: 3 1 0 2<br>I: 4 5 7 4<br>I: 7 6 4 4<br>I: 0 1 5 8<br>I: 5 4 0 8<br>I: 1 3 7 16<br>I: 7 5 1 16<br>I: 3 2 6 32<br>I: 6 7 3 32<br>I: 2 0 4 64<br>I: 4 6 2 64<br>ENDGROUP | |
| **Anim_Ctrl:** N | The <**Anim_Ctrl:**> tag may be followed by N (meaning there is no animation data) or by Y (there is animation data). In the latest case animation data follows, otherwise nothing follows. |
| }<br>}<br>ENDGEOMETRY | |

AAM characters may also contain animation frames. The animation data may be expressed for all the animation frames (samples: values are directly specified in the AAM file for each frame) or only for a limited number of key frames (keyframes: values are specified in the AAM file for keyframes, the other are calculated at runtime using some kind of interpolation).
Note that for characters the <**Animation_mode:**> is always **Keyframe**. This is not related to the presence of animation data (which is specified by the tag <**Anim_Ctrl:**>) neither to the type of data (samples or keyframes, which is specified by the tags **PsSam**, **RtSam**, **PsBez**,…, etc.).

| AAM file line | Description |
|---|---|
| MATERIALS<br>MatCount: 1<br>Mat# 0<br>{<br>Name: Color0<br>Class: Color<br>Am: 0.3412 0.8784 0.5608<br>Di: 0.3412 0.8784 0.5608<br>Sp: 0.0 0.0 0.0 | |

```
   Tr: 0.0
   Sh: 0.0
   Tx: N
   TS: N
}
ENDMATERIALS
GEOMETRY SmGEnabled
NObj: 1
NFrames: 11
Animation_mode: Keyframe
Frame: 0
{
   Obj: 0 Box01
   Par: -1
   {
   Piv: 0.0000  0.0000 0.0000
0.0000
           0.0000 0.0000 0.0000
   Mir: 0
   TM: 1.0000 0.0000  0.0000 0.0000
1.0000 0.0000 0.0000 0.0000
1.0000 -2.5223  -10.3093 0.0000
1.0000 1.0000 1.0000
   MatID: 0
   V_List: 8
      -29.5533 -30.5842 0.0000
      29.5533 -30.5842 0.0000
      -29.5533 30.5842 0.0000
      29.5533 30.5842 0.0000
      -29.5533 -30.5842 49.4845
      29.5533 -30.5842 49.4845
      -29.5533 30.5842 49.4845
      29.5533 30.5842 49.4845
   TV_List: 0
   I_List: 12 1
     NEWGROUP: 0
     I: 0 2 3 2
     I: 3 1 0 2
     I: 4 5 7 4
     I: 7 6 4 4
     I: 0 1 5 8
     I: 5 4 0 8
     I: 1 3 7 16
     I: 7 5 1 16
     I: 3 2 6 32
     I: 6 7 3 32
     I: 2 0 4 64
     I: 4 6 2 64
     ENDGROUP
   Anim_Ctrl: Y
```

In this case the <**Anim_Ctrl:**> tag is followed by Y, therefore animation data follows. One or more or the following tags may follow:

**PsSam: RtSam: ScSam:** (position/rotation/scale values expressed as samples)

**PsBez: RtBez: ScBez:** (position/rotation/scale values expressed as keyframes, to interpolate with the Bezier method)

**RtTcb:** (rotation values expressed as keyframes, to interpolate with the TCB method)

**RtIPA:** (rotation values expressed as keyframes, to interpolate linearly).

Each one of the above mentioned tags, is followed by the number of the frames whose data are subsequently specified.

```
   PsBez: 2
```

Two keyframes are present for Position, and the interpolation mode is Bezier.

