

Plataforma de Desarrollos 3D Distribuidos (P3D)

Lucas E. Guaycochea, Javier E. Luiso, María V. Galán, and Horacio A. Abbate

Instituto de Investigaciones Científicas y Técnicas para la Defensa (CITEDEF)
lguaycochea,jluiso,mgalan,habbate@citedef.gob.ar

Resumen Presentamos un proyecto para la consolidación y ampliación de una plataforma de desarrollo que actualmente da soporte a la implementación de simuladores de entrenamiento distribuidos. Esta plataforma ha permitido el desarrollo de varios simuladores en funcionamiento en distintas instalaciones del Ejército Argentino. Un proyecto que permite convertir dicha plataforma en un producto robusto, confiable y de fácil uso permitirá la reproducción de proyectos de producción de simuladores o aplicaciones 3D interactivas.

Keywords: simulación, entrenamiento, sistemas distribuidos, computación gráfica, motores gráficos

1. Introducción

En un gran número de disciplinas el factor humano es uno de los más importantes y su entrenamiento para las tareas pertinentes constituye un área de gran relevancia. El avance tecnológico ha permitido que la simulación sea un recurso cada vez más utilizado en el entrenamiento, mejorando su impacto y calidad.

Se han desarrollado simuladores de entrenamiento donde se recrean ambientes virtuales donde se lleva a cabo dicho entrenamiento. Este tipo de simuladores son empleados en diversas áreas como el campo militar, la aeronáutica, la minería, la navegación, etc. En la actualidad, no sólo interesa el entrenamiento individual, sino también, el entrenamiento de varios individuos interactuando en forma colaborativa en una misma realidad simulada.

La experiencia adquirida en el desarrollo de varios simuladores ha permitido obtener una solución de software reutilizable que se encarga de la representación gráfica y visualización de la escena virtual. Esta solución fue extraída como factor común de las necesidades de los distintos simuladores y fue desarrollada como un módulo independiente. Inicialmente la solución consistió en un servidor gráfico únicamente con funcionalidades de renderizado de escenas. Luego, la misma ha evolucionado en la forma de una plataforma que fue adquiriendo más funcionalidades y resolviendo distintos aspectos comunes en el desarrollo de aplicaciones 3D distribuidas.

En la actualidad se cuenta con una versión madura y estable que es de uso interno del equipo que la desarrolló. Con el proyecto que se describe a lo largo del

presente artículo se pretenderá no sólo mejorar y robustecer dicha plataforma, sino también, darle una forma de producto que pueda ser utilizado por distintos equipos de trabajos en sus proyectos particulares.

2. Situación - Problema u Oportunidad

La división Computación Gráfica y Visualización¹ ha desarrollado el simulador SIMOA² (simulador para observadores adelantados) en sus versiones I (año 2001), II (año 2004), III (año 2013). Actualmente están instalados y en funcionamiento en el Colegio Militar de la Nación (CMN) en Palomar, en la Escuela de las Armas (EDA) en Campo de Mayo, y 11 Grupos de Artillería a lo largo todo el país. Este grupo de trabajo también desarrolló el software de la primera versión del simulador de combate para tripulaciones del tanque TAM³, NeoNahuel (2012) (Fig. 1), que está a la espera de la finalización de la construcción de las cabinas para proceder a su instalación.



Figura 1. Visión del usuario conductor en un ejercicio del simulador NeoNahuel.

Todos estos productos descansan sobre una versión básica de un motor gráfico. Este motor gráfico es un módulo de software que permite crear una representación virtual de escenarios reales donde tienen lugar las diferentes situaciones a simular. Desde sus comienzos, la división ha tenido la visión de desarrollar una herramienta con la funcionalidad mencionada en un módulo con servicios genéricos que puedan ser reutilizados en cada simulador o producto que desarrollaría.

