REAL TIME VISUALIZING TOOLS: DESCRIPTION OF AN EXPERIENCE WHERE GAME ENGINES AND INTERNET BROWSERS ARE USED AS MEANS OF EXHIBITION AT AN ARCHAEOLOGICAL SITE

Rocio Ruiz-Rodarte
Instituto Tecnológico de Estudios Superiores de Monterrey,
campus Edo. de México
Department of Architecture
Carretera Lago de Guadalupe Km. 3.5, Atizapán de Zaragoza,
Estado de México, C.P. 52926
caruiz@itesm.mx

Fernando García Cuspinera
Universidad Autónoma Metropolitana
Department of Architecture
Av. San Pablo No. 180, Col. Reynosa Tamaulipas,
C.P. 02200, México, D.F.
fgcaeam@mexico.com

José Larios
Universidad Nacional Autónoma de México
Engineering School
Anexo Ingenieria, Ciudad Universitaria
joselarios@lycos.com

Abstract

This paper presents the use of Virtual Reality (VR) environments as didactic means of exhibition. It approaches VR field by non traditional tools of visualization -as game engines- and describes the process and particular issues faced during its development. The examples shown in this document are part of an archaeological project under current development which faced budget solutions for equipment and its maintenance. The lack of experience outside the traditional VR field (SGI), lead the research team to try different game engines and VRML format in explore solutions to issues as performance, complexity, appearance, visualization, different needs of programming skills, file format compatibility, archiving and promotion applications. The experience delivered to replicate the process for other cultural projects can be useful also to architects and designers that are willing to experiment with VR as means of visualization during different stages of their projects.

1. Introduction

A VR reproduction of the ancient Maya city of Calakmul (Mexico) was created to show visitors to their ruins how that place originally looked. A virtual mock-up will provide visitors a unique view of the archaeological site since the real buildings, within a protected biosphere, are partially hidden by trees up to 100 feet height.

The project is part of a virtual exhibition currently under development which will operate free as an on-site regular service. The tool is needed to preserve the trees and help visitors to get a previous knowledge of the urban distribution. No special budget for equipment, maintenance or personnel is given, so all proposed deliverables should work over regular PCs.

See Figure 1.

2. Three Dimensional Modeling Process

The digital survey developed for the project was done from accurate archaeological data. It was done at the same time that the findings were made and were used also as archaeological reports. The accuracy needed was met with AutoCAD software using three dimensional solid models' method.

Absolute coordinates were observed even when modeling individual buildings. This means that every building is placed in their exact position, orientation and scale in reference to an absolute origin for all files. Topographic measurements and GPS data were acquired to place every building in their exact position.

Observing absolute coordinates in every individual file, allows a simpler handling of the archives, its weight and

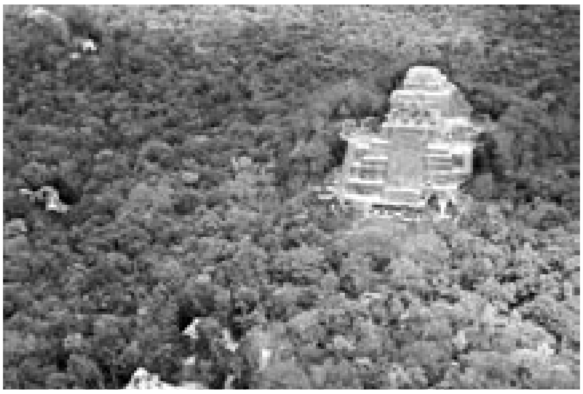


Figure 1: Aerial view of Calakmul, Mexico

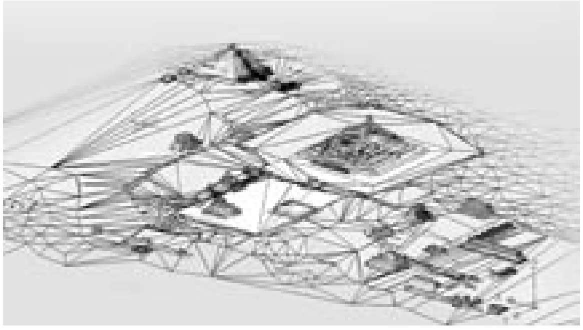


Figure 2: AutoCAD survey of Calakmul

eventual upgrades. When VR final files are composed, each imported structure will take its exact position.

