RenderWare Graphics

Artist Guide

Maya™ Reference Guide

Contact Us

Criterion Software Ltd.

For general information about RenderWare Graphics e-mail info@csl.com.

Developer Relations

For information regarding Support please email devrels@csl.com.

Sales

For sales information contact: rw-sales@csl.com

Contributors

RenderWare Graphics Development and Documentation Teams

Table of Contents

1. Introduction

RenderWare Graphics is a 2D and 3D graphics library. It is used by programmers to create real-time 3D graphics applications, such as computer games and simulations.

RenderWare Graphics has exporter plugins for 3ds max, Character Studio and Maya which enable artist to create compatible artwork for programmers to place into games.

1.1 What do the exporters support?

The high performance exporters for Maya include support for:

- lights, cameras, meshes, nurbs surfaces and splines.

- hierarchical, blend shape, morp, skin and UV animation.

- inverse kinematics (IK) and constraints.

- vertex prelighting.

- multiple UV sets.

- instancing.

- custom attributes and blind data.

- texture filter and addressing modes.

- material effects - bump, environment, normal maps and dual pass.

- RenderWare Graphics specific material with Xbox and GameCube hardware shader setup.

1.2 What's New in 3.7

Visualizer Connection Wizard

This new wizard greatly simplifies the process of setting up remote connections for use by RenderWare Visualizer.

Lightmap improvements

The algorithms used when the Native Lightmap Export option is enabled have been improved leading to better lightmap UV packing. It is also now possible to mix lightmapped and non-lightmapped materials in the same exported asset. In order to differentiate between lightmapped and non-lightmapped materials you now need to either:

Enable the new *Set Light Mapped* flag for a layered texture using the *Object Settings* dialog.

Set the lightmap flag on a RenderWare material which has a dual pass texture setup.

These steps were not necessary in previous RenderWare version so be sure to perform them when updating old artwork to RenderWare 3.7.

Background color

The camera background color is now preserved when performing views. The color is taken from the background color of the first visible camera found in the scene. If no cameras are visible it is taken from the camera for the currently selected viewport. If no viewport is selected black is exported. The background color can also been added to exported files (see the Technical Artist Guide for more information on enabling export of this data).

1.3   How is this document organized?

This document is split into sections:

• Installation and setup of RenderWare Graphics for Maya.

• Modeling Reference

• Material References

• General References

• Exporting

Maya 4.5 screenshots are used throughout this document.

1.4   Other Documentation

• MayaTutorials.pdf - the tutorials document is organized around a series of tutorials that take you through the basics of RenderWare Graphics as well as some of the issues you need to think about as you create 3D worlds. If you're new to RenderWare Graphics this is where you should start. .

- [TechnicalArtistGuide.pdf](#) - this document is aimed at advanced/technical users (artists/programmers)of the RenderWare Graphics Exporters. It describes in depth a process of customizing export templates and export process itself.

- [OptimizeStaticGeom.pdf](#) - this document is a case study of how to optimize static geometry using knowledge of the PlayStation 2 architecture.

- Three viewers can be used to view artwork exported using the RenderWare Graphics exporters. The viewers are: RenderWare Visualizer; Clump View and World View. There are two viewer documents describing the controls and setup of these viewers [RenderWareVisualizer.pdf](#) and [ClmpviewWrldview.pdf](#).

- RenderWare Graphics has a range of documentation material aimed mainly at developers but still useful for the artist. After installation, take a look at additional documents in the docs directory. The User Guide in particular should be useful to you as it covers a lot of material relevant to the artist.

- RenderWare's Fully Managed Support Service (FMSS) contains RenderWare Graphics Art Examples, which are available for download. In the FMSS https://support.renderware.com/, click *Downloads* on the left of the screen.

PDF format: Most RenderWare Graphics documents are in PDF format, which is a self-contained document format from Adobe. You'll need to install the (free) Acrobat Reader to view and print these. In some cases the quality is better in the printed form than on-screen.

1.5 Multi-platform

Direct3D, OpenGL, Sony PlayStation 2, Microsoft Xbox and NINTENDO GAMECUBE platforms are all supported. These platforms are very different which means that in some situations you'll need to adapt your working methods to suit the platform.

1.6 Software Supported

The examples and details in this document are based on Maya 4.5 and 5.0, and RenderWare Graphics 3.7. The tutorials assume that you have good knowledge of 3D animation concepts and are familiar with basic Maya operation. Windows refers to Windows XP or Windows 2000.

1.7 Known Issues

8 bit `TIFF`s do not work correctly when exported from Maya. This is a known Alias|Wavefront issue.

2. Installation and Setup

You will normally use the RenderWare Graphics setup program to install the Maya exporter. Assuming that the setup program finds your Maya installation directory correctly (or you point it there) it will copy all the exporter files to the correct places.

2.1 Files

For your reference the installed RenderWare Graphics files are as follows:

- `RwExport.mll` to `Maya\bin\plug—ins`.

- `RenderWareEditor.mel`, `RenderWareMenu.mel`, and `RenderWareNodeExport.mel` to `Maya\scripts\others`.

- `AERenderWareExportTemplate.mel`, `AERenderWareHintTemplate.mel`, `AERenderWarePartitionTemplate.mel` and `AERenderWareShaderTemplate.mel` to `Maya\scripts\AETemplates`.

- `USERRENDERWARE.BMP` and `USERVISUALIZER.BMP` to `Maya\extras\icons`.

- `render_RenderWareShader.xpm` to `Maya\icons`.

An additional set of common exporter files are installed in your main RenderWare Graphics directory and added to your path environment variable. The exact location depends on where you choose to install RenderWare Graphics to but the default is `c:\Rw\Graphics\export\bin`. The files stored here are as follows:

- `ExGrid.dll`, `ImgMagick.dll`, `RwAxCtrls.dll`, `RwComm.dll`, `RwExp.dll`, `RwExpMgr.dll`, `Rwgnull.dll`, `RwInstance.dll`, `RwMI.dll`, `RwRf3.dll`, `RwXML.dll` and `RwMaya45.dll` or `RwMaya5.dll`

`Maya` in this instance refers to the full path name of your Maya folder.

2.2   Maya Setup

After installation there are some one-time changes needed.

Plug-in Setup

Before you can run the exporters you'll need to tell Maya to load the plug-in files. To do this, start up Maya and open the plug-in manager using *Window→Settings/Preferences→Plug-in Manager*. Add checkmarks to the *loaded* and *auto load* boxes for RwExport.mll. Make sure any previous RenderWare Graphics plug-ins are not loaded.
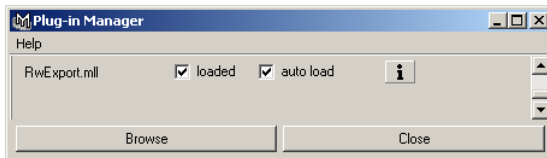
Display Backface Culling

By default Maya does not backface cull the polygons rendered in its viewports. This means that all your surfaces look double-sided. Since RenderWare Graphics does backface cull by default we recommend you enable culling in Maya to ensure that the exported geometry matches what you see in Maya. This is set in the *Custom Polygon Display Options* dialog, found under *Display→Custom Polygon Display* options. To enable backface culling for all polygons set the *Objects Affected* option to *All*, the *BackFace Culling* drop down to *On* and click *Apply*.
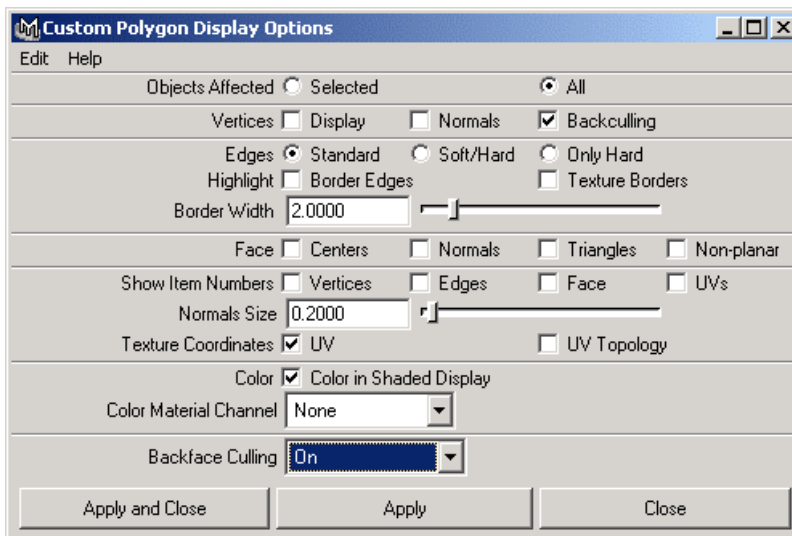
2.3 RenderWare Exporter

- To access the RenderWare Exporter use the *RenderWare* menu. The menu options available are:

  - Export RenderWare Assets

  - View RenderWare Assets using Visualizer

  - Object Settings

  - Advanced Settings

  - Create sectorization hints and particles (part of the FX Pack)

  - Tools for RenderWare Triangulate, Batch Exporting and Template Updates

  - Plugin Manager listing all the RenderWare plugin files used

  - Help for all RenderWare Graphics exporter documentation

From the command line

To access the Advanced Settings dialog from the command line type

```
RenderWareAdvancedSettings
```
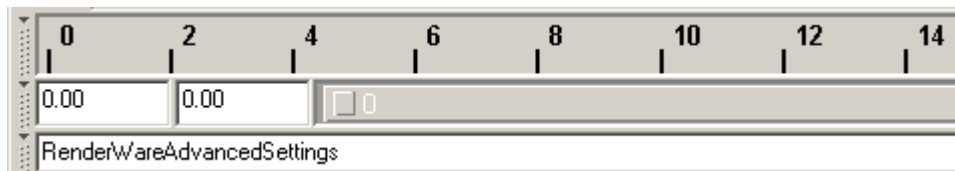

From a button

You may want to add an *Advanced Settings* button to your Maya shelf. Do this by typing *RenderWareAdvancedSettings* in the command line selecting it and middle-click dragging it to the shelf area. This should create a default MEL button on the shelf. To change the button from the default graphic, open the shelf editor (*Window→ Settings/Preferences→ Shelves*) select *RenderWareAdvancedSettings* in the *Shelf Contents* tab and use *Change Image* to select USERRENDERWARE.BMP in Maya\extras\icons.
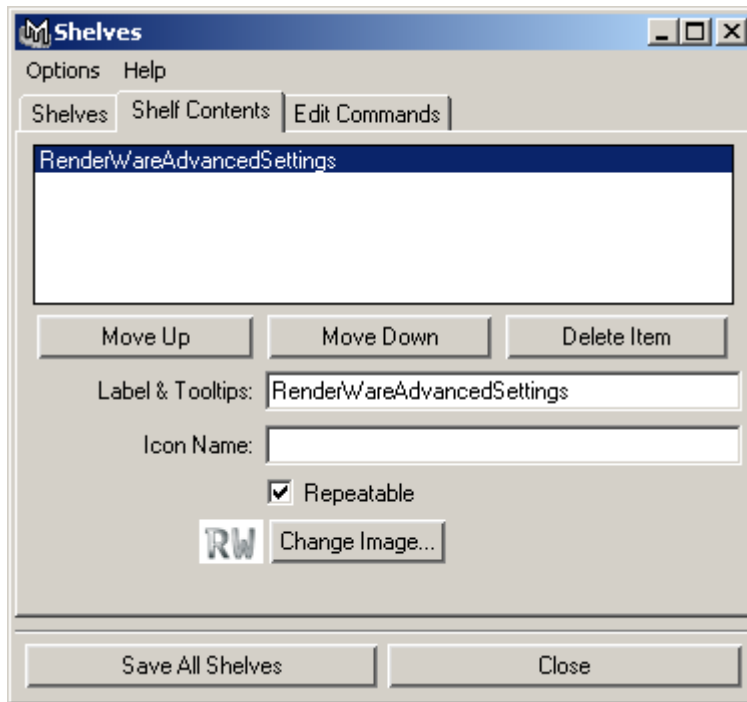
The button should now appear in Maya as shown below.
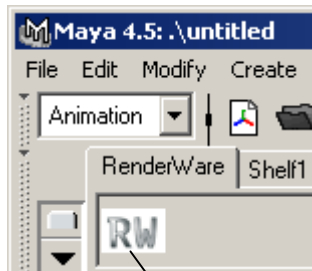
RenderWare Advanced
Settings button

RenderWare View

You may find it useful to add another common RenderWare command to the menu. This command is *RenderWareViewExport* and is used to view the current scene in RenderWare without manually setting up any export nodes. As with the *RenderWareEditor* command, simply type *"RenderWareViewExport –a"* in the command line, select the text and middle click drag it to the shelf area. The installation includes an icon file you can associate with the command, USERVISUALIZER.BMP in Maya\extras\icons.
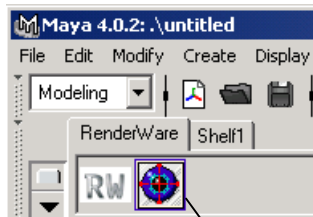
RenderWare View
button

2.4 Installing the Exporter Manually

To install the exporter by hand, you need to setup your environment variables, update the Windows registry and copy the exporter into your Maya directory. This section tells you how to do this.

Setting up the environment variables

Setup your `PATH` environment variable with the directory path that contains the exporter DLLs: Rwg.dll, RwExp.dll, RwComm.dll, RwExpMgr.dll etc. (this is usually rw\graphics\export\bin):

1. Select Start -> Settings -> Control Panel.

2. Double-click on System, and then select the Advanced tab.

3. Click on the Environment Variables button, select `PATH` in the list of variables and then click Edit.

4. Add the path to the directory that contains that exporter DLLs (Rwg.dll, RwExp.dll, RwComm.dll, RwExpMgr.dll) to the list of paths, and then click Okay.

To test this configuration, open a new command prompt and type `path`. All your `PATH` settings will be displayed, including the one you added for the exporter. You may need to restart your computer, or at least log on and off, to ensure the registry is updated.

Updating the registry

The following COM controls should be registered using the Microsoft Register Server, `regsvr32`:

*   `ExGrid.dll`, `RwAxCtrls.dll` and `RwRf3Translator.dll`, which are in the `rw\graphics\export\bin` directory.

*   `OutputPanel.dll` which is in `Program Files\Common Files\RenderWare\`

Copying the exporter to your plugins directory

To add the exporter plugin to Maya, copy RwExport.mll into your Maya\bin\plug-ins\ directory. You also need to copy the exporter's MEL scripts into the Maya scripts directory:

- Copy `AERenderWareExportTemplate.mel`, `AERenderWareHintTemplate.mel`, `AERenderWarePartitionTemplate.mel`, and `AERenderWareShaderTemplate.mel` into your `Maya\scripts\AETemplates\` directory.

- Copy `RenderWareEditor.mel`, `RenderWareMenu.mel` and `RenderWareNodeExport.mel` into your `Maya\scripts\others\` directory.

3. Modeling Reference

This section explains how you can model your artwork for RenderWare Graphics exporters so that you can obtain the best possible results. While there is an option of conditioning geometry for static worlds (see the TechnicalArtistGuide.pdf), it should not be seen as a replacement for good initial artwork.

Most of the general principles of good modeling for games apply to RenderWare Graphics. Examples are: minimizing vertices, keeping texture map sizes small, tri-stripping etc. There are a few specifics worth keeping in mind and these are listed below.

3.1 Triangle Strips (Tri-strips)

Tri-strips are an efficient way of storing geometry. Tri-strips consist of adjacent triangles which share vertices. Artwork created using tri-strips can reduce the number of vertex repetitions by a factor of three.

The series of adjacent triangles below use tri-stripping. Vertices 1, 2 and 3 draw the triangle A. Triangle B requires only one vertex to define it, vertex 4, as it uses vertices 2 and 3 which have already been drawn.
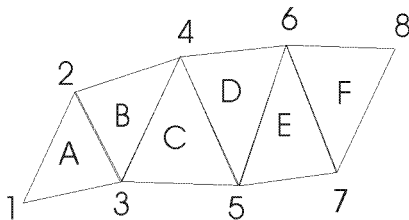
Improving tri-stripping performance

RenderWare Graphics calculates the number of triangles and the number of triangles actually processed by the hardware. The difference between these two statistics is the number of degenerate triangles inserted between the end of one tri-strip and the beginning of the next. To improve tri-stripping performance and reduce the number of degenerate triangles there are a few golden rules:

1. Avoid using tri-fans, or T junctions in meshes. These topologies are difficult to tri-strip. Where possible create surfaces using quads with each quad split into two in the same direction. Breaks in surface geometry will terminate tri-strips and this in turn will have an impact on rendering performance.

2. Use as few materials for each model as possible as tri-strips can not be created across material boundaries. Only one texture is allowed per material. Materials or normals applied to an individual triangle within a tri-strip, will break the tri-strip.

3. For single pass texturing RenderWare Graphics vertices can only have one UV coordinate. UV coordinates should be shared on vertices wherever possible as vertices in the same position with different UV coordinates can not be considered for tri-stripping. This involves using the same texture on as many adjacent polygons as possible. Ideally one texture is applied around a whole object, and the UV coordinates mapped onto the mesh. Texture coordinates should be laid out with tri-stripping in mind. A typical example is, a series of quads tiled with same texture, but with each polygon having its UV coordinates in corners of the texture. In RenderWare Graphics the programmer should ensure that the texture is set to `rwTEXTUREADDRESSWRAP` and the texture coordinates of the quads continue across the repeating texture rather then starting at the beginning for each quad.

4. When applying lightmapping, the lightmapping plugin processes the geometry creating UVs for each triangle to use. The plugin tries to use UVs so that the lightmap textures are used as efficiently as possible. This means that these UVs may not be the same for two adjacent triangles (i.e. the shared vertices will have 2 UVs each). This will stop the tri-stripper from being able to create a single tri-strip from them. The lightmap creation code is designed to minimize this problem as much as possible, but there is a trade-off between good tri-stripping and small texture sets for the lightmaps.

5. RenderWare Graphics vertices can only have one normal. Where two vertices at the same position have different normals, these cannot be considered for tri-stripping, therefore it is important to smooth your models (especially if you are using pre-lighting, as normals are not used). If the models are built with vertices that can not be shared, tri-stripping will try to connect strips by building extra polygons between the disconnected ones. If the models are very disjointed, these extra polygons can add large amounts of data to a scene. In this instance, tri-lists may be more suitable

6. For untextured objects, or prelit objects where the vertex colors vary across the surface of the mesh, the color at shared vertices should be the same. Two vertices at the same position with different colors cannot be considered for tri-stripping.

3.2   Triangle Lists (Tri-lists)

A tri-list is like a tri-strip but instead of sharing vertices a tri-list uses three vertices per polygon in every case. This is usually a lot less efficient than tri-strips if the models are well built, but if the models don't tri-strip very well then it can lead to improvements. This is because tri-stripping a bad model forces RenderWare Graphics to create lots of extra (degenerate) linking polygons.
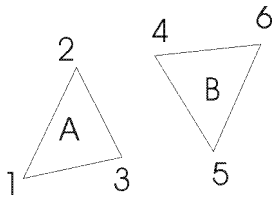
3.3   Triangle fans (Tri-fans)

A tri-fan is a way of creating a surface by placing a central vertex and having the individual polygons radiate from that. Because of the way consoles deal with these shapes they tend to be slow so try to avoid them in favor or a more grid like tri-strip approach.
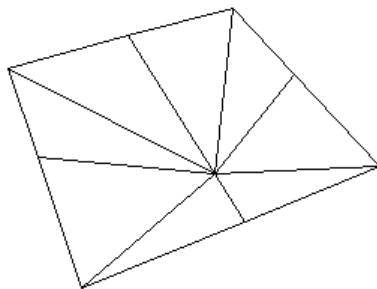
3.4   Connect Vertices

In your models all the vertices should connect and there should be no gaps, overlaps or T-junctions. T-junctions, like the central vertex shown below, will cause problems along the horizontal seam.

Problem T-junction
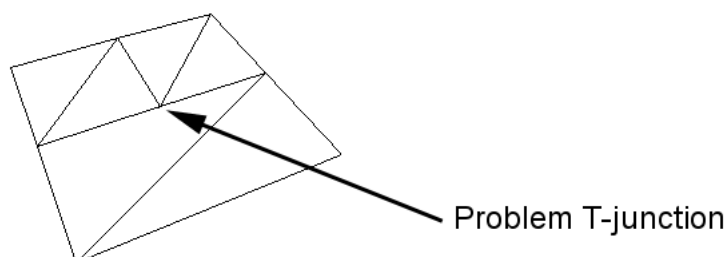
To optimize geometry, all objects that are to be exported as a single *Static World* file should be welded together into one object.

3.5 Scale and Clipping Planes

There are some hard-coded limits to where the clipping planes can be set in the viewers. This can cause problems with very large models. Generally, use a near/far clip ratio of not more than several thousand to one. Making the ratio any greater can lead to problem artifacts in the game (from the low z-buffer resolution). For example, using a near clip plane of 0.01 and a far clip plane of 10000 will probably cause problems, whereas using a ratio of 10:10000 should work well. Scenes should be scaled with this in mind. Note that the exporters can scale scenes during the export process.

3.6 NURBSpline Curves

The RenderWare exporters have basic support for exporting NURBS from Maya into RenderWare as RpSpline assets. RpSpline curves in RenderWare are implemented using Uniform B-Splines. When exporting from Maya NURBS to RenderWare B-Splines, only the edit(knots) or control points will be exported, not both. This can lead to visual differences between the Maya NURBS and the RenderWare Uniform B-Splines.

The recommended way to preserve the appearance of NURBS, when exporting to RenderWare, is to create them using the EP Curve tool. Set the curve degree to 3 Cubic, with a uniform knot spacing and do not edit the control points. This will give you the closest approximation to a RenderWare Uniform B-Spline.

Also available is the Rebuild Curve functionality within Maya that allows you to rebuild any problem curves to suit RenderWare B-Splines. Again, it is best to have Rebuild Type of Uniform and Keep NumSpans checked, to preserve the same number of edit points. The Degree should be 3 Cubic. The Rebuild Type Curvature is also useful as it creates extra edit points on the curve areas. It uses a threshold to determine the interpolation amount. However, too much interpolation will create excessive edit points and make the exported spline less optimal.
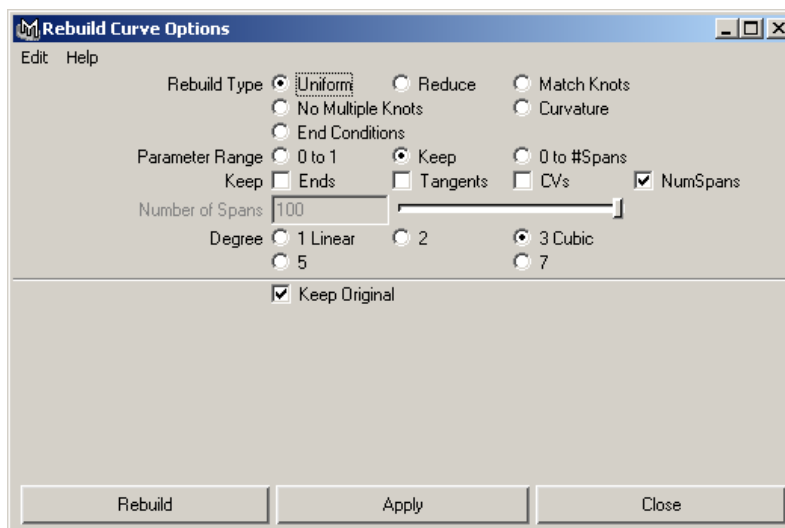
Curves can be exported using either the edit points or the control points using the spline asset template option, "Spline Vetex Type". This defaults to Edit.

3.7   NURBS Surfaces

When exporting using a *Static World* template, all NURBS surfaces are converted to polygons. However, to ensure maximum control over the degree of tessellation we recommend you use the Maya *NURBS to Polygons* functions yourself before export.

