

## Artists' Guide for using 3DS Max with RenderWare3

### ***Table of Contents.***

Introduction. ....	3
1. Modeling. ....	4
Basic RenderWare3 Geometry Conventions. ....	4
Model with care. ....	4
Creating geometry.....	4
Collapse all Edit Stacks. ....	4
Keep Geometry Continuous. ....	5
Avoid Creating 'Long Thin Triangles'.....	6
Attaching the Parts. ....	7
Welding the Parts. ....	7
Working with Possible Visible Sets (PVS). ....	8
2. Texturing. ....	10
Generating UV coordinates. ....	10
The Material Editor. ....	10
Using 'Cropped' Textures.....	10
Using Embedded Alpha Data. ....	11
3 Animation.....	12
Morph Target animation.....	12
Keyframe animation.....	12
Multiple Keyframe Sequences.....	12
Export as Skin and Bones.....	13
Character Studio.....	13
4. Lighting. ....	14
Pre-Lighting World Geometry in Max. ....	14
5. Exporting data from 3DS Max. ....	16
Clumps and Atomics (DFF Files). ....	16
Scale. ....	16
Animation. ....	16
Character Animation. ....	17
Geometry Creation.....	17
Export Space.....	18
Texture Naming. ....	18

UV Range Checking.....	18
Static 3D Worlds (.BSP Files).....	18
Conclusion. ....	20

## **Introduction.**

This guide has been created as the definitive artists' reference, for use when creating and exporting geometry from 3DS Max for use with real-time applications powered by RenderWare3. It will be updated from time to time as more plugin tools and techniques are developed by Criterion Software. For those artists not using 3DS Max, most of the same basic principles apply although clearly the methods of how they are achieved will differ.

All the explanations and illustrations contained within this document refer to 3DS Max release 2.5, but are also applicable to earlier versions as well as release 3.0. However, in terms of workflow and the tools available, it is recommended that release 2.5 and upward is used.

All the techniques described here make use of the standard Max tools. The only additional plugin required is the RenderWare3 exporter supplied with the SDK.

The document is divided into five key sections. These are intended to loosely mirror the typical 3D design process. These are: Modeling, Texturing, Animating, Lighting and Exporting.

## 1. Modeling.

### ***Basic RenderWare3 Geometry Conventions.***

Supplied with RenderWare3 is a 3DStudio Max file export plugin (rw30exp.dle). Once this is placed in your 3DSMax\Plugins Folder, and you have restarted Max, the following will appear as file export options:

**RW3.0 BSP** for exporting static 3D worlds, levels, or scenes.

**RW3.0 DFF** For the export of 'clumps' - individual objects or hierarchies of objects for use within the 'world'. These can have keyframed, or morph target animations embedded within them.

Details of the various export options can be found in '**Exporting**' - Section 5 - of this document.

### ***Model with care.***

There are a few conventions to consider when modeling geometry for use in RenderWare3. Whilst they may seem a little constricting, they will ultimately save time and keep your workflow jogging along smoothly toward those all-important deadlines.

We spend a lot of time here at Criterion, repairing and altering stuff to improve game performance. If things are built with the basics in mind up front, it can save a lot of pain and tedium later on.

Sound advice is: test work in RenderWare3 often. Install the RenderWare3 viewer, and use it to keep a check on performance and appearance.

### ***Creating geometry.***

It may seem obvious but don't create unnecessary geometry. It may seem like just a few triangles here and there, but they really start to add up as the scene grows. I won't try to recommend actual maximums, that depends on target platforms, graphics cards etc, and this is evolving constantly. You'll soon start to get a feel for what works and what doesn't.

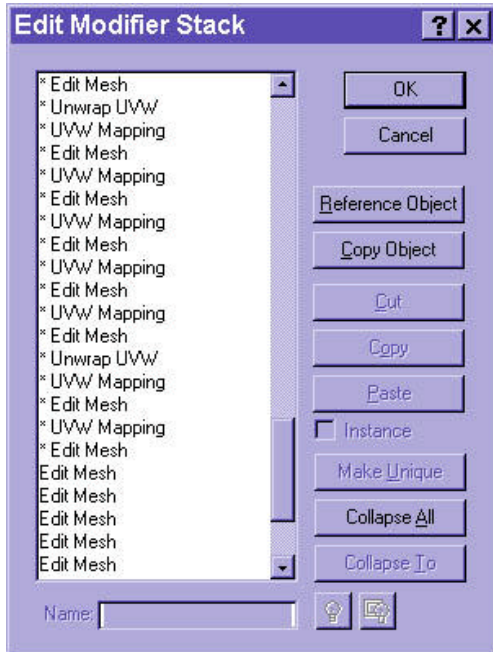
Avoid over detailing things, try to let the textures do the work (One good texture is worth a thousand polygons, and all that stuff!). After all, if you're racing round a world at 90mph, you won't see all that stunning detailed modeling anyway. Remember this is *real-time*, not FMV (not yet anyway!). Delete any unseen polygons, you just don't need them, and every poly counts. If the cameras are going to follow a spline, build with that in mind. Present your geometry to the camera like a film set - a solid looking façade.