| | |
|---|---|
| 0 -2.5223 -10.3093 0.0000<br>0.0000 0.0000 0.0000 0.0000 0.0000<br>0.0000 6912 | Data of the first position Bezier keyframe. |
| 10 94.5301 -10.3093 0.0000<br>0.0000 0.0000 0.0000 0.0000 0.0000<br>0.0000 6912 | Data of the second position Bezier keyframe. |
| **RtBez:** 2 | Two keyframes are present for Rotation, and the interpolation mode is Bezier. |
| 0 0.0000 0.0000 0.0000 0.0000<br>0.0000 0.0000 0.0000 0.0000 0.0000<br>6912 | Data of the first rotation Bezier keyframe. |
| 10 0.0000 0.0000 0.0000<br>0.0000 0.0000 0.0000 0.0000 0.0000<br>0.0000 6912 | Data of the second rotation Bezier keyframe |
| **ScBez:** 2 | Two keyframes are present for Scale, and the interpolation mode is Bezier. |
| 0 1.0000 1.0000 1.0000 0.0000<br>0.0000 0.0000 0.0000 0.0000 0.000<br>0.0000 0.0000 0.0000 0.0000 6912 | Data of the first scaling Bezier keyframe. |
| 10 1.0000 1.0000 1.0000<br>0.0000 0.0000 0.0000 0.0000 0.0000<br>0.0000  0.0000 0.0000 0.0000<br>0.0000 6912<br>}<br>}<br>ENDGEOMETRY | Data of the second scaling Bezier keyframe. |

The possible configurations of the animation data are subsequently reported, following this convention:
<nf>: means a sequence of n floats
<i>: means one integer
<f>: means one float

**Bezier Interpolation:**

| **PsBez** | |
|---|---|
| <i> | Frame number |
| <3f> | Position values (x, y, z) |
| <3f> | In tangent (vx, vy, vz) |
| <3f> | Out tangent (vx, vy, vz) |
| <i> | Flag (ignored) |

| **RtBez** | |
|---|---|
| <i> | Frame number |
| <3f> | Rotation values (rx, ry, rz), expressed as Euler angles xyz. |
| <3f> | In tangent (vx, vy, vz) |
| <3f> | Out tangent (vx, vy, vz) |
| <i> | Flag (ignored) |

| **ScBez** | |
|---|---|
| <i> | Frame number |
| <3f> | Scaling values (sx, sy, sz) |
| <4f> | Scaling rotation: quaternion representing the orientation of the scaling reference frame. |
| <3f> | In tangent (vx, vy, vz) |
| <3f> | Out tangent (vx, vy, vz) |
| <i> | Flag (ignored) |

**No Interpolation:**

| **PsSam** | |
|---|---|
| <i> | Frame number |
| <3f> | Position values (x, y, z) |

| **RtSam** | |
|---|---|
| <i> | Frame number |
| <4f> | Rotation values (ax, ay, az, angle), expressed in AxisAngle representation |

| **ScSam** | |
|---|---|
| `<i>` | Frame number |
| `<4f>` | Scaling rotation: quaternion representing the orientation of the scaling reference frame |
| `<3f>` | Scaling values (sx, sy, sz) |

**TCB Interpolation:**

| **RtTcb** | |
|---|---|
| `<i>` | Frame number |
| `<4f>` | Rotation values (ax, ay, az, angle), expressed in AxisAngle representation |
| `<f>` | Tension |
| `<f>` | Continuity |
| `<f>` | Bias |
| `<f>` | Ease IN |
| `<f>` | Ease Out |

**Linear Interpolation:**

| **RtIPA** | |
|---|---|
| `<i>` | Frame number |
| `<3f>` | Rotation values (rx, ry, rz), expressed as Euler angles xyz. |

## MULTIMATERIALS

AAM files, in addition to ordinary materials, may contain Multimaterials which are materials, identified as usual by a unique ID, containing several submaterials. Multimaterials are useful if an object is composed by several subsets linked to different materials. In this case the object will be assigned the multimaterial, and each subset will be assigned one of the contained submaterials. In this case, for each subset the related ID, corresponding to the index of the associated submaterial, will follow the tag **Newgroup:** as described in the following example.

