



**UNIVERSIDAD
NACIONAL
AUTÓNOMA
DE
MÉXICO**



**FACULTAD DE INGENIERÍA
DIVISIÓN DE INGENIERÍA ELÉCTRICA**

**RECONSTRUCCIÓN VIRTUAL DE LA ZONA
ARQUELÓGICA DEL TEMPLO MAYOR**

**T E S I S
QUE PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACIÓN**

**P R E S E N T A N :
URIEL VLADIMIR MARTÍNEZ FLORES
MANUEL SANGINÉS HUITRÓN**

**A S E S O R D E T E S I S :
ING. JOSÉ FRANCISCO SALGADO RODRIGUEZ**

CIUDAD UNIVERSITARIA, SEPTIEMBRE 2007



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

ÍNDICE

ÍNDICE

AGRADECIMIENTOS	III
INTRODUCCIÓN.....	1
CAPÍTULO I ANTECEDENTES.....	5
CAPÍTULO II MARCO TEÓRICO.....	14
CAPÍTULO III MARCO TECNOLÓGICO.....	25
CAPÍTULO IV INGENIERÍA DE SOFTWARE.....	51
CAPÍTULO V ANÁLISIS Y DISEÑO.....	60
CAPÍTULO VI DESARROLLO.....	94
CAPÍTULO VII PRUEBAS Y MANTENIMIENTO.....	141
CAPÍTULO VIII MANUAL DE USUARIO.....	145
CAPÍTULO IX CONCLUSIONES.....	155
BIBLIOGRAFÍA.....	157

Introducción

El siglo XX será recordado por tres grandes innovaciones: medios sin precedentes para salvar, prolongar y mejorar la vida, medios sin precedentes para destruirla (hasta el punto de poner por vez primera en peligro nuestra civilización global) y conocimientos sin precedentes sobre nuestra propia naturaleza y la del universo.

Carl Sagan

Mientras creas e intentas encarnar un proyecto, es muy importante para ti pensar en los recuerdos de tus experiencias.

Shigeru Miyamoto

Nuestro objetivo es desarrollar una reconstrucción computacional gráfica de la Zona Arqueológica del Templo Mayor, que permita su visualización completa como lo era en la época prehispánica, así como interactuar con el ambiente mediante un recorrido virtual y un juego. Utilizando herramientas de desarrollo tridimensional.

Se plantea el aprovechamiento de recientes herramientas comerciales en el campo de la computación gráfica, para ser aplicadas en la reconstrucción de un sitio histórico nacional que ayude a visualizar al público en general, como era el sitio que hoy ocupa la capital de México, permitiéndoles realizar un recorrido por las diversas construcciones del recinto sagrado.

Al contar con computadoras más veloces, de menor costo, memoria RAM de mayor capacidad y la miniaturización de los componentes, junto a los avances del software gráfico han hecho aparecer ante nuestros ojos, mundos que solo en nuestros sueños e imaginación hubiésemos encontrado, mundos totalmente artificiales, así como sitios reales desaparecidos y otros de difícil acceso. Y la tecnología que hace esto posible es la Realidad Virtual (RV)¹.

Esta expresión “Realidad Virtual” cada vez es más común, esto gracias a sus múltiples aplicaciones que van desde el apoyo a la educación e investigación científica en todas sus ramas, hasta la industria del entretenimiento como el cine y la televisión, sin embargo, en donde ha tenido un mayor realce en los últimos años es en la tan menospreciada industria del “entretenimiento interactivo” (EI)² o mejor conocida con el nombre de industria de los videojuegos.

Pero donde adquiere una mayor importancia el desarrollo de esta tecnología es en la educación. Pero ¿De qué forma puede ayudar la RV a la educación? Las formas son innumerables, las cuales van desde una mejor comprensión de las ciencias, en especial de las ciencias exactas, visualizando conceptos abstractos que en el pizarrón lo son aún más, situaciones en diferentes escalas y lugares peligrosos como el estar en el interior de un volcán o transportarse a lo largo del cuerpo humano en un submarino miniatura, esto quiere decir que podemos estar en cualquier lugar y sobre todo en cualquier sitio ubicado en cualquier momento de la historia de la humanidad, como en una máquina del tiempo que nos ayuda a visualizar el pasado y porque no, el futuro.

Con las capacidades de esta tecnología los estudiantes lograrán un mejor dominio de los conocimientos adquiridos y así aplicarlos, lo cual es el objetivo primordial del aprendizaje humano. Por esta razón múltiples instituciones educativas ponen en marcha

¹ A partir de este capítulo nos referiremos a la Realidad Virtual por las siglas RV.

² A partir de este capítulo nos referiremos a Entretenimiento Interactivo por las siglas EI

programas para la investigación y desarrollo de aplicaciones de RV. Pero esto no solo se hace exclusivamente para la educación superior, se tienen también actividades programadas para la educación primaria y secundaria, etapas más importantes según algunos expertos, ya que son las etapas del ser humano en las que se forma el carácter y el interés hacia actividades científicas.

En esta época el medio de RV más utilizado es el de la industria del EI, esto debido a que cada vez son más las personas que utilizan este medio no solo para jugar, si no para crear toda una comunidad y cultura a su alrededor, incluyendo medios inmersivos para una mayor interacción a través de dispositivos cada vez más elaborados.

¿Y cuál es la fórmula del éxito de este medio?, la respuesta es sencilla, le da al cliente lo que pide, ya que tenemos múltiples formas de interactuar que van desde manejar vehículos que sería imposible tener, hasta artefactos que solo estarían en una película de ciencia ficción, además la cantidad de lugares por visitar son tan innumerables como la imaginación humana lo permita; como factor extra, el juego en línea permite el poder estar y al mismo tiempo no estar, se oye bastante raro pero es posible, este tipo de experiencia permite interactuar con cualquier persona sin importar el lugar.

El tiempo en que se subestimaba a este medio clasificándolo como un simple pasatiempo terminó, en la actualidad es una forma de vida y es algo en lo que la ingeniería en computación ha logrado avances impresionantes; en cuestión de ventas es el software más vendido en el mundo, aún por encima de los sistemas operativos y otros tipos de software.

Pero esto no es solo por los alcances económicos; ha ayudado a que la sociedad se acostumbre a cosas que aún no son posibles de conseguir en la actualidad, por ejemplo el funcionamiento de un sistema de localización geográfica, lo cual es muy común en los juegos donde se demuestran los objetivos en un mapa y así sea más fácil llegar a ellos y cumplir con el objetivo.

Esta tecnología no es exclusiva de países del primer mundo, en nuestro país las instituciones educativas tanto públicas como privadas, cuentan con programas de desarrollo de software interactivo que van desde recorridos virtuales hasta la simulación del manejo de vehículos motorizados. Cabe destacar que la Universidad Nacional Autónoma de México a través de la Dirección General de Servicios de Cómputo Académico (DGSCA) cuenta con una sala para el desarrollo y visualización de ambientes virtuales totalmente en tercera dimensión, mediante el uso de lentes estereoscópicos activos, esta es la sala Ixtli. También en el Universum se cuenta con “La Ventana de Euclides” un espacio diseñado para crear ambientes virtuales.

Además en nuestro país también se cuenta con concursos y eventos que permiten conocer a gente desarrolladora de este tipo de tecnología, es decir no es exclusivo de las universidades, estos laboratorios independientes trabajan en el desarrollo de paseos virtuales, creación de juegos de video, animaciones en tercera dimensión, así como a la aplicación de efectos especiales en el cine. Para lo cual es necesario que las futuras generaciones de profesionales del cómputo se desarrollen en el uso de técnicas de software interactivo, y no solo eso, que estas técnicas sean mejoradas para así contar

con ambientes cada vez más reales, y en los que la interacción sea cada vez mejor, lo cual haría que el costo de trasladarse de un lugar a otro se reduzca.

En el caso de los paseos virtuales como el que realizaremos en este trabajo de tesis, también sirve para llegar a lugares que por motivos históricos y de espacio-tiempo ya no es posible visitar, en este caso se trata de la zona arqueológica del Templo Mayor, situada en el corazón del imperio Mexica, símbolo de la gloria de nuestros ancestros indígenas y también símbolo de su caída bajo la mano de nuestros también ancestros españoles.

Con los alcances antes mencionados no solo se pueden reconstruir los edificios y paisajes, si no también a las personas que vivían ahí e interactuar con todos estos elementos. Queremos lograr que las personas que interactúen con este software comprendan mejor nuestro pasado y que puedan vivir la experiencia de retroceder en el tiempo, para aprender más sobre nuestra cultura.

Otro atractivo con el que cuenta este proyecto es la incorporación de un juego de recolección de piezas contra reloj, es algo que va ayudar a que el software tenga un mayor interés sobre este lugar, sus secretos y lo que aún falta por descubrir.

Capítulo I

Antecedentes

I. 1 Realidad Virtual

¿Qué es la Realidad Virtual?

El definir RV, o su término en inglés Virtual Reality (VR) es algo complicado. Tenemos tantas definiciones como desarrolladores en un área, que crece y evoluciona a un ritmo tan acelerado que no ha permitido tener una sola definición.