Try to save polygons for where they will really count. For example if you are using pre-lighting put extra triangles where the light gradients occur. This will make your shadows far more detailed and help keep them free of artifacts. If you have characters, give them the lions' share, as they will be the focal point of the scene.

### ***Collapse all Edit Stacks.***

Collapse your Edit Stacks. Whilst not critical, it is certainly desirable (and good working practice) once modeling and texturing is completed to collapse the stacks (after all you can always keep the uncollapsed version!). This avoids unwieldy files, which take ages to load, save and process. I've seen 20meg files come down in size to around 1.7meg after collapsing the stacks!

A good example of potential problems created by stacks was this: We needed to re-work some existing geometry that had been originally produced in Max1.2. The artist had used some long defunct third party modifiers and left the geometry uncollapsed. For us using Max2.5 it was impossible to load any of the files. Not good when your deadline is yesterday...!



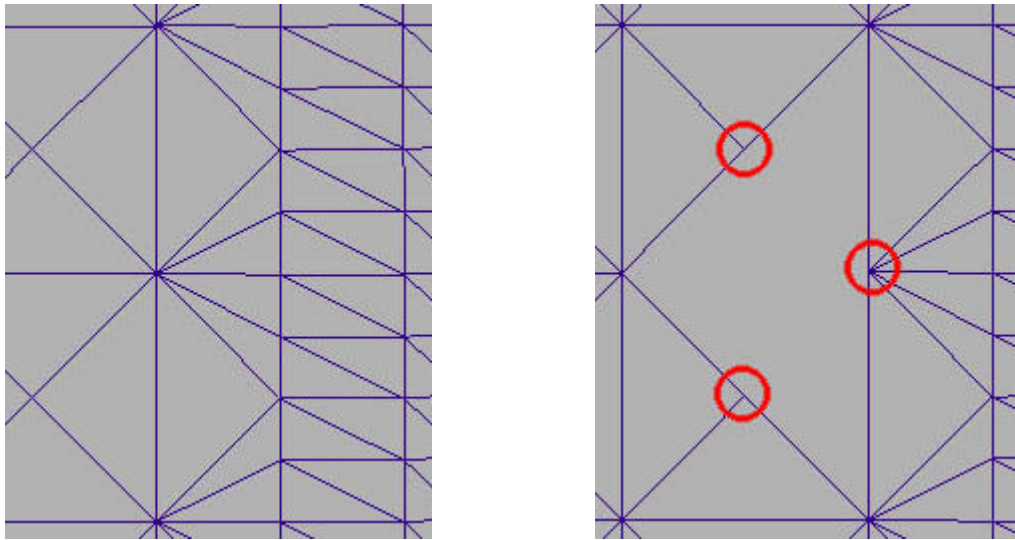
*Avoid stacks like these. (Note where the scroll bar is!!)*

Use the Collapse Utility in Max's Utilities Section to deal with large scenes quickly. It will be time well spent.

A word about splines. Be careful when you collapse geometry not to have any splines selected. These will, likewise, be collapsed to 'Editable Meshes' – not ideal! If your splines do have edit stacks it is worth collapsing these too, as they tend to contain large amounts of data which balloons Max files.

### ***Keep Geometry Continuous.***

When building your geometry, especially adjacent parts of a world, be sure to keep the meshes continuous with no gaps or mismatches between vertices or edges. This is because when you have finished constructing you will need to weld the various parts together into a continuous 'water tight' skin. Any discontinuities that do exist will manifest themselves as cracks or tears in the RenderWare3 world. This will have the QA and Games Testers complaining. When the PVS plugin is used, these can upset its operation significantly. (For an explanation of what PVS does, see **Working with Possible Visible Sets**)

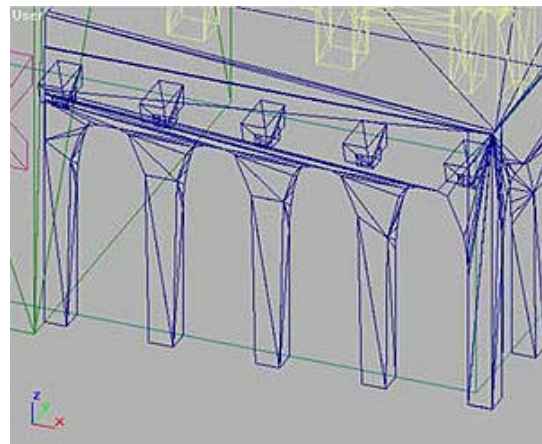
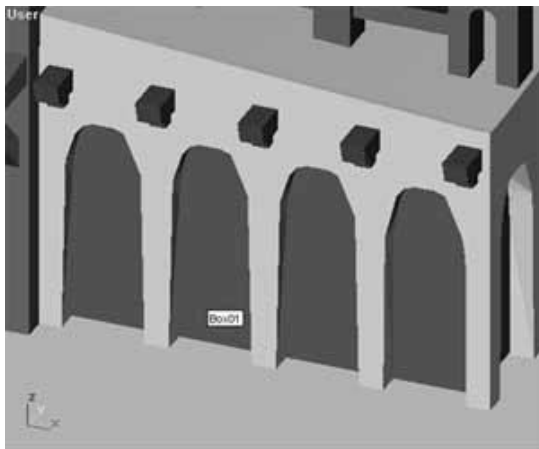


*Always go from lower to higher tessellation smoothly and avoid any discontinuities like the above.*

### ***Avoid Creating 'Long Thin Triangles'***

A popular method of building geometry is by creating a spline 'outline' for an object then extruding it into 3D.

