

3d modelling for Virtual Reality

Tutorial #3 – VRML & panoramic tips!



Some extra information on working with VRML/panoramic models.

Contents:

1	Improving visual performance.	2
1.1	Modelling geometry using primitives:	2
1.2	Texture map sizes.	2
1.3	Browser VRML client (Cortona player) settings.	3
1.4	Shadows and reflections in VRML.	3
1.4.1	Preparing a shadow texture map in 3ds max.	3
1.5	Other VRML nodes to consider – billboard and lod.	3
2	Navigation tips.	4
2.1	Cameras.	4
2.2	Anchors.	4
2.3	Invisible objects.	4
3	Panoramics from the 3ds max model.	5
3.1	Creating the still images from 3ds max.	5
3.2	Creating more vertical visibility.	5

1 Improving visual performance.

VRML worlds can sometimes be based on quite complex geometry, (high polygon counts) and use many large and detailed digital images as texture maps, etc. These all contribute to excessive 'end use' frame rates when the world is being interacted with.

To minimise the load that the model puts on your computers graphics resources there are some basic rules to follow.

1.1 Modelling geometry using primitives:

The VRML standard supports 'primitive' nodes. These are pre-defined 3d objects such as spheres, boxes and cones, which are converted/created from the 3d model when exporting from 3ds max. They provide a very efficient way of representing the shapes in the VRML file without having to provide all of the co-ordinate code data for each and every defining vertex.

To create primitive objects when exporting to VRML simply select the **Primitives** option in the VRML 97 exporter window.

The primitive object is then re-defined in VRML using a default 'optimised' number of vertices. If the quality (resolution) of the resultant geometry is not acceptable then you will have to deselect that **Primitives** option and the object will be exported as any other non primitive object. To smooth out the individual faces of multifaceted objects ensure that the **Normals** option is selected in the VRML 97 exporter window.

Note for AutoCAD modellers:

If you model a sphere in AutoCAD, using for example the sphere command, and then import into 3ds max, that sphere is not actually recognised as a sphere primitive. It will not be possible to export it as a sphere primitive in VRML. You will have to model the sphere in 3ds max. The 3ds max sphere will be recognised by the exporter and can be exported as a primitive object.

1.2 Texture map sizes.

For optimal image rendering always save your texture maps (digital images) to a size in pixels which are multiples of powers of 2.

Widths and heights: 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, etc.

Some graphics systems may also render square format images more efficiently than rectangular, but the main advise is use images saved with widths/heights as above, in pixels.

Image formats must be *.jpg, *.gif or *.png.

1.3 Browser VRML client (Cortona player) settings.

The VRML client that your browser uses may be able to use the different graphics rendering software that is installed on your machine. Access the preferences and to set this. In the Cortona player simply right click in the display area, select preferences and then select the Render tab. Here you can chose from a list of the available graphics renderers. You may need to experiment with these. Some renderers support certain features differently, such as fog, lights and anti-aliasing.

1.4 Shadows and reflections in VRML.

There are limitations with the visual realism possible using VRML. The format does not support the real time rendering of either shadows or reflections.

Reflections are difficult, as they change as the observers viewpoint changes.

Shadows however can be cheated in VRML, provided objects and light sources remain static. To do this you can use a pre-prepared image of the shadow cast on to the surface as a texture map for the VRML model surface.

1.4.1 Preparing a shadow texture map in 3ds max.

Use the same model that you have created to export to VRML.

Have the lights and materials all set up and tested.

Capture the image of the shadow on the surface by rendering a viewport that is oriented square on the surface.

Remove the object that casts the shadow from the rendered image by de-selecting 'Visible to camera' in its properties (select the object, right click > properties.....).

For the lights casting the shadows experiment with the different shadow types, either using the **Light lister** or by selecting the lights and clicking on the **Modify** tab. The most realistic may be **Area Shadows**. Try increasing the **Shadow integrity** and **Shadow quality** under the **Area Shadow** properties.

Save the image. In a graphics package you can enhance the image as you like. Follow the guidelines in section 1.2 for formats and image sizes.

Once saved, apply a new material to the surface and apply this new image as a map on the diffuse channel.

1.5 Other VRML nodes to consider – billboard and lod.

See module web page, the 3ds max to VRML document.

2 Navigation tips.

Ease of moving around the VRML world is critical to the quality of the virtual reality experience. There is the basic **navinfo** node of course, which every VRML world/model must contain. This really is necessary and defines parameters such as eye level height, collision distance, etc. Other techniques can be used to make navigation easier as well, such as using cameras (viewpoints), proximity sensors or simple objects with anchors and invisible objects.

2.1 Cameras.

The easiest way to provide good access to any location/view point in your VRML world is to create a selection of carefully placed cameras. As a guide create at least four cameras in the 3ds max model. These will then appear as viewpoints in the exported VRML world. To make smooth transition between navigation walking and jumping to viewpoints, ensure that the cameras are at the same height as the **terrain** setting in the navinfo node.

2.2 Anchors.

The anchor node provides a very useful and easy way to jump to pre-defined viewpoints (cameras) or any other web page or VRML world. This jump can be instigated in various ways, for example by clicking on an object anchored to a camera or file.

2.3 Invisible objects.

You can use the collision behaviour defined by the navinfo node to prevent access to certain areas of your VRML world simply by modelling extra geometry to act as fences or floor surfaces and apply a totally transparent material to them. The user will not see the objects but collision detection will.

3 Panoramics from the 3ds max model.

A good way to produce QuickTime panoramic movies from your cad model is to use the Stitcher software to stitch together still images rendered in 3ds max. Use of Stitcher software is covered by documents on the module web page.

3.1 Creating the still images from 3ds max.

- Create a free camera and give it an appropriate name for future reference, such as panoramic.
- Move the camera to the desired location in the model and make the necessary adjustments to the direction. Switch the **Perspective** viewport to the new panoramic camera view to help set up the camera view.
- Under the Modify tab of the camera adjust parameters, such as field of view.
- In the **Top** viewport animate the camera rotating through 360°, using all of the default 100 frames.
- Observe the camera view as you play or 'time scale slider' the camera motion and make any required camera adjustments.
- Render the animation, making two important settings in the Render window first:
- Configure to render only every **nth** frame. Choose a number which gives you images that overlap by about 20%.
- Create a folder for the rendered output and under the files settings select this folder location and a still image format such as *.bmp, *.png, *.tiff or *.jpg (watch for quality loss with *.jpg).
- Render.

Now open Stitcher. Load these images and proceed as usual.

3.2 Creating more vertical visibility.

These images should give you a 360° panorama, possibly with some freedom to 'look' up and down. If you would like more or total freedom to look anywhere you will need to render more images. So far you have created a single row of images at a specific eye level. You need to render images from cameras set with higher and lower eye levels in order to obtain more rows of images for stitching.

You could simply change the direction of viewing of the existing camera and render over a set number of frames again. A better way however is to copy the panorama camera, give it a new name and then use it for the higher eye level views. Obviously repeating for the lower level views. You may, depending on the field of view of the camera, need to render single vertical up/down images as well, to complete the scene.

One important thing to remember is not to change any of the camera parameters or its actual location of the copied cameras. Only change the direction they 'look'.