When exporting using the *Animated Hierarchy* template you have control over whether the NURBS surfaces are converted to polygons or to RenderWare Graphics Bézier patches. This is set using the *Export Patches* option in the *Animated Hierarchy* tab in Advanced Settings. If you do intend to export as RenderWare Graphics patches we recommend you ensure your Nurbs surfaces are modeled from the outset as Bézier patches in Maya. If you already have Nurbs surfaces that you would like to convert to Bézier patches you can do this using the *Edit NURBS→Rebuild Surfaces* tool. You should set the options as below:
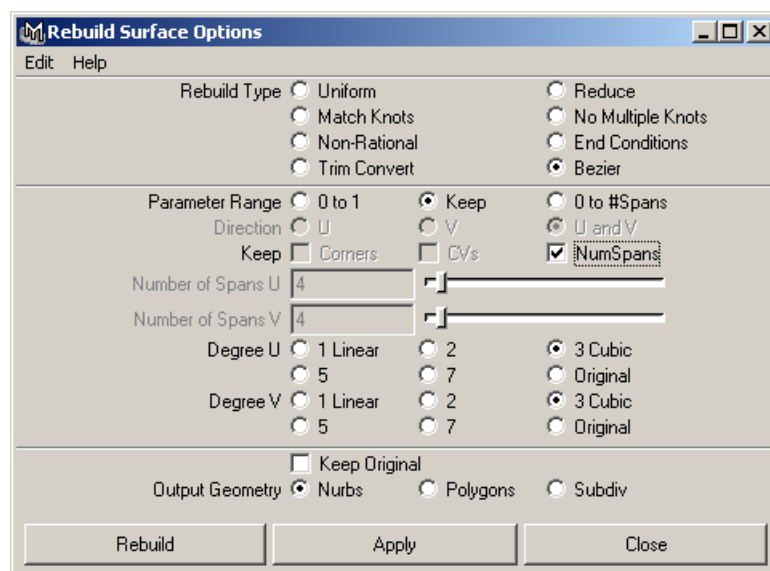
It is particularly important that you do this conversion yourself if you wish to export skinned patch objects. When an already skinned NURBS surface is converted to patches during export the skin weights may not match exactly to the new surface and skinning errors can result. We recommend you use the above conversion and re-bind your skin. This will ensure that the result you see in Maya closely matches the result in RenderWare Graphics.

3.8   Normals and Smoothing

RenderWare Graphics supports a single normal per vertex. In Maya, normal smoothing makes it possible to have multiple normals per vertex. In these situations the exporter has to duplicate vertices to maintain the different normals. So to maintain the hard edged look RenderWare Graphics has to create three or more times as many vertices. In addition, it's not possible to tri-strip across the edges as the vertices are not common.

If you want to retain a low vertex count and good tri-stripping you will need to ensure that normals are shared between faces. This can be achieved by going to *Edit Polygons*➔*Normals*➔*Soften/Harden* and setting the value of *Angle* to 180.

The screenshots below show a pair of default Maya cubes before (above) and after (below) the normals have been adjusted. Note that the lower pair of cubes have only one normal for each vertex and that as a result the shading is smoothed around the corners.
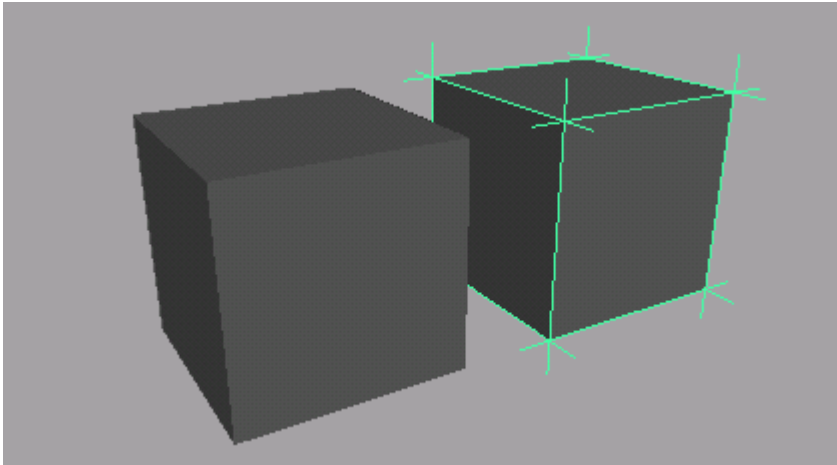
Before: three normals per vertex – notice the sharp corners
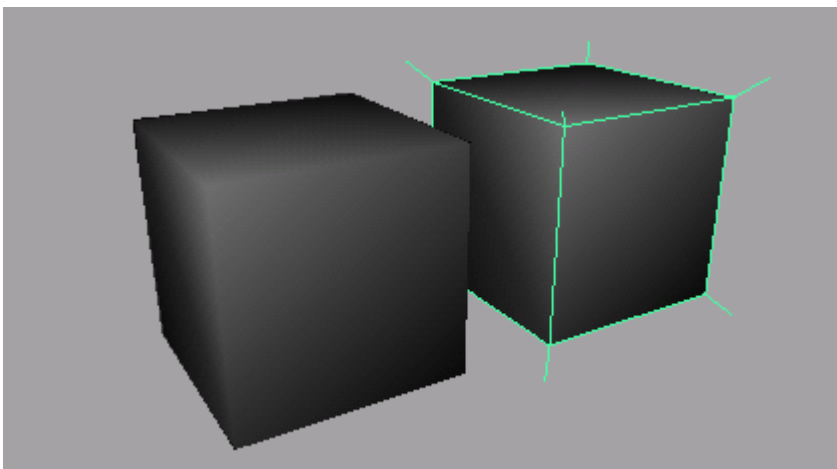
After: one normal per vertex – notice the smoothed corners

3.9   Level of Detail

RenderWare Graphics has the capability to switch models based on distance to the camera. All that is needed from the artist are separate models each exported individually with the required level of detail.

3.10 Instances

It's often useful to use instances in your scenes. When using the *Animated Hierarchy* template the RenderWare Exporter will attempt to preserve instancing, rather than exporting multiple copies of the geometry. Whilst Maya allows the changes to material assignments on instanced geometry this is not supported by RenderWare Graphics. If you want instancing to be preserved when exporting to RenderWare Graphics you will need to ensure material assignments are not changed between instances.

RenderWare Graphics does not support instancing in *Static World* assets so Maya instances are always converted to copies when exporting in this way.

3.11 Potentially Visible Sets (PVS)

RenderWare Graphics uses PVS to decide what can be seen by the user. It checks visibility from each part of the model. The parts that are *not* visible are culled reducing processing time. PVS culls all unseen or occluded geometry.

For example, a camera placed inside a modeled world may not be able to view all visible geometry. This geometry could be behind the camera or blocked by other geometry and is culled.

PVS data is stored in a database known as Potentially Visible Sets (PVS), calculated once per model.

> Not all models benefit from PVS.
>
> Example 1:
> A landscape with few occluders will not benefit from PVS because every sector will be visible from every sector. The more occluders the better the PVS result.
>
> Example 2:
> The generation and efficiency of PVS is affected by cracks and T-junctions.

Visualizer

To view PVS in the Visualizer, open your model and:

- disable Triangles

- enable Wireframe

- enable World Sectors

- enable Use PVS

Your model should display a cyan wireframe for the sector the camera is in. The other colors represent non-visible sectors.
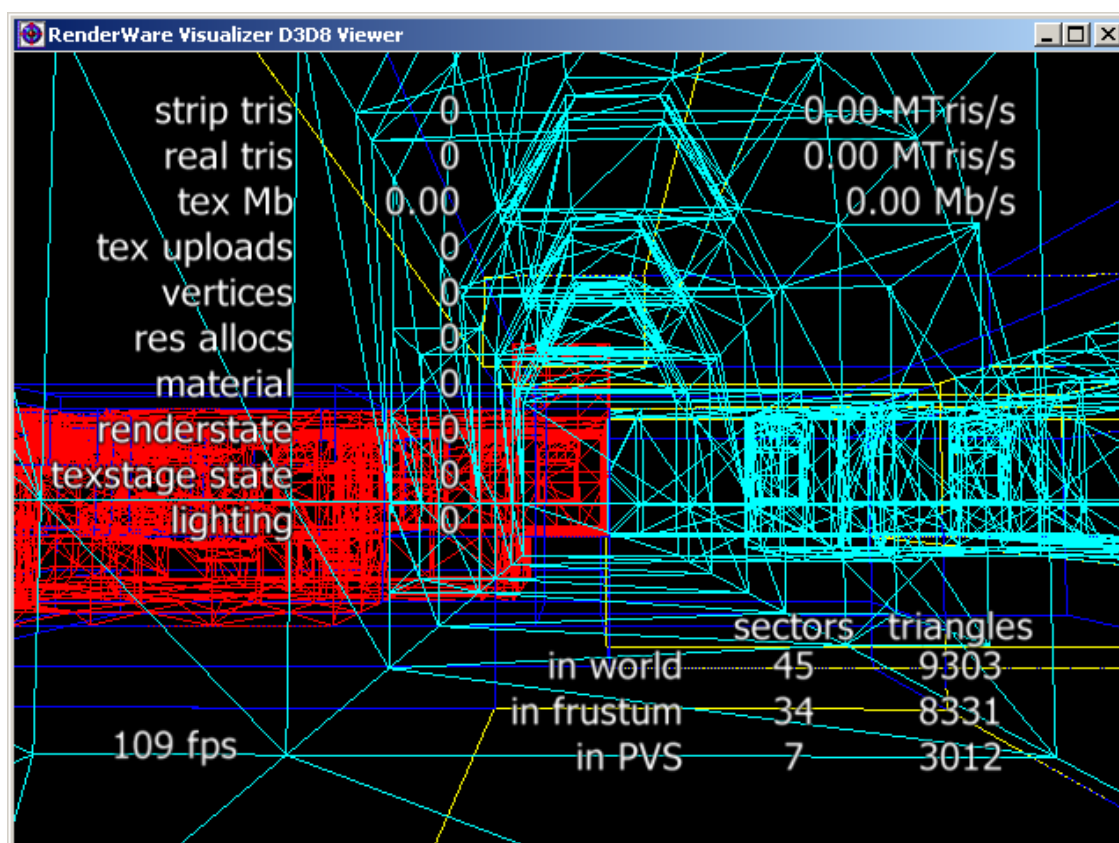
PVS Editor

RenderWare Graphics ships with a PVS Editor. The PVS editor is a tool that allows the PVS data in a world to be edited. The PVS Editor is located: `RW\Graphics\tools\pvsedit\PVSEdit_<platform>.exe`. The PVS Editor has a `.chm` which details keyboard shortcuts and can be accessed from within the PVS Editor. A separate pdf: `RW\Graphics\docs\tools\PVSEdit.pdf` explains the use of the PVS Editor in more detail.

The PVS Editor below, displays color coded sectors. If you are inside the selected white sector, then you can potentially see the blue sectors but you are unable to see the orange sectors. This speeds up performance as unseen sectors will be culled.

✎   PVS Editor does not support `.rws` files.
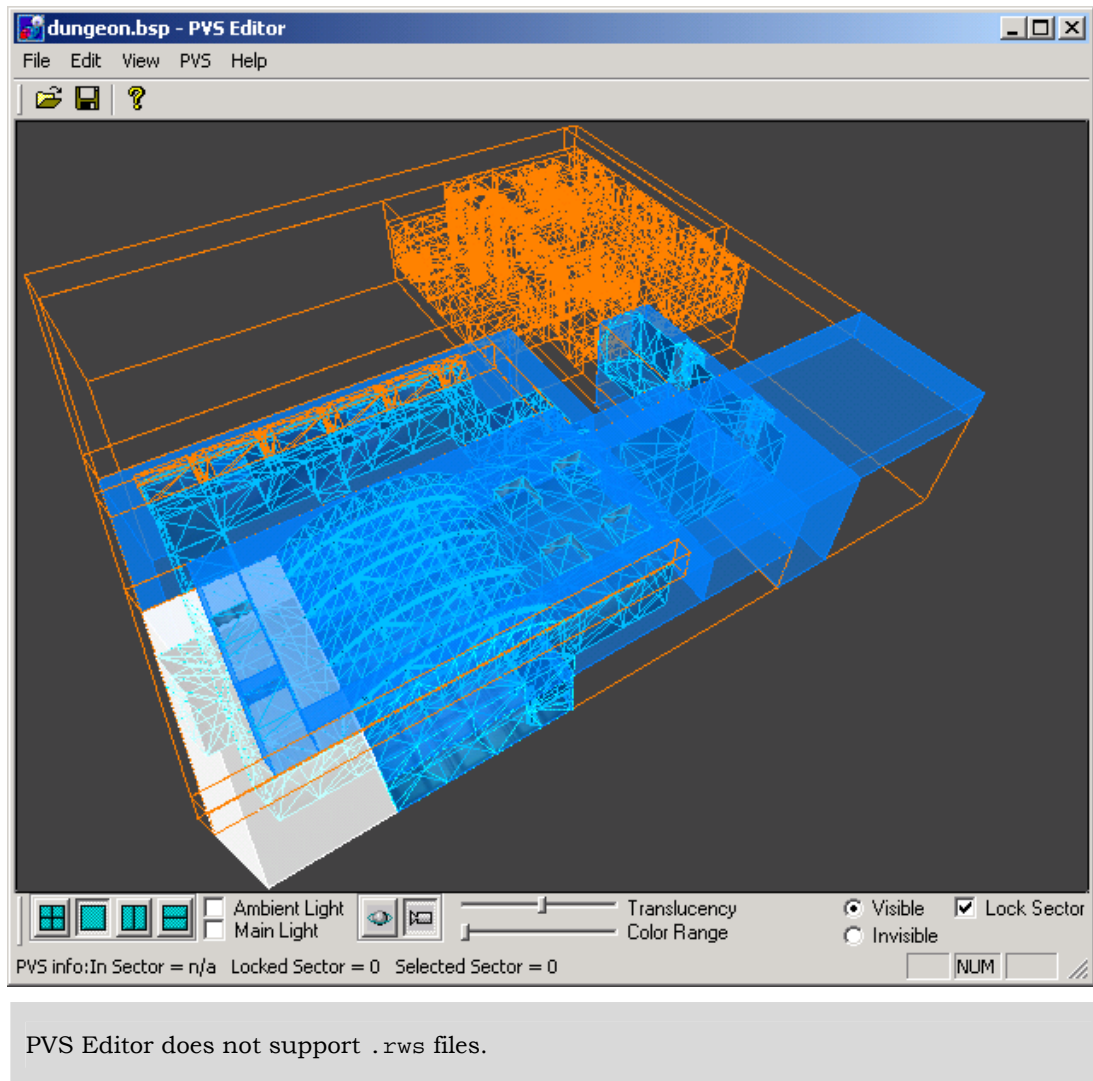
3.12 Hints

While many of the considerations concerning the partitioning of a world are covered in the 3.13 Partitions section the 'hint' object that affects this stage is a Modeling concern.

What are hints

Hints are a special type of primitive that affects the sectorization of the scene on export but do not get exported as geometry to the final model.

Just a few hints placed strategically in the scene can significantly improve the sectorization, and they provide a way to affect the partitioning more controllably than before.

The hints are used in conjunction with the automatic partitioning of a scene.

Hint types

There are two types of hint: Shield hints and partition hints. In addition, they come in different shapes, all orthogonal: Boxes and three plane types (XY, YZ, XZ).

Shield hints

Shield hints attempt to stop partitions cutting through them, and so they can be placed around small, complex geometry, such as a statue, to give the partitioning process a 'hint' as to where to avoid cutting. Shield hints are color coded blue.

NB: In many of the exporter partitioning schemes, a space culling algorithm is applied, and since this, by definition, does not cut through any geometry, it is permitted to cut through shield hints, if necessary!

Partition hints

Partition hints are an additional option that tells the partitioning process the locations where a partition would be appropriate. A bounding box partition hint might have its faces surrounded by partitions, while a plane partition hint would just have its single face coplanar with a partition. Partition hints are color coded red.

Combining hints

Since partitions hints and shield hints are not mutually exclusive, a hint can be assigned both as a shield hint and a partition hint! Combined hints are colored magenta.
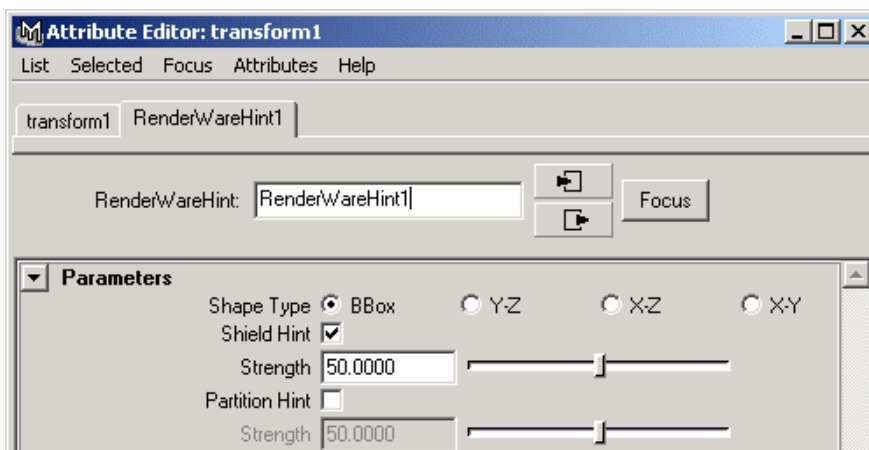
Creating hints

A hint can be created by selecting *RenderWare→Create→RenderWare Sectorization Hint.* Its parameters can be altered using the attribute editor.

The hints attributes dialog box appear as below:

A hint object is drawn just like the box object. In addition, it has a number of options that make it act different, subtly affecting the partitioning process.

Initially, the hint is a bounding box. This can be switched to a plane type by selecting one of the radio buttons governing shape type. Note: A bounding box with a zero dimension can behave differently to an explicitly planar hint, so it is advisable to select the shape of the object before changing its dimensions.

Initially, the hint type is a shield. This can be deselected using the shield hint check box. The partition hint check box can be set likewise. Note, if both hint types are deselected, the hint will not affect the partitioning process at all.

Hint strengths

Each hint type has a strength associated with it. Even further control over the partition process will be gained using them strategically. As a general rule, it is best to leave the strengths at their default value. Only when a particular partition or shield hint should work differently should its value be altered.

For additional feedback, the strength of the hint is associated with the brightness of it on the screen.

Here are some insights into the use of hints and their strengths:

- A partition hint with a high strength will be picked in favor of one with a lower strength. This means it will most likely be a parent of a lower strength one if that one is picked at all.

- Partition hints are affected by shield hints, but regardless of what they cut through they will be picked (to respect the artists intentions).

- If there are two partition hints and one of them cuts a shield hint, the other one will be picked in favor of the latter.

- A partition with a strength of x that cuts through two shield hints with powers of y and z with have a resulting strength of x-(y+z)

Hint Example

Here, we see a staircase that has had a set of hints applied to it. (Note, normally, such a scene would require fewer if at all any hints.)
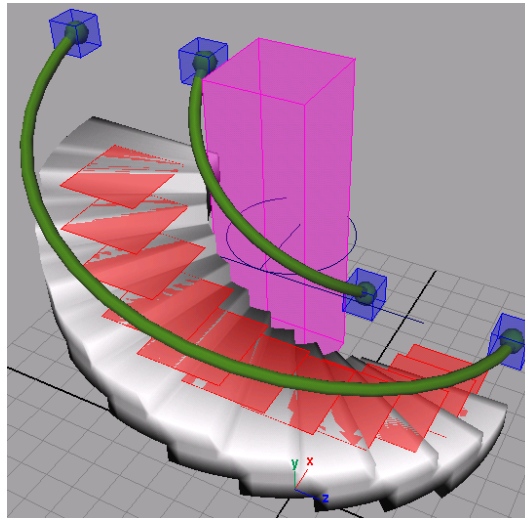
Shield hints (blue) have been placed around each banister knob in order to stop them being split; Partition hints (Red) have been placed on each step so that the world is partitioned into step-high sectors; and a combined hint (magenta) has been placed in the void so that it is partitioned around but not occupied by partitions. In the 3.13 Partitions section we'll see how this affects the partitioning of the world.

3.13 Partitions

The *Visualize Partitions* option in *Static World* asset properties shows the partitions/sectors that are generated as a result of the export process.

Whilst the partitioning of an exported world can be viewed in the viewer, it is sometimes useful, especially when using hints, to view them in the art package.

Note, the partitions are a read-only mechanism and are locked in place. No manipulation of them can be used as a feedback to the export process and they should only be used to study the final result of the export.

Initially, they are drawn as sectors, colored based on each sectors depth in the BSP tree (blue being root-most through the leaf-most red). The display options can be changed so that the partitions (which define the sectors) are displayed instead.

Once created, the partition attributes can be modified using the *Attribute Editor*.

The partitions attributes dialog box appears as below:
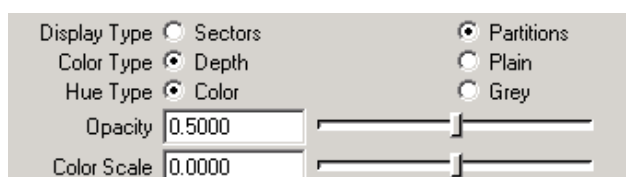
For additional visual information there are two sliders that affect the opacity and the color-range of the visualization. Opacity can be set to any value between 0 and 100%. The color-range default is zero, this means that in a scene whose BSP tree has seven depths, each partition is assigned a color of the rainbow; sliding the color-range bar to the left shifts the range to the red-end of the spectrum, and to the right the blue-end. This is useful for identifying where in the BSP tree a partition lies, and provides further color enhancement.

If the visualization is overwhelming, it can be toned down: The coloring can be set to 'plain' which displays the partitions in a single color only. The hue type can be set to gray-scales which removes the color component.

These are two views (colored partitions and gray-scale sectors) of the example scene, from the 3.12 Hints section:
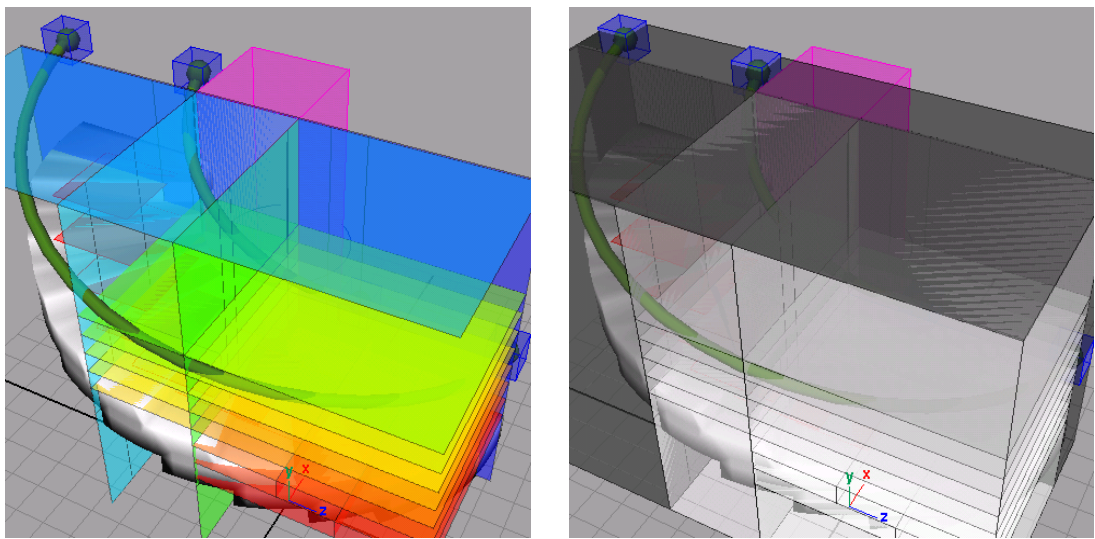
We can see how the partitions are visualized. Furthermore, we can see how the hints have been honored with the shield hints being uncut, and the partition hint being used to create some of the sectorization.