In Max this can be bad news because although the shape looks nice and simple in Max's viewports, it can in fact be very complex. By displaying edges you can see that Max has created lots of 'long thin triangles'.

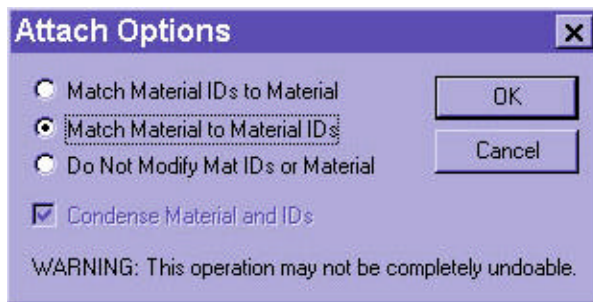


These are not good for RenderWare3 and are one of the pitfalls of letting Max "do it's own thing". They will cause artifacts and be difficult for RenderWare3 to cull nicely. They will also get split in strange ways when the Exporter creates the BSP file.

My suggestion would be not to create geometry in this way, or to rework it by hand and reconstruct the faces in a more efficient way. Also I would have less definition in the curves of the arches and let surface smoothing do the work of creating the appearance of a smooth surface.

### ***Attaching the Parts.***

When you're ready use Max's *Attach* tool and start to join all the adjacent parts of the world's surface geometry. If you are creating an interior space this will include the walls and ceilings. In the *Attach Options* dialogue select the middle option – *Match Material to Material IDs* – this will keep the texture material assignments of the various components intact.

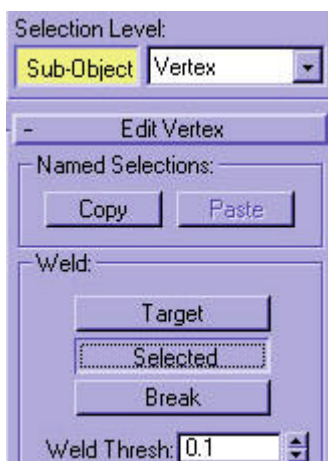


Once all the component parts are attached, go into the material editor and select the eyedropper tool. Click on the object and you will select a newly created Multi Sub-Object Material. This new material has no name, so give it one now. Again, this will pay off later, because if you create or merge more 'attached' geometry you won't get two objects with un-named Multi Sub-Object Materials. If you do, it really throws the texture assignments badly, and can cost you loads of time reassigning materials.



### ***Welding the Parts.***

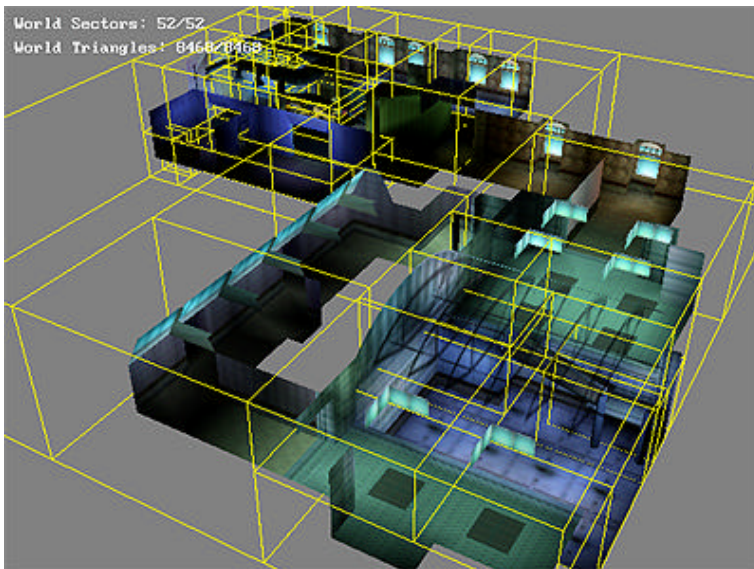
The next job is to weld all the various sections' adjacent vertices together. Choose *Vertices* in *Sub-Object* Mode, and group select some vertices along a seam. Note how many you have selected, then go to *Weld Selected*. If the number of selected vertices halves you know that you have welded them all successfully. If not undo, and step up the *Weld Threshold* value and try again. You should eventually arrive at an optimum setting, which you can stick to for all the other welds.