Desglosando el concepto en cada una de las palabras que lo conforman, Realidad y Virtual, también resultan ser conflictivas a la hora de buscarles una definición única y exacta. No cabe duda que la realidad resulta ser altamente relativa para los diferentes puntos de vista que se tienen, y que el término virtual se refiere a algo que tiene existencia aparente y no real. Debido a esto nos vamos a enfocar en dar más de una definición, donde cada una entrega una idea de lo que se debe entender por RV.

Empezaremos por dar una definición de cada una de las palabras que forman este concepto:

Realidad: Calidad de lo que tiene una existencia real, es decir física y medible. Cosa real. Cosa o hecho que resulte de valor práctico o genere determinadas consecuencias. En filosofía, término que señala la existencia independiente de algo, frente a la apariencia o la ilusión.¹

Virtual: Que tiene virtud para producir un efecto, aunque no lo produce de presente. Frecuentemente en oposición a efectivo o real. Implícito, tácito. Que tiene existencia aparente y no real.²

A partir de la unión de estas dos definiciones surge el concepto de RV a continuación damos varias definiciones que intentan explicarlo.

Realidad Virtual

- Es una simulación por computadora, dinámica y tridimensional, con alto contenido gráfico, acústico y táctil, orientada a la visualización de situaciones y variables complejas, durante la cual el usuario ingresa, a través del uso de sofisticados dispositivos de entrada, a espacios altamente interactivos de origen artificial.
- Es una manera mediante la cual los humanos visualizan, manipulan e interactúan con computadoras y datos extremadamente complejos.
- Es un ambiente altamente interactivo donde el usuario participa a través del uso de la computadora en un mundo virtualmente real. Es una simulación tridimensional por computadora durante la cual el usuario resulta inmerso tan completamente que esta realidad, de origen artificial, aparenta ser real.³

¹ Diccionario Enciclopédico SALVAT, Tomo 22 (pol-red), Barcelona 1986.

² Diccionario Enciclopédico SALVAT, Tomo 26 (u-zwo), Barcelona 1986.

³ <http://www.monografias.com/trabajos11/realitua/realitua.shtml>, 27 de Agosto 2007.

- Consiste en simulaciones tridimensionales interactivas que reproducen ambientes y situaciones reales.
- Es un entorno de tres dimensiones sintetizado por computadora en el que participantes acoplados de forma adecuada pueden atraer y manipular elementos físicos simulados en el entorno y, de alguna manera, relacionarse con las representaciones de otras personas pasadas, presentes o ficticias, o con criaturas inventadas.
- Es el medio que proporciona una visualización participativa en tres dimensiones y la simulación de mundos virtuales, siendo dichos mundos el elemento fundamental de un sistema de RV. La RV es un entorno generado por computadora en el que los participantes pueden entrar físicamente e interactuar con él desplazándose por su interior o modificándolo de cualquier forma. En su forma más simple, un mundo virtual podría estar compuesto por un edificio tridimensional por el que podríamos desplazarnos, aunque sin modificar nada. Sin embargo, con el equipamiento adecuado, los usuarios podrían ver, desplazarse e interactuar a través de estos entornos gráficos generados por computadora.⁴

I.2 Cronología del Entretenimiento Interactivo

A través de los años, el avance de la tecnología ha permitido elaborar sistemas de cómputo cada vez más sofisticados, los cuales han permitido el desarrollo del EI. En esta cronología se toman en cuenta a los principales actores dentro de la industria, en un principio fue el arcade (maquinitas), después a las consolas caseras, y por último los juegos de computadora en cuanto a popularidad, aunque todo el software es desarrollado en una PC, misma que cuenta con su propio mercado. A continuación se enlistan los eventos más sobresalientes en la historia de los videojuegos.

1958

William A. Higinbotham, del Brookhaven National Laboratory, laboratorio de investigación nuclear en Upton, New York, crea un tenis en osciloscopio para entretener a los visitantes en una feria de ciencias.

1961

Estudiantes del MIT crean Spacewars⁵

⁴ Aukstakalnis, The Silicon Mirage, 1992.

⁵ Petri Kuittinen, History of Videogames, Diciembre 1997.

1972

Se crea el Magnavox Odyssey, considerado el primer sistema de videojuegos casero de carácter comercial. Los componentes electrónicos eran todos analógicos, con aproximadamente 40 diodos y transistores. Como los gráficos eran muy primitivos, la consola se vendió con cubiertas de plástico para la pantalla de televisión con el fin de hacer los juegos más interesantes. Los jugadores llevaban la puntuación en papel, pues la máquina no tenía memoria.

Se funda Atari por Nolan Bushnell.

1975

Atari Pong, esta es la versión casera del popular juego de acción Pong. Tuvo un gran impacto cultural y dió origen al florecimiento de los videojuegos. El juego tenía dos controles incorporados y sólo se podía jugar Pong.

Entre 1975 y 1977 se lanzaron varios sistemas Atari Pong diferentes, en los que se incorporaban innovaciones como el color o puntuaciones digitales en la pantalla.

1977

El Atari VCS (VÍdeo Computer System) fue la primera consola de videojuegos de 8 bits de Atari, a la que posteriormente bautizó como la 2600. Este producto revolucionó el mercado de los videojuegos al refinar el concepto de una máquina que utilizaba cartuchos intercambiables. El Atari 2600 estuvo en el mercado hasta 1990, lo que la convirtió en la consola con mayor tiempo en el mercado en toda la historia.

1980

Nintendo nombre en japonés de la compañía que en español significa “en las manos del cielo” es fundada en 1889, como una compañía que se dedicaba a los juegos de cartas, pero que vió en el EI la oportunidad de trascender y explotar el mercado con productos de calidad.

Shigeru Miyamoto, un joven programador de Nintendo creó para arcade, el hoy también clásico Donkey Kong. El protagonista era un personaje llamado en principio Jumpman que tenía que ir saltando barriles hasta llegar a su amada Princesa, secuestrada por un Gorila. A raíz del tremendo éxito cosechado por el juego, Jumpman cambió su nombre por Mario, el cual se convertiría en icono de la industria del EI siendo uno de los personajes más famosos en el mundo de los videojuegos y la mascota de Nintendo desde entonces.⁶

⁶ Petri Kuittinen, History of Videogames, Diciembre 1997.

1985

La consola Nintendo Entertainment System (NES), lanzada en Japón en 1983 como la Famicom (Computadora familiar), marcó la segunda era de los videojuegos y el comienzo de la hegemonía de Nintendo en este mercado. El NES era un sistema de 8 bits con 52 colores y sólo 2kb de RAM. Fue un éxito total y se convirtió en el sistema de videojuegos más vendido desde la Atari 2600. El juego Súper Mario Bros. 3, lanzado en 1988, vendió más de siete millones de copias. La popularidad del NES convirtió a Nintendo en la compañía japonesa más exitosa para 1990.

1991

La entrada de Nintendo en el mundo de los 16 bits llegó de la mano del juego “Súper Mario World”. El Súper NES y Sega compartieron el liderazgo en el mercado de los videojuegos durante los cinco años siguientes, hasta la llegada de los juegos de 32 bits. Esta época de competencia tuvo el efecto de aumentar la selección de juegos disponibles, incluso con productos de terceras partes.

1993

Doom es un videojuego de acción en primera persona creado por ID Software en 1993 por un grupo dirigido por John Romero. Dicho juego funcionaba bajo el sistema operativo DOS, utiliza 4 MB de memoria RAM y requiere al menos un procesador 386 a 33 MHz o 486 a 25 MHz. Este es conocido como el primer FPS (First Person Shooter), el cual es el género mas popular en nuestros días para la PC.

1995

El PlayStation de Sony comenzó su vida como un lector de CD que se incorporaba al Super NES. Nintendo y Sony no consiguieron ponerse de acuerdo sobre la comercialización del nuevo aparato, por lo que esta última decidió desarrollar su propio producto hasta convertirlo en una consola. El PlayStation salió al mercado estadounidense en septiembre de 1995, inaugurando la era de los 32 bits en esa área. Para mayo del 2000 uno de cada cuatro hogares en Estados Unidos tenía un PlayStation.

1996

Nintendo pasó la era de los 32 bits, rechazando las amenazas de las consolas de Sega y Sony mientras trabajaba en un aparato de nueva generación. El N64, lanzado en Estados Unidos en 1996, trasladó los éxitos de la compañía a los 64 bits, enfatizando la sensación de una tercera dimensión. El Hardware para el video del N64 fue producida por Silicon Graphics, la compañía de computadoras que llevó a las pantallas éxitos cinematográficos como “Jurassic Park”.⁷

⁷ Petri Kuittinen, History of Videogames, Diciembre 1997.

1998

The Legend of Zelda: Ocarina of Time es un videojuego de la saga The Legend of Zelda desarrollado por Nintendo para la consola Nintendo 64. Este videojuego catalogado como el mejor de todos los tiempos mezcla elementos de juego de acción y aventuras en tercera persona y juego de rol (RPG).