Esta herramienta ha ido evolucionando en el contexto de diferentes proyectos cuyos objetivos eran obtener un simulador específico. Esta situación implicó que su desarrollo nunca fue el objetivo principal de los proyectos, sino que el esfuerzo empleado en el mismo haya sido secundario buscando cubrir las necesidades gráficas de cada simulador específico. Como consecuencia, la incorporación de nuevas funcionalidades siempre fue detrás del desarrollo de un simulador particular. De esta manera, la plataforma cubre actualmente las expectativas visuales

¹ Perteneciente al Departamento de Guiado y Simulación, del Instituto de Investigaciones Científicas y Técnicas para la Defensa (CITEDEF).

² SIMOA: Simulador de entrenamiento para Observadores Adelantados, usada para la instrucción y entrenamiento en el Arma de Artillería del Ejército Argentino.

³ TAM: Tanque Argentino Mediano

y de performance de mínima, es decir, renderiza satisfactoriamente una escena con una cantidad moderada de objetos 3D opacos y/o transparentes. Este módulo necesita una dedicación exclusiva para obtener escenarios más realistas y más poblados, trabajando en técnicas de optimización e implementando efectos visuales como sombras, entre otras.

A pesar del contexto presentado, el motor ha alcanzado una versión madura y estable, en un estado avanzado de depuración. Como se dijo, las funcionalidades que ofrece han permitido crear escenarios virtuales con una calidad aceptable para los simuladores de entrenamiento desarrollados. Además, en los últimos años se le ha incorporado soporte para desarrollar simuladores sobre un ambiente distribuido (equipos de trabajo interconectados en una red local interviniendo en el mismo escenario virtual)[1].

Finalmente, los antecedentes descriptos, el estado actual y la tendencia futura en el campo de los simuladores presentan una situación de oportunidad que el grupo de trabajo está en condiciones de abordar. El proyecto busca como objetivo transformar la herramienta existente en un producto de software estable, confiable y cerrado. Un producto capaz de dar soporte y facilitar el desarrollo de simuladores futuros de nuestra División, y que también pueda utilizarse en proyectos ajenos a la misma. En síntesis, se desea obtener un producto equivalente a los motores gráficos comerciales.

3. Solución

Este proyecto se propone cumplir con dos objetivos simultáneos, la obtención de un producto robusto, confiable y bien documentado, y la implementación de nuevos algoritmos para lograr resultados visuales cada vez más realistas así como también dotarlo de la capacidad de trabajar con escenarios extensos y de alto nivel de detalle. El primer objetivo busca que pueda ser utilizado en forma satisfactoria en la generación de una amplia gama de simuladores que el Ejército Argentino necesita para su entrenamiento así como también por distintos grupos de trabajo en el ámbito del desarrollo de simuladores y aplicaciones con representaciones 3D virtuales. El segundo objetivo tiene por fin mantener un producto actualizado y de características y cualidades equivalentes a otros productos de este tipo que pueden encontrarse en el mercado. Es decir, incorporar algunas de las últimas tecnologías disponibles a la plataforma, para dotarla de mayor potencialidad y mantenerla actualizada respecto del avance tecnológico.

3.1. Etapa de consolidación: Un producto robusto

En principio se realizará una exhausta depuración y prueba de la plataforma en el estado (funcional y operativa) que se encuentra en la actualidad para resolver los bugs del sistema que ocasionalmente alteran el correcto desarrollo de las simulaciones. Asimismo, se pondrá foco en generar documentación de la plataforma con una descripción de la API, tutoriales y también proveer ejemplos sencillos. Por último, se desarrollarán herramientas para facilitar la interacción

de los usuarios con la plataforma, tanto usuarios desarrolladores como animadores y artistas, proveyendo una herramienta de conversión de modelos 3D de los formatos estándar al formato optimizado que es utilizado por la plataforma, una herramienta que permita la visualización rápida de los modelos 3D transformados, un editor de escenarios para el armado de escenas sobre la plataforma de manera interactiva y visual siguiendo el concepto WYSIWYG (*what you see is what you get*).⁴