4. Material Reference

When producing artwork for RenderWare Graphics within Maya you have two options for how to create materials. The first method is to use the built in Maya materials (Blinn, Lambert etc). The exporter will convert these materials to RenderWare Graphics materials at export time. The second option is to use the exporter specific RenderWare Shader material.

There are a number of advantages to using the RenderWare Shader:

- The shader restricts material options to those that RenderWare Graphics supports. This makes it clearer what settings the exported material will have.

- The shader allows you to link to platform specific effect files. Effect files contain a low level description of the material state for a given hardware platform. Effect files are created using a standalone, platform specific editor tool (FXEditor for Xbox and GameCube). They are complex to design but present the full capabilities of the target platform without translation.

It's a good idea to use large areas of a single material in a sector for good tri-stripping.

More information about the RenderWare Graphics' plugin `RpMatFX` and `RpWorld`'s `RpMaterial` can be in the RenderWare Graphics API Reference and in the User Guide chapters *The Material Effects Plugin* and *World and Static Models*.

4.1 Maya Materials

General Settings

The color of the exported RenderWare Graphics material on an object is based on the Color parameter in the Maya material. If a file texture is applied to the Color channel the texture filename (stripped of path and extension) will be used to set the RenderWare Graphics texture name. In this case the material color is taken from the Color Gain attribute of the file texture node.
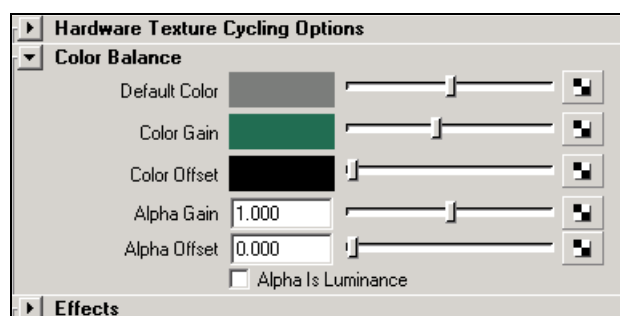
Only file textures are exported, all procedural textures will be ignored.

If a texture is applied to the color channel the transparency channel will also be queried for a texture to use as an opacity map. Because Maya uses transparency whereas RenderWare Graphics uses opacity you need an inverted map. The easiest way is to use an opacity map is to *Invert* under the file nodes options in the *Effects* attributes.
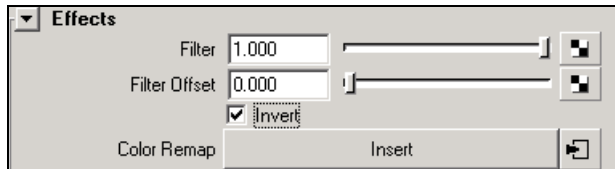
If no color texture or no transparency texture is found the transparency value is read from the Maya material and inverted for use as an alpha value in the color, otherwise the alpha value is set to 255.

Transparency

Transparency is straightforward in RenderWare Graphics. Simply create a material as normal and then in Transparency, in the material's attribute editor, either adjust the level of gray using the slider bar or load in a gray scale map file.

When a RwTexture is created from the diffuse and transparency maps, the maps are merged into a single RGBA texture (with alpha). The name of this new RwTexture is the name of the original diffuse texture. RenderWare uses texture names, so the merged texture is used for each instance of the diffuse texture. This can cause the textures in exported scenes to be different than those expected.

Bump Mapping

The bump mapping effect in RenderWare Graphics works only with file textures. When applying a texture to the Bump Mapping channel of the material be sure to choose *2D Textures→File* in the *Create Render Node* dialog.

The file texture applied to the Bump Value of the bump2d node is the one that will be exported. The Bump Depth value is exported to give you some control over the height of the bumps, but RenderWare Graphics uses a different bump-mapping algorithm than Maya so check your work often.
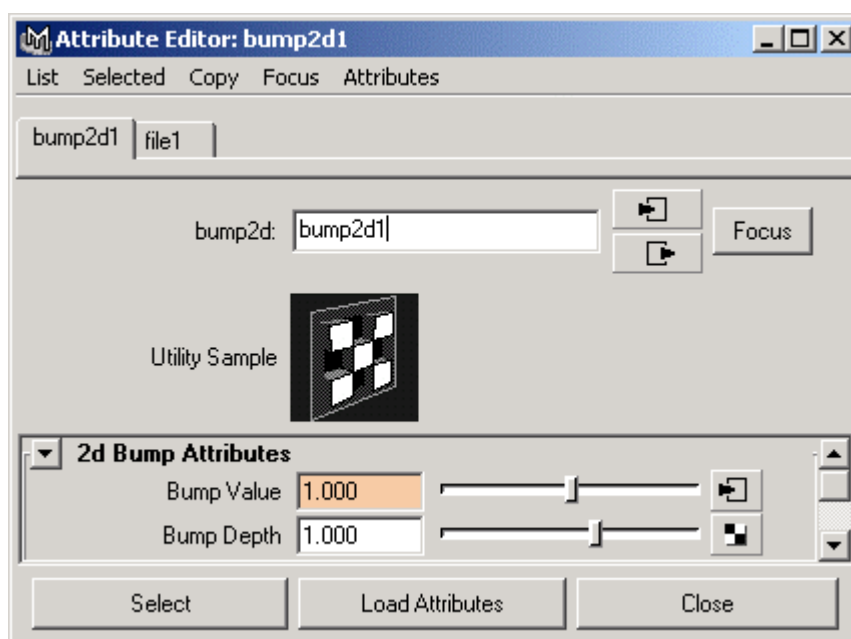
Note that the bump texture must use the same texture co-ordinates as the color texture. In fact the RenderWare Graphics Maya exporter exports only the texture co-ordinates for the color map, the bump texture co-ordinates are assumed to be the same.

Also note that you may not have a color map with an alpha channel at the same time as bump mapping. This functionality is not provided because the performance implications are severe.

You may have color, bump, and environment textures all on the same object and they will export correctly. Unfortunately, having the bump mapping effect the surface normals used for environment mapping would have severe performance implications, so this functionality is not provided.

Environment Mapping

To export a RenderWare Graphics environment mapping effect you should use a Blinn, Phong or other material with specular shading attributes. Add an environment map to the material's reflected color attribute. The environment mapping in RenderWare Graphics most closely resembles Maya's *Env Ball* environment mapping, so it's best to choose that type of environment texture. However, be warned that it is far from a perfect match so check your work often in RenderWare Graphics. The name of the environment texture is picked up from *Env Ball's* image attribute. Note that RenderWare Graphics only accepts file textures as images for environment mapping.

The material's reflectivity parameter is used by RenderWare Graphics to control how the environment map is blended with the color map. The process is additive so a value of 0 is 100% color map and 0% environment map; a value of 1 is 100% color *and* 100% environment map, i.e. the colors will be brighter. All other specular parameters are ignored by the exporter.

Dual Pass

To export a RenderWare Graphics dual pass texture effect (for example, for lightmaps) you need to apply a *layeredTexture* node to the color channel of the material. You should use exactly two textures in the layered texture. The first texture layer (the leftmost one) will be exported as the dual pass texture. The second texture (the rightmost one) will be the base texture.

The blend mode specified in the layered texture node is ignored by the exporter. Instead the blend mode must be set from the RenderWare Graphics 6.2 Object Settings (page 75). When a layered texture node is selected the Layered Texture Settings options are enabled and you can set an explicit RenderWare Graphics blend mode.

Enable the "Set Blend Functions:" option and select your src and destination `RwBlendFunctions`. For example:
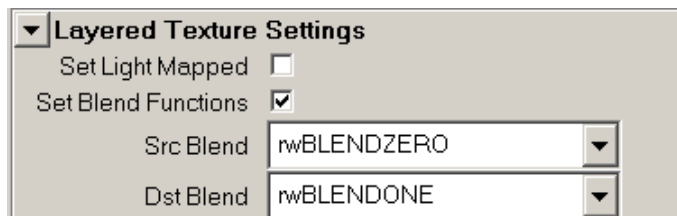
The texture nodes in the layer texture node dialog "Layered Texture Attributes"; are ordered so that the texture node on the left is the top layer and the texture node on the right is the bottom layer.
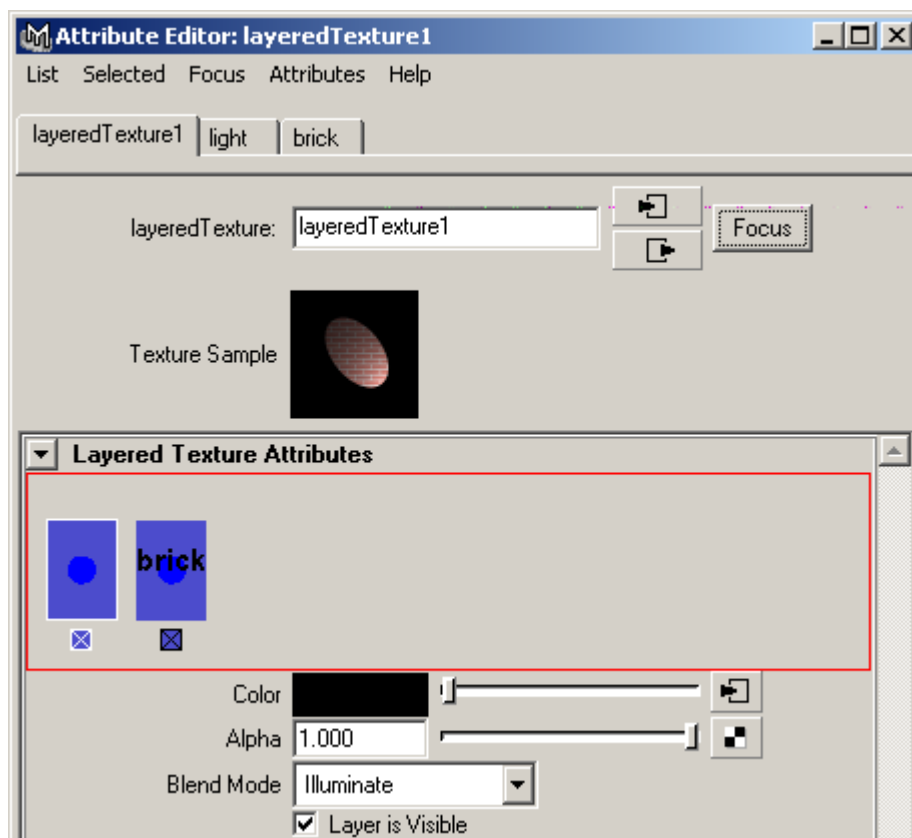
The exporter detects which UV sets the layer textures are connected to and export them as needed.

Texture UV Animation

You can export texture UV animation to RenderWare Graphics. To do so, simply select the color texture that you want to animate in the attribute editor, go to the `place2dTexture` tab, and then add key frames to anyone of the Repeat UV, Offset or Rotate UV fields in the Place 2D texture attributes section.
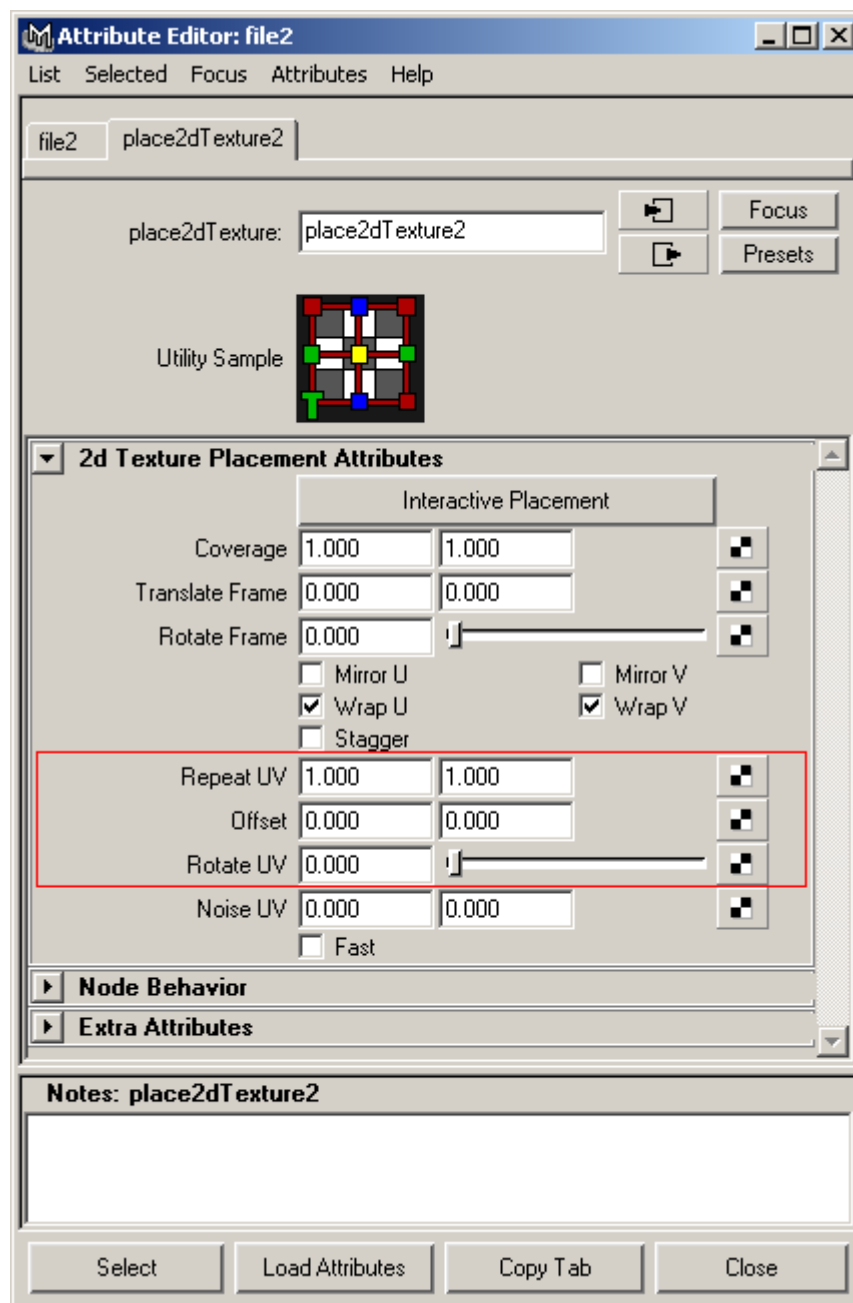
Note that animating the Coverage, Translate Frame or Rotate Frame is not supported, and might lead to wrong interpolation schemes.

You can animate one texture, using a standard color texture, or two dual pass textures, using a layered color texture.

Combining Special Effects

RenderWare Graphics supports any combination of bump mapping and environment mapping but dual pass rendering may not be combined with either. If dual pass and either bump mapping or environment mapping is defined on a single material then dual pass will take preference.

In addition, if you animate the color texture UVs, you cannot use bump or environment mapping. Make sure that these textures are disabled before using an animated texture. If you do setup both bump or environment mapping and UV animation then the bump or env effect will take preference and UV animation will not render.

4.2 RenderWare Shader

It is recommended you use the RenderWare Shader rather than the built in Maya shaders. It provides direct settings for the RenderWare Graphics material and no translation of these settings is performed on export.

To use the RenderWare Shader, in Hypershade, click RenderWare Shader.
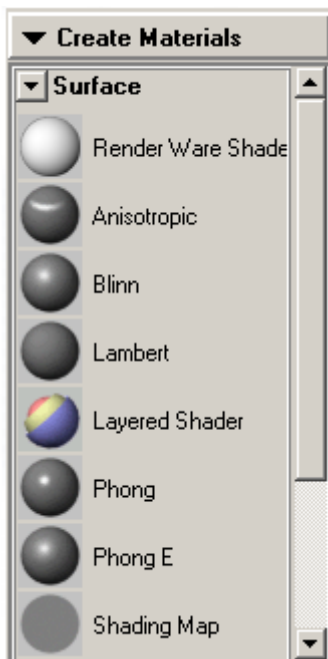
General Settings

The *Default Material Options* frame provides direct settings for the material options available in RenderWare Graphics. No translation of these settings is performed when exporting to RenderWare Graphics.

Clicking on the *Material Color* canvas brings up a Maya Color Chooser dialog where both the color and alpha values for the material may be set.

The *Coefficient* values are used to control how the material reacts to light in the scene. At present the *Specular Coefficient* value is not supported on most platforms.
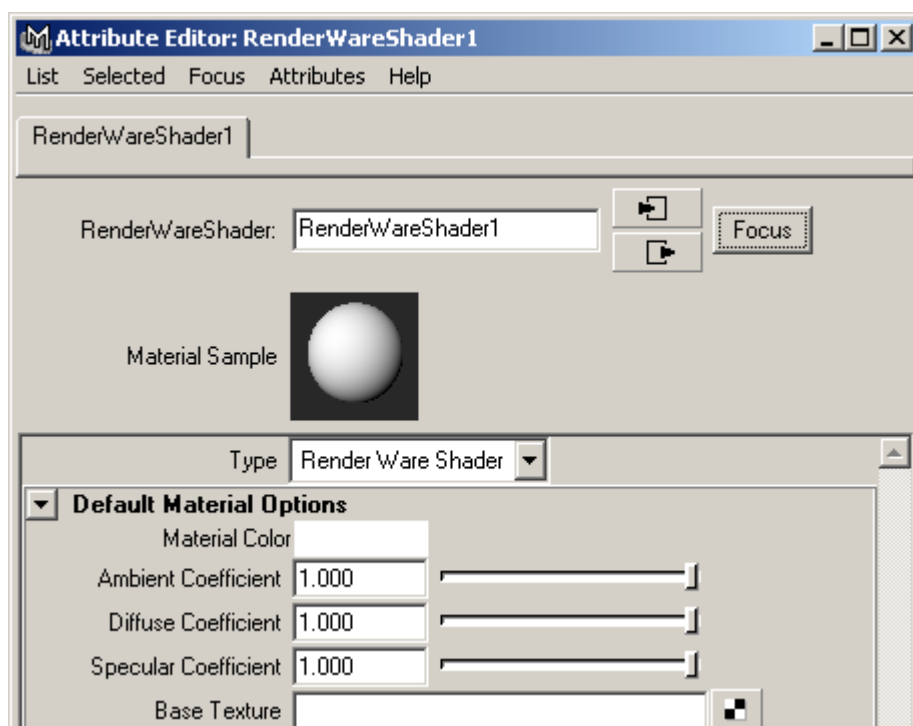
The *Base Texture* control is used to specify a file texture that may be exported as the base texture in RenderWare Graphics. Only Maya file texture nodes are supported. The base texture color is always modulated by the material color in RenderWare Graphics.

MatFX settings

These settings control the MatFX settings for the RenderWare Graphics material. MatFX is used to setup special effects such a bump mapping and dual pass rendering. Depending on the MatFX effect selected, the various options beneath will be enabled. The texture channels can only be used with file texture nodes. When using the dual pass effect, the blend mode used is set explicitly using the *Src Blend* and *Dst Blend* controls. The blend settings in the RenderWare Graphics Node Properties dialog are not used. The *Bump* and *Env Value* settings default to the 0.0 – 1.0 range but any number may be entered using the text fields.

MatFX Options

MatFX Effect	rpMATFXEFFECTNULL ▼
Bump Texture	▣
Bump Value	1.000 [────────]
Env Texture	▣
Env Value	1.000 [────────]
Dual Texture	▣
Src Blend	rwBLENDZERO ▼
Dst Blend	rwBLENDZERO ▼
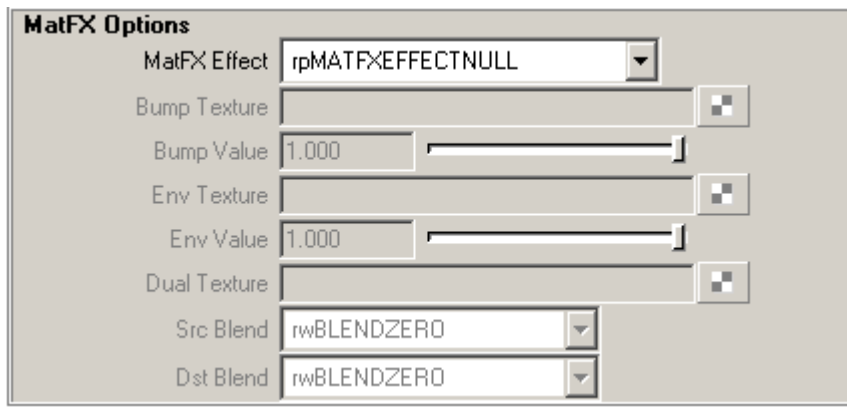
Effect Types

Effect Types relates to the function `RpMatFXMaterialSetEffects()` in the RenderWare Graphics API.

- `rpMATFXEFFECTNULL` has no material effect.

- `rpMATFXEFFECTBUMPMAP` enables bump mapping on the chosen material.

- `rpMATFXEFFECTENVMAP` enables environment mapping on the chosen material.

- `rpMATFXEFFECTBUMPENVMAP` enables both bump and environment mapping on the chosen material.

- `rpMATFXEFFECTDUAL` enables dual pass texturing. This option enables blending modes.

- `rpMATFXEFFECTUVANIM` single pass texture coordinate animation. You can use this effect without setting the UV animation on the texture.

- `rpMATFXEDFFECTDUALUVANIM` dual pass texture coordinate animation. You can use this effect without setting the UV animation on the texture.

Environment and Bump Mapping

Environment mapping is used in RenderWare Graphics to add reflectivity to objects. It can be used to simulate effects such as chrome plating, glass or water. We recommended that you use the RenderWare Shader for environment mapping as it provides the direct settings provided by RenderWare Graphics.

Bump mapping can be used to simulate rough surfaces such as stone or tree bark.

✎ **PlayStation 2**

In RenderWare Graphics for PlayStation 2, `RpMatFXMaterialSetEnvMapFrameBufferAlpha()` is used to modulate the environment mapping by frame buffer alpha (for those video modes that support it). For the bump-environment mapping effect, the frame buffer alpha contains the bump alpha value and the result gives the appearance of the reflection being affected by the bumps.

Dual Pass

This effect works by combining the material's own texture with a second texture according to specified combination flags.

The setup function is `RpMatFXMaterialSetupDualTexture()` and sets the following properties:

- The second texture (the first is defined by the `RpMaterial` object)

- The blend function for the source data

- The blend function for the target data

Platform Specific Settings

The RenderWare Shader provides platform specific settings for a number of targets. At present Xbox and GameCube are supported.

Each platform has an *Enable* control. If this is disabled (the default) there will be no specific settings exported for this platform. There are then a number of texture channel boxes that may be linked to file texture nodes. The number of texture channels varies depending on the capabilities of the target platform. Finally, there is a browser control for a material effect file.
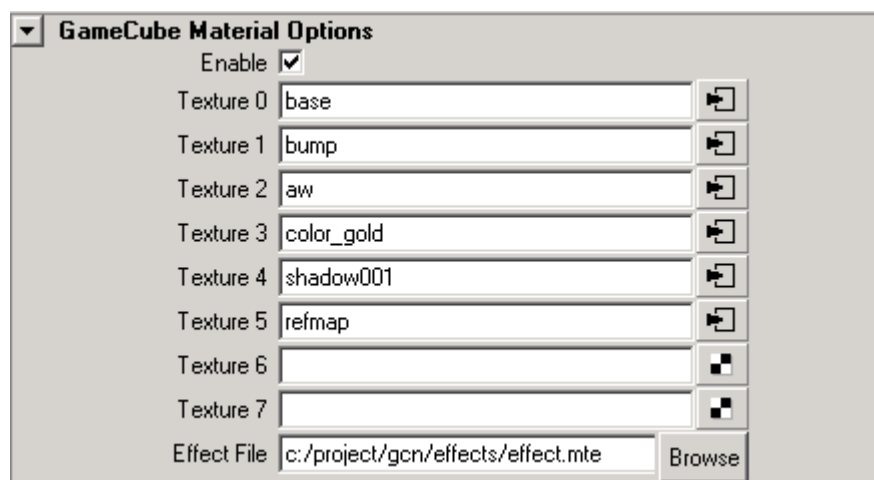
Effect files contain a low level definition of the material hardware state for the platform. Effect files are created and edited using a standalone application (FXEditor for Xbox and GameCube). The FXEditor application is installed as part of the main RenderWare Graphics SDK. It is only provided with the version of the RenderWare Graphics SDK relevant to that platform.