1999

El Dreamcast fue la primera consola de videojuegos con arquitectura de 128 bits. Vino equipada con hardware que sólo puede encontrarse en una poderosa computadora personal de la época. Contaba con un procesador de 200 MHz un chip de sonido de 64 canales y 26 megas de RAM. Todo este hardware se tradujo en gráficos más veloces, más detallados y con mayor calidad. También se le incorporó un modem de 56K para aprovechar la creciente popularidad de los juegos en línea.

2000

El PS2 fue uno de los productos de juegos más esperados. Esta consola contaba con mejores características de velocidad, mejor calidad de imagen y capacidad para reproducir películas en DVD. El PS2 contiene un compartimiento de expansión con capacidad suficiente para un disco duro de 3.5 pulgadas, así como un adaptador de red para acceder a Internet.

2001

El GameCube es la incursión de Nintendo en la generación de 128 bits. Con un procesador de 405 MHz. Los juegos se incorporaron por primera vez para esta compañía en discos de tres pulgadas en un formato incompatible con los lectores estándar de CD y DVD, en parte para combatir la piratería. Entre sus características figura la introducción de controles inalámbricos y un modem para jugar a través de Internet.

2002

Microsoft entra en la guerra de las consolas con el XBOX la cual incorpora muchos de los componentes de una computadora personal. Tiene un sistema operativo Windows, un procesador de 700 MHz, un disco duro de ocho gigas, 64 megas de memoria, puerto de Ethernet y lector de DVD.⁸

⁸ Petri Kuittinen, History of Videogames, Diciembre 1997.

Actualidad 2007

En la actualidad Microsoft, Nintendo y Sony pelean el mercado tan demandante que tiene esta industria, la más importante en cuanto a software se refiere. Y esto se comprueba con la incursión de estas compañías en dos nuevos campos, el primero dominado por Nintendo en los últimos años, las consolas portátiles con su Nintendo DS y SONY con su PSP (Playstation Portable), y el segundo y uno de los más importantes la interacción del EI entre usuarios de diferentes regiones del mundo a través de la red, es decir, el juego en línea, ampliamente dominado por Microsoft tanto para su consola Xbox 360 como para los usuarios de la PC con sistema operativo Windows.

Esto lo permite el mejoramiento del hardware a través de los años, ya que ahora contamos con verdaderos “monstruos” que pueden realizar prácticamente cualquier tarea gráfica (como el PS3), que anteriormente eran inimaginables. En esta generación contamos con consolas incluso más poderosas (gráficamente) que algunas computadoras personales, o con controles que permiten un mejor nivel de inmersión, como es el caso del Wii de Nintendo, ya que se puede tomar el control y transformarse en un bisturí, una espada, un bat, una raqueta, etc.

I.3 Historia de Tenochtitlan

Tras la caída de la civilización tolteca que había florecido principalmente en Tula entre los siglos X y XI, oleadas de inmigraciones inundaron la meseta central de México, alrededor del lago de Texcoco. Debido a su tardía aparición en el lugar, los aztecas (mexicas) se vieron obligados a ocupar la zona pantanosa situada al oeste del lago. Estaban rodeados por enemigos poderosos que les exigían tributos, y la única tierra seca que ocupaban eran los islotes del lago, rodeados de ciénagas.

El hecho de que, desde una base tan poco esperanzadora, los aztecas fueran capaces de consolidar un imperio poderoso en sólo dos siglos, se debió en parte a su creencia en una leyenda, según la cual fundarían una gran civilización en una zona pantanosa en la que vieran un nopal sobre una roca y sobre él un águila devorando una serpiente. Los sacerdotes afirmaron haber visto todo eso al llegar a esta zona. En 1325 fundaron la ciudad de Tenochtitlán ubicada donde se encuentra la actual ciudad de México, capital del país.

Los aztecas convirtieron el lecho del lago en chinampas, las cuales eran jardines muy fértiles, contruidos con un armazón de troncos que sostenían arena, grava y tierra de siembra, atados con cuerdas de ixtle, para lograr islas artificiales donde se cultivaban verduras, flores y se criaban aves domésticas. Se hicieron calzadas y puentes para conectar la ciudad con tierra firme; se levantaron acueductos y se excavaron canales por toda la ciudad para el transporte de mercancías y personas. Las construcciones religiosas (gigantescas pirámides escalonadas recubiertas de piedra caliza y estuco de vivos colores, sobre las que se construían los templos) dominaban el paisaje.

La ciudad floreció como resultado de su ubicación y del alto grado de organización. A la llegada de los españoles, el gran mercado de Tlatelolco atraía a unas 60,000 personas diarias. Las mercancías llegaban a manos aztecas gracias a los acuerdos sobre tributos establecidos con los territorios conquistados. Muchas de esas mercancías se exportaban a otras zonas del Imperio azteca y a América Central.

Tenochtitlán se dividía en cuatro sectores que simbolizaban las cuatro direcciones del mundo, con el Recinto Ceremonial en el centro como quinta dirección. De su inmensa plaza principal partían cuatro calzadas de las que la occidental, la septentrional y la meridional conducían a ciudades menores, mientras la oriental lo hacía a los almacenes y al lago (hoy en día desecado). Cada sector se dividía en barrios. Aunque cada barrio tenía su propio templo y su divinidad, así como sus edificios administrativos. Los templos más importantes se encontraban dentro del Recinto Ceremonial, que no solo incluía pirámides–santuarios de los dioses más reverenciados, sino el terreno de un tlachtli (juego de pelota), la piedra de los sacrificios (cuauhxicalli), el friso de las calaveras, piscinas para baños ceremoniales, escuelas, un jardín botánico, un zoológico, una biblioteca y viviendas para los sacerdotes.

La estructura de la ciudad se apoyaba en el Recinto Ceremonial, entorno al Templo Mayor, y en las calzadas que iban a Tepeyac, a Iztapalapa, a Tacuba, a Texcoco y al acueducto de Chapultepec. Aun hoy, en el plano de la actual ciudad de México, se pueden apreciar estos grandes ejes de la estructura urbana precolombina.

Por los cronistas españoles de la época de la conquista sabemos que el Recinto Ceremonial agrupaba no menos de 78 edificios, de los que en la actualidad, sólo conocemos escasamente el Templo Mayor, algunas estructuras que se han puesto al descubierto a raíz de diferentes obras efectuadas entre los años de 1978 a 1982, distintas excavaciones practicadas en el subsuelo de la catedral metropolitana y numerosos monumentos escultóricos hallados en la zona desde el siglo XVIII. Aun así, esos escasos restos dan una idea precisa de la importancia y belleza que llegó a alcanzar este centro sagrado, verdadero corazón de la ciudad de Tenochtitlán.

Más allá del Coatepantli (muro decorado con serpientes que rodeaba el Recinto Ceremonial), se encontraba la ciudad, cruzada por multitud de canales, calzadas y callejones. En ella había dos tipos de vivienda: unas poseían chinampas y otras no. Estas últimas se localizaban principalmente en la zona más próxima al Recinto Ceremonial, área residencial de la nobleza, mientras que las viviendas con chinampas se localizaban en la periferia, donde habitaban los campesinos o macehualtin (plural náhuatl de macehualli). En la zona sur del lago se localizaba la llamada zona chinampaneca, es decir, la región en la que se encontraban los cultivos sobre chinampas, con cuyos productos agrícolas se alimentaban la ciudad y el valle circundante.

La mayor parte del limitado territorio urbano que poseía la ciudad estaba ocupado por las viviendas de la nobleza y por las de los plebeyos. Las primeras eran más grandes y estaban construidas con piedras y vigas de madera, frente al adobe utilizado por el pueblo para fabricar las suyas. Las viviendas se agrupaban en torno a patios, siguiendo un modelo cuyos orígenes podrían remontarse a la época de esplendor de la ciudad de Teotihuacán. El sistema administrativo de Tenochtitlán llegó a tener una complicadísima estructura.

Las condiciones geográficas de su emplazamiento obligaba a que los abastecimientos básicos (alimentos, agua potable y materias primas) se realizaran forzosamente desde tierra firme. El mercado era una institución fundamental para la vida de la ciudad. A través de él se obtenía el 40% de los alimentos. Allí se vendían animales vivos, joyas de oro y plata, y también adornos de plumería y objetos de uso corriente, como vasijas, cuchillos de obsidiana, sal, carbón o flores. Durante su celebración había alguaciles

encargados de la vigilancia para evitar fraudes y robos, así como un cuerpo de jueces que tenían jurisdicción sobre nobles y plebeyos. Los servicios de limpieza de la ciudad eran también dignos de ser destacados ya que no menos de un millar de personas estaban encargadas de barrer y regar las calles y, naturalmente, todo el complejo sistema de canales y acueductos debía ser vigilado, limpiado y reparado para que funcionase a la perfección, ya que todos los transportes y comunicaciones dependían de ello.⁹

Tenochtitlan maravilló a los conquistadores pero no los apartó de su objetivo militar, el sitio de Tenochtitlan duró 75 días. Los mexicas defendieron con heroísmo su ciudad. Pero el hambre, la sed y las enfermedades terminaron por vencerlos. El 13 de agosto de 1521, Cuauhtémoc, último señor mexica fue capturado y la ciudad fue saqueada. Así comenzaron tres siglos de dominación española y el territorio conquistado fue nombrado como Nueva España.

