2005-01-01

Using the Crytek game engine in the Dublin Institute of Technology

Hugh McAtamney
Dublin Institute of Technology, hugh.mcatamney@dit.ie

Brendan O'Shea
Dublin Institute of Technology, brendan.oshea@comp.dit.ie

Fredrick Mtenzi
Dublin Institute of Technology

Recommended Citation

McAtamney, Hugh and O'Shea, Brendan and Mtenzi, Fredrick Japhet: Using the Crytek game engine in the Dublin Institute of Technology. 7th International Conference on Computer Games: AI, Animation, Mobile, Educational & Serious Games,CNBDI at Magelis, Angoulême, France, 28-30th. November, 2005.

Using the Crytek Game Engine in the Dublin Institute of Technology

Hugh McAtamney, Bryan Duggan,
Dr. Fredrick Japhet Mtenzi
School of Computing,
Dublin Institute of Technology,
Kevin St., Dublin 8.

Email: { hmcatamney, bryan.duggan,
fred.mtenzi}@comp.dit.ie
Web: http://www.comp.dit.ie/hmcatamney

Abstract

It is envisaged that within the next 10 years. the Dublin Institute of Technology (DIT), Ireland's largest third level university, will move to a new campus in Grangegorman, located in the north inner city of Dublin This site is currently being used as a hospital. It has twelve listed buildings and is located in a densely populated urban community. This paper describes how the Crytek 3D Game engine is being used to create a game *MOD* (modification) of the current hospital site, how it will be used in the construction and public consultation process of the new campus and how students in the DIT are learning computer science in an new way.

1 Introduction

3D models and computer-generated environments are often used in the construction industry to help people visualise what something or someplace might look like in the future. These *walkthroughs* traditionally take the form of a pre-programmed, pre-rendered movie and almost never involve user interaction. An advantage of this type of 3D movie is that it does not require any user experience to operate the movie viewer and thus can be easily distributed. The 3D model can also be rendered to a very high quality. However it does have disadvantages. For example, it does not allow a local resident, to look at the new buildings from their home window or from other arbitrary locations. It does not facilitate the modification of different proposed building designs as each design would have to be rendered individually, thus taking a long time.

3D gaming engines can get around these problems. Modern PC's can now render high quality graphics very quickly, gaming technology is very sophisticated and many tools exist for designers to create complex and intelligent interactive 3D virtual environments. We felt that there was no reason why these tools could not be used in the planning and consultation process. In 2005 DIT chose a 3D gaming environment over a pre-rendered movie to aid the planning and consultation process.

In computer science courses, students usually take subjects, which are important to the development of computer games, for example, 3D graphics programming, Artificial intelligence and maths. These subjects are usually taught as stand alone and separate subjects. By developing this MOD we hope that we can help put some computer science courseware into context.

2. The Crytek Engine

The award winning [Gamespot 2004] first person shooter (FPS) 'FarCry' was released in 2004 and was developed by the German interactive entertainment company Crytek [Crytek 2004]. FarCry is unique in that its game levels feature wide expansive outdoor terrains with luscious foliage [figure 1]. Many similar FPS games [Half life 2, Doom 3 2004] base their levels around physical objects and buildings with terrain and foliage being used for aesthetics [figure 2] This suited us, as Grangegorman is a large open area with many buildings and lots of foliage.

It also ships with a relatively robust 'all in one' level editor, called *Sandbox* [Crytek 2004]. It is more common for these type of level editing tools to be downloaded separately through community forums. It is also unusual to have all the level editing requirements such as terrain generation, object database, AI scripting, physics and deployment tools packaged into one program. It was for these reasons that we chose to use FarCry and Sandbox.

Figure 1 – FarCry

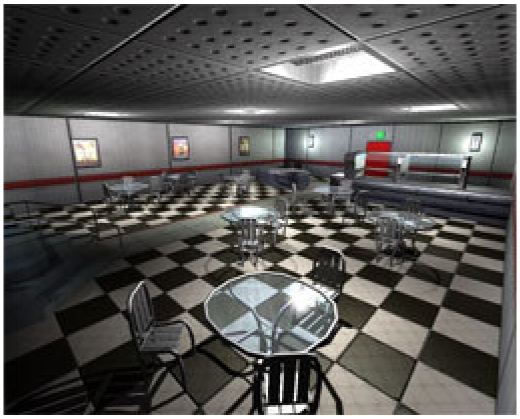

Figure 2 – Doom 3

This will also prove useful for teaching students game programming.