The exporter does not actually load the effect file you specify, it simply stores the effect file name (without a path but with an extension). You do not need to use the *Browse* button to select a file, you can simply enter an effect name in the text field. You will always need to do this if the effect you wish to use is part of an effect file dictionary. Effect file names are limited to 31 characters. If the name you use is longer it will be truncated on export and you will be given a warning message. When your exported assets are loaded into RenderWare Graphics the effect file will be found and loaded. You can view effect files rather like textures. They are edited externally to Maya, referenced by name from the Maya scene and loaded by RenderWare Graphics at runtime.
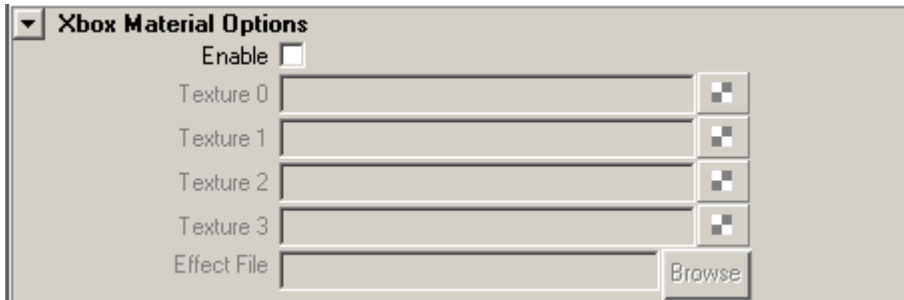
Toon

Toon is a plugin in RenderWare Graphics called `RpToon`. It's a chargeable optional extra in the FX Pack. Toon provides rendering of outlines and "cel shaded" texture based lighting for RenderWare Graphics objects to achieve cartoon-like 3D graphics. An art path is provided to allow the artist wherever possible control over the properties of the lines and shading, offering functionality similar to that of non-real time toon rendering plugins.

It's recommended that you read the Toon documentation to understand toon in more detail.

Toon Paint names can be specified per material to override an object's default paint if you wish.

Checking the Enable box allows the paint name entered in the box below to override the default paint for the object for just this material.

Light Maps

The lightmap options in the exporters provide a means to render geometry in RenderWare Graphics using detailed, full color, static lighting information from lightmap textures.

The lightmaps plugin has limited support for alpha. It works with a one bit alpha, as long as there are totally transparent regions. Sometimes for fine pieces of geometry lightmapping is not the ideal solution and you may find it better to use vertex lighting on detailed objects.
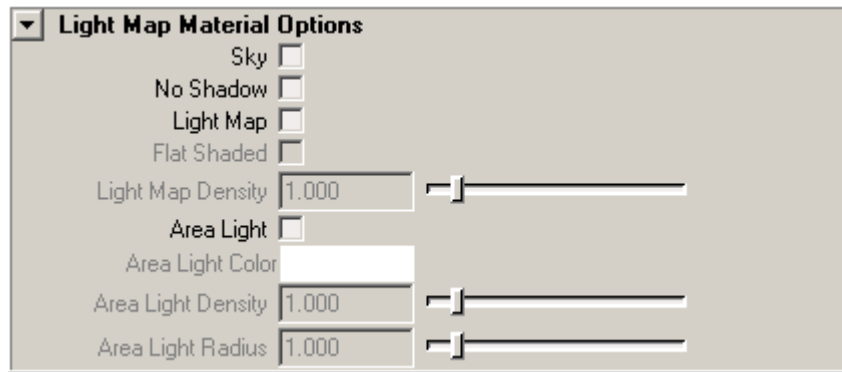
Sets the lightmap options used in RenderWare Graphics. These settings are used by RenderWare Graphics to generate the lightmaps.

The lightmap options for a material are:

Sky - polygons of this material have directional lights shining through them.

No Shadow - polygons of this material do not cast shadows.

Light Map - enables lightmapping for this material.

Flat shaded - enables flat shaded polygons of this material when lightmapping.

Light Map Density - density of lightmaps for polygons of this material.

Area Light - polygons of this material will emit light.

Area Light Color - the color of the area light emitted by polygons of this material.

Area Light Density - polygons of this material are covered by the density of temporary RpLights.

Area Light Radius - the radius of the area light emitted by polygons of this material.

Lightmaps are covered in more detail in the LightMaps section on page 55.

Normal Maps

RenderWare Graphics provides facilities for using normal maps on Xbox and D3D9. Normal maps are textures that store a normal per pixel encoded as a color, converting the components of the normals from the range [-1..1] to [0..255]. These packed normals are later unpacked in the pixel shader and are used to calculate the light contribution per pixel.
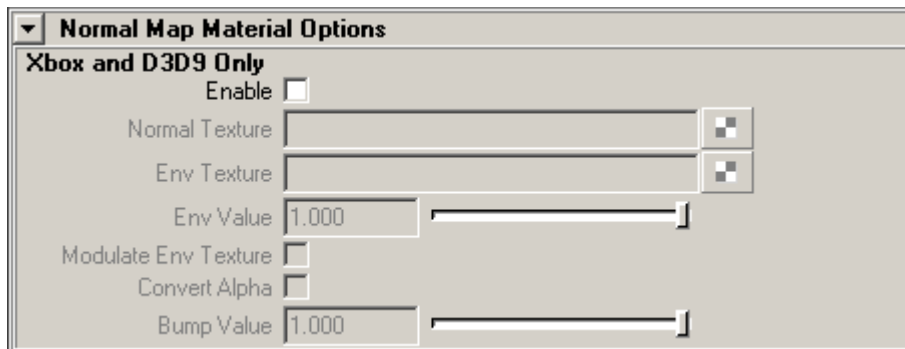
Enable: Enables normal mapping.

Normal Texture: Loads a normal map file.

Env Texture: Environment mapping performed on a pixel.

Env Value: Sets the brightness of the environment map.

Modulate Env Texture: This option controls whether the environment texture should be modulated by the base texture color.

Convert Alpha: Convert standard image to normal map. To use add a standard bump map texture for Normal Texture and select this option. The value that follows is a scale factor for converting to normal maps and calculates the normal pixel.

Normal maps are covered in more detail in the Normal Maps (Xbox and D3D9 specific) section on page 58.

Hardware Preview

The hardware preview options allow you to control which texture channel is viewed in the Maya viewports. To enable hardware texturing the Hardware Texturing option must be enabled in the Maya viewports.

By default the *Software Render Sample* setting is selected. With this setting Maya will render a sample of the material to a small texture and use this in the viewports. This sample is low resolution and gives a blocky appearance in the viewports. Since the RenderWare Shader only supports the base texture channel in software renders this is the only channel you will see. The other settings allow you to pick a single texture channel to view. Maya will use a high resolution version of the texture. If you pick a texture channel that does not have a file texture applied there will be no preview.

The Preview Channel setting is not saved with the RenderWare Shader and will default to *Software Render Sample* when a scene is loaded.

UV Sets

RenderWare Graphics supports up to eight UV sets on export. The Maya UV sets exported are controlled by the UV linking to the RenderWare Shader texture channels. The exporter processes the texture channels and for each that has a file texture, checks which UV set they are linked to. If it is linked to a UV set that hasn't already been referenced, a new RenderWare Graphics UV set will be created. If more than eight unique UV sets are used by the material (possible if more than one specific platform is enabled) then the extra channels will all map to UV set zero.

The order that the texture channels are processed, and hence the order in which RenderWare Graphics UV sets are created, matches the order in which they appear in the attribute editor. Specifically - *Base Texture, Bump Texture, Dual Texture, Xbox Textures 0-3, GameCube Textures 0-7*. The *Env Texture* channel never generates a RenderWare Graphics UV set.

When generating UV data from a RenderWare Shader the exporter does not process any *place2dTexture* nodes connected to file textures. The UV data that is exported is based solely on the Maya UV set data. This is different from the processing of the built in Maya materials.

4.3   Common Material Topics

Texture Maps

Only bitmap texture maps are exported, all procedural texture maps will be ignored.

When creating image maps it is generally recommended that you:

- Keep file names to eight characters or less in the 8.3 format, because some consoles do not support long filenames;

- Use lower case;

- Do not use spaces;

- Save as `.bmp`;

- Ensure that masks are the same size as the base texture.

It's more efficient if you can use the same texture across large parts of your model. Putting different textures on adjacent polygons will force RenderWare Graphics to create lots of extra vertices that will slow down your game.

Texture Size

If textures are loaded into RenderWare Graphics from image files (such as `bmps` or `pngs`) rather than a texture dictionary, the image data needs to be converted to a format that RenderWare Graphics understands. This may involve resampling the image to a different width, height and/or depth which can slow load times. Some graphics hardware requires textures to have a width and a height that is a 'power of two' (i.e. 1, 2, 4, 8, etc). If the source images are not a power of two in either dimension, RenderWare Graphics may resample them. It is perhaps more preferable to resample the images in a paint package since RenderWare Graphics executes a very simple resampling algorithm.

If a textures is of a size that may cause problems on the platform being exported to the exporter will issue a warning in the RenderWare Output Window.

Texture Repeats

On some platforms there are strict limits to the texture UV range that may be used. For instance, on PlayStation 2, for artifact-free rendering the size of your maps and the number of repeats should be restricted so that the product of the map size and the UV value has a magnitude less than 2048. For example, a 256x256 map should not have UV values outside the range plus or minus eight. Rendering outside this range will introduce texturing errors.

The exporters have controls to enable the UV values output to be limited to specified ranges. When UVs are detected outside the specified range they will be shifted towards zero. The downside to this is that there will be discontinuities in previously smooth UV ramps. For example, rather than running 0-1-2-3-4-5-6-7 across polygons the UVs will now run 0-1-2-3-4 0-1-2. This will not affect the rendering but will cause vertices to be duplicated at the discontinuities and result in poorer tri-stripping. If a single polygon is detected with a UV range outside the limit the exporter will not be able to fix the problem (in other words the exporter does not attempt to tessellate geometry to restricted UV ranges).

In Maya there is a tool called Unitize *(Edit Polygons→ Texture → Unitize UVs)* that will set texture coordinates for polygons in the range 0-1, this can help texture coordinate accuracy problems with sharing UVs, but will produce bad tri-stripping.

Texture Alpha

There are two ways of including alpha in your textures. The first way is to use a Maya material with file textures applied to both the color and transparency channels. The texture on the transparency channel will be used as a mask on the color texture. At runtime RenderWare Graphics will load the mask texture and store it in the alpha channel of the color texture. For additional information, see the Transparency section.

The second, and preferable way, is to use a texture bitmap which has embedded alpha.

If you're exporting textures directly to RenderWare Graphics texture dictionaries then all the texture formats supported by Maya are also supported fully by RenderWare Graphics. If you export your assets without a texture dictionary and load the textures later you will find that the only file format the RenderWare Graphics viewers support that includes embedded alpha is png but Maya does not support png. The recommended method in this case is to use a format such as tiff or tga within Maya and convert the texture to png for use in RenderWare Graphics viewers. Since the exported RenderWare Graphics texture links do not store a file extension the png version of the texture will be picked up at runtime.

You should be careful when deciding how to export transparent objects from your scene. RenderWare Graphics does not guarantee that transparent objects are sorted before rendering and therefore transparent polygons may not always render as you expect. You should talk to the programmers on your project about how they want to handle transparency as it may affect how you need to model your scene. It is quite likely you will need to split transparent objects into separate *Animated Hierarchy* or *Static World* assets so they can be rendered after the opaque parts of the scene.

Texture Addressing Modes
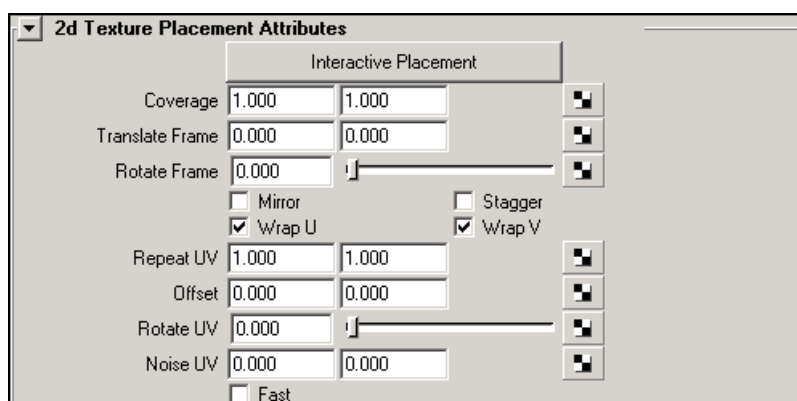
The texture addressing mode of the exported RenderWare Graphics texture is controlled by the *Wrap U*, *Wrap V* and *Mirror* options in the *Placement Attributes* of the *place2dTexture* node. The table below illustrates how the Maya options affect the RenderWare Graphics texture addressing mode. Wrap V affects the RenderWare Graphics V addressing mode correspondingly.

| WRAP U / WRAP V	MIRROR	RENDERWARE UV ADDRESSING MODE
Off	Off	Clamp
On	Off	Wrap
Off	On	Clamp
On	On	Mirror

Texture Filter Modes

The filter mode of the exported RenderWare Graphics texture can be controlled in two ways. The first is to use the Filter Type setting on the Maya texture file node.
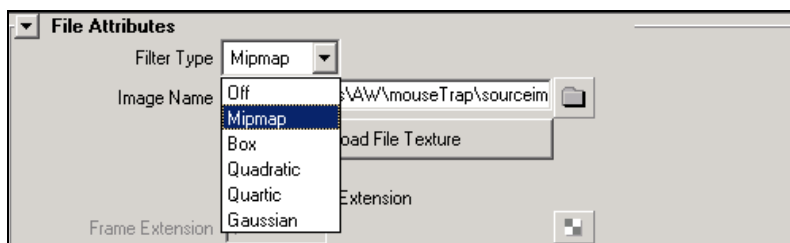
Because the Maya filter modes do not map exactly to RenderWare Graphics ones not all RenderWare Graphics settings are available. The table below illustrates how the Maya Filter Types map to the RenderWare Graphics filter mode.

FILTER TYPE	RENDERWARE GRAPHICS FILTER MODE
Off	Point sampled (`rwFILTERNEAREST`)
Mipmap	Trilinear interpolation (`rwFILTERLINEARMIPLINEAR`)
Box	Bilinear interpolation (`rwFILTERLINEAR`)
Quadratic	Bilinear interpolation (`rwFILTERLINEAR`)
Quartic	Bilinear interpolation (`rwFILTERLINEAR`)
Gaussian	Bilinear interpolation (`rwFILTERLINEAR`)

The second way of controlling the filter mode is to use the *RenderWare➔Object Settings* dialog. When a file texture node is selected the *File Texture Settings* options will be enabled. If *Set FilterMode* is enabled then the exact RenderWare Graphics filter mode can be set using the drop down box.
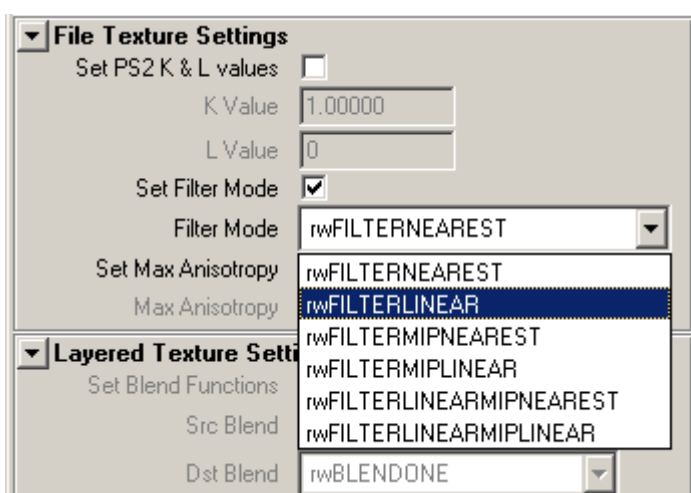
The RenderWare Graphics Filter Modes are documented in the `RwTextureSetFilterMode()` function in the API Reference.

Blend Modes

Texture blending is the technique of combining the colors of a texture with the colors of the surface to which the texture is applied.

Blend modes are set in MatFX settings (page 37) and Layered Texture Settings (page 74) in *RenderWare➔Object Settings*.

Src Blend and Dest Blend are used in the function `RpMatFXMaterialSetDualBlendModes()` in the RenderWare Graphics API. To use blend modes within RenderWare Shader the MatFX effect type should be set to `rpMATFXEFFECTDUAL`.

The diagram below shows how the screen color is created.

For the red part of the pixel, the red source color is multiplied by the given red source blend color. The result is added to the red destination color multiplied by the red destination blend color. The final result of this is the red of the pixel on screen. This occurs for each color.
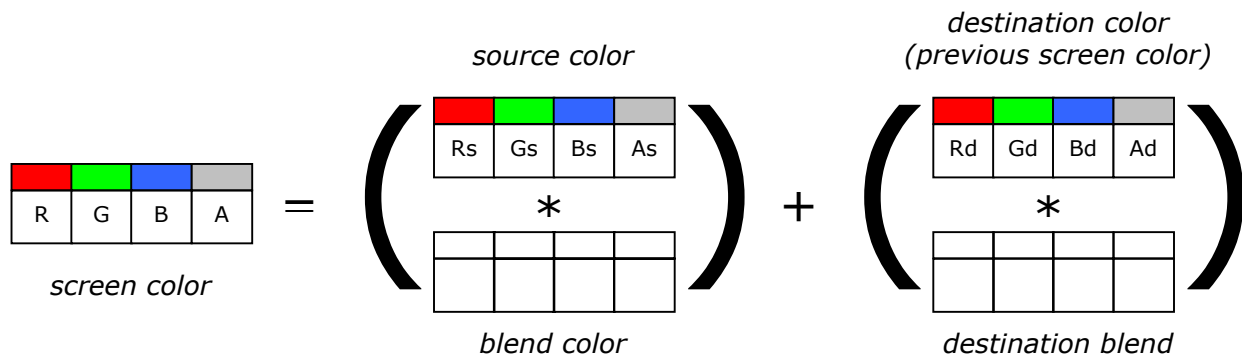
R = red, G = green, B = blue, A = alpha, s = source, d = destination

Substitute the source blend and destination blend with values of the `RwBlendFunctions` used. A description of how each color component is blended in texture blended is listed below:

- `rwBLENDZERO` - RGBA channels set to zero

- `rwBLENDONE` - RGBA channels are set to 1

- `rwBLENDSRCCOLOR` - source RGBA only

- `rwBLENDINVSRCCOLOR` - inverse of source RGBA only

- `rwBLENDSRCALPHA` - source alpha only on all channels

- `rwBLENDINVSRCALPHA` - inverse source alpha only on all channels

- `rwBLENDDESTALPHA` - destination alpha only on all channels

- `rwBLENDINVDESTALPHA` - inverse destination alpha only on all channels

- `rwBLENDDESTCOLOR` - destination RGBA values only

- `rwBLENDINVDESTCOLOR` - inverse destination RGBA only

- `rwBLENDSRCALPHASAT` - source alpha (saturated)

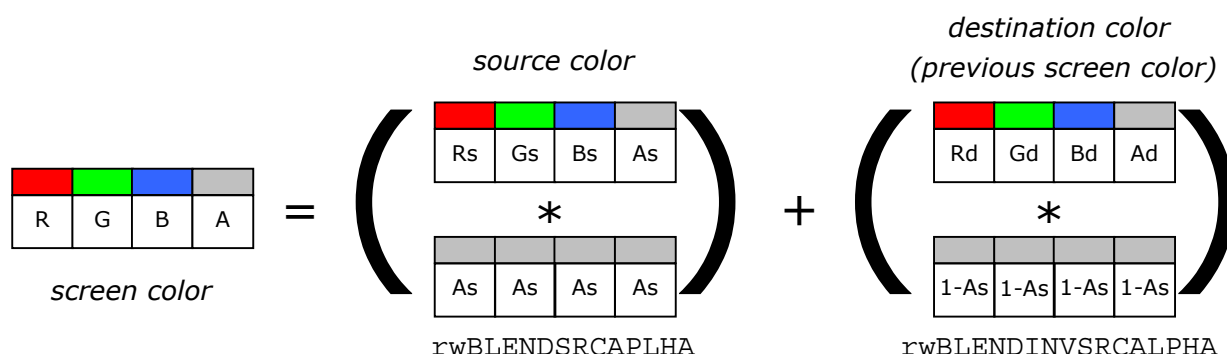These values are only valid for dual pass using MatFX.

With dual-pass texturing, you can consider the first texture to be the destination and the second texture the source (you are blending the second onto the first).

The resulting pixel color is given by the formula:

```
( srcBlend * srcColor + destBlend * destColor )
```

The example below, shows the most commonly used blend modes.

R = red, G = green, B = blue, A = alpha, s = source, d = destination

The above example will be used when rendering alpha pixels, for example, the glass of a glass table. You would also use this blend mode to overlay a second pass texture, such as a logo onto a base texture.

> Not all blend modes are supported on all platforms. The blend modes supported on each platform are listed in the platform specific section of the API Reference. Also note that programmers can also override the blend modes set using RenderWare Shader.
>
> For further information about blend modes in RenderWare Graphics refer to the blending and matfx examples supplied with the SDK.
>
> `RwBlendFunction` values:
>
> `rwBLENDZERO (0, 0, 0, 0)`
>
> `rwBLENDONE (1, 1, 1, 1)`
>
> `rwBLENDSRCCOLOR (Rs, Gs, Bs, As)`
>
> `rwBLENDINVSRCCOLOR (1-Rs, 1-Gs, 1-Bs, 1-As)`
>
> `rwBLENDSRCALPHA (As, As, As, As)`
>
> `rwBLENDINVSRCALPHA ( 1-As, 1-As, 1-As, 1-As)`
>
> `rwBLENDDESTALPHA (Ad, Ad, Ad, Ad)`
>
> `rwBLENDINVDESTALPHA (1-Ad, 1-Ad, 1-Ad, 1-Ad)`
>
> `rwBLENDDESTCOLOR (Rd, Gd, Bd, Ad)`
>
> `rwBLENDINVDESTCOLOR (1-Rd, 1-Gd, 1-Bd, 1-Ad)`
>
> `rwBLENDSRCALPHASAT (f, f, f, 1) f=min (As, 1-Ad)`

Material Color and Vertex Color

Maya materials and RenderWare Shaders are exported as RenderWare Graphics materials. Prelights are exported as vertex colors. These vertex colors are applied at each vertex of the geometry and then interpolated over the face between the vertices. Few vertices will result in minimal color variation and poor quality lighting. If a geometry has both material and prelights then the final color is calculated by multiplying the vertex color by the material color at runtime.

Double-sided Materials

RenderWare Graphics does not support double-sided materials. To create a double-sided object you will need to duplicate the surface with reversed normals.

Extra Vertices from Multiple Materials

Every time the material changes across a model, RenderWare Graphics has to duplicate the vertices along the boundary. This can lead to a large increase in the number of vertices if adjacent materials change often.