The construction of models was made by layers dividing items by the final color they will have at the VR application. All items sharing a common finishing or material were placed in a specific layer. This meant a layer for red walls, another for white walls, a layer for floors and stair steps, and so on.

The addition of trees and bushes was made directly over the game level editor or by calling isolated tree-files by code. Only few bushes and trees were added in 3DMax files when still renders were needed from small sections of the scene.

2.1. Texturing requirements

To apply texture or color to the models, we use "Standard" materials within 3DMax software. "Multi sub-object" materials do not work with any of the visualization tools proposed in this paper.

Splitting layers by final materials from the source AutoCAD files, obeys to a texturing criteria. When files are imported into 3DMax each layer is identified as an individual item. During the mapping procedure when a material is assigned to each item, all surfaces (e.g. walls) in a same layer take automatically the same material; all floors another one and so on.

The format of the images used to map each surface of buildings and billboards, depend on the final program planned as visualization tool. Internet browsers need .jpg images while Unreal game engine need .pcx or .tif. Nebula game engine reads only bicubical images (whose dimensions are in base 2).

3. Game Engines

For several years universities have tried game engines to make their own VR scenes and provide them with artificial intelligence behaviors (Hooley 2004). In 1999 the project VRND Virtual Reality Notre Dame made with Unreal Tournament, became a fortunate example where game engines mixed entertainment and cultural contents. Examples as the PC game Aztec by France Telecom Multimedia have successfully worked in joint teams with cultural authorities. All mentioned cases have successfully avoided violence and substitute it with cultural contents but, still, there are isolated examples.

One of the main problems to achieve more meaningful VR examples deals with the 3D construction of the environments. Accurate surveys, efficient modeling and the ability to fake some geometries with textures, demand a lot of work. Other problems are the lack of affordable visualization tools and the time consuming tests that require non traditional tools.

The project described in this paper use Unreal, Unreal

Tournament (UT) and Nebula game engines.

3.1. Nebula

The Nebula version of Virtual Calakmul is a day light variety.

Nebula level editor accepts Maya™ Wavefront OBJ files but accurate architectural source files will hardly come from it. This responds more to the 3D modeling of organic shapes and human characters used as avatars within the game and to the well-known animation features provided by Maya ™. This project uses same AutoCAD source files mapped within 3DMax software and then exported to Nebula with a third-party plug-in converter obtained from Nebula's homepage. Only modeling and texturing restrictions previously mentioned were observed. Though, there is a shaders plug-in, Nebula 3DS Max Toolkit which appears in the Map Browser of the Material command and can be used to apply more realistic textures to the environment, specially designed for Nebula's engine.

The possibility of using C++ language with the open source provided with Nebula, gave the project a unique opportunity to build a useful application. Since C++ is a well known programming language, it is easy to find ungraduate students with enough skills who are willing to provide virtual environments with interactivity. This project was useful for two theses of this kind.

It was designed for people that are not very confident to interact with computers. Instead of walking around the environment by moving a mouse or joystick, the user only needs to point over a sensitive map to select a building and the environment takes him/her automatically towards the selected point. It works with two computers connected, one shows an interactive map (computer A) while the other displays an immerse view of Virtual Calakmul (computer B), this is, a view that represent the player eye level standing on the floor in front of the buildings. This view is usually called "camera view" or just "camera" (see figure3).

To run the environment is a free navigating version, the user only needs to forget about the interactive map in computer A and wander by the environment in computer B.