One of the most powerful features of the Crytek engine is that it is completely modifiable. What this means is that every aspect of the gameplay [Rollins, Adams 2004] is scripted using the widely used open source scripting language LUA [Lua 2004]. Technically speaking, it is possible to create games of a different genre using sandbox. By modifying scripts we can control how the enemy AI responds to actions, triggers and events. It also allows us to create interactive movie scenes and new characters. This was attractive to us in the context of the Grangegorman project as it opens up the scope for us to provide information based MODs.

3. Building Grangegorman

There were a number of considerations that were made before the project started. Firstly, it was important that accurate ordinance survey data was collected in order that architects and engineers could build accurate future plans. This information had to be made available in such a way that it could be easily transferred into the gaming system. Furthermore, it was decided that all game objects contained within the model be made to industry standard design formats so that if an alternative game engine or other delivery platform became available we could easily migrate content.

In order to create an accurate model of the entire Grange Gorman site, the first task was to model the terrain topology as accurately as possible. If the terrain was not accurately modelled, there would be no chance of modelling the scene to scale. Estimating the topology from terrestrial photography was considered but it was realised early on that such an approach would not provide the required accuracy.

The approach taken was to capture aerial imagery of the Grange Gorman area. This imagery would then be used to generate a DTM (Digital Terrain Map) of the area. The aerial images were registered using Intergraph's Digital Mensuration software. Control points of the area were captured in order to register the images against a global coordinate system so that the absolute position of any point the scene could be determined relative to this coordinate system.

The Imagestation DTM software was then used to capture elevation points throughout the area. This was performed manually since we wanted the DTM to represent the ground level and did not want to model the building rooftops or the tree tops of wooded areas, of which there are many of in the area.

Once the DTM of the scene was created, a contour map was generated which identified areas of equal elevation. This contour map was then used to generate a height map, which could be used to generate the terrain in the FarCry engine.
[Figure 3 and 4]. [Hegarty 05].

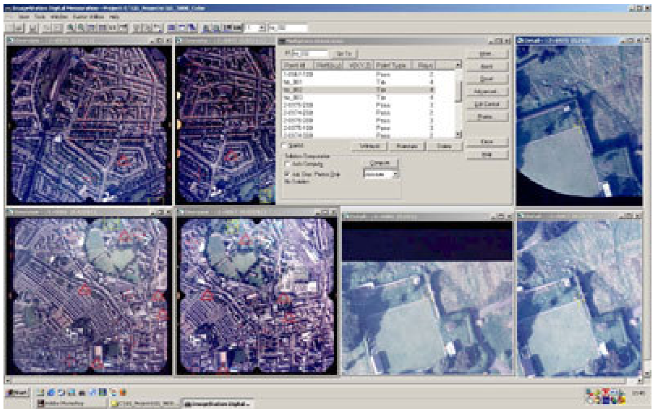

Figure 3 – Taking control points

At this accuracy, architects and engineers are able to construct accurate designs. From our perspective we needed to create a *height map* and import this into the game to create our terrain model. A height map is a greyscale picture, which is used by gaming engines to generate terrains. The whiter the colour the higher the terrain rendered. FarCry can only render terrain at 1meter accuracy so this produced a 'stepping' effect when we imported our map. However smoothing tools were applied to counteract this.

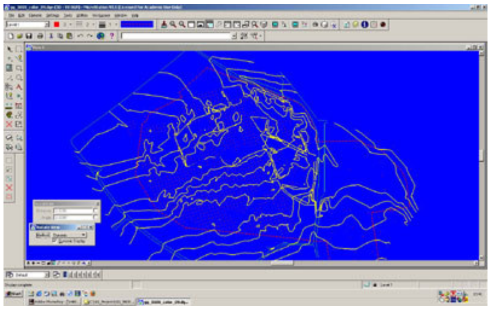

Figure 4 - Generating a contour map

Using a combination of aerial photographs and masks we were able to manipulate the terrain-texturing feature of Sandbox to build road, grass and tree layers [Figure 5 and Figure 6]