Duplicated vertices
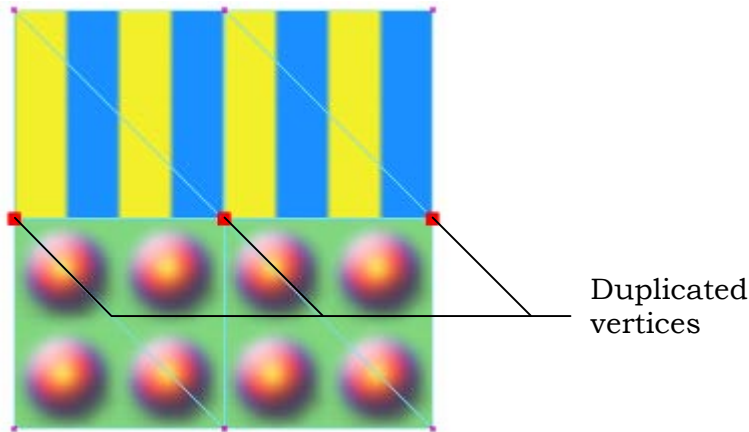
UV Vertices

It is important to think about how UVs are setup in your artwork as Maya stores UVs per vertex per polygon but RenderWare Graphics stores a single UV per vertex. If there are multiple UVs per vertex in Maya artwork the RenderWare Exporter will need to duplicate vertices on export. Duplication of vertices prevents efficient tri-stripping of the artwork. See Texture Repeats on page 44.

The other consideration when assigning UV values in Maya is that your target hardware platform may have limitations on the UV range it can support. If your UVs value exceed this range then rendering artifacts may occur.

A balance needs to be struck between increasing UV values to ensure good tri-stripping and staying within hardware UV limits. The PlayStation 2 exporter has options, which help you to restrict UV values to set ranges and can be set using the *Limit UV* Options in the PS2 Project Template in the RenderWare Exporters.
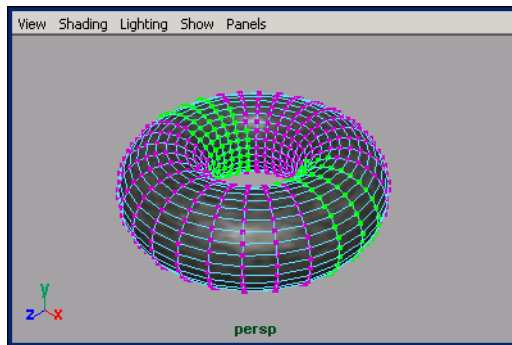
You should use the UV Texture Editor *Window→UV Texture Editor* to check that the UV values are within the ranges you require.
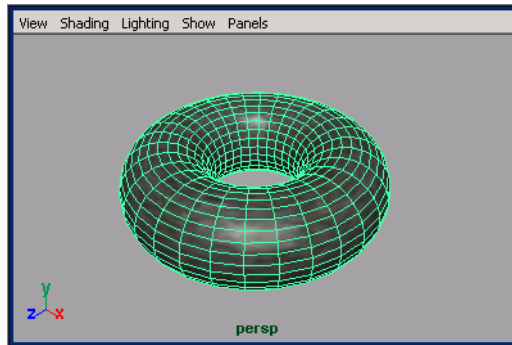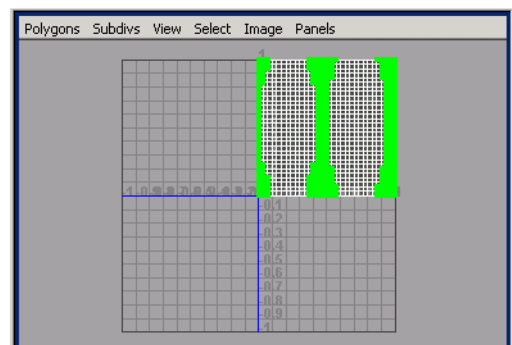
Torus Polygon Primitive Selected
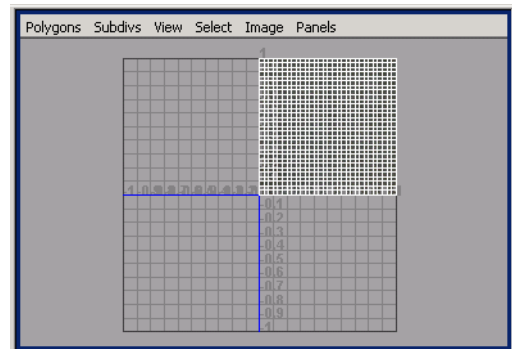
Torus in UV Texture Editor

Torus displaying UVs in purple and selected UVs in green.

Torus in UV Texture Editor with selected UVs in green.

5. General Reference

5.1 Lighting

RenderWare Graphics handles prelights and dynamic lights very differently. Prelights are baked on light effects which are cooked into the model's vertices whereas dynamic lights behave more like real lights. The big difference in RenderWare Graphics is that prelights are fast whereas dynamic lights can be animated. So it's a good strategy to use prelights for stationary parts of your model such as walls and floors, and save the expensive dynamic lights for characters, etc.
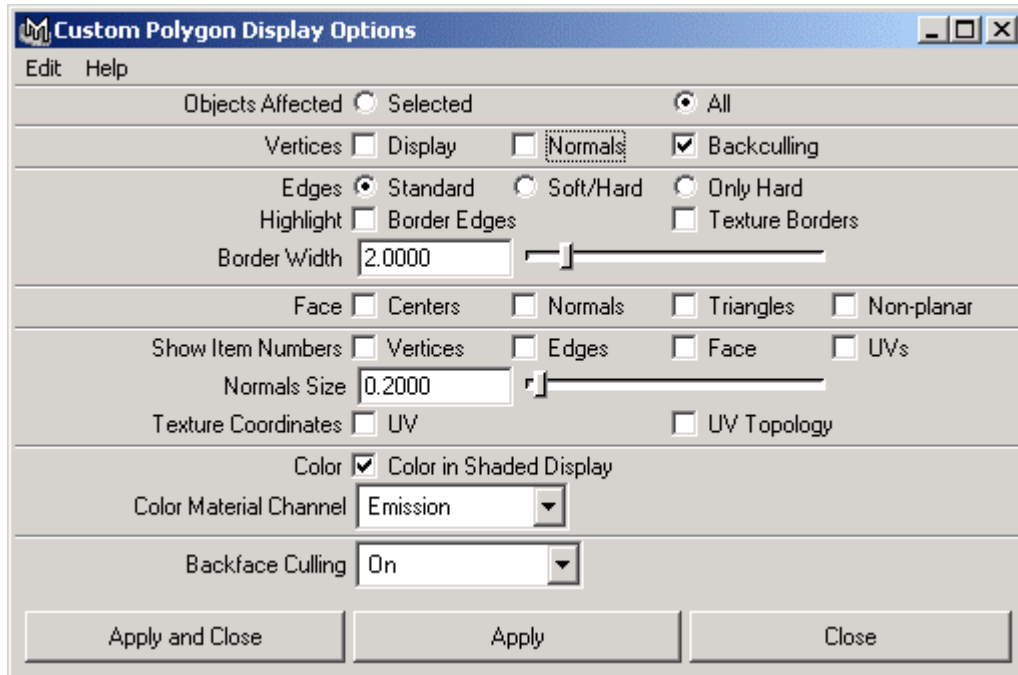
Prelights

RenderWare Graphics geometry prelit using Color Vertex Prelighting

World (static) geometry can be prelit in Maya before it is exported to RenderWare Graphics. This gives the artist control over how the scene will appear in the final application. To create prelights, setup and light your scene as needed. Then use *Edit Polygons→Colors→Prelight* to bake the prelights.

The density of the geometry affects the shading, quality and light fall off. Basically, more triangles equals better shadows. This is an unavoidable overhead so really you need to decide how much you can afford to spend on shading geometry before major modeling work begins. Conversely, the plus side is that you would require many RenderWare Graphics lights to create the same effects, which would require a far greater processing overhead. So all in all, prelighting is a relatively inexpensive solution.

When a scene is prelit in Maya, you need to ensure that you are only viewing the prelight colors. By default, you normally view a combination of prelight color and dynamic lighting.

In the shaded view used, make sure Lighting is set to *Use Default Lighting*. Also in the *Display→Custom Polygon Display* tool make sure *Color* in *Shaded Display* is enabled and *Color Material Channel* is set to *Emission*. You should now be viewing pure prelight colors.

Another issue is that prelight colors in RenderWare Graphics are more similar to prelight intensities. The prelight color is multiplied by the material color and stored in the vertex colors when a geometry is instanced. Any dynamic lighting intensity multiplied by the material color is then added during rendering.

In the case of a prelight example with a mid gray material color, the prelight colors get halved in intensity before being added to the dynamic lighting values. If you change the material color to white then you should get results closer to what you expect. Where you have a file texture the material color is set by the Color Gain attribute which likewise should be set to white.
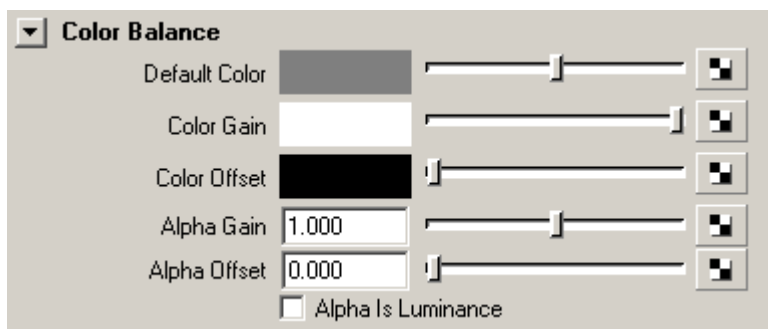
No negative prelights: Prelights cannot be negative in RenderWare Graphics.

Duplicate Vertices

When vertex pre-lighting is applied, *Static World* assets generate duplicate vertices at the same location. This is because two or more differently colored lights flooding adjacent triangles with their respective colors with no overlap cause duplicate vertices to be created.

For example, for a cube, the different faces would be lit with different colors causing vertices to be duplicated as there is lack of continuity. To resolve this, try making all lights the same color, i.e. all black or all white.

Dynamic Lights

Any lights in the export hierarchy will be exported along with any geometry. Ambient, directional, point and spot lights are supported and RenderWare Graphics supports color, intensity, decay rate and spot angle for these light types.

Ambient is the cheapest light to use then in order of cost are: directional, positional, and spot, each of which is roughly twice as expensive as the one before with a soft spot possibly more expensive than a spot on some platforms. It is recommended that excessive numbers of lights are not used as performance will be affected.

LightMaps

Lightmaps store information about the illumination of a surface. Each surface in a scene can have its own lightmap. Lightmaps work by taking a texture and a lightmap and combining them at render time to create a lit texture.

|		
texture	*lightmap*	*lightmapped texture*
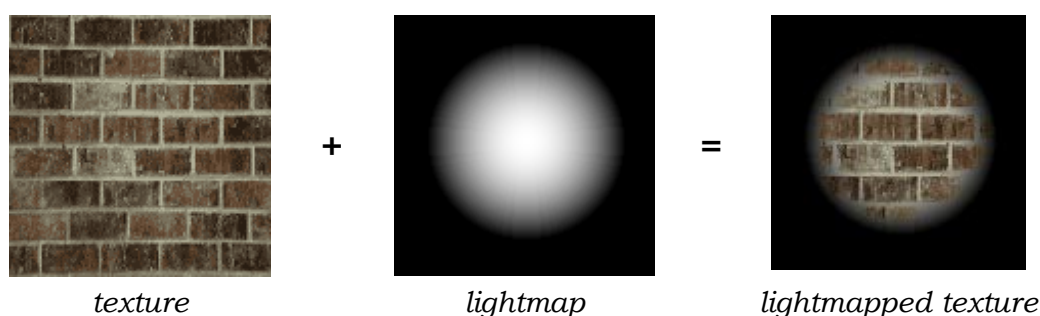
Lightmaps can be created and exported in two main ways. These are:

- Enabling the *Process Lightmaps* option in your asset template to *Export Native Lightmaps*. The exporter will then look for dual pass textures it can export as lightmaps. You can create these by using the RenderWare Shader dual pass settings and enabling the Lightmap flag in the lightmap setting section. You can also do this by using a standard material with a *layeredTexture* node attached to the color channel and enabling the the *Set Light Mapped* flag on the *layeredTexture* in the *Object Settings* dialog. The lightmap Uvs will be re-sewed for better performance at runtime and the lightmap textures will be re-sampled for use by the RenderWare pipeline.

- Exporting light mapping creation data (axis info) for the generation of light maps externally, using the RtLtMap toolkit. You would use standard single texture materials within Maya and lightmaps would be added after export. In this case the *Process Lightmaps* option in your asset template should be set to *Generate RtLtMap Uvs.*

Manually setting up dual pass materials for lightmapping

Please refer to sections 4.1 and 4.2 for details on setting up dual pass materials. For example a lightmap material could be created using these dual pass settings in a RenderWare Material.
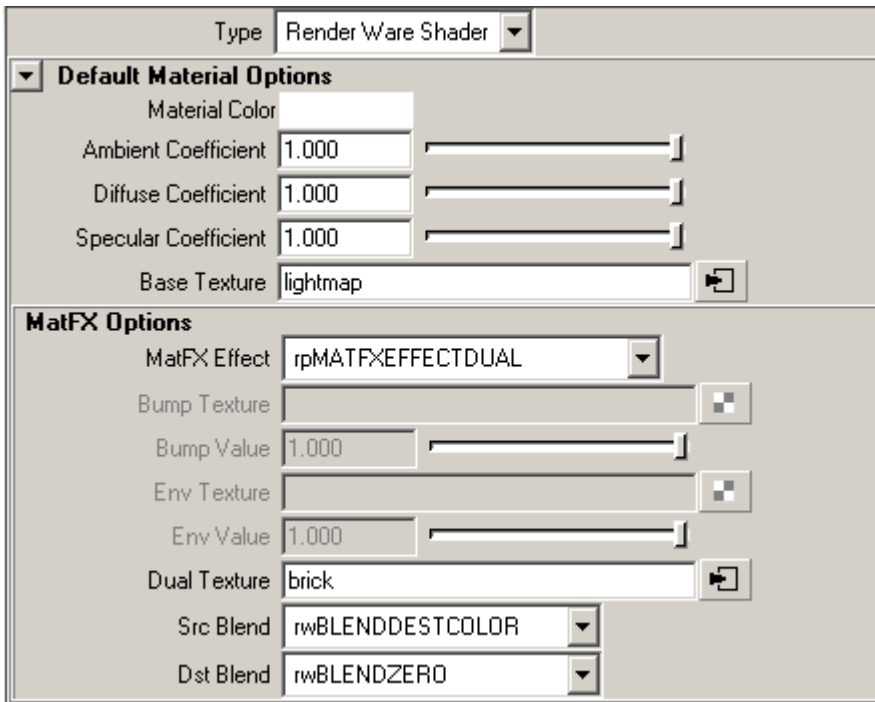

Using the Bonus Game pack to generate lightmaps

The Bonus Game pack for Maya provides a script for generating lightmaps. When run, this script will automatically generate lightmaps and add them to your shaders in a layered texture node. In order to export the generated texture to a RenderWare lightmap material set the *Process Lightmaps* option in the Asset Template to *Export Native Lightmaps*, and ensure that the *Set Light Mapped* flag is set in the *Object Settings* on the *RenderWare* menu for the layered texture node. The light map object flag can be set automatically by adding the following to the addToLayeredTx script of the Bonus Game pack.

```
if (!`attributeQuery -exists -node $layeredTx
"RwLightMapFlag"`)
{
addAttr -ln "RwLightMapFlag" -sn "LF" -at "long"
$layeredTx;
}
setAttr ($layeredTx + ".RwLightMapFlag") 1;
```

The "Generate Light Maps" script works differently between the Bonus Game pack with Maya 4.5 and the Bonus Tools pack with Maya 5.0. In Maya 4.5, when the script encounters a shader with no texture applied to the color channel it will simply connect the lightmap to this channel. To export these as lightmaps you will need to use the "Make Texture Layered" script to convert to a layered texture node and set the Light Mapped flag in the RenderWare Object Settings. In Maya 5.0 a layered texture node will be generated and you can use the suggested modification to addToLayeredTx to automatically set the light mapping flag. However, in Maya 5.0, when the script encounters a shader with a texture on the color channel the lightmap texture is incorrectly placed as the second connection to the layered texture node. To visualize the light map in the viewport and export the light mapped texture, reverse the order in the layered texture node attributes.
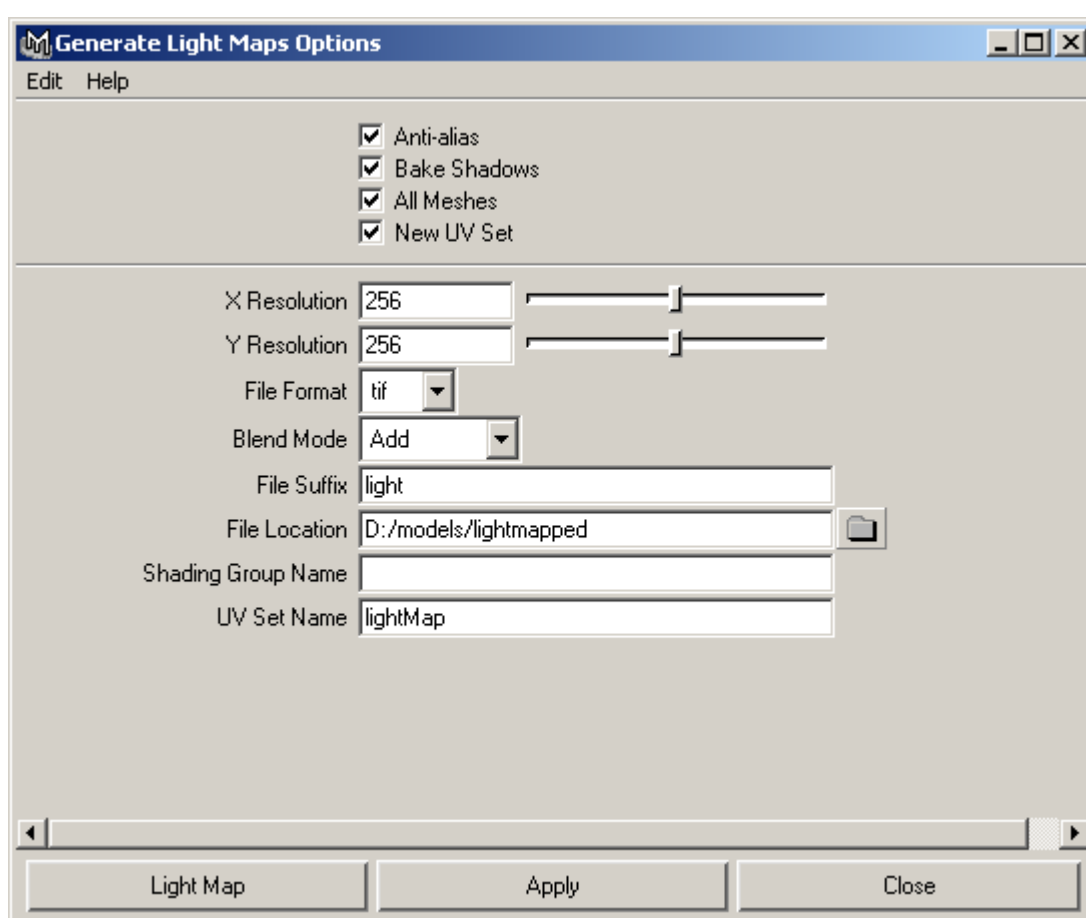
- The Bonus Game script package can be downloaded from Maya Support section of the Alias Wavefront website.

- Due to the way this script works it is not possible to create a layered lightmap texture by using the default Lambert material, a new material must be created first. RenderWare Shaders are also not supported, though the script can be modified to support them.

Generating lightmaps externally

Lightmap settings for generating lightmapping info for creating lightmaps externally for use with RenderWare Graphics are explained in the Light Maps section. The *Process Lightmaps* option in the Asset Template should be set to *Generate RtLtMap UVs* for this type of lightmap support.

Platform specific settings

When exporting lightmaps for use on Playstation 2 it is vital that you use a project template that specifically targets the platform (such as the PS2.rwt that ships with RenderWare). This is because RenderWare lightmapping on this platform works quite differently from the other platforms, it actually uses darkmaps rather than lightmaps. The Playstation 2 template therefore sets options that generate darkmaps rather than lightmaps. If you view lightmap assets that have been exported for Playstation 2 on other platforms they will look incorrect.

Normal Maps (Xbox and D3D9 specific)

Bump mapping describes how the normal vector is perturbed across the surface so that the differing direction of reflection will result in lighting changes. Bump mapping works by creating embossed effects. Using bump maps you are unable to derive true diffuse/specular/reflections from the bumps. Bump mapping can be used to calculate normal maps.

A normal map is similar to a bump map but instead of containing color values (as RGB) it contains normal values (as xyz). This results in better lighting calculations and better results than bump map models. Normal maps can render real bump maps using DOTPRODUCT3.
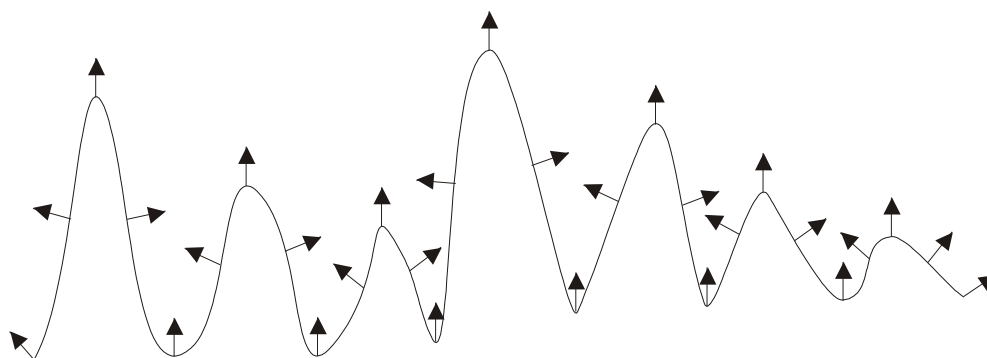
normals - gray scaled bump map

You are able to provide a bump grayscale alpha texture and it will be converted to a normal map by treating each pixel in the picture as a vertex of a triangle. The values are calculated using a scale factor based on relative values between the four surrounding pixels.

The normal map is then normalized using the range [0..1].

Gray scale bump mapped texture

Normal mapped texture

Using normal maps can reduce the number of polygons required to make a model look more realistic.

Setting up normal maps for the RenderWare Graphics exporters is explained in the Normal Map section.

More information about DOTPRODUCT3 and information about the Adobe® PhotoShop® Normal Map Plugin which creates normal maps from height maps for per pixel rendering can be found:
http://developer.nvidia.com/view.asp?IO=Simple_dot3_bm

5.2   Camera

RenderWare Graphics supports the exporting of cameras. There are no camera specific export options in the user interface.

Any cameras in the export hierarchy will be added to the clump. Both perspective and orthographic cameras are supported. The camera parameters exported by RenderWare Graphics are matrix, projection type and clipping planes.

Background color

The background color of a scene can be exported depending on the `Background Color` setting in the `rws` options. The default setting is off, but it is enabled when a scene is viewed rather than exported. The color is taken from the relevant camera's background color. The camera chosen is the first visible camera in the scene. If there are no visible cameras then the currently selected viewpoint camera is used.

5.3   Animation

Smooth Skinning

By binding a skin to Maya's joints it is possible to associate the vertices from the model with the joints. The smooth or soft skinning means that each vertex is tied to multiple joints. This gives a much more natural, organic result than if these vertices were tied to just one joint, like a robot arm would be. RenderWare Graphics supports four joint weights per vertex. If more than four weights are detected at a vertex then the highest four will be used and the weight total normalized to one. If less than four weights are detected the remaining weights will be exported as zero. The exporter will issue a warning in the Output Window when this occurs as the final animation in RenderWare Graphics may not match what you see in Maya. When performing the smooth skin bind operation in Maya we recommend you change the Max Influences option to four. This will ensure the skinning results in Maya match those in RenderWare Graphics as closely as possible.

The PlayStation 2 RenderWare Graphics implementation has a limit of 64 bones in the skeleton hierarchy. On Xbox the limit varies depending on the exact rendering settings used. Other platforms have a limit of 256 bones.

Rigid skinning is not supported by RenderWare Graphics and a warning will be issued if it is detected.

