Why did the **chicken** cross the road?

To see Talisman, the new 3D hardware architecture from Microsoft. Benjamin Woolley looks at its application in real-time graphics, and gets in a twist about special FX.

I first visited SIGGRAPH, America's annual computer graphics megafest, in 1989 when it was held in Boston. I still have the mousemat to prove it, which features a large red lobster (Boston's unlikely choice of mascot). Even in those days, SIGGRAPH was huge, attracting upwards of 20,000 delegates from all four corners of the globe and the computing industry. It was there that I remember Al Gore, then a humble senator, now vice-president of the USA, opening the event with a live-by-satellite speech in which he talked of information "exploding in leaps and bounds". A wonderfully Moulinexed metaphor it may have been, but it accurately captured SIGGRAPH's transformation into one of the computing world's key events.

It was at SIGGRAPH 89 that people started talking excitedly about this newfangled virtual reality idea, and gazed with amazement and amusement at Jaron Lanier, the then fledgling VR industry's chief guru, doing his strange sort of jam session thing during what was otherwise supposed to be a serious technical conference.

These were the first tinglings of excitement that now seem to electrify SIGGRAPH every year as it becomes ever more firmly established as the venue for unveiling the most exciting ideas and developments in visual entertainment. At this year's conference, held in New Orleans, they came in their tens of thousands to get a peek at next year's movie effects, web content and games. Where Boston was full of bearded programmers and conceptual artists, New Orleans attracted the likes of Jeffrey Katzenberg, co-founder (with Stephen Spielberg and David Geffen) of the new computer-literate Hollywood studio, Dreamworks. It is also where you find a welter of new animation, including a strange little cartoon called "Chicken Crossing".

Finger-lickin' good...

Chicken Crossing was neither produced by DreamWorks nor any other studio. It came from Microsoft, a company that did not even attend Boston yet was out in full force in New Orleans. Although amusing enough as a work of entertainment, Microsoft's first attempt at a cartoon had the primary purpose of showing off "Talisman", a new technology being developed by the company's research division. This is, Microsoft states, "a new 3D graphics and multimedia hardware architecture" and if Chicken Crossing is anything to go by, it's the first sign that decent real-time 3D graphics may at last find their way onto the home PC.

First, let us consider what we mean by real-time graphics. In a 3D game like, say, Myst, or a movie with 3D graphic effects such as Twister (*see later*), the computer-generated images you see take hours, sometimes even days, to produce. So, obviously, they have to be done in advance. As a result, a game like Myst cannot strictly be 3D. Rather, it is a slideshow of 2D images with various puzzles determining the order in which they are seen.

A game like Doom is very different, because as you wander around those interminable tunnels (I am not a fan) the images are more or less generated from scratch as you go. This is necessary if the game is to allow you to roam freely through the artificial world it is trying to recreate, because to pre-render and store each possible scene as witnessed from every possible point of view would require impractical quantities of rendering time and storage capacity. Games like Doom deploy

Fig 1 **A particle system in action**

Fig 2 **Three stills from Chicken Crossing**

a special set of graphics tools (known as APIs) which use a variety of nifty shortcuts and compromises to generate each image as and when it is needed.

Several APIs have been developed for this task, one of the best known of which, Reality Lab, was created by a British company called RenderMorphics. Like so many innovative British high-tech companies, RenderMorphics was snapped up by Microsoft which used Reality Lab as the basis for Direct3D, which itself is a subset of a whole library of APIs designed for multimedia content, called DirectX.

DirectX provides the software layer for the Talisman architecture, and Chicken Crossing was supposed to demonstrate what the two could achieve, in combination. According to Microsoft, a Pentium PC with Talisman hardware (which should only cost two or three hundred dollars) could render and display each of the frames you see in Fig 2 and the 6,997 that made up the rest of the Chicken Crossing animation, in the time it takes for the screen to refresh (in other words, around one 75th of a second). This is an astonishing claim, given the richness of the textures and the number of objects: way beyond anything currently achievable on a Pentium system, even one with hardware acceleration.

In an extremely technical paper presented to SIGGRAPH, Microsoft explained how this impressive trick could be pulled off. Talisman, like any graphics technology, works by making compromises, the most important of which is layering. Most 3D scenes are rendered as true three-dimensional spaces, with the shading of each element of the scene calculated according to its position and orientation with respect to the rendered point of view. Talisman instead associates particular objects in a scene with particular layers, and then decides how much work needs to go into rendering each layer. So, for example, a layer comprising an object disappearing into the distance needs very little render time at all. Indeed, the effect can be reproduced in a 2D rather than 3D scene by scaling down the 2D image of the object as it recedes.

On the face of it, this is a clever solution, although how smart Talisman-based software will be when it comes to deciding how to handle layers, remains to be seen.

Another compromise is one that sounds rather obvious, even low-tech. It is graphics compression. For various technical reasons to do with the way a scene is calculated, compression is difficult to achieve with conventional renderers. With Talisman, the scene is rendered in blocks 32 x 32 pixels in size (the process carries the unglamorous name of chunking), which can be compressed using the same sorts of techniques used by the JPEG graphics format.

Microsoft says it will not be making Talisman boards, but will license the detailed "reference" design to hardware manufacturers. The company claims that because the design of the silicon is relatively simple and because many of the main components will be standard parts, boards should retail for less than $300. If this is the case, then that really should set the cat among the crossing chickens.

Particles

Summer is about blockbusters, and nowadays blockbusters are about showing off the latest computer graphics effects. Some of the most impressive were to be found in Twister, a movie about tornadoes. In my opinion, the computer-generated tornadoes were the most realistic feature of the whole movie (far more realistic than the characters) and I began to wonder about how they might have been produced. With the help, it turned out, of the resources of Industrial Light and Magic, AliasWavefront, several very pricey plug-ins and about 20,000 lines of customised code.

Having returned home, I tried to brew up a tornado for myself. Naturally I failed (it looked like an upturned tree trunk) but I did manage a smoking chimney (Fig 1).

The key to such effects is a set of 3D tools called particle systems. These are not yet to be found in cheaper 3D packages but they should trickle down into future releases. There are a number you can buy as plug-ins for mid-range programs: for Lightwave, for instance, you can buy products like Particle Storm for about £300.

I used a 3D Studio Release 4 plug-in called "Vapor" to produce the smoking chimney. It is an unexceptional effect but, believe me, it was not easy to create. All particle systems make enormous demands on the processor, not least because being effects that develop over time, they have to be calculated for each frame of an animation. This means that until you render the animation, which can take ages, you cannot really judge whether you have correctly captured the dynamics of your smoke trail or twisting tornado.

The key to all particle effects is a special class of objects called "emitters". These emit a series of smaller objects (the particles) that are generated at a particular rate and disperse in a particular direction, in a particular formation, at a particular speed.

The Vapor plug-in comes with a series of presets for producing different types of smoke, from a cigarette trail to a steam locomotive's billowing clouds. The latter was not particularly convincing, so I had to fiddle around with the parameters to achieve the effect seen in Fig 1 (which, I hope you will appreciate, looks a lot better when animated). Each change to the size and intensity of such parameters (the "whorl" and "turbulence") produced rather unpredictable results, so it took a good few goes, and hours of rendering time, to tune the effect. It just goes to show that there is no smoke, and no tornado, without toil.

PCW Contacts

Benjamin Woolley, writer and broadcaster, can be contacted at **woolley@illumin.co.uk**. His home page is **www.illumin.co.uk/woolley/**