*Merge all world object geometry and weld all to avoid holes or cracks – as well as looking unsightly, cracks and gaps will upset the PVS, in some cases prevent it working at all.*

### ***Working with Possible Visible Sets (PVS).***

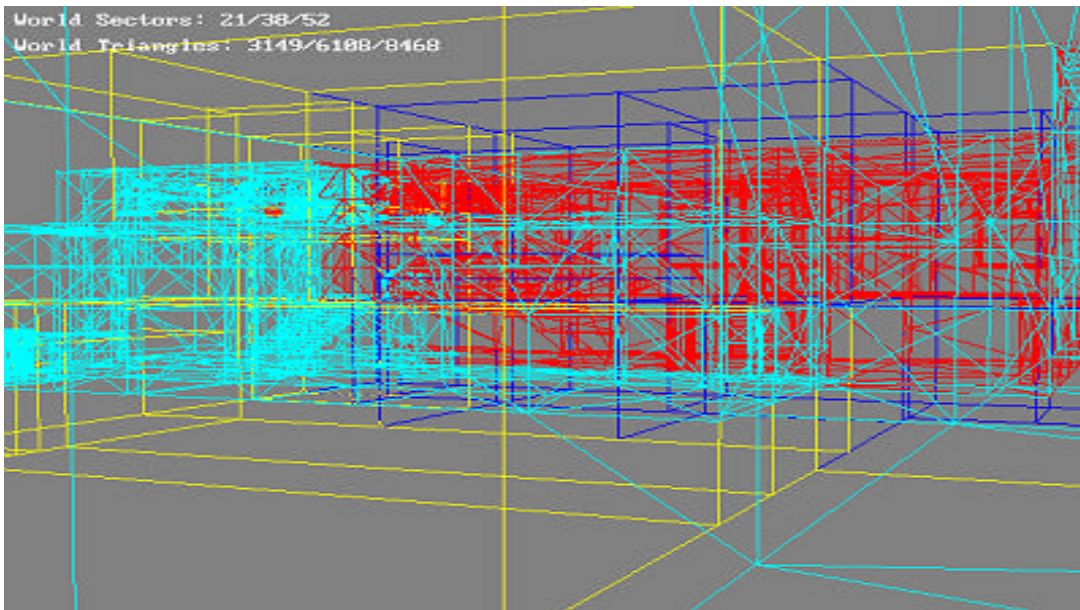
When the exporter creates RenderWare3 BSP static world files it divides them into 'World Sectors'. These are sub divisions of the whole world's geometry and effectively break it down into 'bite sized chunks' that can be handled efficiently by RenderWare3.



*RenderWare3 world with world sectors displayed in yellow.*

What PVS does, in a nutshell, is two things: First it creates a database by moving a virtual camera systematically around from world sector to world sector. It then sends rays from the camera position to establish what other world sectors can be viewed from the current one.





The data collected is used to create a database which can then be accessed by RenderWare3 so that from any given camera position it will know what other world sectors are visible and therefore need to be rendered. This is an effective way of culling unseen geometry sector by sector in real-time. The supplied RW World Viewer enables you to generate and keep an eye on PVS data as shown here:

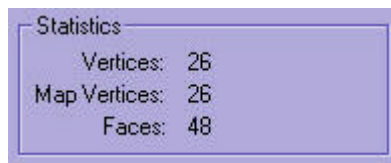
*Wireframe of the same world showing PVS in action.*

The color coding provides visual feedback as follows: Cyan = world geometry. Yellow = world sectors. Blue = PVS culled world sectors. Red = PVS culled geometry. The numerical feedback in the top left of the viewer represents, from right to left: Total world sectors. World sectors in camera view. Rendered world sectors. Below is the equivalent information in triangles. A list of commands for accessing these modes is provided with the World Viewer.

## 2. Texturing.

### ***Generating UV coordinates.***

All the texture UV coordinates generated by the mapping tools in Max should export with no problems to RenderWare3. However try to avoid using 'Box' mapping. This seems to generate large amounts of map vertices which all get exported to RenderWare3. Below are two examples of the same object with regular mapping on the left and box mapping applied on the right. Notice the excessive amount of map vertices generated by the box mapped example.



Statistics	
Vertices:	26
Map Vertices:	26
Faces:	48



Statistics	
Vertices:	26
Map Vertices:	144
Faces:	48

### ***The Material Editor.***

RenderWare3 supports diffuse mapping and Opacity mapping. When naming maps, it is generally recommended to stick to the 8.3 format. Stick to lower case, do not use spaces and use .bmp image type. This is most bullet proof for working across platforms. Power of 2 sizes work best although RenderWare3 can handle odd sizes, as these will be resampled on loading.

The alpha map can either be a separate bitmap in the 'Opacity' material channel, or an alpha component of a .PNG file.

Max procedural textures, bump mapping, shininess, reflection, refraction, specular values, animated, and wireframe materials are not currently supported by RenderWare3.

Self-illuminating materials are supported; with the restriction that when used, the whole of the clump to which a self-illuminated material is applied will be self-illuminating.

Double sided textures are now supported by automatically creating backface geometry on export. There are options to duplicate the UV vertices if required.

The texture map cropping tool values are also taken into account on export, as is the diffuse color value. This should normally be set to white – RGB = 255,255,255. If it is not the underlying color will show through the material giving it a strange hue or making it look 'dark'. There is also a considerable processing overhead if the value is not white (especially on the DreamCast).