Many issues were solved during the design of this VR application. Instead of using artificial intelligence to move the camera-view from one point to another one, nurbs paths were placed from each structure to all of the rest. This trick moves the camera-view from one structure to any other emulating a smooth walkthrough between them but since the view was directed to the structure in front of it, when moving towards the next building, the camera stays facing backwards instead of looking straight ahead during the walkthrough. The solution meant some extra C++ code: the camera was order to point towards the center of the building while steady and was programmed to keep the path direction while it was in movement. As soon as the camera arrives to the end of the path, it was programmed to change once again to focus the target structure.

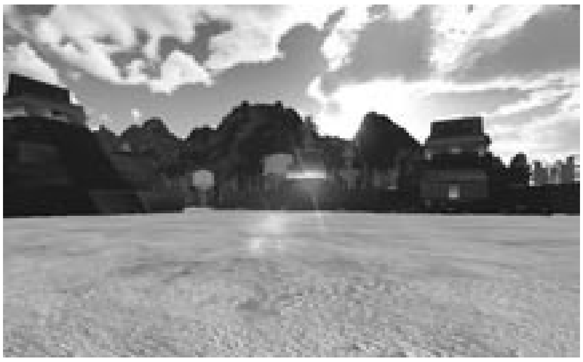


Figure 3: Calakmul's Nebula version

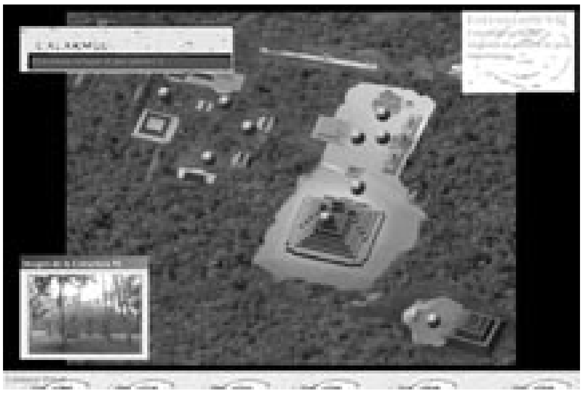


Figure 4: Interactive map

3.2. Unreal II

Unreal's environment is design for a player who is willing to wander all over the place. It presents a nocturnal version of Calakmul. Fire torches all over the landscape provide realism to the scene adding a warm sense of being inhabited. Torches, however, are not enough to illuminate the total scene so during their walk it was difficult for users to appreciate the architecture details of the buildings. Lights were added on specific points in order to emphasize architectural volumes and stairs but the environment was too large and that action caused huge performance troubles. The addition of lights in any 3D modeler software increases the complexity of the rendering because each source of lightning obliges the software to calculate shadows and brightness. Unreal II presented the same issue. To keep the appearance, the environment size was limited to the two most important core yards of Calakmul.

As well as torches, moving water and grasses are part of Unreal II realistic features available for any developer and that is why this game engine becomes a splendid tool for work teams without programmers. Adding these features allowed use to show part of the biosphere attractive and emphasize its preservation aims.

To create this VR application 3DMax files were saved (exported) in .ase format. As previously explained, the individual layers made in AutoCAD are recognized as separate items inside 3DMax. Those items should be exported consecutively to the Unreal level editor. Unreal arranges geometries in packages called Static Meshes. Each item saved as *.ase and imported to the Unreal level editor, takes is own position according to its absolute coordinate reference.

To make available all texture materials, another package is made inside the editor. In this package, called Maps, all images previously applied as textures within 3DMax, are imported. Unreal read only certain image formats. Most common ones are .pcx for regular images and .tif format for those that have an alpha channel to provide transparent areas. This means that all texture images should be translated to .pcx format before importing them into Unreal editor. The editor only reads the name of the file so, even though the mapping procedure has been done with other image format (e.g. brick_1.jpg); as soon as he has the package of textures in their right file format it applies the correspond image (e.g. brick_1.pcx).

A peculiar thing about Unreal is its character height. In order to keep the player eye level to a real proportion, the source files should be enlarged 65 to 1.