Desafortunadamente, este gran recinto desapareció ya que los españoles utilizaron las piedras con las que se encontraban construidas los edificios para levantar la nueva Ciudad de México. Por ello nuestro énfasis en la reconstrucción de este lugar, así podremos conocer la antigua ciudad como en sus días de máximo esplendor, además a través de un videojuego podremos explorar la ciudad en forma más divertida.

⁹ Historia del nombre y de la fundación de México, Gutierre Tibon, Fondo de Cultura Económica, México, 1993.

Capítulo II

Marco Teórico

II.1 Conceptos de RV

Características de un sistema de RV

Para que un sistema de RV sea reconocido como tal es necesario que cumpla con características básicas, las cuales se comprenden con los siguientes conceptos:

- **Punto de referencia o punto de vista:** Es la característica que permite determinar ubicación y posición de observación del usuario dentro del mundo virtual.
- **Percepción:** Esta característica es la que involucra a los sentidos, un sistema de RV ideal debe involucrar los cinco sentidos, es decir, poder oler, ver, gustar, oír y sentir, esto mediante dispositivos sincronizados con la aplicación de RV como lo son los cascos y guantes utilizados en la actualidad.
- **Interacción:** Nos sirve para tener control de la exploración del sistema creado; de no existir sería como estar viendo una simple película o recorrido fijado a priori. Para lograr la interacción existen diversas interfaces hombre-maquina, que van desde teclado y mouse hasta guantes o trajes sensoriales. La interactividad con el mundo virtual supone que el usuario pueda mover objetos, así como moverse en el ambiente para que tales acciones, produzcan cambios en ese mundo artificial.
- **Navegación:** Es la propiedad que permite al usuario cambiar su posición dentro del mundo virtual, siempre y cuando se encuentre dentro de los parámetros permitidos por el sistema de RV.
- **Manipulación:** Es la acción que nos permite manejar o tocar los objetos a través del mundo virtual. Esta manipulación nos permite el control sobre la aplicación, liberando así acciones permitidas dentro del ambiente de RV, que influyen directamente sobre estos objetos.
- **Simulación:** Esta es la característica que nos permite ver que tan real es nuestro mundo artificial, ya que es en donde se establecen las “leyes” o reglas que deberán seguir nuestras acciones.
- **Tridimensionalidad:** debe existir realmente una dimensión de profundidad, similar a la que ofrecen los simuladores de vuelo. Para conseguir este efecto de tridimensionalidad los objetos del mundo virtual, deben tener asociada una tercera dimensión que marque su profundidad en la pantalla.

II.2 Clasificación

Una de las desventajas que tiene la RV es en cuanto a los dispositivos necesarios para cumplir su cometido, es decir el hardware, el cual es bastante variado. En función de estos elementos involucrados, se pueden clasificar en:

- **SISTEMAS DESKTOP**

Se han definido como sistemas de RV sin Inmersión.

Algunos sistemas utilizan un monitor convencional para mostrar el mundo virtual. Estos sistemas son conocidos como WOW (Window on a World) y también como RV de escritorio.

Estos sistemas tratan de hacer que la imagen que aparece en la pantalla luzca real y que los objetos, en ella representada actúen con realismo; dentro de esta clasificación entran los videojuegos.

- **SISTEMAS DE MAPEO POR VIDEO**

Este enfoque se basa en la filmación, mediante cámaras de vídeo, de una o más personas y la incorporación de dichas imágenes a la pantalla de la computadora, donde podrán interactuar en tiempo real con otros usuarios o con imágenes gráficas generadas por la computadora.

De esta forma, las acciones que el usuario realiza en el exterior se reproducen en pantalla permitiéndole desde fuera interactuar con el mundo virtual. El usuario puede, a través de este enfoque, simular su participación en aventuras, deportes e incluso entrenamientos policíacos para situaciones de peligro.

Este tipo de sistemas puede ser considerado como una forma particular de sistema inmersivo.

- **SISTEMAS INMERSIVOS**

Los más perfeccionados sistemas de RV permiten que el usuario pueda sentirse "sumergido" en el interior del mundo virtual.

El fenómeno de inmersión puede experimentarse mediante cuatro modalidades diferentes, dependiendo de la estrategia adoptada para generar esta ilusión. Ellas son:

- a. El operador aislado
- b. La cabina personal
- c. La cabina colectiva
- d. La cueva (cave)

Estos sistemas inmersivos se encuentran generalmente equipados con un casco-visor HMD (Head Mounted Display). Este dispositivo está dotado de un casco o máscara que contiene recursos visuales, en forma de dos pantallas miniaturas coordinadas para producir visión estereoscópica y recursos acústicos de efectos tridimensionales.

Una variante de este enfoque lo constituye el hecho de que no exista casco como tal, sino un visor incorporado en una armadura que libera al usuario del casco, suministrándole una barra que permite subir, bajar o controlar la orientación de la imagen obtenida mediante el visor.

Otra forma interesante de sistemas inmersivos, se basa en el uso de múltiples pantallas de proyección de gran tamaño, dispuestas ortogonalmente entre sí para crear un ambiente tridimensional o caverna, en la cual se ubica a un grupo de usuarios. De estos usuarios, hay uno que asume la tarea de navegación, mientras los demás pueden dedicarse a visualizar los ambientes de RV en tiempo real.

- **SISTEMAS DE TELEPRESENCIA**

Esta tecnología vincula sensores remotos en el mundo real con los sentidos de un operador humano. Los sensores utilizados pueden hallarse instalados en un robot o directamente sobre el usuario. De esta forma el usuario puede operar el equipo como si fuera parte de él.

Esta tecnología posee un futuro extremadamente prometedor. La NASA se propone utilizarla como recurso para la exploración planetaria a distancia.

La telepresencia contempla, obligatoriamente, un grado de inmersión que involucra el uso de control remoto, pero tiene características propias lo suficientemente discernibles como para asignarle una clasificación particular.

El juego en línea del EI, es un ejemplo de esto, ya que se pueden elaborar encuentros mediante personajes personalizados que aparenten a los usuarios, para juegos como deportes, disparos y aventura.

- **SISTEMAS DE REALIDAD MIXTA O AUMENTADA.**

Al fusionar los sistemas de telepresencia y RV obtenemos los denominados sistemas de Realidad Mixta. Aquí las entradas generadas por la computadora se mezclan con entradas de telepresencia y/o la visión de los usuarios del mundo real.

Este tipo de sistema se orienta a la estrategia de realzar las percepciones del operador o usuario con respecto al mundo real. Para lograr esto utiliza un tipo especial de HMD de visión transparente, que se apoya en el uso de una combinadota, la cual es una pantalla especial transparente a la luz, que ingresa proveniente del mundo real, pero que a la vez refleja la luz apuntada a ella mediante los dispositivos ópticos ubicados en el interior del HMD.

En este sentido se percibe un prometedor mercado para los sistemas de Realidad Mixta en industrias y fábricas donde el trabajador debe llevar a cabo operaciones complejas de construcción o mantenimiento de equipos e instrumentos, así como de situaciones de peligro o muy costosas lo cual ayuda a reducir costos y a evitar accidentes.

- **SISTEMAS DE REALIDAD VIRTUAL EN PECERA.**

Este sistema combina un monitor de despliegue estereoscópico utilizando lentes LCD acoplados a un rastreador de cabeza mecánico. El sistema resultante es superior a la simple combinación del sistema estéreo con varias pantallas debido a los efectos de movimientos introducidos por el rastreador.

- **SISTEMAS DE REALIDAD VIRTUAL EN SEGUNDA PERSONA**

Son aquellos sistemas en los cuales el usuario sabe que esta en el mundo virtual porque se ve a sí mismo dentro de la escena. Es decir es un integrante “visible” del mundo virtual, por medio de la proyección de su imagen en un fondo o ambiente, las cuales son visualizadas en la pantalla.

II.3 Objetivos de la RV

- Crear un mundo posible, crearlo con objetos, definir las relaciones entre ellos y la naturaleza de las interacciones entre los mismos.
- Poder presenciar un objeto o estar dentro de él, es decir penetrar en ese mundo que solo existirá en la memoria del observador un corto plazo (mientras lo observe) y en la memoria de la computadora.
- Que varias personas interactúen en entornos que no existen en la realidad, sino que han sido creados para distintos fines.

Hoy en día existen muchas aplicaciones de entornos de RV, en estos entornos el individuo solo debe preocuparse por actuar, ya que el espacio que antes se debía imaginar, es facilitado por medios tecnológicos.

La meta básica de la RV es producir un ambiente que sea indiferenciado a la realidad física. Un simulador comercial de vuelo es un ejemplo, donde se encuentran grupos de personas en un avión y el piloto entra al simulador de la cabina, y se enfrenta a una proyección computadorizada que muestra escenarios virtuales en pleno vuelo, aterrizando o en situaciones de peligro como turbulencias. Para la persona en la cabina, la ilusión es muy completa, y totalmente real, y piensa que realmente está volando un avión.

La RV toma el mundo físico y lo sustituye por entrada y salida de información, tal como la visión, sonido, tacto, etc.