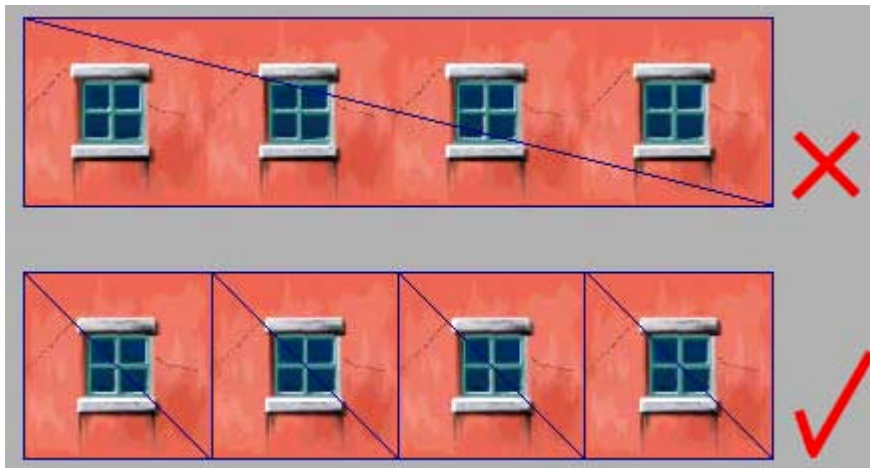
### ***Using 'Cropped' Textures.***

Using the texture-cropping tool in Max is fine, but please bear in mind a couple of important points:

The exporter takes its cropping UV and WH values from those set in the dialogue box by the marquee tool. If the 'Apply' check box is not checked, although you won't see it in Max, the values still remain and will be exported. These values have to be set to zero if you don't want your texture cropped. This is a common cause of 'my textures look wrong'.

*Make sure that if a material has had cropping applied but no longer needs it the UV and WH values are set back to zero. Unchecking 'apply' will not work.*

In RenderWare3 you cannot tile a cropped texture across a single polygon as in the illustration below. It will not work, as RenderWare3 has no means of knowing where the crop should apply.

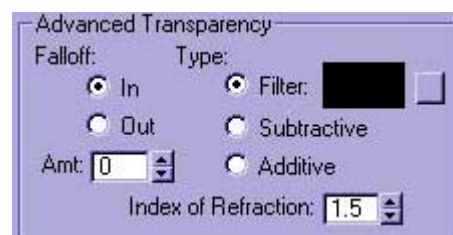
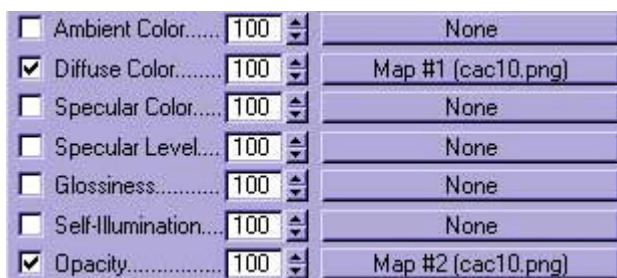


*Don't tile a cropped texture across single polygons. It is illegal in RenderWare3.*

### **Using Embedded Alpha Data.**

It is desirable to use 32bit images and have your transparency information embedded with the diffuse texture map. It saves texture memory and removes the need for having two map files to represent a surface with opacity. To this end RenderWare3 supports the PNG file format.

To set PNG alpha up in the Material Editor, load the same file into both the 'diffuse' and 'opacity' channels and then in the 'Advanced Transparency' section under 'Extended Parameters' set Type to filter and the color to black.



### 3. Animation.

Currently there are five types of animation supported by RenderWare3 – Morph Target, Keyframe, Seq Anim and Skin 'n Bones and now Character Studio.

#### ***Morph Target animation***

Linear morphing between targets is supported by RenderWare3. Because of the nature of linear morphs you may find the need to create more targets to keep object distortion to a minimum. The RenderWare3 exporter samples the geometry's vertex positions at user specified intervals and then interpolates between them. The greater the number of samples the better the fidelity, and the bigger the file sizes. Create morph objects in Max and animate in the normal way using the Morpher modifier.

#### **Keyframe animation**

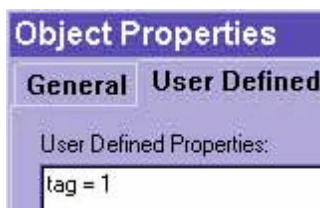
Linear transform and linear rotation keyframes can be exported from Max to RenderWare3 (note: Max tension continuity and bias values if set will be ignored) scaling is not supported by *any* of the keyframe options. Keyframe stuff in the normal way and the animation data will be exported as part of the DFF file.

**Note:** The root object of *any* animated hierarchy must be a *reference* object with *no* keyframes set. This could be a cube or some similar primitive, which will act as an anchor object in RenderWare3. As you will not want this object to be visible to RenderWare3 you make it non-renderable using Max's *Object Properties* dialogue. Right click the object and under 'properties', 'general', 'Rendering Control' uncheck the *Renderable* option. It will now export as invisible geometry.