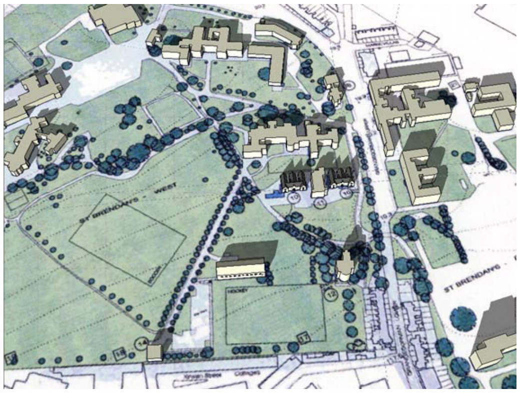

Figure 5 – Generating layers

The next step was to survey the 12 listed buildings. As there were no architects' plans for these, we did our own survey. We used a combination of shadow heights taken from the arial photographs and some 'on the ground' tape measurements.

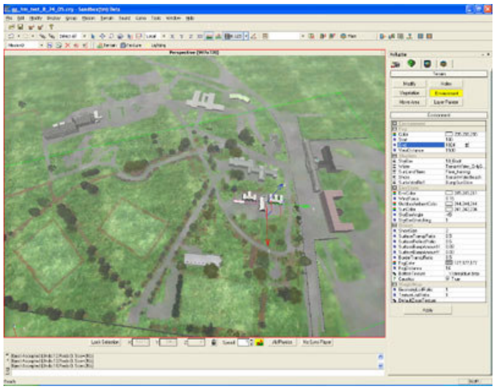

Figure 6 – Generating Layers

Using a program called Sketchup [Sketchup 2004] we create low polygon 3D models. These 3D models were then textured using photographs to give a photorealistic effect [Figure 7]

The photographs (textures) and 3D models were then filtered though a number of processes to make them renderable in the gaming engine.

The textures were transformed into *mipmaps*, which are a type of image format used to provide different levels of detail depending on the viewers distance from the object [Planet Quake 2005]. The 3D models needed to have Crytek specific shaders applied to them in 3D studio max [discreet 2004] via a custom plug-in tool [Crytek 2004].

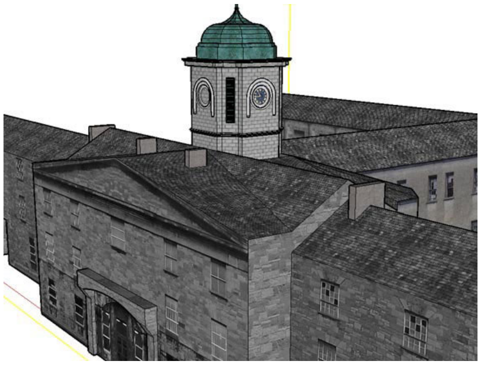

Figure 7 – Photorealistic textures applied

Once this process had been defined and refined it was relatively trivial to add new buildings to the Grangegorman site.

Indeed we were asked to build more buildings once the initial model was presented. This was a reflection on how successful the engine was in conveying a good impression of the Grangegorman site [Figure 8 and 9]

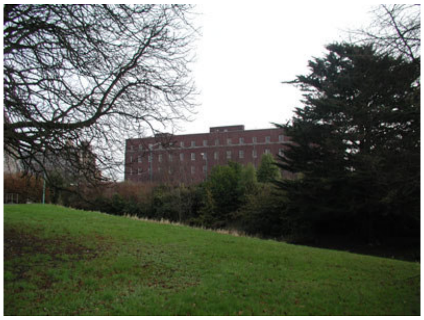

Figure 8 – actual photograph

Another powerful feature of the sandbox tool is the ability to drag and drop buildings, trees and other entities onto the map, change the lighting and sun direction and create a rough rendering with shadows in a matter of minutes. The ability to do this greatly enhanced the workflow and presentation of the project