II.4 Problemas actuales de la RV

En términos del estado actual de la tecnología, existe aún un número de importantes problemas por resolver, para garantizar nuestra satisfacción como futuros usuarios a nivel sistemático y no casual. Estos problemas están siendo atacados en la actualidad por numerosos equipos humanos, a nivel técnico y científico. Entre estos problemas tenemos:

- a. Representación
- b. Demora u alentamientos en tiempo de respuesta
- c. Rango de rastreo
- d. Angulo de visualización
- e. Malestar por uso prolongado

A continuación se explican los términos mencionados y el porque de sus inconvenientes:

Representación

En contraste con el mundo verdadero, constituido en su nivel primario por átomos y moléculas, un mundo virtual está constituido por polígonos, que son los bloques básicos constructivos de la computación gráfica. Los polígonos forman "mallas", que sirven para representar objetos y escenarios. Resultan indispensables en la constitución de mundos virtuales.

A mayor número de polígonos en la descripción de un objeto o escenario, más fina será la imagen que percibimos. Por otro lado, a mayor número de polígonos, mayor exigencia a la velocidad de procesamiento necesaria para presentar la imagen en tiempo real. Ha sido estimado que el representar imágenes del mundo real se tiene una exigencia de entre 80 y 100 millones de polígonos por segundo. En comparación las actuales "máquinas de realidad" pueden, cuando mucho, producir de 7.000 a 10.000 polígonos por segundo.

Visto en abstracto, la escala del problema es inmensa. Sin embargo el ser humano posee una muy adaptable capacidad de percepción. De esta forma, por ejemplo, dibujos animados con un mínimo de 500 polígonos por segundo son ampliamente aceptados.

Pero, en el caso de RV, el problema va mucho más allá, ya que esa imagen debe:

1. Poseer tridimensionalidad.
2. Sincronizar los cambios en perspectiva originados por los desplazamientos del usuario, incluyendo la resolución de problemas de visibilidad de múltiples objetos, muchos de los cuales pueden hallarse en movimiento.
3. La imagen requiere para mantener la ilusión de credulidad, de tratamiento mediante sombras y efectos especiales.
4. Existe una información complementaria de sonido, tacto y fuerza.

Demora u alentamientos en tiempo de respuesta

El Alentamiento es la medida de tiempo entre el momento en el cual una persona se mueve y el momento en el que la computadora registra el movimiento.

Los problemas de demora se refieren a la actualización de la imagen a medida que el visitante se desplaza en el ambiente virtual. Generalmente se expresa en número de cuadros por segundo o frames por segundo (fps, frames per second).

En una situación ideal, cuando se gira la cabeza mientras se usa un casco visor u otro dispositivo para visualización, las imágenes no deberían dar saltos. Pero esto resulta difícil de lograr a nivel del avance actual de la tecnología en el área.

Angulo de Visión

Con respecto al ángulo de visión resulta difícil precisar un campo óptimo de visión en RV, lo que en un caso puede resultar adecuado, en otro puede no serlo. Por ejemplo, si se le ofrece un amplio campo de visión a una persona que necesita concentrarse para cumplir una tarea específica, encontraremos que son más los problemas que se le crean

que los beneficios que se derivarán de esta acción, por cuanto un amplio campo de visión pudiera ofrecerle muchas distracciones. En el otro extremo, si se le da un campo muy estrecho de visión a una persona que está buscando alcanzar una percepción global, lo cual sería poco efectivo.

Los investigadores han encontrado que una forma de evitar el vértigo, es limitar el ancho de campo de visión a no más de 60 grados horizontales. Pero, aún cuando esto sea cierto, es casi imposible simular la sensación de inmersión en un campo tan pequeño.

Molestias por el uso prolongado

Bajo circunstancias ordinarias, los sistemas sensoriales del ser humano operan como una pieza de maquinaria cuidadosamente calibrada. Incluso la aparentemente simple tarea de caminar erguido manteniendo un balance, es logrado a través de relaciones precisas entre los diversos músculos y mecanismos sensoriales.

Se estima en diez por ciento de usuarios de RV los afectados por el malestar derivado, del uso prolongado. Esto se debe a la falta de validación entre los sentidos de estas personas y las señales contradictorias que son recibidas por los ojos y el sentido de posición del cuerpo. Este malestar es inducido por los efectos de inmersión en mundos virtuales, cuando los usuarios se encuentran volando, girando, etc. Sus síntomas se asemejan a los experimentados por astronautas cuando entran en caída libre o por pasajeros mareados a bordo de un barco.

En este sentido, se han detectado síntomas de incomodidad y hasta de náusea durante experiencias de RV.

Las sofisticadas relaciones entre los efectos de las simulaciones sobre el cuerpo y sobre las interpretaciones del cerebro, muy posiblemente se constituyan en una rica fuente de interrogantes durante años venideros.

Las investigaciones actuales muestran que la náusea tiende a ocurrir durante la exposición inicial de un usuario a una simulación específica, especialmente cuando existen mucha información visual. Por ejemplo cuando se generan frecuentes movimientos de arranque y detención y frecuentes cambios en la aceleración, el usuario puede experimentar malestares. En este sentido, la adaptación gradual mediante el período de calentamiento, suministra una clave para reducir el malestar inducido en el usuario de RV.

II.5 Entretenimiento Interactivo o Video Juegos.

Un videojuego es un programa informático, creado en un principio para el entretenimiento, basado en la interacción entre una o varias personas, que se puede ejecutar desde un aparato electrónico mejor conocido como consola, que no es más que una computadora con la función de ejecutar este tipo de software, también se puede realizar esta acción desde una computadora personal (PC). Estos recrean entornos y situaciones virtuales en los cuales el jugador puede controlar a uno o varios personajes (o cualquier otro elemento de dicho entorno), para conseguir uno o varios objetivos por medio de reglas determinadas por los programadores. Estos objetivos pueden contar

con un nivel de dificultad para dar oportunidad a cualquier tipo de usuario (ya sea experimentado o principiante) la oportunidad de probar sus habilidades.

La interacción entre los jugadores y el videojuego puede ser individual o múltiple, mediante el uso de una consola estática o a través de internet (online).

Este software como se explicó anteriormente tiene fines lúdicos, pero también mejoran la coordinación visual, estimulación sensorial, ayuda a la memoria, debido a que se utilizan ambas manos para controlar cualquier objeto dentro del software, lo que ayuda a ejercitar ambos hemisferios del cerebro, es decir ayuda a los a los usuarios de este software a desarrollar habilidades de observación y percepción para la solución de problemas.¹

II.5.1 Géneros de videojuegos

A continuación se presentan los diferentes géneros existentes dentro del EI o Videojuegos.

- **Aventura:** Juegos en los que el protagonista debe avanzar en la trama interactuando con diversos personajes y objetos. Ejemplos: God of War, Hevenly Sword, Ninety Nine Nights.
- **Deportivo:** Son videojuegos basados en deportes, ya sean reales o ficticios, y se pueden subdividir en simuladores o los llamados "arcade" (maquinita). Ejemplos: FIFA , NBA Live, NFL Street, Pro Evolution Soccer, Pc Fútbol, Tony Hawks.
- **Disparo:** También conocidos como "Shooters" o "Shoot'em up" (en inglés, dispárales). En estos juegos el protagonista ha de abrirse camino a base de disparos. Según su temática y desarrollo pueden clasificarse en diferentes sub-géneros, siendo los más populares los de acción en primera persona o "FPS" (First Person Shooter). Ejemplos respectivos: Doom, Quake, Hexen, TimeSplitters, WarRock o Counter Strike como FPS; Galaxian, 1942, R-Type, Thunderforce, Halo; y House of the Dead ó Virtua Cop que son juegos que necesitan de un control parecido a una pistola real.
- **Survival Horror.** En este tipo de juegos el protagonista debe seguir la historia, la cual se desarrolla en un ambiente de terror y suspenso, este tipo de juegos se caracterizan por tener la cámara en una perspectiva en tercera persona. Ejemplos: Alone in the Dark, Sillent Hill, Resident Evil, Fatal Frame, Clock Tower.
- **Educativo.** Juegos cuyo objetivo es transmitir al jugador algún tipo de conocimiento, como la realización de operaciones matemáticas o de lógica teniendo como enemigo el tiempo. Su mecánica puede abarcar cualquiera de los otros géneros. Habitualmente son dirigidos para niños de menos de 10 años, aunque hay excepciones como en nuestro caso. Ejemplos: Big Brain Academy, Sudoku, etc.
- **Estrategia.** Se caracterizan por la necesidad de manipular a un numeroso grupo de personajes, objetos o datos para lograr los objetivos. Según su temática los hay de gestión (ya sea económica o social) y bélicos, mientras que por su mecánica pueden ser en tiempo real, también llamados "RTS" (Real Time

¹ Revista Club Nintendo, Año 16, No. 7, Mitos de los videojuegos, 26/07/2007.

Strategy), o por turnos. Ejemplos: Stratego, Dune 2, Starcraft, Age of Empires, Civilization, Empire Earth, Advance Wars, Fire Emblem, Europa Universalis, Sim City, Zoo Tycoon y la saga Worms y el Command & Conquer .