3.2. Etapa de Mejoras: visuales y de performance

Se refiere a la implementación de algoritmos y soluciones que permitan dotar a la plataforma de efectos visuales más ricos y realistas, así como una mayor capacidad en cuanto a performance y tamaño de los escenarios que pueda manejar con fluidez y resultados visuales de calidad (Fig. 2). Entre las técnicas que se pretenden implementar se destacan al menos las siguientes:



Figura 2. Escenario renderizado con la plataforma P3D. La cantidad de objetos presentes se encuentra en el límite de la capacidad de la plataforma. La incorporación de sombras brindaría una imagen más realista.

Microsoft: DirectX 11®. Microsoft: DirectX 11® es la última versión de la plataforma gráfica de Microsoft. DirectX es una colección de APIs (Interfaz

⁴ Existen herramientas comerciales conocidas como 3D Studio o Blender que permiten armar no sólo objetos individuales sino también escenas, pero suelen existir diferencias entre el renderizado logrado por estas aplicaciones y la P3D.

de Programación de Aplicaciones) creadas para facilitar las complejas tareas relacionadas con multimedia, especialmente programación de juegos y vídeo en la plataforma Microsoft Windows desde 1995. Específicamente hablando sobre gráficos 3D, DirectX 11 incorpora avances en varios aspectos, destacándose en el procesamiento paralelo y la utilización de múltiples núcleos. DirectX 11 podrá administrar los tiempos de ejecución y las cargas de las aplicaciones de forma paralela y completamente asíncrona, aprovechando así los diferentes núcleos que la mayoría de los procesadores en cualquier PC tienen hoy disponibles. Además, la capacidad de multi-threads que incorpora DirectX 11 permite procesamiento en segundo plano que se traducirá en mayor velocidad en las aplicaciones que utilizan esta tecnología. Otro aspecto de las nuevas capacidades de DirectX 11 se encuentra en el proceso denominando Tessellation. Cuando se modela un objeto 3D, la superficie del mismo siempre tendrá cierto grado de imprecisión, pero con la aplicación del proceso de Tessellation, DirectX 11 puede lograr aspectos más suaves y realistas. Este proceso también permite incorporar los algoritmos y técnicas de renderizado con distintos niveles de detalles (LOD) en función de a qué distancia se esté observando el objeto en cuestión. Todo esto permite alcanzar representaciones gráficas con un grado de realismo muy importante[2].

En la actualidad la plataforma utiliza la versión 9 de DirectX la cual ya fue superada ampliamente en cuanto a performance, funcionalidad y paradigma por la versión antes mencionada. La arquitectura de lo desarrollado ha utilizado conceptos y patrones de abstracción de la programación orientada a objetos por lo cual la porción de la misma atada a la versión de la biblioteca gráfica utilizada está perfectamente aislada y puede implementarse la misma funcionalidad en la nueva versión sin que afecte al resto del sistema. Por último, recientemente se encuentra disponible la versión 12 de DirectX con el lanzamiento de Windows 10. Se estudiará y evaluará la estabilidad de la misma y se considerará la posibilidad de que la actualización sea directamente con esta última versión.

Referido a las tecnologías, es de destacar que la plataforma ha sido desarrollada sobre plataforma Windows debido a los requerimientos de los simuladores realizados. De la misma manera que la dependencia a la biblioteca gráfica, la plataforma está desarrollada en ANSI C++, y las porciones dependientes del Sistema Operativo (como el uso de threads) se encuentra también perfectamente aislada. Por lo tanto, en este proyecto también se evaluará la posibilidad de generar una versión para plataforma UNIX y la utilización de OpenGL en su versión más avanzada.