#### ***Multiple Keyframe Sequences***

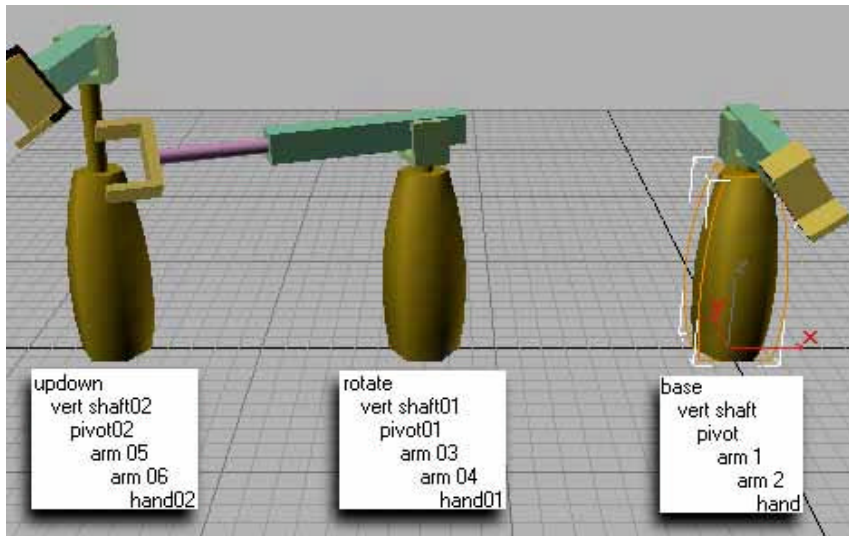
This method enables a given object hierarchy to have several or many, discreet animation sequences linked to it in RenderWare3.

Using the example of the robot arm illustration below, the hierarchy on the right is the 'original' geometry to be exported. Each object is tagged in order to associate it with the animated 'copies' when exported to RenderWare3. To tag an object, right click on it and open the 'Object Properties' dialogue box. Click the 'User Defined' tab and enter a tag name and number.



Start with the root object then step through the hierarchy tagging each part in turn incrementing by one each time.

The cloned copies to the left, contain two animation sequences. These are identified by re-naming the base object to the name you want the sequences to have, in this case 'rotate' and 'updown'. When the 'base' hierarchy is exported with 'Save Anim Sequences' box checked in the RWexporter the keyframe sequences of the clones will be exported as part of the DFF and receive the same names as the base objects.



*Multiple keyframe sequences set up in Max showing copied hierarchies.*

### ***Export as Skin and Bones***

This is similar to multiple keyframe sequences except that the root hierarchy acts as a bones system to deform a skin. There is a separate tutorial for this supplied with the SDK.

### ***Character Studio***

Character Studio is now supported. This is a major upgrade to the artist's armoury. Skinned characters can now be driven by motion capture data and animation sequences managed by a popular and powerful set of tools.

Currently, RenderWare3 Pipeline 1 only supports rigid vertex assignment in Physique.

RenderWare3 PowerPipe users, however, can now use weighted skinning. Each vertex can be shared between up to 4 bones and each skeleton consist of up to 64 bones.

To export, select the 'skin' and check the Character Studio check box in the exporter. Currently only the active Biped animation is exported. If you wish to export sequences edited in the Motion Flow Editor export them as a .bip file first and then re-import as a single sequence into Character Studio.



## 4. Lighting.

### *Pre-Lighting World Geometry in Max.*

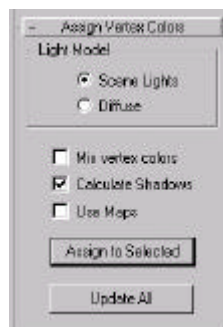


*RenderWare3 geometry pre-lit by 3DS Max using Color Vertex Pre-lighting.*

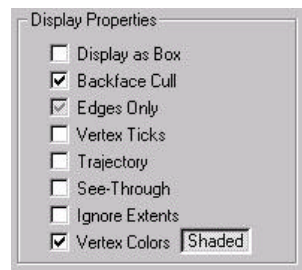
World (static) geometry can be pre-lit in Max *before* it is exported to *RenderWare3*. This gives the artist control over how the scene will appear in the final application. The density of the geometry affects the shading and quality and light fall off. Basically more triangles equal better shadows. 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 *RenderWare3* lights to create the same effects, which would require a far greater processing overhead, so all in all, it is a relatively inexpensive solution.

With the addition of the Assign Vertex Colors Utility in 3DS Max 3.0 it is now possible to 'bake' light values into vertices in Max instead of the *RenderWare3* exporter needing to perform this task. This can be found in the Utilities section of 3DS Max and should be set as follows when baking lights to geometry:



In order to view the results in the viewports you will need to set up the geometry's properties to display vertex colors:



This has several key advantages. All 3DS Max light types can be used. All their parameters are taken into account. Shadow casting can be calculated. The result should remain consistent between 3DS Max and RenderWare3.