- **Pelea:** Juegos basados en el combate físico. Se dividen en juegos de 1 contra 1 o "versus" y juegos de avanzar y pegar o "beat'em up" (derrótalos a todos). Ejemplos: Mortal Kombat, Virtua fighter, Dead or alive, Tekken, Street Fighter, Fatal Fury, King of Fighters, Dragon Ball Z Budokai Tenkaichi, Killer Instinct, Soul Calibur; y Golden Axe, Final Fight.
- **Plataformas:** Juegos en los que el protagonista ha de avanzar a través de un entorno con múltiples alturas en el que debe saltar las plataformas (de ahí el nombre) para llegar al objetivo. Ejemplos: Mario64, Sonic the Hedgehog, Alex Kidd, Megaman, Crash Bandicoot, Jak and Daxter.
- **Puzzles:** Juegos basados en los reflejos y diferentes formas de la inteligencia formal. Es un género muy amplio, abarcando juegos con concepciones y funcionamientos muy diferenciados. Ejemplo: Tetris, Columns, Bubble Bobble, Lemmings, Snow Bros. Algunos de ellos no tienen fin; quiere decir que no hay niveles, sólo aumenta el grado de dificultad en un sólo escenario, es el caso del Tetris.
- **Rol:** También llamados RPG (Role Playing Games), se basan en los juegos de rol clásicos, donde el protagonista interpreta un papel y ha de mejorar sus habilidades, mientras interactúa con el entorno, objetos y otros personajes. Ejemplos: Final Fantasy, Dragon Quest, Legend of Zelda, The Elder Scrolls, Chrono Trigger, Legend of Mana, Suikoden.
- **Musicales:** Su desarrollo gira en torno a la música ya sean de tipo "karaoke" como Singstar Party, tocar un instrumento, como Guitar Hero o en los que se debe bailar, como en DDR Dance Dance Revolution Ejemplo: Pump It Up, Electroplankton, Rhythm Tengoku, Sound Voyager, Space Channel.
- **Simulación:** Juegos basados en la reproducción, generalmente de forma realista, del funcionamiento de alguna actividad. Diferentes subgéneros son los simuladores de vuelo o de conducción. Así como el raro subgénero de la "simulación social" como en The Sims. Ejemplos: Ace Combat, Silent Hunter, IL2 Sturmovik, Microsoft Flight Simulator, Animal Crossing, Ship Simulator, Trainz Simulator.
- **Carreras o Velocidad:** Son juegos en los que se pilotan diferentes vehículos, ya sean reales o ficticios, para ganar en diferentes carreras. Dentro de este apartado se pueden distinguir dos variantes, arcades como Crash Team Racing y simuladores como Gran Turismo. Otros ejemplos: Mario Kart, Wipeout, Top Gear, TrackMania, Burn Out, Need for Speed.
- **Acción:** Son juegos en los que se mezclan el shooter o juego de disparos, simuladores y carreras. Están ambientados en la guerra, son juegos en los que se tiene que hacer un objetivo, pudiendo ser desde capturar una bandera (o base), hasta eliminar a un determinado personaje o grupo del videojuego. En este tipo de videojuego suelen haber tanques, barcos, aviones, coches, etc. Suelen ser de mapas (o ambientación) muy grandes. Ejemplos respectivos: Ghosts'n Goblins, Metal Slug, Battlefield, Operation Flashpoint, Golden Eye, Devil May Cry.
- **Hit n Run(golpea y corre):** Se caracterizan por ser juegos en los que se puede hacer lo que el jugador quiera, como viajar libremente por el mapa del juego, e interactuar con casi todo lo que este a su disposición. Estos juegos son una mezcla de disparos, peleas y carreras. Son muy conocidos por su tema de "antihéroes" ya que la mayoría de estos tienen como protagonistas a criminales,

asesinos y ladrones. Ejemplos respectivos: Grand Theft Auto, Just Cause, Saints Row, Scarface, The Godfather, True Crime, Bully.

II.5.2 Clasificación de los videojuegos

Este tipo de Software es reglamentado por la Entertainment Software Rating Board (ESRB) la cual es una entidad auto-regulada establecida en 1994 por Entertainment Software Association (ESA), la anteriormente llamada Interactive Digital Software Association (IDSA). ESRB realiza en forma independiente clasificaciones, entrega lineamientos y principios de privacidad para la industria de los videojuegos.

ESRB clasifica los videojuegos según su contenido de violencia física o verbal y otros elementos como el sexo. Esta clasificación orienta y ayuda a los padres y consumidores a escoger los juegos de video que sean correctos para su familia.

La clasificación es la siguiente:

- EC; Early Childhood ("Infancia temprana"): Material con contenido apto para edades a partir de 3 años. Los juegos que entran dentro de esta categoría son específicamente desarrollados para niños pequeños y usualmente son de orientación educacional. Sin embargo, algunos juegos con problemas más complejos (tales como la serie del Dr. Brain) pueden ser catalogados como "E".
- E; Everyone ("Para todos"): Abarca temas aptos para edades de 6 años en adelante. Los títulos comprendidos dentro de esta categoría, pueden contener algo de animación, fantasía o violencia moderada y/o el uso de palabrotas suaves. Algunos ejemplos de juegos que entran en esta clasificación son: Sonic Advance, Super Mario Advance 4, Luigi's Mansion, Pokémon Rubí, Zafiro y Esmeralda, serie FIFA y la mayoría de juegos sobre deportes y acertijos.
- E10+; Everyone 10 and older ("Para todos mayores de 10 años"): (También conocido como Pre-adolescente) Material idóneo para edades de 10 o más años. Las obras dentro de esta categoría contienen animaciones, fantasía o violencia media, palabrotas regulares y/o sangre en temas sugerentes. El icono se añadió a los que usa la ESRB el 2 de marzo del 2005. Donkey Kong Jungle Beat fue el primer juego en recibir esta valoración. Otros ejemplos con esta distinción son: Shadow the Hedgehog, Project Gotham Racing 3, Kingdom Hearts 2 y Lego Star Wars II: The Original Trilogy.
- T; Teen ("Adolescentes"): Contenido que puede ser adecuado para edades de 13 años o más. Los productos de este género contienen de manera limitada violencia, temas sugerentes, humor crudo, sangre, juegos de azar simulados, y/o uso de lenguaje fuerte. Ejemplos de juegos dentro de esta tipificación son: Star Wars: Rogue Squadron, Prince of Persia: The Sands of Time, Tekken 5, Battlefield 1942, y Super Smash Bros. Melee.
- M; Mature ("Maduro"): Artículos que pueden contener material pertinente para edades de 17 años o más. Las obras de esta categoría muestran violencia intensa, sangre y horror, temas sexuales y/o palabrotas fuertes. Éstos son ejemplos de juegos que caen dentro de esta categoría: Dead or Alive 4, Duke Nukem 3D, Doom, Mortal Kombat, Grand Theft Auto 3, Conker's Bad Fur Day, Resident Evil, y Devil May Cry 3.

- AO, Adults Only (Solo para Adultos), este tipo de juegos tienen contenidos recomendados para personas mayores de 18 años. Los títulos de esta categoría pueden incluir prolongadas escenas de violencia y/o contenidos con sexo y desnudez. Generalmente estos títulos solo están disponibles para PC. Como ejemplo: Singles, The Guy Game, Manhunt 2.
- RP; Rating Pending ("Clasificación Pendiente"): La llevan productos que han sido sometidos a evaluación ante la ESRB y están en espera de una categorización final. Este símbolo aparece solo en demos y publicidad de juegos próximos en salir a la venta.²

² <http://www.esrb.org/index-js.jsp>, 28 de Agosto del 2007.

Capítulo III

Marco Tecnológico

III.1 Software

El software que se utiliza para crear la ilusión de RV constituye lo que se ha denominado "máquina de realidad" ("reality engine"). La cual consta de un modelo tridimensional detallado de un mundo virtual, el cual es almacenado en la memoria de la computadora y codificado en forma de "bits". Cuando el usuario levanta su vista o mueve su mano o la del personaje que interactúa por el, el software de RV actualiza los datos que fluyen de las acciones antes mencionadas para producir la simulación tridimensional.

Este software es el corazón de cualquier sistema de RV porque procesa y genera mundos virtuales, incorporando a ese proceso una o más computadoras. El Software obedece a instrucciones destinadas al ensamblaje, procesamiento y despliegue de los datos requeridos para la creación de un mundo virtual, debiendo ser lo suficientemente poderosa para cumplir tal tarea en "tiempo real"¹, con el objeto de evitar alentamientos entre los movimientos del personaje y las reacciones de la máquina a dichos movimientos. Este software puede operar a nivel de computadoras personales, estaciones de trabajo y supercomputadoras. El software de RV maneja tres tipos de tareas:

- a. Entrada de Datos
- b. Salida de datos
- c. Generación, operación y administración del espacio virtual, así como de los objetos dentro de este espacio.

Es también a través de este software que la computadora convierte su modelo digital de un mundo, en el patrón apropiado de puntos de luz, sonido e interacción con los objetos dentro del espacio tridimensional; que mezclados en la forma apropiada ayudan a convencernos que nos encontramos dentro de un mundo.