Técnica de Nivel de Detalle. Aquellos objetos que se encuentren más lejos del observador en una escena tridimensional pueden representarse con un menor detalle ya que se verán más pequeños. En Computación Gráfica, la técnica conocida como Nivel de Detalle (LOD, Level of Detail) involucra disminuir la complejidad de la representación geométrica de un objeto tridimensional en función de la distancia a la cual es observado, lo cual es linealmente proporcional con el tamaño relativo que el mismo tendrá en la pantalla. Con el mismo objetivo, se puede considerar la velocidad con la cual el objeto se mueve en la porción de

la escena visualizada y así disminuir también el nivel de detalle de la geometría con la cual se representa. La aplicación de esta técnica en los motores gráficos incrementa significativamente la eficiencia de los mismos y disminuye la carga de trabajo en las sucesivas etapas que conforman el Pipeline Gráfico. A pesar que originalmente esta técnica surge mediante la disponibilidad de más de una representación geométrica de un mismo objeto, recientemente han aparecido variantes que mediante el uso de los Shaders permiten aplicar esta técnica a nivel de pixel en las etapas de texturado.[3][4][5]

Framework de Efectos - Shaders. Hasta el momento la plataforma renderiza haciendo uso del pipeline gráfico fijo. Existe la posibilidad de customizar el renderizado de distintos tipos de objeto a través de un lindeo dinámico de un programa en distintos niveles de procesamiento del pipeline. Estos programas son llamados Shaders, y pueden intervenir en la etapa de transformaciones geométricas de los vértices (vertex shaders), como en la etapa de definir el color de cada píxel (pixel shader), entre otras que se han incorporado en los últimos años. Estos programas de alto nivel serán ejecutados por procesadores dedicados que poseen las tarjetas gráficas modernas o GPU (graphics processing units).

La plataforma a través del uso de shaders ya implementa los efectos de lluvia y niebla. Estos efectos son apenas algunos de los deseables, para obtener un mejor renderizado fotorealista de la escena se deben incorporar efectos como sombras y explosiones, humo, polvaredas, etc. Para una correcta y flexible implementación de estos efectos a partir del uso de shaders, que permita la escalabilidad y manteniendo la robustez, se plantea realizar un framework para el uso de shaders ya en la última versión de la biblioteca gráfica seleccionada.[6][7]

Framework de Representación de terrenos de grandes extensiones. Un desafío con los que los motores gráficos y de juegos deben lidiar particularmente es con la representación detallada de grandes extensiones de terrenos debido al volumen de datos que deben administrarse en tiempo real[8]. Esto consiste tanto en la administración dinámica del nivel de detalle (LOD) de la geometría con la que se renderiza la porción del terreno visualizada, como el manejo de las texturas con las que se colorean las superficies. En los simuladores actuales se utilizan texturas que parten de la imagen satelital de la región del país donde ocurren los ejercicios de entrenamiento que son mejoradas manualmente por un diseñador gráfico para dotarles de un mejor aspecto al ser utilizadas en un simulador. Para facilitar la tarea del diseñador referente al detalle que puede obtenerse mediante el manejo de texturas, se pretenden estudiar y aplicar técnicas como Perlin Noise, Bump Mapping y Alpha Blending.

También, se estudiarán e implementarán algoritmos y técnicas para la representación y reproducción de vegetaciones como pasto y árboles para obtener un mayorrealismo considerando que éste es un tema que también debe encararse de manera particular ya que son un tipo de objeto particular de la escena.

4. Innovación e Inédito

En el mercado mundial existen productos comerciales enlatados de estas características propiedad de empresas extranjeras, algunos son utilizados en el desarrollo de videojuegos de PC o consola y otros en el desarrollo de simuladores de entrenamiento.

En nuestro país no conocemos la existencia de un desarrollo nacional similar. A partir de este proyecto se adquirirán las capacidades para consolidar el desarrollo de esta plataforma logrando un producto de calidad. El desarrollo de este proyecto pretende realizarse siguiendo los lineamientos de la serie de estándares internacionales, normas ISO 25000, llamadas Requisitos y Evaluación de Calidad de Productos de Software (SQuaRE)[9]. De esta manera, se podrán desarrollar simuladores de entrenamiento militares o civiles homologables que se ajusten a los estándares de calidad que correspondan.