| AAM file line | Description |
|---|---|
| `MATERIALS` | |
| `MatCount: 1` | |
| `Mat# 0` | |
| `{` | |
| `Name: 01 - Default` | |
| `Class: Multi` | **`Multi`** indicates that the material is a multimaterial, therefore the description of its submaterials will follow. |
| `NSubs: 6` | The tag **`<NSubs:>`** is followed by the number of the submaterials of the current material. |
| `Sub 0` | The description of each submaterials starts with the tag **`<Sub>`**, followed by the ID of the submaterial. Then a section follows indicating the properties of the submaterial (this section is the same as in the case of normal materials). Although not advisable, each submaterial may be a multimaterial. |
| `{` | |
| `Name: Material #26` | |
| `Class: Standard` | |
| `Am: 0.5880 0.5880 0.5880` | |
| `Di: 0.5882 0.8000 0.7333` | |
| `Sp: 0.1000 0.1000 0.1000` | |
| `Tr: 0.0000` | |
| `Sh: 0.1000` | |
| `Tx: N` | |
| `TS: N` | |
| `}` | |

```
   Sub 1
   {
      Name: Material #27
      Class: Standard
      Am: 0.5880 0.5880 0.5880
      Di: 0.6353 0.6118 0.5882
      Sp: 0.1000 0.1000 0.1000
      Tr: 0.0000
      Sh: 0.1000
      Tx: N
      TS: N
   }
   Sub 2
   {
      Name: Material #28
      Class: Standard
      Am: 0.5880 0.5880 0.5880
      Di: 0.5882 0.6157 0.8196
      Sp: 0.1000 0.1000 0.1000
      Tr: 0.0000
      Sh: 0.1000
      Tx: N
      TS: N
   }
   Sub 3
   {
      Name: Material #29
      Class: Standard
      Am: 0.5880 0.5880 0.5880
      Di: 0.9529 0.7098 0.5922
      Sp: 0.1000 0.1000 0.1000
      Tr: 0.0000
      Sh: 0.1000
      Tx: N
      TS: N
   }
   Sub 4
   {
      Name: Material #30
      Class: Standard
      Am: 0.5880 0.5880 0.5880
      Di: 0.8588 0.8627 0.5882
      Sp: 0.1000 0.1000 0.1000
      Tr: 0.0000
      Sh: 0.1000
      Tx: N
      TS: N
   }
   Sub 5
   {
      Name: Material #31
      Class: Standard
      Am: 0.5880 0.5880 0.5880
      Di: 0.8980 0.5882 0.6745
      Sp: 0.1000 0.1000 0.1000
      Tr: 0.0000
      Sh: 0.1000
      Tx: N
      TS: N
   }
}
ENDMATERIALS
GEOMETRY SmGEnabled
NObj: 1
NFrames: 1
Animation_mode: None
```

| | |
|---|---|
| Frame: 0 | |
| { | |
| Obj: 0 Box01 | |
| Par: -1 | |
| { | |
| MatID: 0 | The ID of the material is related to a multimaterial, therefore each Newgroup: tag will be followed by the ID of the associated submaterial. |
| V_List: 8 | |
| -45.0172 -56.0137 0.0000 | |
| 31.9588 -56.0137 0.0000 | |
| -45.0172 34.0206 0.0000 | |
| 31.9588 34.0206 0.0000 | |
| -45.0172 -56.0137 63.2302 | |
| 31.9588 -56.0137 63.2302 | |
| -45.0172 34.0206 63.2302 | |
| 31.9588 34.0206 63.2302 | |
| TV_List: 0 | |
| I_List: 12 6 | |
| NEWGROUP: 0 | The triangles belonging to this subset will be associated to the submaterial 0 of the material 0. |
| I: 4 5 7 4 | |
| I: 7 6 4 4 | |
| ENDGROUP | |
| NEWGROUP: 1 | The triangles belonging to this subset will be associated to the submaterial 1 of the material 0. |
| I: 0 2 3 2 | |
| I: 3 1 0 2 | |
| ENDGROUP | |
| NEWGROUP: 2 | … |
| I: 2 0 4 64 | |
| I: 4 6 2 64 | |
| ENDGROUP | |
| NEWGROUP: 3 | … |
| I: 1 3 7 16 | |
| I: 7 5 1 16 | |
| ENDGROUP | |
| NEWGROUP: 4 | … |
| I: 0 1 5 8 | |
| I: 5 4 0 8 | |
| ENDGROUP | |
| NEWGROUP: 5 | The triangles belonging to this subset will be associated to the submaterial 5 of the material 0. |
| I: 3 2 6 32 | |
| I: 6 7 3 32 | |
| ENDGROUP | |
| } | |
| } | |
| ENDGEOMETRY | |

TOADD:
Avatar documentation
C_list
Shaders