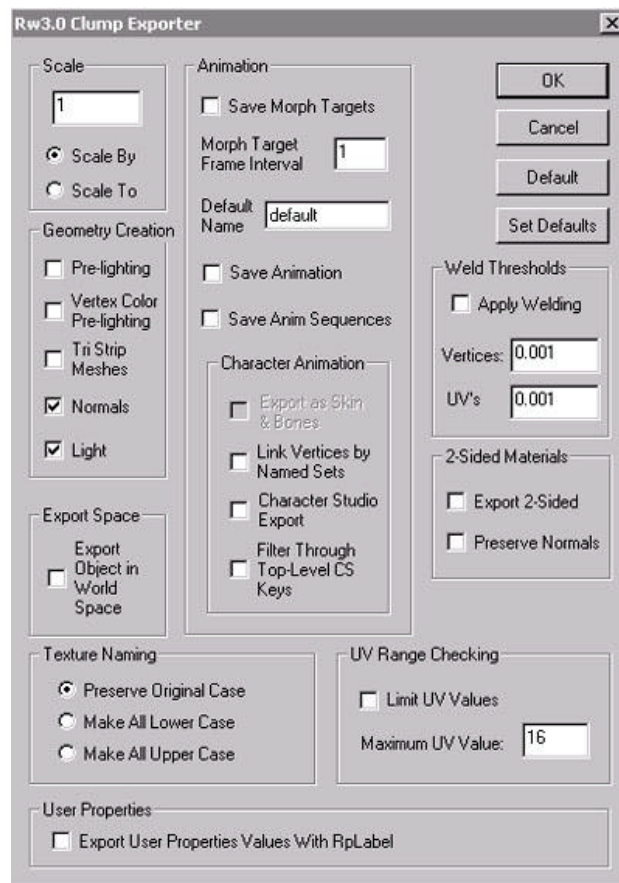
When duplicating lights that perform a particular function (in the case below *fill1*), clone them as *instances*. Now you only have to tweak the original to affect all the 635 others. This way you can globally alter the lighting in your worlds quickly and relatively easily.

To view the results of exports quickly drag and drop exported BSP files onto Advview, the artwork viewing tool supplied with the RenderWare3 SDK. The L Key toggles the default lighting on and off. You will need this to be turned *off* in order to see your pre-lit geometry.

## 5. Exporting data from 3DS Max.

### ***Clumps and Atomics (DFF Files).***

Below is the dialogue box you see when exporting clump data from Max. Below that is an explanation of each function.



#### **Scale.**

**Scale By** – Enter a value here to scale your clump. A scale value of 2 would be twice the size of the original. A scale value of 0.1 is one-tenth the size etc.

**Scale To** – The value here will scale an object to *RenderWare3* units. In other words, if you're unsure of the relative size of your geometry, this can be used to scale it to fit the game.

We generally try to stick to one scale factor for an entire project, although this may not always be possible.

#### **Animation.**

Currently, there are three animation export options. There follows a brief description of each:



**Save Morph Targets** -will sample any morphing created in Max and sample the geometry at whatever the value the 'Morph Target Frame Interval' box is set to. The lower the value the higher the animation's resolution (and resulting file size).

**Save Animation** – saves the keyframes with whatever hierarchy you have selected as a .dff and calls the anim sequence 'default'.

**Default Name** – for RpAnim all animations are named, when exporting a single animation we always used to name it "default" since it's very difficult (read impossible) to rename animations this lets you select the name to use for the single animation.

**Save Anim Sequences** –is a bit more complex. If you want a given hierarchy to have different animation sequences associated with it, create duplicates and animate them. When 'Save Anim Sequences' is checked, the original object hierarchy is exported, and the animations associated with the duplicates are exported, but linked to the original by *RenderWare3*. These anims are given the name of the **root** objects of the duplicate hierarchies.

### Character Animation.

These animation options are specifically included to allow for flexibly 'skinned' character export types and include:

**Export as Skin & Bones** – *This is in some ways similar to 'Save Anim Sequences' above, but builds on the idea in the following ways. As you can see from the illustrations below the 'bones' and 'bounding hulls' do not have to have the same proportions. As long as the axes are coincident the transforms you create with the 'bones' will work fine once they are translated by RenderWare3 into Hulls. This is just as well as animating the thinner bones will give a much better picture of how the final character will behave than would the bulky container hulls.*

*'Skin' object contained within bounding box hierarchy.*

*Animated 'Bones' corresponding to box hierarchy.*

*Skinned mesh animated by the bones in RenderWare3.*

**Link vertices by Named Sets** – *This is a further option for Skin and Bones where named selection sets of vertices can be assigned to a particular bone. **Note:** you *must* have the skin in Sub Object Mode, with edit vertices active when you export for this to work.*

**Character Studio Export** - This allows the export of skins attached by the 'Physique' Max plugin to Character Studio 'Biped' skeletons. Only **rigid** vertex binding types are supported by RenderWare3 using PipeLine 1. RenderWare3 PowerPipe users have access to weighted vertices. Any vertex can be weighted between up to 4 bones, and the maximum number of bones in a skeleton hierarchy is 64.

### Geometry Creation.

**Pre-lighting** – Unlike BSP files this will *not* pre-light the geometry using Max lights. Instead it will allocate 'space' within the exported geometry for lighting values to be added by *RenderWare3*.

**Vertex Color Lighting** – Color Vertex Pre-lighting enables the export of pre-light information created in Max using the Assign Vertex Colors 3DS Max Utility. It also allows for the painting of Vertex colors to be supported using the 3DS Max Paint Vertex Modifier.

**Tri Strip** – This is a method of Geometry creation which allows vertex sharing between triangles and is therefore a more efficient way to generate data. It does not suit all applications and is therefore an export option.

**Normals** – If this is checked we include normals in the export, if not we don't.

**Light** – This sets the RenderWare lighting flag in the exported data, if this is left unselected the exported geometry doesn't get lit by real time lights.