Con la culminación del proyecto, el grupo de trabajo buscará ofrecer el producto y de esta manera generar vínculos con otras instituciones y grupos de trabajo del ámbito nacional. Esto no sólo permitirá la creación de nuevos proyectos de desarrollo de simuladores o aplicaciones interactivas con representación de escenarios 3D virtuales, sino también la posibilidad de obtener un feedback de la utilización del producto. Esta retroalimentación, permitirá tomar las sugerencias y pedidos de mejoras y/o nuevas funcionalidades de los grupos vinculados para generar futuras versiones de un mejor producto.

5. Beneficiarios

Este proyecto permite identificar impactos científicos y tecnológicos, impactos operativos e impactos socio-económicos.

Desde el punto de vista científico y tecnológico, el proyecto permitirá ampliar el conocimiento de los actuales y futuros integrantes, tanto en el área de la computación gráfica y visualización, como en el soporte dado para el desarrollo de productos como los simuladores y otras eventuales aplicaciones 3D interactivas. Además el proyecto dará lugar a publicar los resultados obtenidos en congresos y publicaciones nacionales y/o internacionales.

En cuanto a los impactos operativos, el producto implica posibilidades de su uso y aplicación para el desarrollo de cualquier simulador o aplicación interactiva que requiera representaciones realistas de escenarios creados virtualmente. El trabajo propuesto sobre el producto implicará la mejora de la calidad de dichos simuladores proveyendo a los usuarios de una mayor inmersión en la situación simulada. Además, el producto facilitará el desarrollo de nuevos simuladores, acortando los tiempos de los proyectos relacionados.

En referencia al aspecto socio-económico, la plataforma redunda en beneficios para el desarrollo de simuladores. Los simuladores permiten un mejor entrenamiento de procedimientos y tácticas con una considerable disminución de costos, sin dañar el medio ambiente y permitiendo entrenar situaciones que no podrían realizarse en la realidad sin riesgo para personas y elementos.

Por último, con el producto propuesto es factible poder obtener beneficios de su venta, licencias de uso, mantenimiento y soporte a usuarios.

6. Viabilidad Técnica, Financiera y Política Organizacional

La División Computación Gráfica y Visualización de CITEDEF cuenta con los recursos materiales y los recursos humanos para llevar adelante este proyecto.

Además, se pretende obtener financiación para la contratación, en forma de becarios u otras, de estudiantes avanzados de las carreras de informática o computación de las universidades del país. La formación de estos estudiantes en materias como Computación Gráfica o similares facilita su incorporación en un proyecto en esta temática.

Este proyecto se alinea con una de las líneas estratégicas del Ministerio de Defensa de los últimos años, la simulación. Es reconocida la necesidad y las virtudes económicas y prácticas del uso de simuladores para el entrenamiento del personal de las fuerzas armadas. Este alineamiento y concordancia de objetivos garantiza la viabilidad política y organizacional del mismo.

7. Facilidad de Reproducción

Es un objetivo del proyecto disponer de un producto de software robusto y usable por distintos grupos de trabajo. Esto permitirá que puedan formarse nuevos equipos de trabajo que puedan iniciar proyectos de desarrollo de simuladores o de aplicaciones 3D interactivas. Estos proyectos contarán con una gran parte de la solución resuelta, la visualización de una escena 3D en un entorno distribuido, que les permitirá la obtención del producto final en menores tiempos y de una manera más sencilla. Esto es, no necesitarán perfiles técnicos en el área de la computación gráfica o de los sistemas distribuidos ya que la plataforma resuelve de manera transparente estos aspectos.

