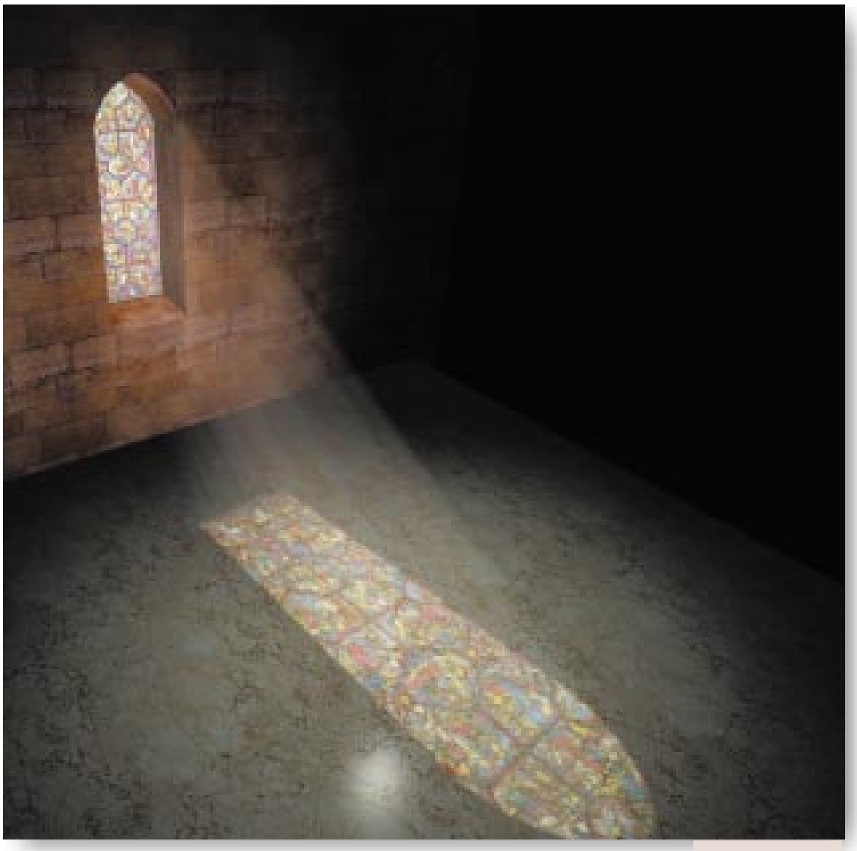
Glass act

Benjamin Woolley creates transparency and light through stained glass.

A window (of the sort you find in a building rather than on a computer screen) is an essential architectural object. For the 3D artist it also represents an interesting challenge; one that raises important questions about the way light works in the virtual world of 3D graphics. So, I have decided to devote this month's column to creating a window — no ordinary window, mind, but a rather beautiful stained glass one [Fig 1].

I chose stained glass for three reasons: you cannot see through it, so I did not have to build an exterior; it provides an interesting exercise in using relatively simple modelling techniques to build what appears to be a complex object; and it shows how 3D software simulates — or, more accurately, fails to simulate — light passing through transparent materials. I also hope to throw some light *(sorry — couldn't resist it!)* on a range of topics relating to lighting, which will prove useful in a variety of contexts.



▲FIG 1 THE STAINED GLASS WINDOW, WITH TEXTURED WALL AND FLOOR

Firstly, here are a few tips on constructing the window itself and fitting the glass. The stained glass was scanned in from a picture I have of the window of Saint Thomas in Bourges cathedral in France — it is a vibrant, medieval masterpiece.

To fit the window, I needed to punch a hole the right shape in the wall, add the tapering surround and create the window pane itself. I did all this with the same basic 2D outline of an archway, traced from the scanned image of the window. Some drawing packages (e.g. CorelDraw) include tools for tracing around bitmaps and turning their outlines into 2D shapes which can be exported to your 3D authoring package. Check first that there is an appropriate format for exchanging files between the packages. I used my 3D package which allows you to place a bitmap as a background in a viewport and then use the drawing tools to trace the outline. The resulting shape was a simple arch made up of lines connected by just five vertices. Keeping the number of vertices down was important because if there were too many, some of the following operations may have produced unpredictable results. I extruded this to create a 3D model with its cross-section matching the shape of the window and then tapered the resulting object to create the surround. I used this to cut a hole in the wall, using a Boolean subtract operation.

...I used this to cut a hole in the wall, using a Boolean subtract operation

The 2D outline of the stained window image was also used to create the window pane, which was simply a flat plane with a matching cross-section. This meant that the image, applied as a texture, fitted perfectly.

Then came the tricky bit. Transparent objects created by most 3D packages behave differently to their physical counterparts. For example, it is possible with many packages to apply a bitmap to an object as an 'opacity map'. Such a map is a texture which varies the transparency of the object according to the intensity of colour in the image — darker areas are treated as more transparent and lighter areas as more opaque. You might therefore imagine that if you used the stained glass image as an opacity map or, indeed, used a texture that was completely transparent, any light that was shining at the opposite (i.e. external) side of the wall would shine through the less opaque parts of the window and illuminate the interior of your scene.