## Export Space.

**Export Object in World Space** – By default the Exporter removes all Translation and rotation data from the root of an object hierarchy. In other words if an object is upside down in Max it will appear the right way up in *RenderWare3*. This function over-rides this so that the same object will have the same orientation when exported to *RenderWare3*.

## Texture Naming.

These options allow you to convert the names used for textures (and hence bitmap filenames etc) into all upper or lower case (in case your platform has restrictions).

## UV Range Checking.

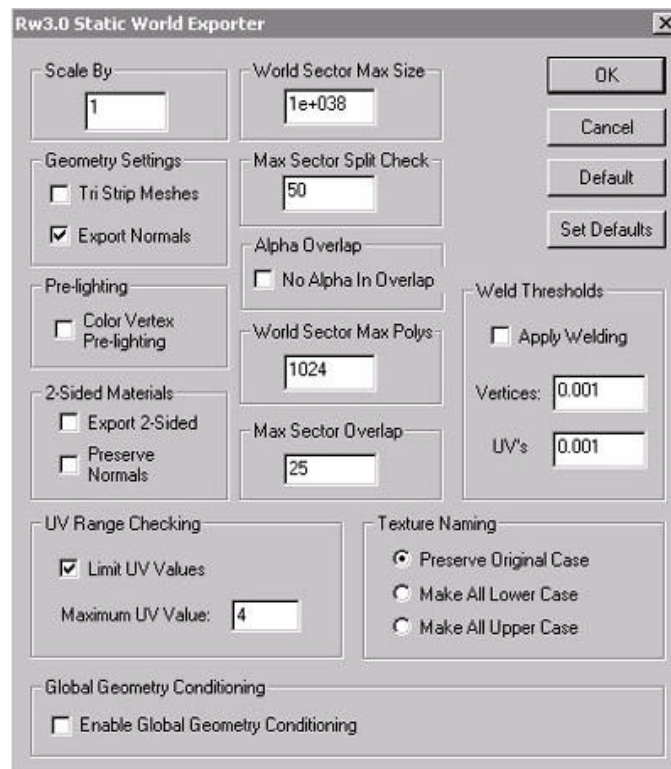
**Maximum UV Value** – When checked will clamp all UV values of a model to the specified maximum. This helps keep UV's under control when designing for specific platforms or graphics systems.

## User Properties.

**Export User Properties Values with RpLabel** – Allows text strings defined in a Max object's User Defined Properties dialogue to be exported for use in a *RenderWare3* application which uses the RpLabel plugin.

## Static 3D Worlds (.BSP Files).

Below is the dialogue box you see when exporting static world data from Max. Below that is an explanation of each function.



**Scale By** - As with the Clump Exporter this controls the relative scale of the exported geometry. A value of 2 will be twice the size of the original, 0.1 is one-tenth etc.

**Geometry Settings** – Tri Strip Meshes: This is a method of Geometry creation which allows vertex sharing between triangles and is therefore a more efficient way to generate data. But does not suit all applications and is therefore an export option. Export Normals: when checked will enable the world geometry to receive dynamic lighting.

**Pre-lighting** – Color Vertex Pre-lighting enables the export of pre-light information created in Max using the Assign Vertex Colors 3DS Max Utility. This allows the artist to make use of all Max lighting types and parameters and take account of shadow casting. Also the appearance of the model when rendered should be identical in both Max and RenderWare3.

**2-Sided Materials** – Export 2-Sided: will automatically create backfacing triangles where geometry has a 2-Sided Material defined in Max. Preserve Normals: will make sure that the normals of the backfacing geometry are identically opposite the mirrored faces.

**Texture Naming** - Allows the artist to over-ride texture map names on export and force the RenderWare3 file to have mixed, all upper, or all lower case names. This helps when exporting for particular target platforms.

**World Sector Max Size** – This controls the maximum dimension of a world sector.

**World Sector Max Polys** – This controls the maximum number of polygons in a world sector. The default value of 1024 should be fine for most worlds. Going higher will reduce the number of world sectors but with the risk of world collision data being lost. This would manifest itself in objects falling through the world in *RenderWare3*. Going lower increases the world sector generation with consequent increase in geometry.

**Max Sector Split Check** – controls the number of planes that are tested when trying to determine the best plane to divide a sector whilst generating BSP data. A higher figure generally generates fewer polygons or less splits. Too high though, and it will start to falloff. Basically it is a case of experimentation to find the optimum value.

**UV Range Checking** – When checked will clamp all UV values of a model to the specified maximum. This helps keep UV's under control when designing for specific platforms or graphics systems.

**Weld Thresholds** - Similar to Weld Threshold in Max except that it operates at export time. I prefer to do this in Max so I can keep close control on what is happening.

**Global Geometry Conditioning** – This runs a global conditioning algorithm over world geometry, it will potentially optimize the geometry (less polys/verts) but may not maintain the level of tessellation setup by the artist hence lighting may not be as required.

## **Conclusion.**

This document is meant only as an overview for using 3DS Max with *RenderWare3*. As such it's generic nature will no doubt leave many specific points you may have unanswered. Additional support is available from our online forum on the *RenderWare3* website at:

<http://developer.renderware.com>.

Updates of this document will also be made available from the *RenderWare3* site as will a version for Maya users.