8. Ambiente de Hardware y Software

La Plataforma de Desarrollos 3D Distribuida permite modelar mundos virtuales, llevando a cabo tanto su evolución en el tiempo como su visualización o renderizado. Al ser una biblioteca, puede ser utilizada por una aplicación que interactúe con un jugador y administre las tareas específicas de la situación simulada. La aplicación usuaria puede crear escenarios tridimensionales instalando objetos, configurando condiciones de iluminación, estableciendo cámaras desde donde se visualizará la realidad simulada. También, puede instalar trayectorias o movimientos a los objetos, cámaras o luces. Además, su propiedad distribuida, permite desarrollar un simulador o una aplicación distribuida, donde distintos usuarios pueden interactuar en la misma realidad simulada desde distintas computadoras.

La P3D, consta de 3 capas de software (Fig. 3) [10]. La capa inferior se corresponde al motor gráfico. La capa intermedia o Capa de Distribución es la responsable del manejo de la comunicación y la distribución. Por último, la capa superior o API de la plataforma reproduce las funcionalidades brindadas por el motor gráfico agregando la posibilidad de crear contextos de simulaciones distribuidas.

El motor gráfico es el encargado de la representación de escenarios virtuales, llevando a cabo la evolución en el tiempo y la visualización o renderizado.

La Capa de Distribución permite que sea usada exitosamente para el desarrollo de simuladores de entrenamiento distribuidos. Un simulador distribuido requiere brindar a distintos usuarios en distintas computadoras la posibilidad de interactuar en la misma realidad simulada. Esta capa se encarga de resolver la comunicación entre las distintas computadoras para lograr ese objetivo. Para esto, utiliza un middleware[1] para abstraer los aspectos concernientes a la distribución, encapsulando las comunicaciones y la sincronización. De esta manera, las aplicaciones presentes en cada computadora pueden trabajar sobre el mismo escenario virtual sin tener que preocuparse por la distribución de las acciones que ocurren en cada una de ellas.