Unreal experience can fulfill all the expectation of every programmer and non programmer developer. Though, there is a huge disadvantage in its use: Unreal level editor 3.0 works for Unreal II, Unreal Tournament 2003 and Unreal Tournament 2004, but files made within an engine cannot be opened in another version. Since lights, shadows, special effects and vegetations are added inside Unreal's editor, too much work should be repeated to transfer the environment.

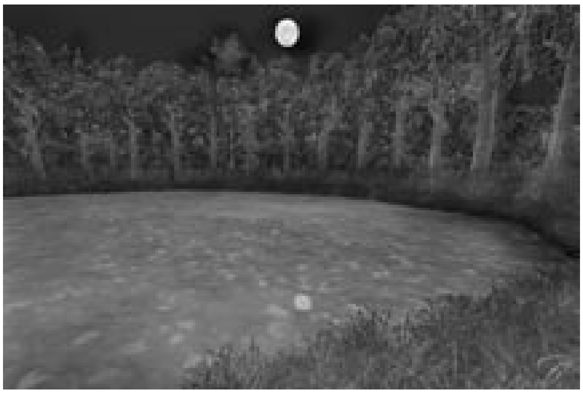


Figure 5: Water appearance in Unreal II

4. VRML environments for Internet Speed.

When modeling 3D objects, it is a common procedure to check repeatedly if materials are looking fine and if they are well applied (orientation, scaling, tiling). Renders only give a fixed view of the model and delay considerably, so a practical task is to take a look of the scene in a visualizing tool. An Internet browser can be turned into a splendid visualizing tool that every computer has already installed without any extra charge.

As HTML, VRML Virtual Reality Modeling Language is a standard language which describes 3D scenes to

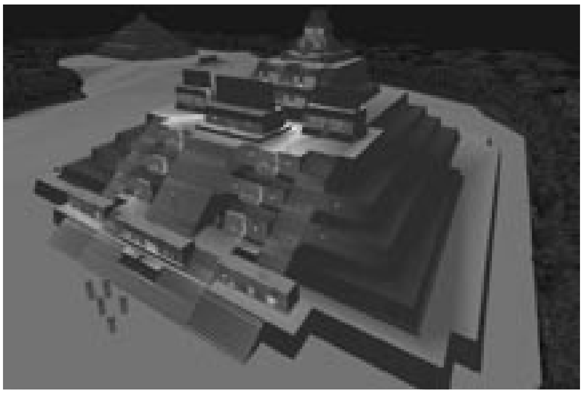


Figure 6: Calakmul's Unreal version

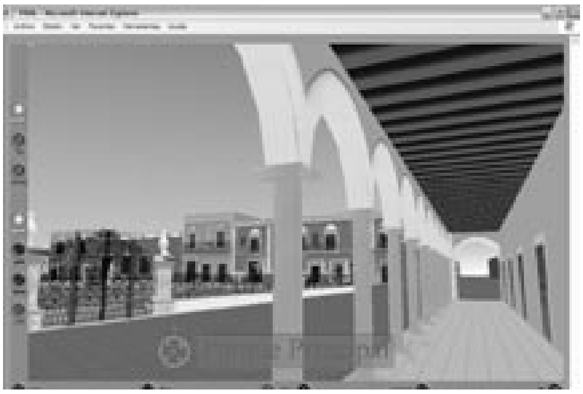


Figure 7: View of Virtual Campeche

make them readable in all Internet browsers. By opening the VRML file with the Internet browser a 3D model is displayed with its final look (materials) and it can also be examined or navigated.

VRML is a "born free" file format created to avoid the incompatibility between software and platforms which eventually could turn the 2D interface of the Internet into a 3D world. Because it is free, most modeling software offer it among their native saving formats. To display VRML files on the Internet browser a free plug-in is needed. Pre-established views can be acquired by adding cameras to the 3D model. By pressing Page Down key, the scene experiment interpolations between cameras, emulating smooth roams over the environment.