> The bone limit can also be found in the Project Template tab of RenderWare Graphics' Exporter's Advanced Settings.
>
> Alternatively, please refer to the *Skinning* Chapter in the RenderWare Graphics User Guide or the Xbox RenderWare Graphics API Reference.
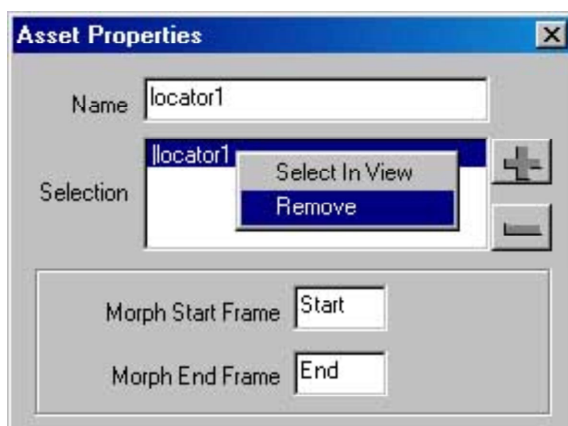
Sub-Hierarchy Animations

Sub-hierarchies can be exported using the Sub-Hierarchy Animation option in the Animation assets' dialog.

By default, an animated hierarchy will be created from the root. To export a sub hierarchy, the selection needs to be changed.

For example, an animation contains a character running with his head moving from side to side. You may want to export just the legs separately as a sub-hierarchy. This can be done as follows:

1. Create an asset for the sub-hierarchy animation. By default, the animated hierarchy will be created from the root, regardless of the selection.

2. Double-click the newly created asset and amend the animation selection by right-clicking the selection and choosing remove (or clicking on the selection and then clicking):

3. In Hypergraph (*Window→Hypergraph*), select the node containing the sub hierarchy and click on in the Animated Hierarchy Asset Properties dialog.
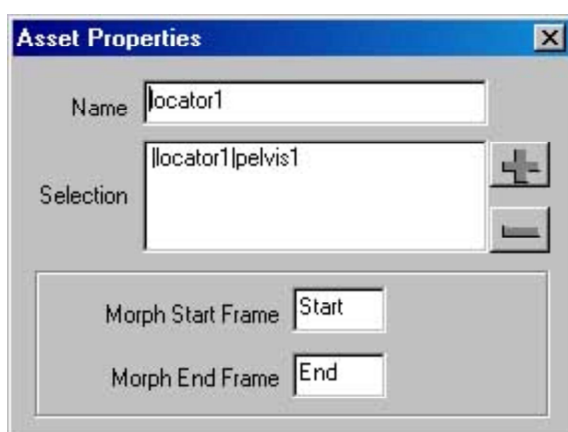

✎ The selection above will not have any affect on the exported hierarchy unless the sub-hierarchy animation option is selected in the Animation Asset properties dialog.

It is recommended that a node tag is assigned to the sub-hierarchy, so that it can be identified post-export.

4. Keeping the node selected in the scene view, assign a node tag to the sub-hierarchy. To assign a node tag, *RenderWare→Object Settings*, check the Set Node Tag option and create a tag.
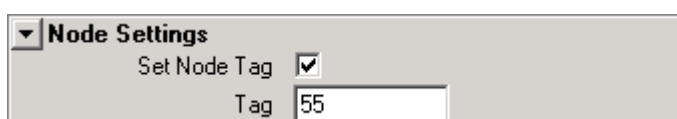
5. Make a note of the tag used. The programmer will need to know about this tag so that the tag can be accessed through code.

6. In the Animation assets' properties, check Sub-Hierarchy Animation.
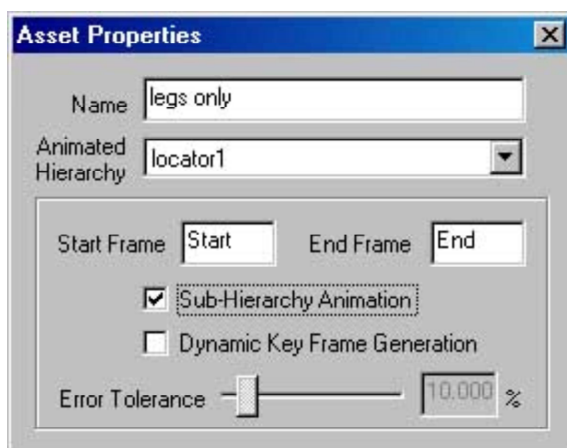
7. Disable the Animated Hierarchy asset containing the sub hierarchy node by right clicking the asset and choosing *Disable*. This means that only the sub-hierarchy animation file (.anm) will be exported:
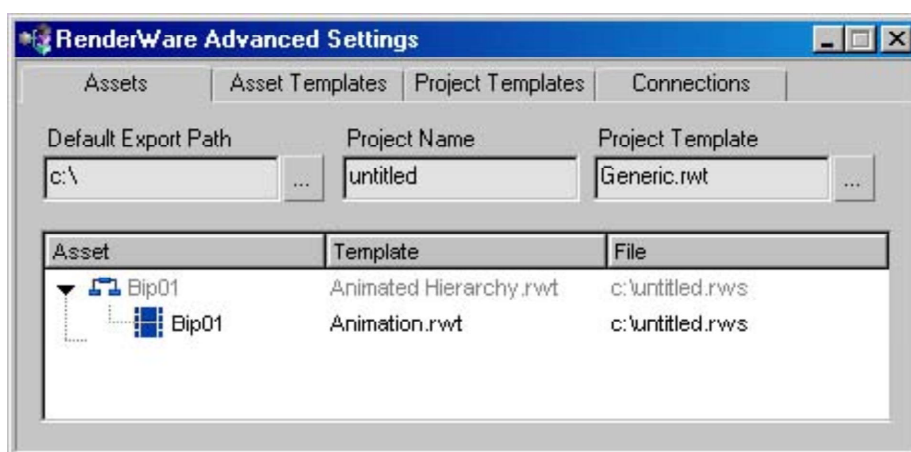
8. Export the sub-hierarchy animation in the same way that you would export any animation (*RenderWare -> Export...*).

✎ The Visualizer can not view sub-hierarchy animations. If you need to view the effects of your sub-hierarchy animation, see your programmers for a suitable tool.

Animating the Root Transformation

When exporting a hierarchy, with an animated root node, the animation for the root is generated relative to the root's initial frame. If you create multiple animations, each one with a different initial frame, mixing and matching them on the same hierarchy might to lead to inconsistent world space jums. To avoid this problem, you can create an additional dummy transformation at the top of your hierarchy, causing it to be your new root, and allowing the generation of absolute animation data on your old root.

Morph Target Animation

When the *Export RpMorph Targets* option is enabled the RenderWare Graphics exporter samples the geometry's vertex and normal positions at user specified intervals and later RenderWare Graphics interpolates between them. The greater the number of samples the better the fidelity but the bigger the file sizes. One issue, however, is that interpolation between targets is linear. Because of this you may need to create more targets to keep object distortion to a minimum.

There are two ways in which the RenderWare Exporter can decide what times to sample the geometry data. By default the exporter will look for any animation curves related to the geometry in the dependency graph. It will combine the keyframe times from all such animation curves and sample at these times. If the *RpMorph Sample* option is set then the exporter will instead sample the geometry at a set frame interval. This interval is controlled by the Sample Interval option in the exporter.

Note that morphing is relatively more expensive (i.e. slower) to process than skinning. Because of this it makes sense to divide your model into morphing and non-morphing parts if possible and then animate only the parts that change.

RpDMorph Animation

Delta Morph Animation is a style of animation which closely matches *Blend Shapes* in Maya and is enabled by the *Export RpDMorph* option.

Rather than sampling geometry at specific times the exporter looks for Maya Blend Shapes attached to geometry and creates a RenderWare Graphics delta morph target for each. Each target is simply a per vertex list of position and normal deltas from the original positions. Any animation curves attached to the Blend Shape weights are also processed by the exporter.

At runtime the position and normal deltas for each target are multiplied by the weight factor for the target and added to the original geometry. The final result is a blend of the original geometry and any delta morph targets with non-zero weight. This type is animation is particularly useful for facial animation. Since the target weights can be modified at runtime it also allows dynamic effects (such as a characters expression changing according to gameplay).

There are no limitations on the number of RpDMorph targets that can be used in RenderWare Graphics. The runtime performance is affected by the number of targets than have non-zero weights at any one time though.

Animation Curves

RenderWare Graphics only supports linear interpolation between animated keys. So it's best to use *linear tangents* in Maya, so that your animation will export exactly to RenderWare Graphics.

The only way to create ease-ins, ease-outs, etc. is by adding keyframes. You can do this manually or you can use the exporter's *Dynamic Key Frame Generation* option in the Animation Asset Properties. This automatically adds extra keys to reproduce the spline motion, although this can add a lot of extra keys.
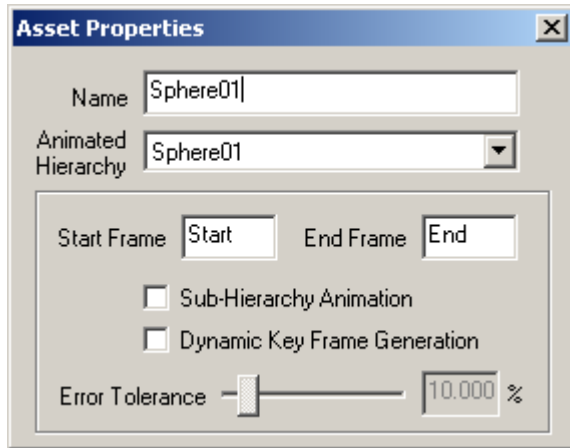
Rotation Keyframes

Animating 360˚ rotations with a single keyframe can cause problems due to the math involved. A better solution is to use multiple keys each with a smaller rotation. For example, to rotate an object 360˚ over 30 frames place one 180˚ key at frame 15 and a second key with a 360˚ value at frame 30, rather than a single key at frame 30.

Animation Constraints

The exporter looks for constraints applied to the objects being exported and follows through the constraint connections. The keyframe times of the animation curves on the constraining objects are applied to the constrained object. This ensures that motion introduced by the constraints are not lost during export. The constraint nodes supported are *aimConstraint, geometryConstraint, normalConstraint, orientConstraint, pointConstraint, poleVectorConstraint, scaleConstraint* and *tangentConstraint.*
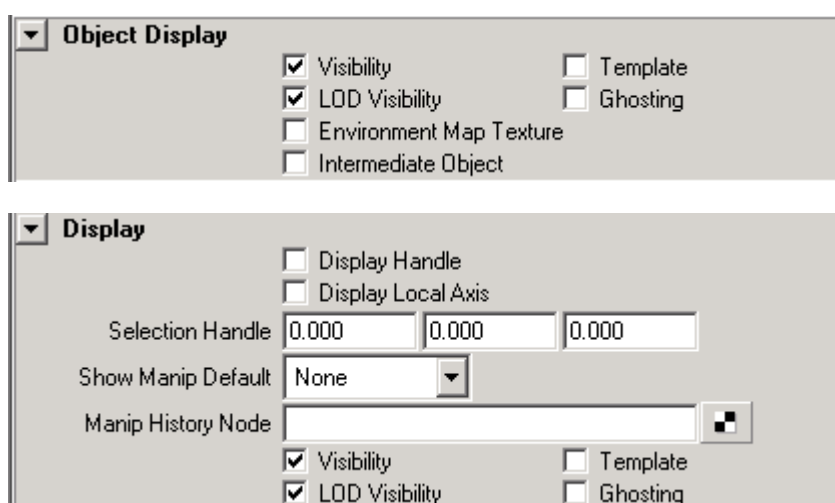
5.4 Visibility

Nearly every Maya node has a attribute called "visibility". If this is off/false/no then the node will not export, and nor will it's children.

For transform nodes this can be changed in the "Channel Box". There is a text field for "visibility", check this is "on".
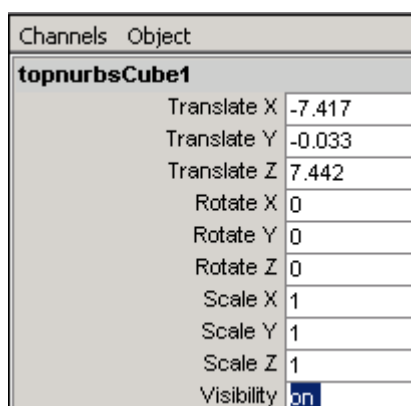
For most nodes the Attribute Editor includes rollout "Object Display" (or "Display" on transform nodes) check that the "visibility" checkbox is checked.

5.5   Layers

In the same way that nodes which are hidden using the visibility flag are not exported, nodes that are members of hidden display layers are not exported. Render layers have no effect on exports. You can add objects to layers and edit layer visibility using the Layers Editor dialog in Maya.
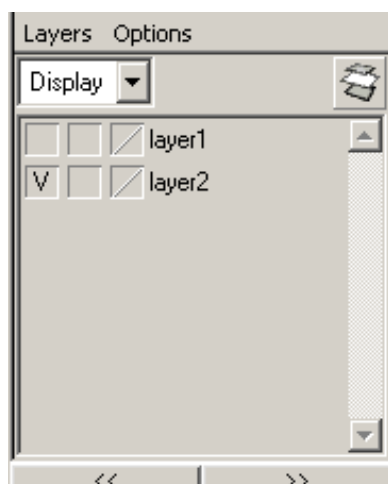
5.6   Blind Data

Blind data is typically used to tag areas of scenes with user defined attributes.

This is explained in more detail in the TechnicalArtistGuide.pdf.

6.   Exporting

The options available in the export menu are fully explained in
TechnicalArtistGuide.pdf. This document explains the basics to take your
assets and export them to RenderWare Graphics formats.

The exporters, by default, export to `.rws` files. This is the RenderWare
Graphics file format for exporting any type of RenderWare Graphics data.
RenderWare Visualizer can be used to view `.rws` files.

The file formats `.anm`, `.dff`, `.bsp`, `.spl` are now considered legacy. The binary
format of these files will continue to be supported, and RenderWare Graphics
will continue to read them. There is no need to re-export existing artwork to
`.rws` file format. However, it is recommended that you export to `.rws` files as
these file formats *may* be removed in future releases.

File Formats

`.rws` files contain RenderWare Graphics stream information and may include animated
hierarchy, static world, animation and spline information as well as embedded
textures.`.rws` files can contain the contents of multiple `.dff`, `.bsp` files etc.

The file formats `.anm`, `.dff`, `.bsp`, `.spl` are now considered legacy and may be removed in
future releases. It is therefore recommended that you export to `.rws` files.

`.dff` files contain Animated Hierarchy (`RpClump`) information.

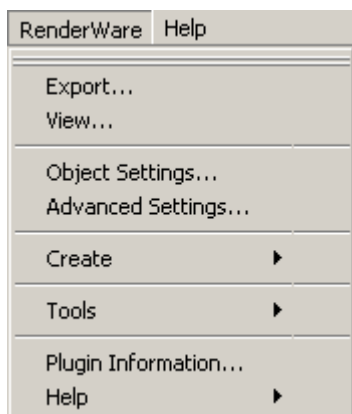`.bsp` files contain Static World (`RpWorld`) information.

`.anm` files contain Animation (`RtAnimAnimation`) information.

`.spl` files contain Spline (`RpSpline`) information.

6.1   RenderWare Menu

The RenderWare Menu has the following options. These are described below.

Export

The Export option in the RenderWare menu uses the Export Wizard to automatically create assets and export them to a `.rws` file.

The first time you perform an export of a scene the wizard will appear and ask for the following information:

1. State your target directory for your exported files
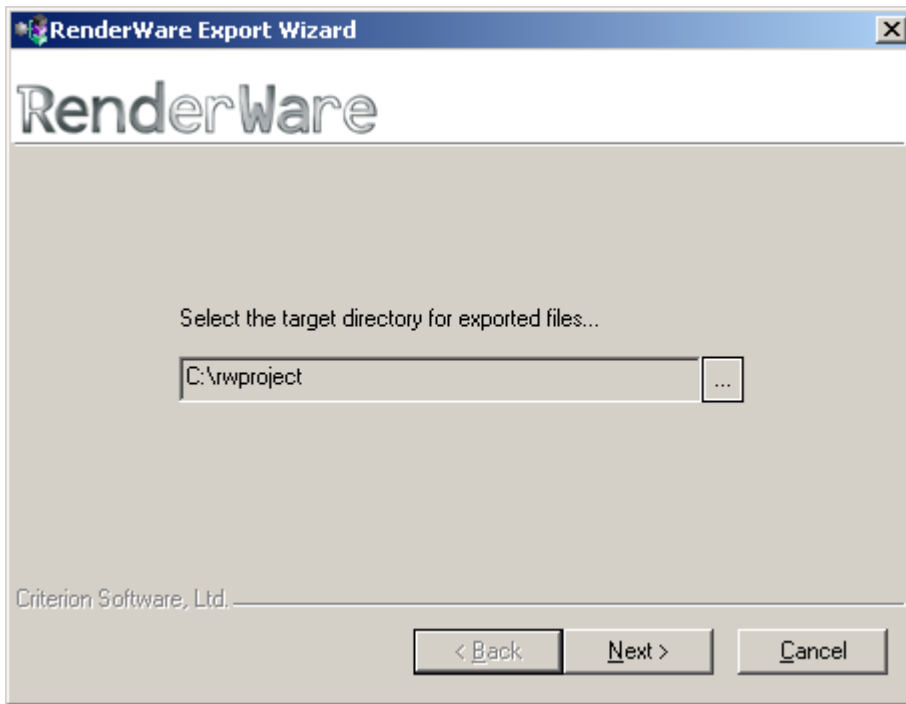
2. Select the project template required. There are four default project templates which relate to the four supported platforms:

    – `Generic.rwt` used for exporting files to the PC (e.g. D3D8 or OpenGL)

    – `GameCube.rwt` used for exporting files to GameCube

    – `PS2.rwt` used for exporting files to PlayStation 2

    – `Xbox.rwt` used for exporting files to Xbox
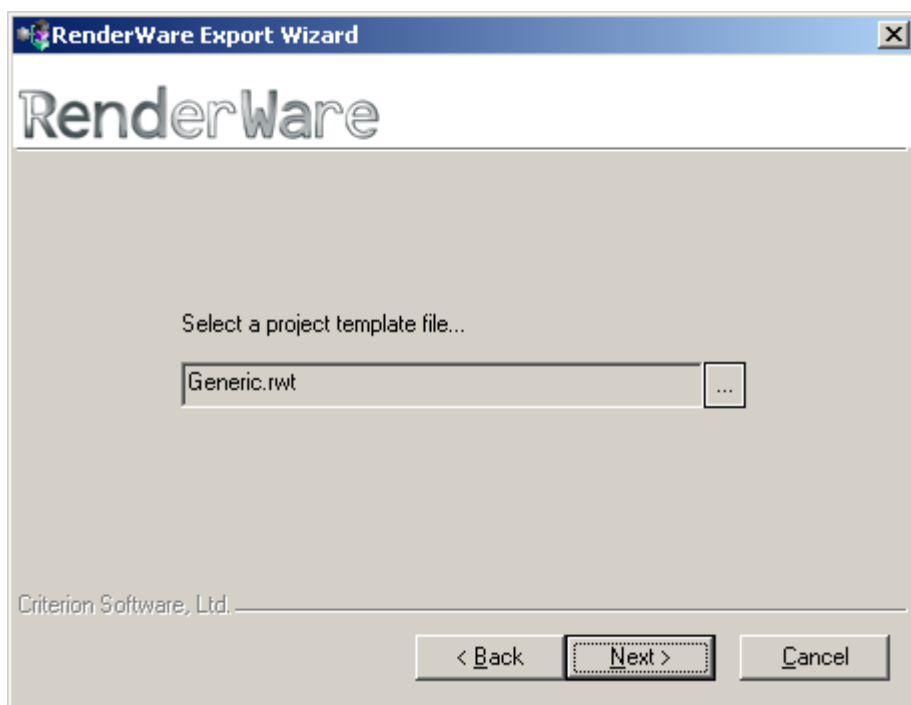
3. The export summary displays the export directory and project template used. Selecting "Use as default" will ensure that these settings are stored as the default.

Your scene is exported using the settings in the project template file you selected. For the default templates the output file type will be `.rws`. The contents of the `.rws` (your scene) are displayed in the Output Window.
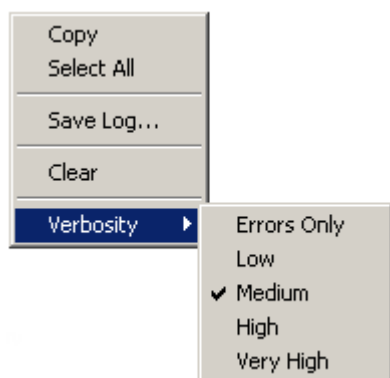
View

The View option in the RenderWare menu will export the assets in your scene and display them in RenderWare Visualizer. The view operation will override the output settings in your project template and always export a temporary `.rws` file with textures. The contents of the `.rws` (your scene) are displayed in the Output Window.

Output Window

When a scene is exported or viewed the output is recorded in the Output Window. The output window contains useful information about the type of asset exported including animation settings, geometry stored, tri-stripping, frames, lighting. The output window is useful for debugging and comparisons.

```
RenderWare Output                                          _ □ ×

(i) Building scene graph...
(i)
(i) Exporting assets for platform Generic
(i)
(i) Exporting asset: locator1
(i) Starting Animated Hierarchy export
(i) Assigned Guid :1cd6232d-cb79-457d-9d5b-548f42517873
(i) Initializing animated objects
(i) Precomputing polygon limits
(i) Building optimized hierarchy
(i) Merging meshes
(i) Applying export handlers
(i) locator1: Exporting RwFrame
(i) joint1: Exporting RwFrame
(i) joint2: Exporting RwFrame
(i) joint3: Exporting RwFrame
(i) polySurfaceShape2: Exporting skinned geometry
(i) polySurfaceShape2: Tristrip Quality: 78% (tris/vertices)
(i) polySurfaceShape2: Exporting RpDMorph data
(i) polySurfaceShape2: Processing Target #0 = "polySurface1"
(i) Checking Nodes
(i) Clump has 5 frames, 0 lights, 0 cameras, 1 geometries, 0 patches, 1 skinned and 0 un-skinned atomics.
(i) Finished Animated Hierarchy export
(i)
(i) Exporting asset: locator1
(i) Starting Animation export
(i) Initializing animated objects
(i) Computing scene bounding boxes
(i) Applying export handlers
(i) Extracting key frames
(i) Sampling animation data
(i) Building optimized animation object
(i) Checking Nodes
(i) Assigned Guid :930085e6-6fff-43c9-81c9-9dcab4d6af74
(i) Finished Animation Export
(i)
(i) Streaming assets
(i) Streaming 3 assets to c:\rwproject\DMorphAndSkin.rws
```

There are several options available from right clicking the output window:

| Copy |
| Select All |
| Save Log... |
| Clear |
| Verbosity ▶ |

| Errors Only |
| Low |
| ✔ Medium |
| High |
| Very High |

Copy: copies the selected text to the clipboard

Select All: selects all the text in the output window

Save Log: saves the content of the output window to a log file
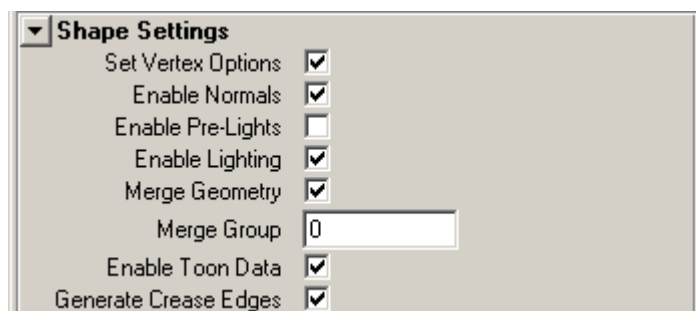
Clear: clears the output window

Verbosity: changing the verbosity dynamically changes the contents of the output window. Higher levels contain more detailed information useful to advanced users.