Además del propio software de RV, se necesita el apoyo de otro tipo de software con características especiales, que ayudan a que el mundo virtual sea lo mas cercano a la realidad o lo mas cercano a lo que queremos que sea nuestro espacio virtual, ya sea que queramos estar en un sitio en el pasado, en el futuro o en un lugar el cual solo existe dentro de nuestra imaginación.

El software en el cual nos apoyaremos para crear nuestro mundo virtual debe ser del siguiente tipo:

- Software para el modelado tridimensional
- Software para el tratado de texturas
- Software para edición de sonido y video
- Software para la simulación

¹ En una descripción muy corta, una tarea en tiempo real es aquella capaz de procesar un dato antes de que ingrese al sistema el siguiente dato. <http://www.alegsa.com.ar/Dic/tiempo%20real.php> 15/julio/2007

III.1.1 Tipos de Software**- SOFTWARE PARA EL MODELADO TRIDIMENSIONAL**

Este software es necesario para crear nuestro mundo virtual, ya que es aquí en donde damos forma a todo lo que se encuentra en nuestra imaginación, como lo son edificios, personas, escenarios, etc. Es decir todo lo que queremos tener en nuestro espacio hay que construirlo, y es con este tipo de software, que podemos lograrlo, además de integrar texturas, animaciones, iluminación, etc.

- SOFTWARE PARA EL TRATADO DE TEXTURAS

Es aquí donde se aplica la parte artística de un proyecto de RV, el trabajo de texturizado se ve ayudado en gran parte por este tipo de software. Con él podemos lograr esa sensación de realidad, debido a que se pueden integrar fotografías u otro tipo de imágenes 2D, para crear materiales que nos ayuden a recrear escenarios fantásticos y reales. Todo esto aplicando transformaciones sobre la imagen a tratar para que después esta sea incorporada a un modelo en 3D.

SOFTWARE PARA LA EDICION DE SONIDO Y VIDEO

Habiendo construido y texturizado el mundo virtual, tenemos que agregar un medio más, el audio, con este tipo de programas podemos editar sonidos que utilizaremos dentro del software de RV.

El editor de video puede ayudarnos, si el software necesita algunas escenas introductorias, las cuales podrían ser para explicar algunas partes del recorrido en el espacio virtual, y así editar dichas escenas en un video que puede ser activado después de cumplir alguna situación dentro del mundo virtual.

- SOFTWARE PARA LA SIMULACIÓN

Este tipo de programas ayudan a que la experiencia de la RV sea aun mejor, ya que son capaces de procesar señales que vienen directamente de dispositivos entrada/salida como lo son cascos y guantes, para crear un ambiente inmersivo. También ayudan a que se comparta esta experiencia entre más usuarios añadiendo más dispositivos de entrada y computadoras conectadas en una red, donde se va a estar ejecutando la simulación. Entonces el software se encarga de crear y de actualizar la base de datos de todos los objetos que se encuentran dentro de nuestro espacio virtual.

III.1.2 Computación grafica

La computación gráfica es la rama de las ciencias de la computación que se encarga del estudio, diseño y trabajo del despliegue de imágenes en la pantalla de una computadora con la ayuda de herramientas proporcionadas por materias como la física, óptica, geometría, etc.

La computación gráfica permite una mejor interacción con los usuarios de una computadora, así como un método para visualizar y entender objetos y fenómenos de cualquier dimensión. Los ambientes virtuales simulados van un paso más allá introduciendo fenómenos dependientes del tiempo. Sin embargo, la computación gráfica y la RV son tareas computacionalmente costosas pero como mencionamos anteriormente, esto ha ido reduciéndose con la evolución de la PC gracias a procesadores y al uso de tarjetas gráficas cada vez mejores.

La información gráfica puede ser un bitmap o mapa de bits, que es una imagen en la pantalla de la computadora, que esta formada por píxeles, los cuales son pequeños puntos blancos, negros o de color dispuestos en filas. Las palabras, los números y las imágenes que vemos no son más que patrones de píxeles creados por software.

Este tipo de imágenes y los programas para esta materia están orientados para el procesamiento de la fotografía digital o para crear dibujos sencillos o planos, es decir 2D. Una de las principales características es la manipulación, ya que se pueden tener cambios modificando la tonalidad de los puntos, pero la limitación sería la resolución y precisión.

Con respecto al formato del archivo digital es variado, entre los que se encuentran GIF, JPEG, PCX, CGM, BMP, PNG y TIFF.

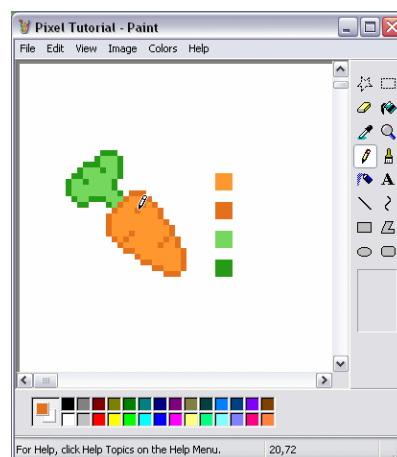


ILUSTRACIÓN III.1

Imagen hecha a través de píxeles

Otro tipo de información gráfica se representa mediante la implementación geometría vectorial, utiliza el sistema de coordenadas cartesianas (x, y), logrando de esta manera guardar datos gráficos de una forma más precisa, además de que cada figura conserva sus puntos externos o sea los que se encuentran en su contorno y pueden ser representados en cualquier escala, y la calidad de la imagen también pueda ser modificable.

La complejidad es algo con lo que se tiene que tratar para el diseño de imágenes utilizadas en esta técnica, pero nos permite trabajar con dimensiones calculadas y con la ventaja de que al agregar un eje más a nuestro sistema, podemos trabajar en tres dimensiones es decir (x,y,z). Los programas que utilizan este sistema son conocidos como de diseño asistido por computadora o CAD (computer-aided desing).

Algo que debemos tener en mente es que aunque los datos nos den la sensación de encontrarnos en un espacio 3D, nuestro periférico de salida, es decir, la pantalla nos muestra imágenes planas. Lo que nos ayuda a visualizar esta imagen plana y aparentar tridimensionalidad, es la perspectiva, la cual es la técnica que consiste en representar la profundidad espacial en una superficie plana, dándonos como resultado una representación volumétrica.

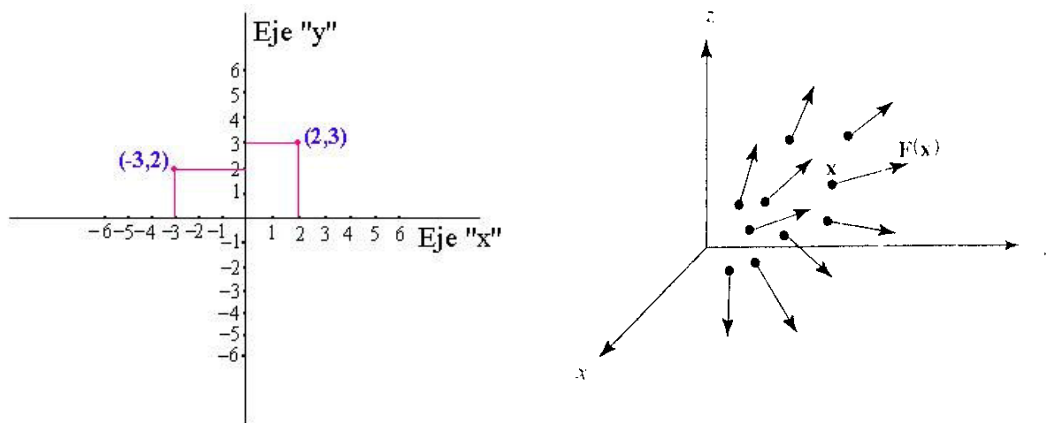


ILUSTRACIÓN III.2

Geometría Vectorial en 2D y 3D

Con el software se pueden crear diseños en pantalla de productos tan diversos como chips para computadora o edificios públicos. En la actualidad el software es mucho más complejo que los bosquejos básicos y los gráficos orientados a objetos, permitiendo a los usuarios elaborar modelos tridimensionales “sólidos” con características físicas, como peso, volumen y centro de gravedad. Estos modelos pueden rotarse y observarse desde cualquier ángulo.

Cuando el diseño esta terminado, se aplican formulas de iluminación y color que ayudan a producir imágenes realistas. También se les ha incorporado a estas herramientas la capacidad de animar si así lo requiere.

Un punto importante referente a la computación grafica son los tipos de software que se pueden utilizar para el manejo de estas técnicas:

- Paquetes generales de programación y de aplicaciones para propósitos especiales

Con ayuda de estos programas podemos ayudarnos con librerías con funciones graficas que pueden ser utilizadas mediante lenguajes de programación como C.

- Programas para el diseño asistido por computadora (CAD)

En este grupo se diseñan las aplicaciones para que no sea necesario saber programar para crear y diseñar, esta especialmente diseñado para personas con conocimientos básicos para el manejo de programas.