A companion application of Calakmul's project is Virtual Campeche which came out as a need of promotion by Internet (see figure 7). Campeche City is the capital of the State to whom Calakmul belongs and where the nearest international airport is located. It is a joint effort between the ITESM in Mexico and the Czech Technical University in Prague (Zara et al. 2004).

To optimize the performance of the real time rendering by any Internet connection, this environment has special features: First, common 3D geometries usually done for modeling scenes were substitute with texture details. Most buildings are plain cubes mapped with photographs from the real building they represent. The work in this case deals more with photograph retouching than with the model construction.

Secondly, the complete environment was divided in sections that are downloaded as they are needed depending on the direction that the user takes while walking through the scene. Each separate section is called by the inline VRML node as the user approaches to certain points. This feature gives the web site a quick response without the need of proprietary software.

Since VRML was created for the Internet, scenes could be nested in complete websites with information making this kind of environment a unique didactic means.

5. Conclusions

This paper describes Virtual Reality applications using game engines and Internet browsers as real time visualizing tools where costly equipment is avoided. They serve as virtual mock-ups to present big scenes in an immersive way and could be applied for meaningful didactic devices at museum exhibitions and Internet websites.

It has been explained certain considerations during the modeling and the texturing phases to ensure that same architectural files could remain compatible during the development of VR environments and reduce any extra work during the process.

The examples describe additional features that require scripting to improve interaction with a user for work teams that have strong programming skills. It is sensible to learn about the possibilities of each visualization tool although work teams are not interested in implementing scripting at the moment.

Eventually, easier and affordable tools for non

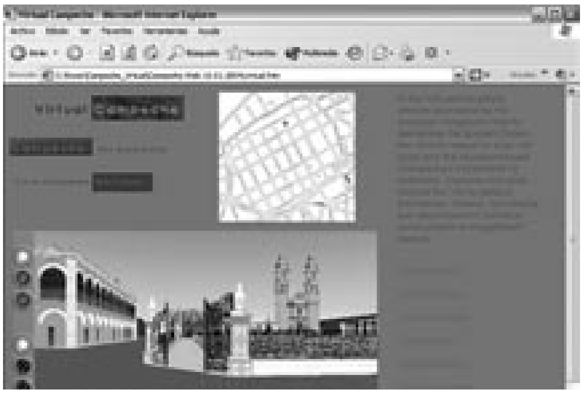


Figure 8: VRML inside a Website

programmer developers as designers and architects will be delivered by some PC games enterprises.

Acknowledgements

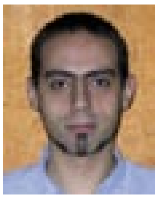
This project is partially supported the Ambassador's Fund for Culture Preservation from the US Government and by the National Institution of Anthropology and History (INAH) in Mexico:
National Coordination of Museums, INAH
National Coordination of Diffusion, INAH
INAH-Campeche Center
Archaeological Project of Calakmul, INAH.

References

Hooley, T., Hunking, B., Henry, M., Inoue, A. (2004).``Generation of Emotional Behavior for Non-Player Characters: Development of EmoBot for Quake II , San Jose, CA. ‘’ Proceedings of the 19th National Conference on Artificial Intelligence (AAAI04), pp.954-955.

Zara, J. Benes, B., Ruiz-Rodarte, R. (2004). Virtual Campeche: A Web Based Virtual Three Dimensional Tour. Colima: IEEE, Encuentro Internacional de Ciencias de la Computación. ENC-2004.

Rocio Ruiz-Rodarte *Industrial Designer, Phd*
Areas of interest: virtual reality, augmented reality, virtual archaeology, heritage, virtual museography, VRML



Fernando Garcia-Cuspinera *Architect, Graduate student*
Areas of interest: virtual reality, game engines, virtual archaeology

José Larios *Computer Engineer, Graduate student*
Areas of interest: programming, computer graphics, game engines, real-time rendering