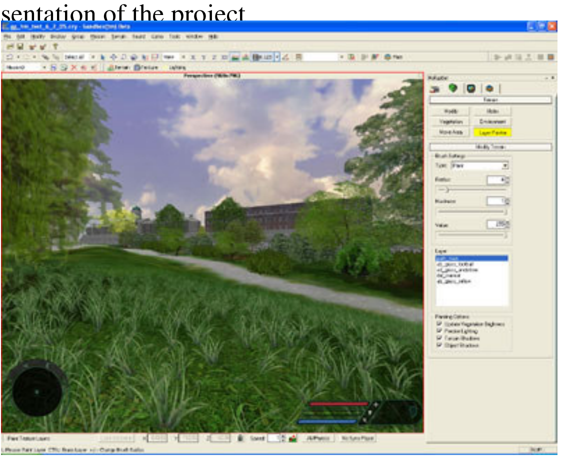

Figure 9 –in game screen shot of the same spot

The 3D model is now almost complete. The next stage is to add interactive elements, which are all programmable through LUA. These will include:

- Creation of an intelligent non-player character within the 3D environment.
- Ability to compare new and old buildings side by side.
- Computer science projects.

Many of the interactive features will form the basis for student projects. It is hoped to be able to incorporate these projects into the 3D environment as a sort of showcase for student work.

4. Using the Game in Education

This project is a follow on from previous work done by DIT in 3D and Virtual Reality [McAtamney 2004]

Computer science students in the games programming elective that is debuting in DIT school of computing this year [Duggan, McAtamney & Mtenzi 2005] are using the Crytek engine to learn game programming principles. This will be achieved through the use of the LUA scripting language to explore and demonstrate the concepts of gameplay and game balance. Students will also learn how to apply the principles of Artificial Intelligence in a gaming context

This elective was set up in response to a government report on industry needs [FAS 04]

FarCry also allows game designers to access C++ functions through LUA. Early in the year students will learn how to create a game engine from the bottom up using C++ and direct X. [Duggan, McAtamney & Mtenzi 2005]. This will have relevance later on the year when they combine these functions with the game scripting.

Already there have been a number of benefits to come out of this project.

The elective itself is the most popular elective and has proven to be effective in improving student retention.

What is also interesting is the fact that the creation of computer game requires good knowledge of AI, 3D programming, physics and Maths. Students must also need to know how to script triggers and game events. Essentially each of these subject areas can be put into a different context that students can relate to.

5. Conclusions and recommendations

We are looking at expanding the gaming elective from one to two or even three years if appropriate.

Online multiplayer games and mobile gaming are on the increase and this approach to learning could prove to be useful in other subject areas such as network programming and JAVA.

Computer games could also be used to illustrate some of the basic computing principles such as triggers, events and loops to first year students. Furthermore, complex technology contained in some of the more recent games could be applied to completely different subject areas such as biology or chemistry in the form of virtual laboratories and simulations.

Following a recent presentation to a large construction firm, interest has also been shown, in some of these ideas, by the schools of engineering and architecture. We are also currently looking at developing a 3D health and safety-training tool using the same technology.

References

[Gamespot 2004] : http://www.gamespot.com/pc/action/farcry/index.html?q=far+cry , accessed September 2005

[Crytek 2004] : http://www.crytek.com , accessed September 2005

[Half life 2, Doom 3 2004] : http://www.doom3.com/, http://www.planethalflife.com/, accessed September 2005

[Luna, 2003] F. D. Luna: *Introduction to 3D Game Programming with DirectX 9.0*. Wordware Publishing, Inc.; 1st edition, June 2, 2003

[Rollins, Adams 2004] A.Rollins and E.Adams: *on Game Design* New Riders, May 2003

[Lua 2004] : http://www.lua.org, accessed September 2005

[Hegarty 2005] : J Hegarty: *Masters Thesis*

[Sketchup 2004] : http://www.sketchup.com, accessed

September 2005

[Discreet 2004]: http://www.discreet.com, accessed September 2005

[Duggan , McAtamney & Mtenzi 2005] : B.Duggan, H.McAtamney, F.J.Mtenzi, *Learning Games Programming with "Dalek World"*, September 2005

[McAtamney 2004]: H.McAtamney *If you build it, will they come ?,* Masters Thesis, Dublin Institute of Technology 2004

[Fas, 2004*]: Skills Requirements of the Digital Content Industry in Ireland: Phase 1*, December 2004, FAS