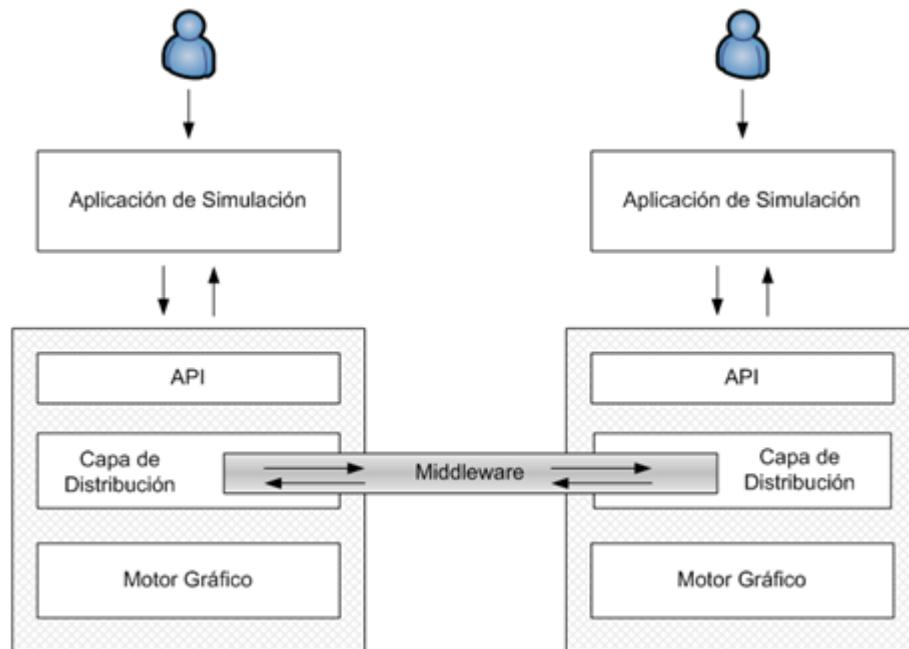


Figura 3. Arquitectura de la Plataforma de Desarrollos 3D Distribuida

9. Conclusiones

El objetivo de este proyecto es consolidar y ampliar la plataforma existente de modo de poder satisfacer las necesidades planteadas en el desarrollo de aplicaciones 3D. Esta es una herramienta de software fundamental en las aplicaciones que involucran la interacción de uno o más usuarios inmersos en un escenario virtual 3D con un alto grado de realismo. Similar a un motor de juegos, permite modelar escenas tridimensionales resolviendo una visualización foto-realista y simular los movimientos de objetos, cámaras y luces dentro de la misma.

Con la incorporación de los nuevos desarrollos la Plataforma quedará preparada para soportar particularmente el modelado de escenarios de exteriores con amplias extensiones de terreno natural. Se incorporarán capacidades que permitan representar de una manera eficiente y realista el avance de la simulación en esas condiciones con resultados gráficos más ricos. Las mejoras implicarán que se contará con sombras, efectos climáticos, efectos que den sensación de viento en el ambiente y mejores versiones de humos, fuegos, polvaredas y otros componentes que otorgan realismo en las situaciones de combate que se simulan.

Este proyecto permite además la formación de los integrantes actuales y futuros en el área de la computación gráfica, permitiéndoles investigar y estudiar el estado del arte de los algoritmos y técnicas actuales, con el objetivo de implementarlas en la plataforma P3D que tiene un uso concreto en productos como simuladores y teniendo la posibilidad de contribuir a dicho estado del arte. Este objetivo implica que es fundamental realizar un permanente trabajo de actualización de los desarrollos para mantener la plataforma actualizada con el constante avance tecnológico que se experimenta actualmente en el área.

Por último, el desarrollo de la plataforma permite no sólo a los participantes desarrollar el *know-how* y formarse, sino también la independencia de productos de terceros cuyas licencias son costosas y no se posee facultad para introducir o modificar a medida funcionalidades deseables al momento que sean necesarias.

Referencias

1. Guaycochea, L., Luiso, J. and Del Rio Garcia, F.: Middleware P2P para la Sincronización de Eventos Discretos en una Simulación Distribuida de Sistemas que evolucionan en el tiempo. XVIII Congreso Argentino de Ciencias de la Computación, CACIC 2012. Bahía Blanca, Argentina, (2012)
2. Direct3D 11 Features. MSDN. [https://msdn.microsoft.com/en-us/library/windows/desktop/ff476342\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/ff476342(v=vs.85).aspx)
3. Clark, James H.: Hierarchical Geometric Models for Visible Surface Algorithms Communications of the ACM, October 1976 Volume 19 Number 10. Pages 547-554.
4. de Boer, W.H.: Fast Terrain Rendering using Geometrical Mipmapping in flipCode featured articles, October 2000
5. Clasen, M., and Hege, H.: Terrain Rendering Using Spherical Clipmaps. Eurographics ¡EuroVis, 2006
6. Kilgariff, E., and Fernando, R.: The GeForce 6 series GPU architecture. ACM SIGGRAPH 2005 Courses on - SIGGRAPH '05, 2005.

7. Luebke, D., and Humphreys, G.: How GPUs Work. Computer. 2007;40(2):96j100
8. Guaycochea, L., Abbate, H.: Error-Bounded Terrain Rendering Approach based on Geometry Clipmaps. IX Workshop Computación Gráfica, Imágenes y Visualización (WCGIV), XVII Congreso Argentino de Ciencias de la Computación (CACIC), 2011
9. ISO/IEC 25000, SQUARE. iso25000.com
10. Luiso, J., Guaycochea, L., Abbate, H.: Simuladores de entrenamiento distribuidos: Plataforma de desarrollo para ocultar los aspectos de la distribución. 1er Workshop Argentino sobre VideoJuegos, WAVI 2010, 2010