6.2   Object Settings

Contains a list of object settings for a specific shape, file texture, layered texture and node settings.
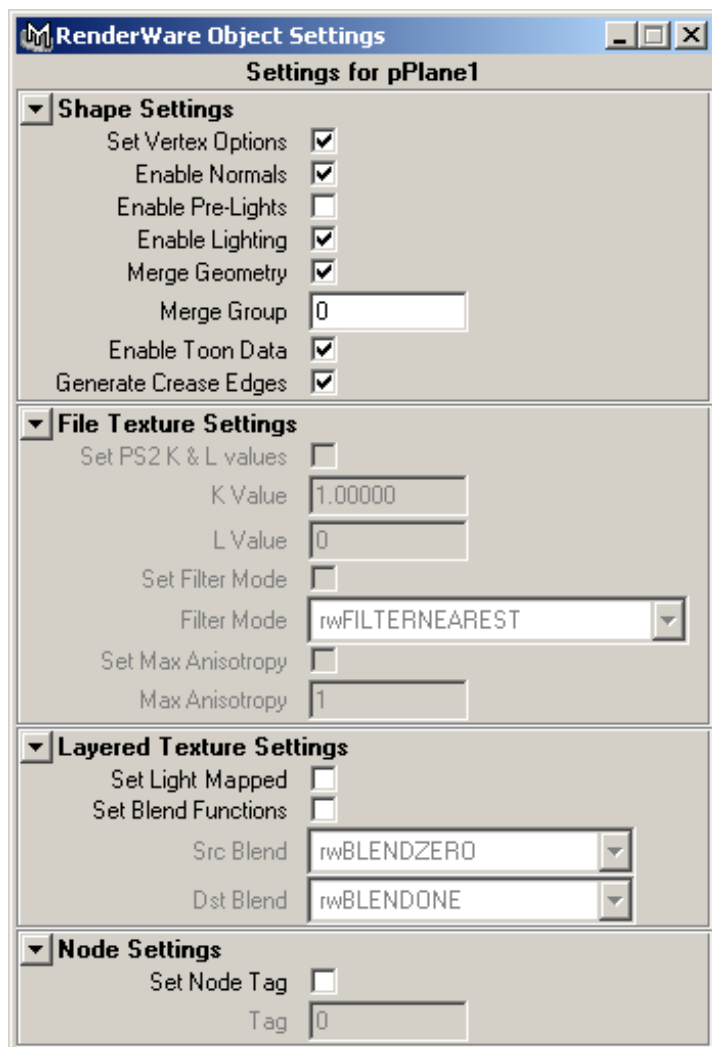
Shape Tags

These options are only available if the currently selected node is of type transform or shape.

Set Vertex Options: The option enables and disables *Export Normals* and *Export Pre-Lights*.

Enable Normals: This option controls whether normals will be extracted from this object when exporting. The Vertex Normals template option must be enabled for this option to have any effect on the final export data. If this option is disabled and the template option is enabled default normals will be stored in the exported data.

Enable Pre-Lights: This option controls whether pre-light colors will be extracted from this object when exporting. The Vertex Prelights template option must be enabled for this option to have any effect on the final export data. If this option is disabled and the template option is enabled default prelight colors will be stored in the exported data.

Enable Lighting: Controls whether the shape will be lit or not.

Merge Geometry: This option controls whether any geometry created from this node should be merged with other geometry in the scene.

Merge Group: This option is grayed out unless the *Merge Geometry* option is selected. The value is an integer describing the Merge this node is a member of. The geometry from all nodes with the same Merge Group will be combined.

For two nodes to be merged they must be in the same Merge Group, have the same lighting flag, both be skinned or not, and both be textured or not.

Enable Toon Data: This option controls whether this object will be toon shaded. The Export Toon Data template option must be enabled for this option to have any effect on the final export data. The option only affects assets exported with the Animated Hierarchy template.

Generate Crease Edges: If this option is not checked and you export a cube, only the silhouette edges are rendered. If this option is checked, each edge that lies between polygons with different smoothing groups is drawn as a crease edge (you must also export vertex normals for the object for this work). The Generate Crease Edges template option must be enabled for this option to have any effect on the final export data and the option only affects assets exported with the Animated Hierarchy template.

File Texture Settings
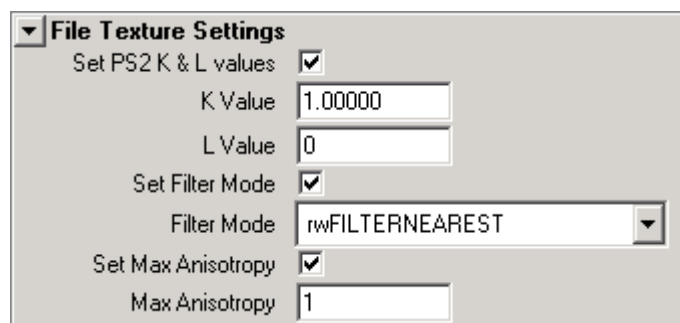
These options are only available if the currently selected node is of type *file*.

Set PS2 K & L values: This option controls whether any RenderWare Graphics textures created from the selected file node will have specific PlayStation 2 K & L values set. Otherwise they will default to standard values. K & L values control the biasing of Mipmap levels on PlayStation 2. You should consult the programmers on your project whether specific K & L values need to be set on your textures.

K Value: This option is grayed out unless the *Set PS2 K & L values* option is set. The floating point value is set in the *K Value* field.

L Value: This option is grayed out unless the *Set PS2 K & L values* option is set. The integer value is set in the *L Value* field and should be between 0 and 3.

Set Filter Mode: When this option is enabled the texture filter mode of the currently selected file node is set based in the *Filter Mode* option. This setting takes precedence over the Maya filter mode.

Filter Mode: This option is grayed out unless the *Set Filter Mode* option is enabled. The drop down box allows you to specify an explicit RenderWare Graphics filter mode. The filter mode is set in RenderWare Graphics by calling `RwTextureSetFilterMode()`. These options in Texture Filter Modes on page 46.

Set Max Anisotropy: When this option is enabled the texture file node currently selected will have anisotropic filtering enabled when exported to RenderWare Graphics. Anisotropic filtering is not supported on all hardware platforms. You should consult the programmers on your project to see if Anisotropy is supported on your platform. The mipmap function is derived from the filter mode.

Max Anisotropy: This option is grayed out unless the *Enable Anisotropic Filtering* option is enabled. The value set here controls the maximum anisotropy of the exported texture. Higher values give better quality but slower performance. You should consult the programmers on your project to see if Anisotropy is supported on your platform. Values should be in the range 1-4.

Layered Texture Settings
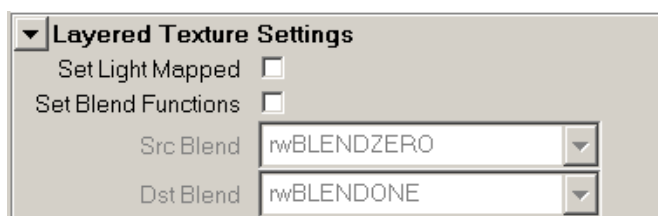
These options are only available if the currently selected node is of type *layeredTexture*.

Set Light Mapped: When this option is enabled the selected *layeredTexture* is set as a light mapped material. The *Set Light Mapped* and *Set Blend Functions* options are mutually exclusive.

Set Blend Functions: When this option is enabled the selected *layeredTexture* node will have *src* and *dst* blend values based on the values set here. The blend values set within the *layeredTexture* node are never used, therefore this control should always be used to define blending.

Src Blend: The source blend value to use.

Dst Blend: The destination blend value to use.

Src Blend and Dst Blend are used in the function `RpMatFXMaterialSetDualBlendModes()` in the RenderWare Graphics API.

Node Settings
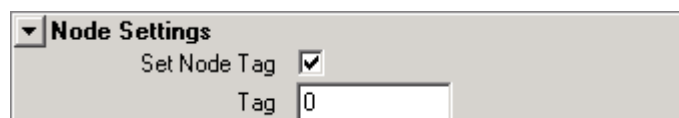
Tagging is a process by which you assign a unique number to certain nodes in the object hierarchy you wish to export. Tagging information is only exported when using the *Animated Hierarchy* template exporter. A typical use would be to tag the hand joint of a character. The tag number assigned to the bone would then be passed onto your programmers so that they can find the tagged node during the game and dynamically attach objects to it.

You should ensure that the tag numbers you assign to nodes within a single hierarchy are all unique. Any nodes that you do not assign tags to will be given an automatically generated unique tag number.

6.3   Advanced Settings

Advanced Settings are settings which enable the user to customize assets, asset templates, Visualizer settings and platform specific settings.

Advanced Settings are accessed through *RenderWare→Advanced Settings*.

The TechnicalArtistGuide.pdf contains a full explanation of the Asset Templates and Project Templates.

The Advanced Settings dialog has four tabs:

- Assets (page 76) - the assets of a scene, for example, a skinned object

- Asset Templates (page 84) - the settings for an asset

- Project Templates (page 86) - the project settings used for exporting assets to different platforms. Supported platforms are: PC; GameCube; PlayStation 2 and Xbox

- Connections (page 87) - the connection settings for RenderWare Visualizer

Assets

Assets are RenderWare Graphics nodes stored in your scene that represent a single item of export data. They link a part of your scene with the settings to use on export. For example, a game may contain a skinned animated character. Two assets are created, an asset for the character and a child asset containing the animation data.

There are two modes of operation for asset generation. When the Customize option is disabled the exporter analyses your scene and generates assets automatically for you. As objects are added or deleted in your scene the assets are automatically updated. When the Customize option is enabled you have manual control over how assets are created. In a new scene the default mode is non-customized assets.

Using the Export Wizard (*RenderWare→Export* or *RenderWare→View*) the assets can be exported and viewed quickly. All assets present in the scene are exported. Using the Asset dialog of the advanced settings window you can export and view assets individually.

All assets use an asset template. There are four default asset templates and these read-only templates contain a specific set of options and settings.

✎
> Assets were referred to as batch nodes in RenderWare Graphics release 3.3 and 3.4.

Asset Dialog

The asset dialog contains settings which are important for customizing existing assets and creating your own assets.

the name of the project and the default file name exported to

the project template used for exporting to different platforms (PC, GameCube, PlayStation 2 and Xbox)

the default directory files will be exported to

the name of the assets

the Asset Templates used

preview of the file exported for each asset

The above example shows automatically generated assets. Right-clicking over an asset enables you to export or view the selected asset. Right clicking below the assets enables you to be able to export and view all assets.

The Assets dialog contains assets which relate to objects in your scene. There are four types of default asset:

- Animation

- Animated Hierarchy

- Static World

- Spline

Customizing Assets

The default behaviour for the asset manager on new scenes is a mode in which assets are automatically created for you as you model your scene. Whilst this is very convenient as you start to model a scene or get to know the RenderWare tools it is likely you will eventually need finer control over your assets. You can take control of asset generation by entering customized asset mode:

RenderWare→*Advanced Settings*→*Assets* right click and choose *Customize*

This activates the customize mode and the Customize option in the right mouse menu is checked.

You are now able to manually edit your assets. The exporter will no longer automatically generate assets as you modify your scene. You can leave customized asset mode by re-selecting the Customize option from the right click menu. The exporter will warn you that any asset modifications you have made will be lost.

Adding Assets

To create your own assets based on a selection:

RenderWare→*Advanced Settings*→*Assets* right click and choose *Add*

This option enables you to create your own assets. Modifying the objects selected

Auto Create Assets

Whilst in customized asset mode you can still have the exporter re-analyse your scene and create assets for you:

RenderWare→*Advanced Settings*→*Assets* right click and choose *Auto Create*

This is similar to automatically generating assets. However, this option does enable you to keep your existing customized assets, if desired.

Adding a Child

An animation is a child of an animated hierarchy. When an animated hierarchy asset is created, an animation child asset is also created. This option enables you to add your own child to an animated hierarchy.

RenderWare→Advanced Settings→Assets right click and choose *Add Child*

This option can be used on a selected asset which can be customized.

Deleting an Asset

Right click the asset and choose *Delete* or select an asset and press *DELETE* on the keyboard.

Only customized assets can be deleted.

Exporting and Viewing Assets

To export and view assets using Advanced Settings:

- Right click below the assets in the Advanced Settings Assets tab and choose *Export all* or *View all* to export or view all assets

- Right click an asset and choose *Export* or *View*. This exports or views the selected asset only.

✎ Multiple assets can be selected and exported or viewed. Use the Ctrl key or rubberband drag with the mouse to select multiple assets.

Enabling and Disabling Assets

During development stages of a scene you may want to create differing variations of export options. Assets can be enabled and disabled giving greater flexibility to the artist.

- Right click an asset and choose *Disable* an asset. This asset will not be exported when all assets are exported or viewed.

- Right click and asset and choose *Enable* an asset. This asset will be exported individually or as part of a full export or view. By default this option is on.

Export Location

The files that the exporter generates during an export are dependent on various settings. The type of files exported is always dependent on the project template in use. By default the file names and locations are calculated as indicated by the following table:

EXPORT FILE TYPE	EXPORTED FILES
RWS	*Default Export path* + *Project name* + `.rws`
RF3	*Default Export path* + *Project name* + `.rf3`
Legacy Files (anm, bsp, dff, spl)	*Default Export path* + *Asset name* + `.anm` or `.bsp` or `.dff` or `.dma` or `.spl` (depending on the asset type)

The file column of the asset manager provides a preview of the file that will be generated when the asset is exported. When the column displays multiple files with the same name this indicates that the assets will all be added to the same file. In some situations, such as when the project template is setup to export an RWS file and legacy files, the exporter may not be able to provide a suitable preview of the files that will be exported.

The export file location (but not file type) can be overriden per-asset when in customized asset mode. This is done by clicking on the asset filename. You can either type an export filename directly or use the browse button to navigate to a location. If you enter an absolute path it will be used as is. If you enter a relative path it will be displayed and exported relative to the Default Export Path location. Any extension you specify will be ignored as the file type is still controlled by the project template in use. To export multiple assets to one file, define the same export file for each asset.
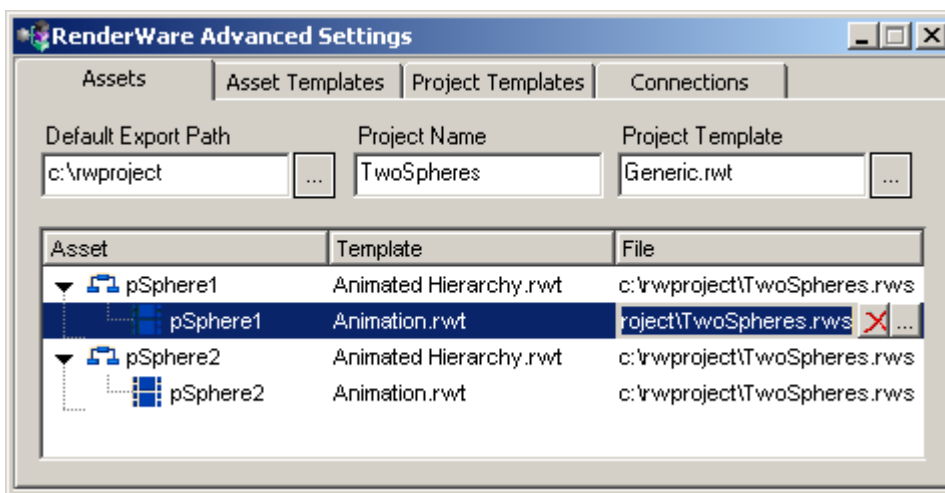
After the output file has been customized the filename will appear in bold in the filename preview column. This lets you quickly identify which assets have customized output files.

If you later wish to return the asset to its default output file you can either enter an empty string or use the cancel button to clear the custom file.

Asset Properties

The four default types of assets have differing asset properties. These assets are based on the four asset templates.

Different symbols are used for the different types of assets. The table below refers to the different asset templates available and the RenderWare Graphics term they relate to.

	ASSET TEMPLATE	RENDERWARE GRAPHICS TERM
	Animated Hierarchy	`RpClump`
	Animation	`RtAnimAnimation`
	Static World	`RpWorld`
	Spline	`RpSpline`

Each asset has properties. To display the property dialogs, double click the asset or right click the asset and choose *Properties*.

Animated Hierarchy Asset Properties

An Animated Hierarchy asset contains a single hierarchy as a RenderWare Graphics container object, called `RpClump` (see the *Fundamental Types* and *Dynamic Models* chapters of the User Guide for more details on `RpClump`).

Depending on the options selected, in the Animated Hierarchy Asset Template, the exported file saved can contain the object hierarchy, hierarchical animation, skinned animation, and morph target animation.
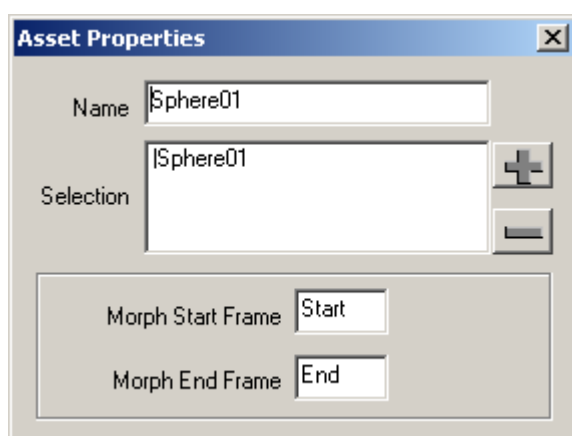
Name: Name of the asset.

Selection: Objects contained within the asset.

Morph Start and End Frames: Sets the start and end range of the animation to use in exports. "Start" and "End" will query the values from the time slider. Alternatively, you can enter numbers in the fields but ensure that only positive numbers only are used.

Animation Asset Properties

An Animation asset stores only animation data and is always a child of an Animated Hierarchy asset which contains the geometry. It is recommended that the asset names of the Animation and the Animated Hierarchy are identical. The selection used in the animation export comes from the selection property of the parent Animated Hierarchy asset.
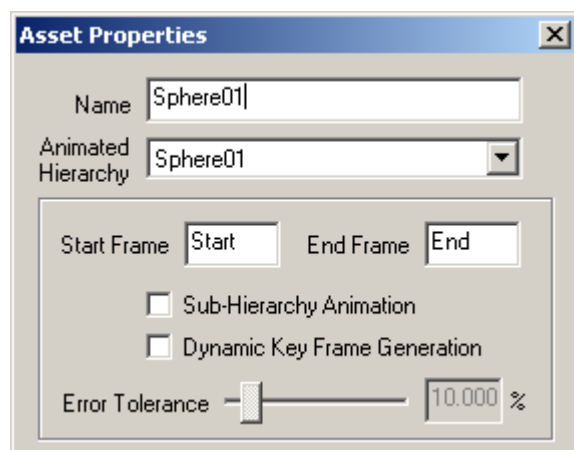
Name: Name of the Asset.

Animated Hierarchy: The name of the parent node.

Start and End Frame: In Maya these options set the start and end range of the animation to use in exports. "Start" and "End" will query the values from the time slider in Maya. Alternatively, you can enter numbers in the fields but ensure that you use positive numbers only.

Sub-Hierarchy Animation: This option is used to export part of an animation by selecting the node at the top of a node hierarchy. Exporting with this option set will export the animation from the selected node down to the bottom of the selected tree. The selection used comes from the parent Animated Hierarchy asset.

Dynamic Key Frame Generation: This option tells the exporter to create extra keyframes based on the error of exported animation compared to the Maya curve. This may be useful if you have used AnimCurves with spline tangents at the keys (current RenderWare Graphics animation systems perform linear interpolation only).

Additionally you can turn this option on for error checking. This option then adds extra keyframes to your animation although it maybe preferable for the artist to manually add extra keyframes on the most important objects.

Error Tolerance: This option is grayed out unless the *Dynamic Key Frame Generation* option is selected. It controls the percentage error allowed at integer keyframe times before the exporter will add a new keyframe in the exported animation.

Static World Properties
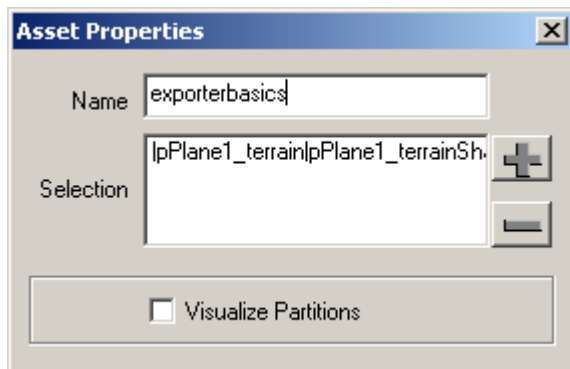
Name: Name of the Asset

Selection: Objects contained within the asset.

Visualize Partitions: shows the partitions/sectors that are generated as a result of the export process.

Whilst the partitioning of an exported world can be viewed in the viewer, it is sometimes useful, especially when using hints, to view them in the art package.

For more information about partitions and hints refer to 3.13 Partitions (page 28) and 3.12 Hints (page 25).

Spline Asset Properties
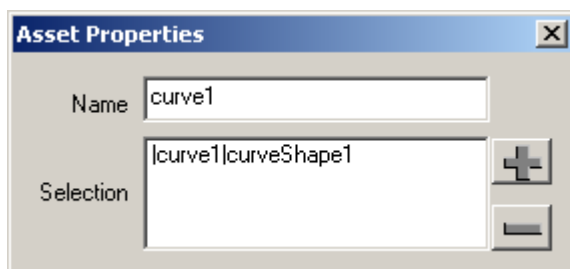
Name: Name of the Asset.

Selection: Objects contained within the asset.

Asset Selection

The example below explains the steps required to manage asset selection.

1. Create 3 boxes in a scene

2. *RenderWare→Advanced Settings→Assets* tab
   An asset has been automatically generated.

3. Right click in the Assets tab and choose *Customize.* This enables you to customize your assets.

4. Delete the automatically created "untitled" asset.

5. Select 2 boxes in your scene.

6. In Advanced Settings Assets tab, right click and select *Add.* This creates an asset based on your selection.

7. Double click the "untitled" asset to access the Asset Properties.

8. Rename the asset "selectionsets".

9. The dialog lists the objects included in the asset, for example CubeShape1, and Cube3.
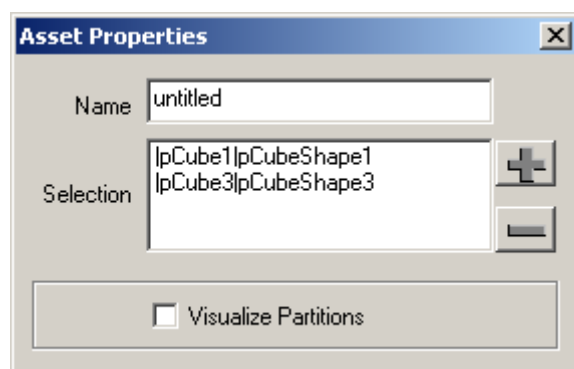
- ⊞ enables you to add an object to the selection set. To add an object, select an object in your scene and click this button to add the object to the selection. For example, you might want to add CubeShape2.

- ⊟ enables you to remove an object from an asset. To remove an asset select the objects in the dialog and click this button to remove the object from the selection set. For example, you might want to remove CubeShape3.

Asset Templates

Asset Templates contain the options settings for exporting an asset in a particular format.

The directory containing the asset templates. Click on the browse button to change the asset template path.

Two options:
Common - contains asset information which can be customized.