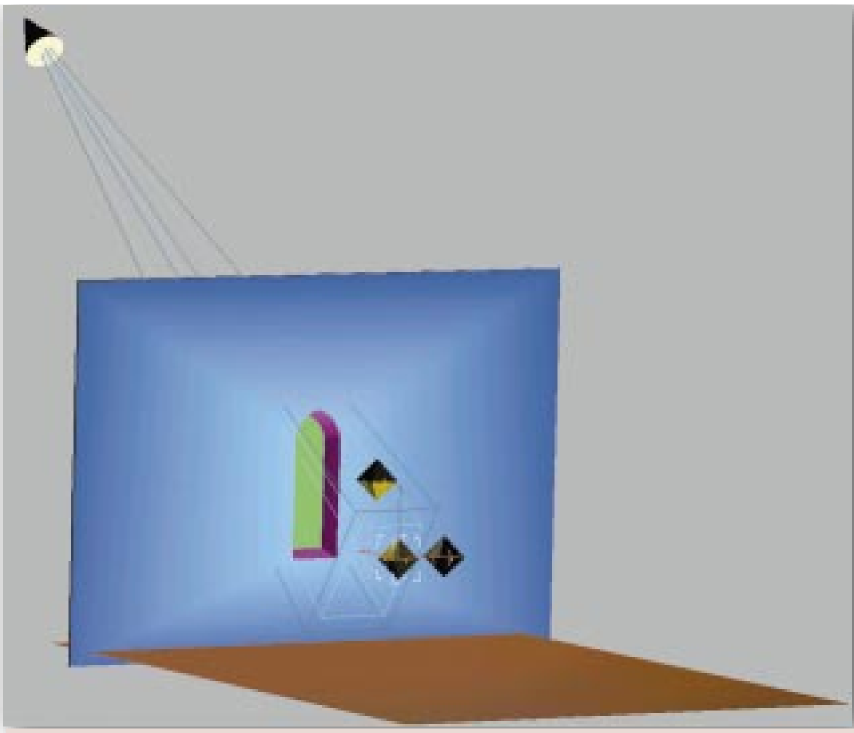
However, with the standard renderers used in many 3D packages, you will find that transparency is not quite the two-way process it is supposed to be. If you make a window out of a transparent material to simulate glass, either stained or clear, you can see out but a light source does not shine in. In other words, it is as though you have X-ray eyes which enable you to see through objects that are in all other respects solid and opaque.

Raytrace renderers, which are slower but generally more accurate, should handle transparency better. However, when I used a raytraced material in the scene created for this exercise, the light projected into the interior through the window still failed to reach the floor, even when I made the window material completely transparent.

To create the effect shown in Fig 1, I decided to do what you nearly always have to do when creating 3D scenes — cheat. The light source shining through the window is a spotlight set behind the wall [Fig 2]. The light needed to be of the shadow-casting type (some are not; the light they cast passes through the illuminated objects as if they were not there) so that the silhouette of the window surround was cast on the floor.

However, to prevent the light now being blocked by the 'transparent' window pane I had to exclude it from the list of objects the light illuminated. This meant that the light from that spot passed through the pane as though it were not there. So, to get the stained glass pattern onto the floor, I made the spot a projector light, using the original bitmap of the stained glass window as the projected image. With just the spot to light the scene the interior was still invisible except for where the image of the stained glass window was projected onto the floor, so I had to add extra lights: one to light the interior of the window pane, to simulate it glistening in the sunlight, and two others to light the floor and the wall, simulating the ambient light reflected by the sunlight passing into the building.

Finally, because it made the rendering time a great deal longer so I wanted to get everything else right first, I added a very feint volumetric lighting effect to simulate the sunlight picking up dust in the atmosphere.



▲ FIG 2 THE BASIC ELEMENTS OF THE SCENE. I HAVE PLACED DUMMY OBJECTS IN THE POSITIONS OF THE LIGHTS, TO SHOW WHERE THEY ARE. THE CONE IN THE TOP LEFT-HAND CORNER IS THE VOLUMETRIC SPOTLIGHT AND THE BLUE LINES INDICATE THE DIRECTION OF ITS ILLUMINATION. THE THREE TETRAHEDRONS CLUSTERED IN THE CENTRE OF THE SCENE ARE THE FILL LIGHTS — OMNIS AS OPPOSED TO SPOTS — WHICH CAST THEIR LIGHT IN ALL DIRECTIONS

The lessons to be learnt from this exercise are that when you are creating an architectural interior you have to treat the light coming through a window, and the lighting and texturing of the window itself, as two separate items. And, the more you try to simulate a physical scene by matching the characteristics of the materials within it, the less convincing the result is likely to be.

▲ FIG 3 THE TORII OF ITSUKUSHIMA TEMPLE MONUMENT CAN BE DOWNLOADED AS A 3D MODEL FROM WWW.GREATBUILDINGS.COM

■ View great buildings for free

If you are interested in architecture, I would recommend that you head over to www.greatbuildings.com. There you will find not just a huge database of some of the world's finest buildings but also several that can be downloaded as 3D models. The list of constructions available in 3D ranges from the familiar Colosseum in Rome, to the less familiar but elegant Sir John Soane museum in London, from the ancient and exotic Torii of Itsukushima [Fig 3] to the modern and minimalist Villa Savoye in France. To examine these models, you will need a utility that can deal with files compressed using the Apple Mac StuffIt file format, and DesignWorkshop. The latter is described by its publisher, Artifice, as an 'architectural modeler'. You can download an evaluation 'Lite' version for free via the Great Buildings website but be warned, it is 7Mb.

PCW CONTACTS

Benjamin Woolley welcomes your comments on the 3D Graphics column. He can be contacted through the PCW editorial office (address p14) or by email at 3d@pcw.co.uk