Platform - contains platform specific settings.
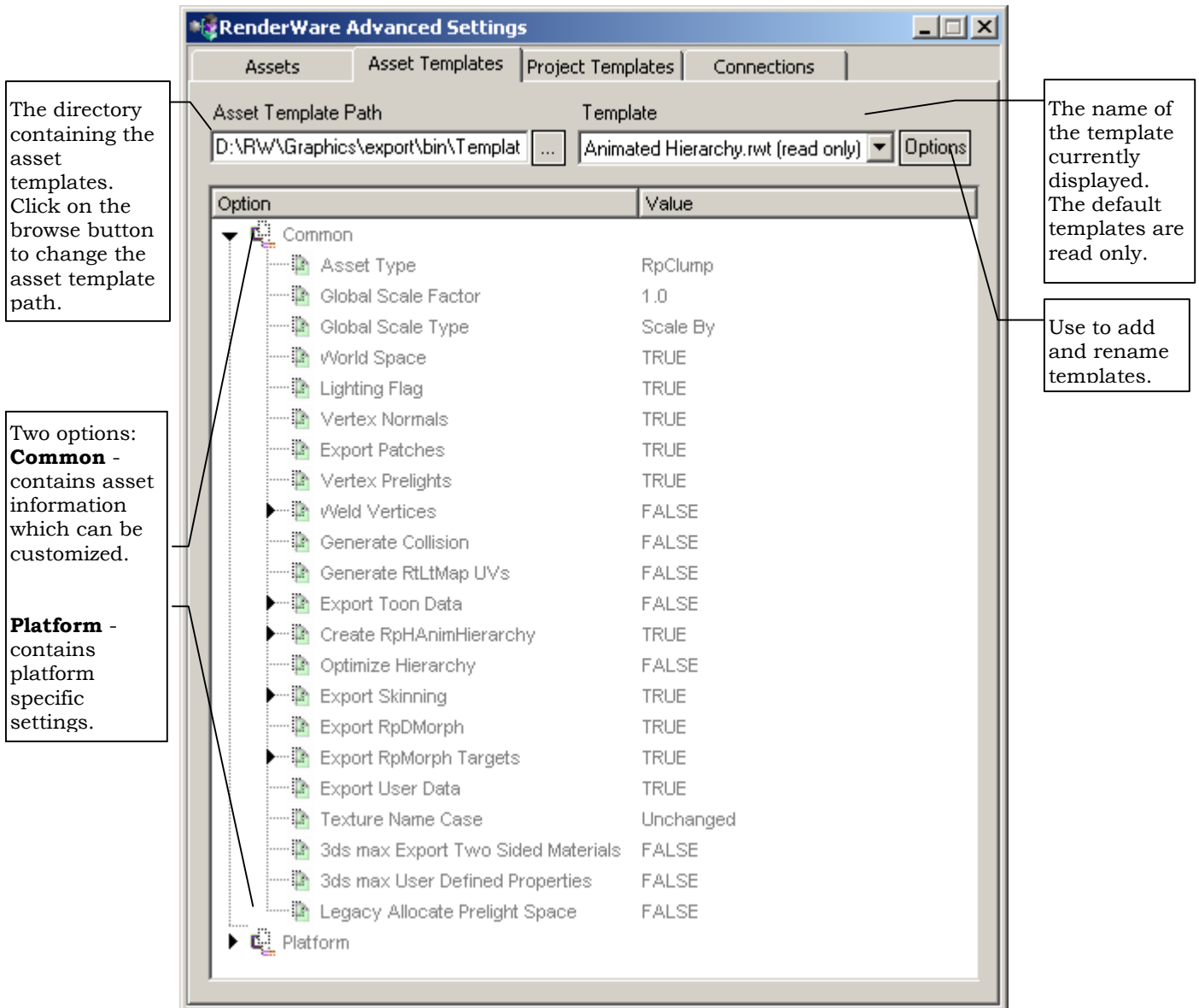
There are four default Asset Templates which you can use or you can create your own templates. The default templates are read-only and we recommend that you set up your own templates based on the default templates. The default templates are:

- Animated Hierarchy - used to export a single hierarchy within a scene as a RenderWare Graphics container object, called RpClump.

- Animation - used to export animation data. Requires an animated hierarchy with the same name.

- Spline - used to export spline data. Requires a static world with the same name to contain the world.

- Static World - used to export objects as RenderWare Graphics World, called `RpWorld`. Worlds do not contain any hierarchy or animation information and are typically used for static level geometry in a game.

> Asset Templates were referred to as batch node properties in RenderWare Graphics releases 3.3 and 3.4.

Project Templates

The Project Templates contains the settings that were previously found in Scene Settings in the RenderWare Graphics 3.3 and 3.4 exporters. These options enable the user to export files to `.rws` (the recommended file format), `.rf3` (the new XML file format) and to legacy file formats (`.bsp`, `.dff`, `.anm`, `.spl`).
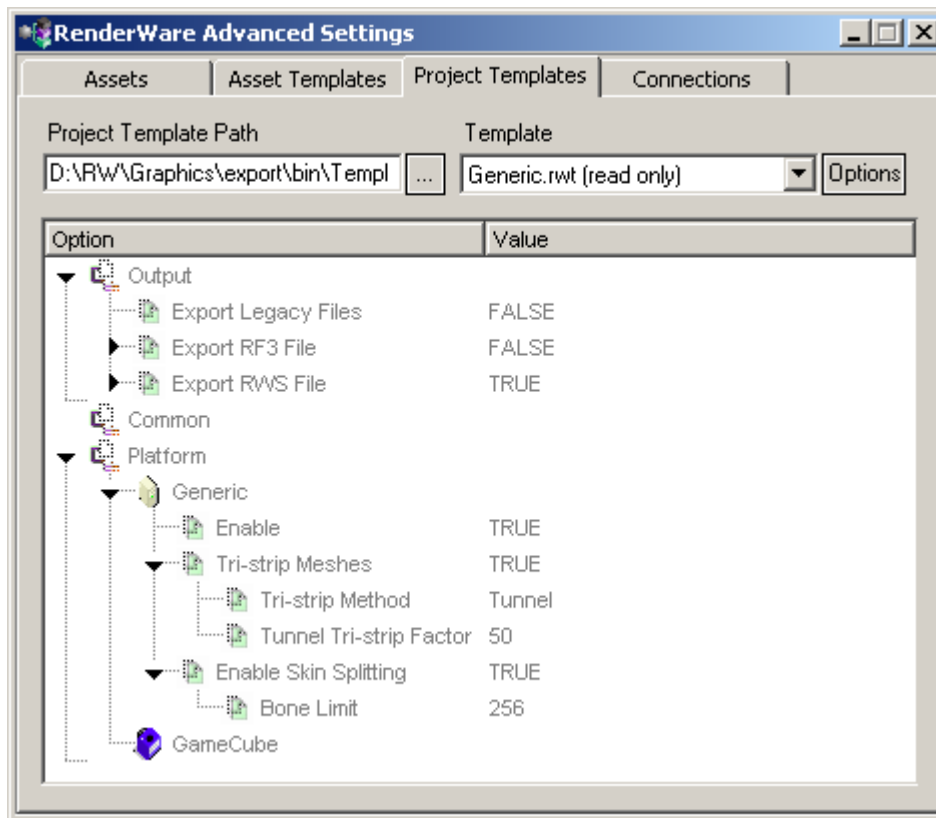
Project Template Path: The directory containing the project templates.

- `Generic.rwt` used for exporting files to the PC (e.g. D3D8 or OpenGL), exports files with the extension `.rws`.

- `GameCube.rwt` used for exporting files to GameCube exports files with the extension `.rg1`.

- `PS2.rwt` used for exporting files to PlayStation 2 exports files with the extension `.rp2`.

- `Xbox.rwt` used for exporting files to Xbox exports files with the extension `.rx1`.

The platforms listed will depend on the RenderWare Graphics platforms you currently have installed. The Project Templates contains the setup information. The actual project template used is specified in the Assets.

The Output and Platform options are explained in full in TechnicalArtistGuide.pdf.

6.4 Connections

Connections stores the connection settings require to view assets in RenderWare Visualizer.

Visualizer

When you run Visualizer from Maya for the first time, a connection wizard will run. Initially, you are given the option of choosing which graphics system to use for local visualization:
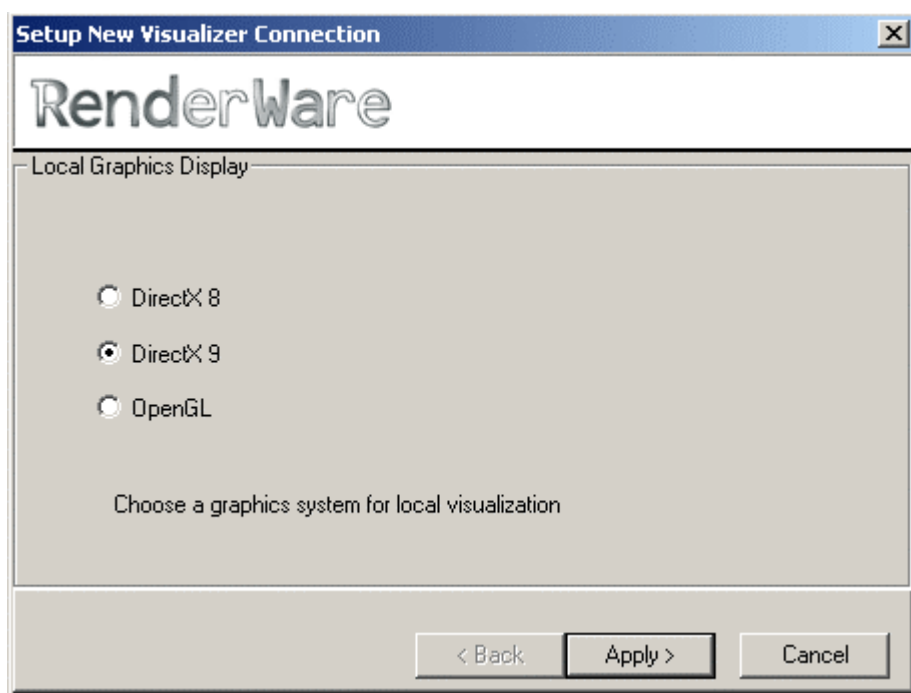
Subsequently, you may setup a connection to a console:
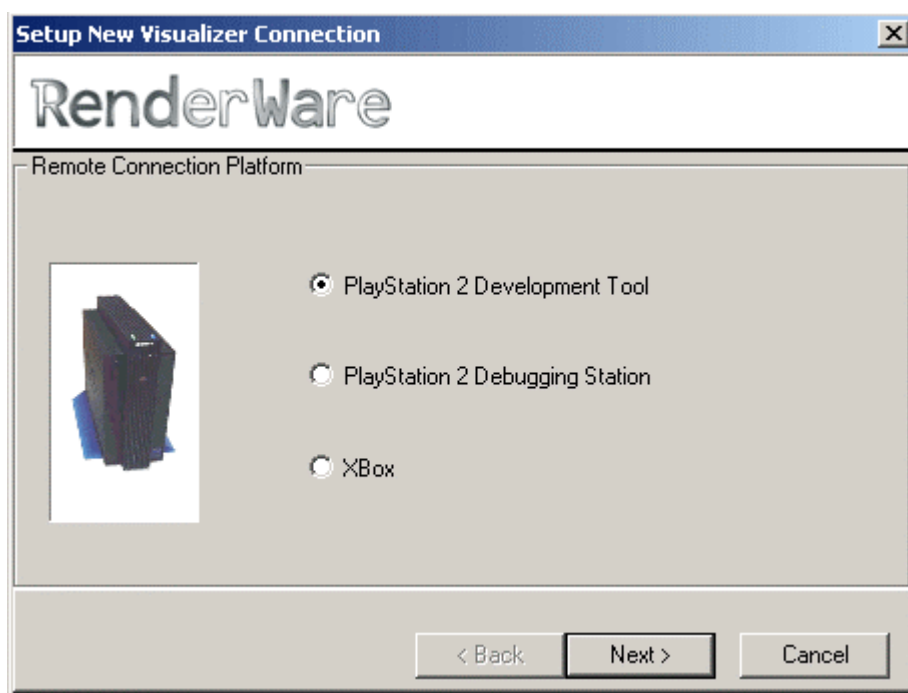
The wizard will guide you through the steps necessary to initialize a connection.

Talk to your system administrator during this phase of configuration. The Visualizer host that runs on the console is able to automatically obtain IP addresses if a DHCP server is present. Failing that, your administrator can provide you with a fixed IP address that can be used by the Visualizer host.

Configuring connections

To change a connection for the PC, select a local connection for D3D8 or OpenGL. Local D3D8 and OpenGL connections will automatically spawn viewers and close them on disconnection.
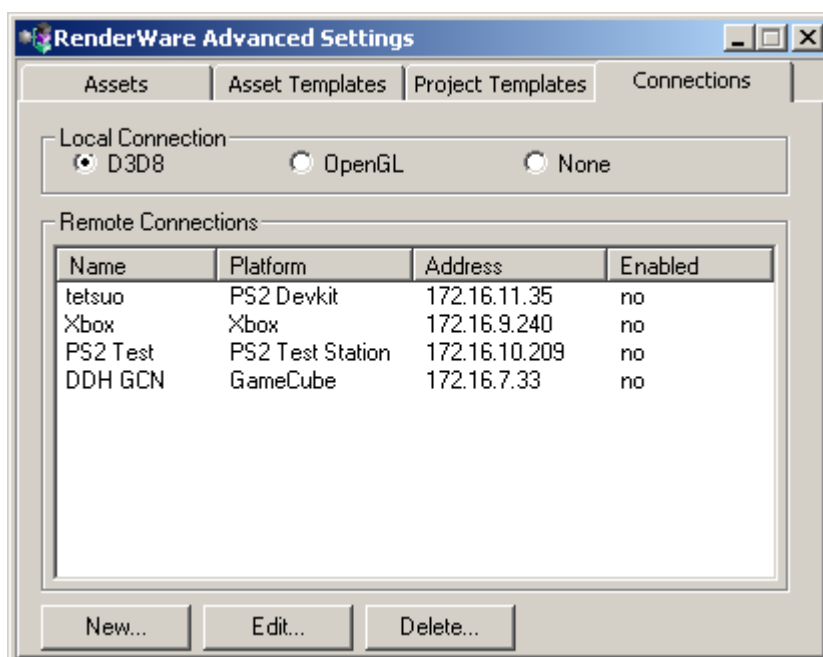
The Connections dialog is divided into *Local Connection* and *Remote Connections* so that you can easily view the rws file on different platforms. Local Connection can be used to view D3D8 and OpenGL on your PC whereas Remote Connections can be used to view rws files on target platforms. More information on setting up the Connection Editor can be found in the *RenderWare Visualizer* viewer document.

New creates a new connection.

Edit edits the selected connection. A connection is selected by clicking on the Connection Name.

Delete deletes the selected connection. A connection is selected by clicking on the Connection Name.

Connection Properties

To create a new connection click on *New* in Connections.
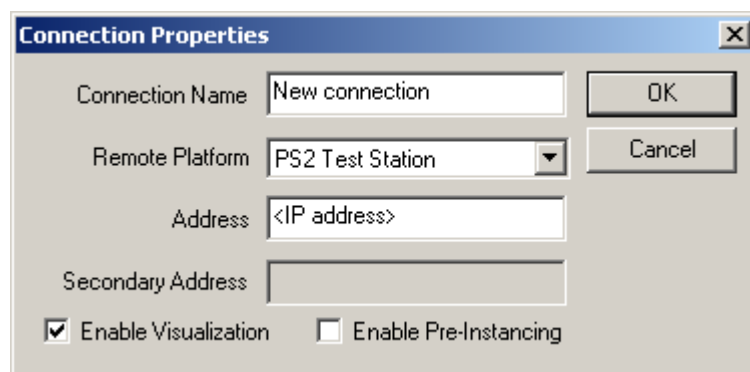
Connection Name the name of the connection to be made. This name is purely for your reference and does not need to be related to the actual name of the target hardware.

Remote Platform the remote target platform.

Address the IP address or machine name of the target hardware.

Secondary Address for the target platform. This option is only enabled for PS2 Devkit and Xbox targets.

Enable Visualization is used to state which platform viewers will display the RenderWare Export Nodes when the *View* buttons are pressed.

When all assets or selected assets are viewed the .rws file is viewed on all enabled connections using RenderWare Visualizer.

When you view assets using the RenderWare Exporter, viewer.exe is run, the registry is accessed and connections are made to the enabled target platforms. The .rws file is then displayed in RenderWare Visualizer.

Enable Pre-Instancing controls whether this connection will be used to generate platform specific data when an `.rws` file is exported with the 'Pre-Instance Data' option enabled. If assets are not being exported to `.rws` files or the 'Pre-Instance Data' option is not enabled in the Project Templates tab then this setting has no effect.

6.5 Create

Sectorization Hint

The Create menu has the following option:
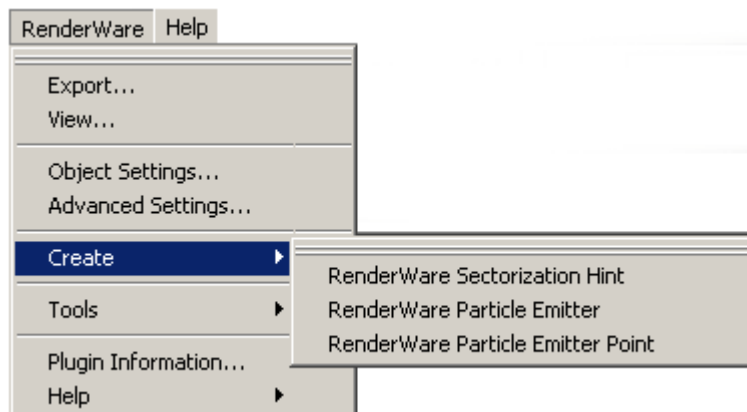
- RenderWare Sectorization Hint adds a hint to the scene.

✎ The menu options "RenderWare Particle Emitter" and "RenderWare Particle Emitter Point" are only available in the menu if the chargeable optional extra ParticleFX (part of the FX Pack) has been installed.

6.6 Tools
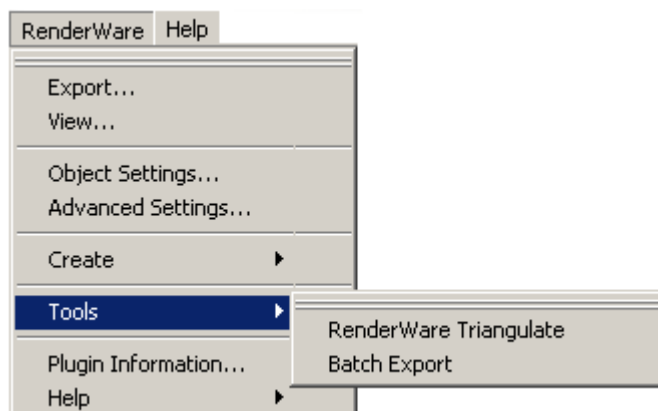
RenderWare Triangulate

This is command which works as the *Polygons→Triangulate* Maya command except it will align all the new edges in the best way possible for tri-stripping. For perfect tri-stripping, vertices have to share: normals, UVs and prelights. RenderWare Triangulate only works on quads and this command can be used for turning quads into triangles.

The command works with multiple selections of objects or polygons.
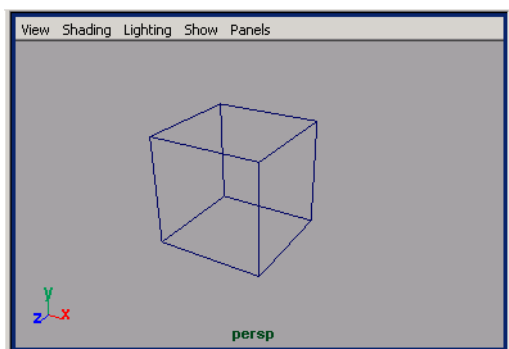
Before RenderWare Triangulate   *After RenderWare Triangulate*
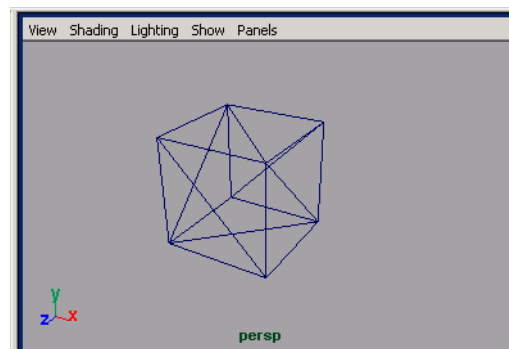
RenderWare Triangulate can be run:

- From the menu:

RenderWare→Tools→RenderWare Triangulate

- From the Command Line

```
RenderWareTriangulate
```

If you get the warning "//Warning: line1: polyFlipEdge: Cannot flip edge xxxx, contained by a concave quad //" it is because of the Maya polyFlipEdge bug.

It should not hit more than 5% of the edges that actually need to be flipped. (much less than the whole number of object edges), so if this does occur tri-stippling will still significantly improve.

This bug has been reported to Alias|Wavefront and will hopefully be fixed in the next main or intermediate (dot) release.
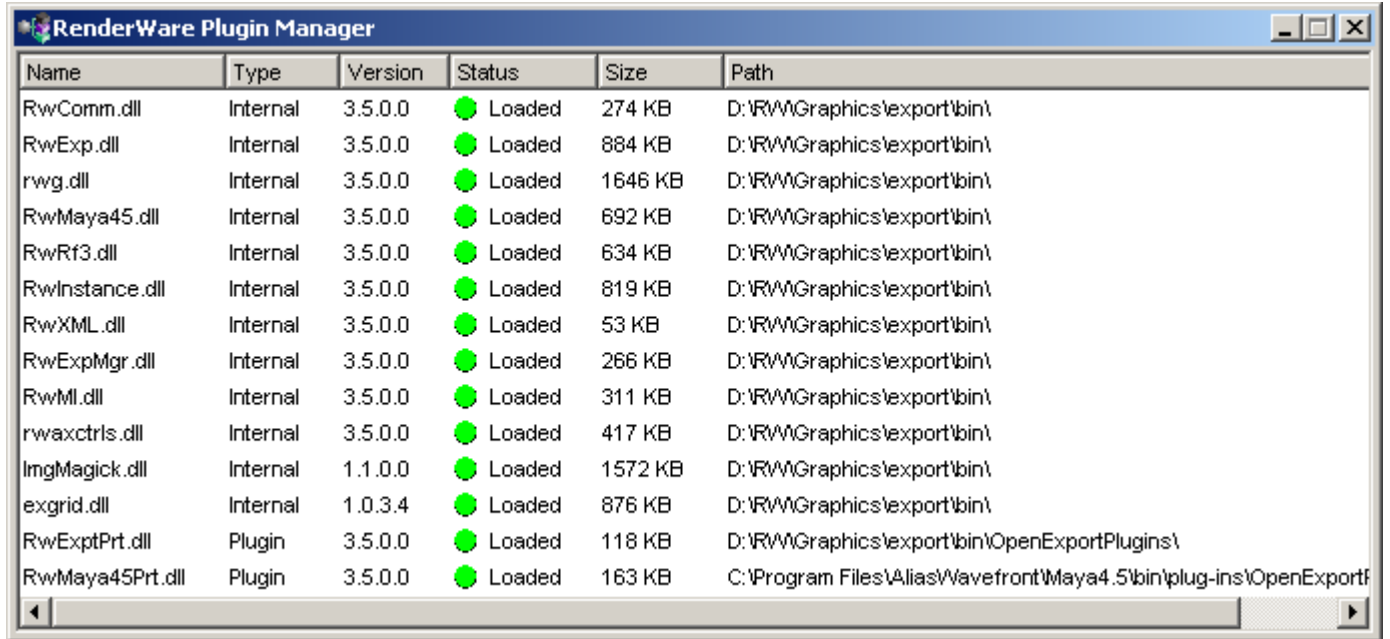
Batch Export

RenderWare→Tools→Batch Export exports all Maya files in a specified directory.

6.7 Plugin Manager

The Plugin Manager is available for information only. It contains the location of the `.dll` files used. The plugins section includes RenderWare Graphics plugins, for example, the Particles plugin which can be bought as an optional extra in the FXPack.

6.8 Other Options

Progress Bar

The progress bar works on a per export basis. For example, if you export a large scene with multiple assets it goes from 0 to 100 once.

Command Line Export

Exports can be done without running the full Maya interface.

The RenderWare Exporter script is stored in `RenderWareNodeExport.mel`.

Index