Así podemos ver los alcances de la computación gráfica, la cual se encuentra presente en campos como la industria del entretenimiento (cine, televisión, EI), publicidad, prensa, revistas y diseño de cualquier producto como: edificios, carros, aviones, hasta un simple envase de refresco, etc.

III.2 Construcción Tridimensional

En la construcción geométrica existen diferentes técnicas de creación según los elementos a utilizar para el desarrollo del modelo, entre las más comunes se encuentran:

Modelado por Malla

Una malla es un objeto deformable, constituida por tres vértices que forman un polígono (triángulo), el cual es el elemento más sencillo que se puede crear con esta técnica, a partir de este triangulo se pueden ir creando objetos más complejos. Es útil en modelos de baja resolución, ya que requieren de poca memoria.

Modelado por NURBS

El término se podría traducir como Curva Base Racional No Uniforme (Non-Uniform Rational Basis-Splines), es una técnica de modelado interactivo a base de curvas y superficies. No uniforme se refiere a que la influencia de cada vértice puede variar, esto es útil cuando se modelan superficies irregulares. Racional se refiere a que la ecuación usada para representar la curva o superficie es expresada como un radio de dos polinomios. Curva Base es una forma de construcción de curvas que interpola tres o más puntos.

Modelado por Patch

Es un objeto deformable usado para crear modelos que requieren curvas suaves en su superficie y provee un gran control de detalle para manipular geometría compleja. Cada patch esta formado por un conjunto de tres o cuatro vértices conectados por ejes que definen una superficie.

Modelado Poligonal

Es un objeto deformable parecido a la construcción que se realiza con una malla, solo que el modelado poligonal usa polígonos que tienen más de tres lados. Esta es la técnica más utilizada para aplicaciones en tiempo real como los videojuegos y por lo tanto es la técnica que utilizaremos para el modelado de la geometría requerida en el proyecto.

Cuando modelamos y desplegamos una escena tridimensional, debemos tomar muchas consideraciones, aparte de incluir sólo valores para la tercera dimensión. Entre estas consideraciones, lo que necesitamos para crear un objeto son sus dimensiones obtenidas de sus diferentes vistas, entre las que necesitamos primordialmente son la vista frontal, lateral y la vista superior del objeto, con esta información es suficiente para tener un objeto en 3D.

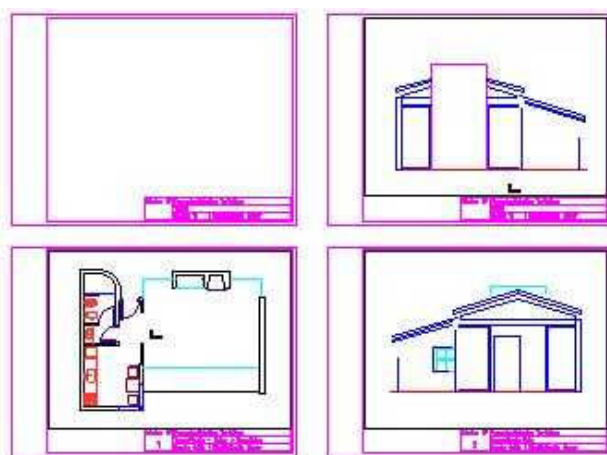


ILUSTRACIÓN III.3

Vistas Lateral, Frontal y Superior de una casa



ILUSTRACIÓN III.4

Modelo 3D de una casa hecha a partir de las vistas anteriores

Lo que queremos con la construcción tridimensional es que nuestro mundo interactivo tome forma, es decir, aquí es donde vamos a crear todos los objetos involucrados, como lo son árboles, edificios, personajes, etc. Para lo cual va a ser necesario utilizar geometrías que representen o se asemejen a las características de los objetos a ser reproducidos, de esta forma a partir de un cubo podemos crear estructuras más complejas como un edificio o un ser humano.

Existen otros métodos que nos ayudan a crear situaciones climáticas, como lo pueden ser la lluvia, el viento, el movimiento del mar o un río, terremotos e incluso efectos como: explosiones, el movimiento del humo, el fuego, etc. Estos métodos podrían ser los fractales o los sistemas de partículas, los cuales son de especial utilidad para describir objetos que cambian con el paso del tiempo, por flujo, oleada, rociada o expansión. Se utiliza el proceso aleatorio para generar objetos en alguna región de espacio definida y variar sus parámetros con el paso del tiempo. Las formas de partículas pueden ser esferas, elipsoides, planos, etc.

Otros cuerpos geométricos utilizados para la construcción tridimensional son: pirámides, cilindros, esferas, toroides, etc.

Lo anteriormente escrito nos lleva a la conclusión de que una mayor cantidad de polígonos nos permite reproducir un objeto con más exactitud, pero esto nos produce un problema debido a que exige mayor procesamiento por parte de la computadora.

III.2.1 Transformaciones Geométricas

Estas transformaciones son necesarias para manipular el objeto dentro del espacio tridimensional. Las transformaciones más utilizadas son:

- a) Traslación: Esta transformación puede entenderse como movimientos directos sin cambios de orientación, es decir mantienen la forma y el tamaño de las figuras u objetos trasladados. La traslación de un punto desde la posición $P1 = (x,y,z)$ a la posición $P2 = (x+dx, y+dy, z+dz)$ se realiza con la siguiente operación de matrices donde d es el valor de la distancia de traslación para los ejes x, y, z respectivamente.²

$$P' = T(d_x, d_y, d_z) \cdot P = \begin{pmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x + d_x \\ y + d_y \\ z + d_z \\ 1 \end{pmatrix}$$

ILUSTRACIÓN III.5

Operación de Traslación en tres dimensiones

² Lehmann, Geometría Analítica, Uteha, Mexico 1969.

Esta operación es válida para cada uno de los puntos que conforman al objeto. Teóricamente, para trasladar un objeto lo que se debe hacer es aplicar la fórmula para todos sus puntos y así no sufra modificaciones; pero en la práctica solo se trasladan los vértices y se redibujan nuevamente los demás puntos.

- b) Rotación: es el movimiento de cambio de orientación de un cuerpo extenso de forma que, dado un punto cualquiera del mismo, este permanece a una distancia constante de un punto fijo. En un espacio tridimensional, para un movimiento de rotación dado, existe una línea de puntos fijos denominada eje de rotación.³

Esta transformación se puede realizar para cada uno de los ejes coordenados. Los ángulos de rotación positivos producen giros en sentido opuesto a las manecillas del reloj.

$$P' = R_x(\theta) \cdot P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\theta) & -\text{sen}(\theta) & 0 \\ 0 & \text{sen}(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \cdot \cos(\theta) - z \cdot \text{sen}(\theta) \\ y \cdot \text{sen}(\theta) + z \cdot \cos(\theta) \\ 1 \end{pmatrix}$$

ILUSTRACIÓN III.6

Operación de rotación respecto al eje x

$$P' = R_y(\theta) \cdot P = \begin{pmatrix} \cos(\theta) & 0 & \text{sen}(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen}(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \cdot \cos(\theta) + z \cdot \text{sen}(\theta) \\ y \\ -x \cdot \text{sen}(\theta) + z \cdot \cos(\theta) \\ 1 \end{pmatrix}$$

ILUSTRACIÓN III.7

Operación de rotación respecto al eje y

$$P' = R_z(\theta) \cdot P = \begin{pmatrix} \cos(\theta) & -\text{sen}(\theta) & 0 & 0 \\ \text{sen}(\theta) & \cos(\theta) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \cdot \cos(\theta) - y \cdot \text{sen}(\theta) \\ x \cdot \text{sen}(\theta) + y \cdot \cos(\theta) \\ z \\ 1 \end{pmatrix}$$

ILUSTRACIÓN III.8

Operación de rotación respecto al eje z

³ Lehmann, Geometría Analítica, Uteha, Mexico 1969.

- c) Escalación: Una transformación de escalación altera el tamaño de un objeto. Se puede realizar esta operación para polígonos al multiplicar los valores de coordenadas (x,y,z) de cada vértice por los factores de escalamiento s_x , s_y y s_z para producir las coordenadas transformadas (x' , y' , z'). El factor de escalamiento s_x escala objetos en la dirección de x, mientras que el factor de escalamiento s_y lo hace en la dirección de y lo mismo para el eje z. Cuando se asignan el mismo valor a S se genera una escala uniforme. Y cuando se asignan valores distintos a S se obtiene una escala diferencial. La operación es representada por el siguiente matriz.

$$P' = S(s_x, s_y, s_z) \cdot P = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \cdot s_x \\ y \cdot s_y \\ z \cdot s_z \\ 1 \end{pmatrix}$$

ILUSTRACIÓN III.9

Operación de Escalación

Además de estas operaciones básicas contamos con las operaciones booleanas. Estas se refieren cuando se utiliza una geometría sólida para operaciones parecidas al álgebra de Boole, las operaciones básicas son: adición sustracción e intersección.

III.2.2 Representación Tridimensional

Existen varios métodos para la representación de objetos tridimensionales, pero lo que debemos entender es que para obtener una imagen de un objeto en 3D, primero se debe tener una posición en el sistema coordenado para situar al observador, y después poner frente a este un plano de visión, en el cual se proyecta la información geométrica de los objetos en tres dimensiones.

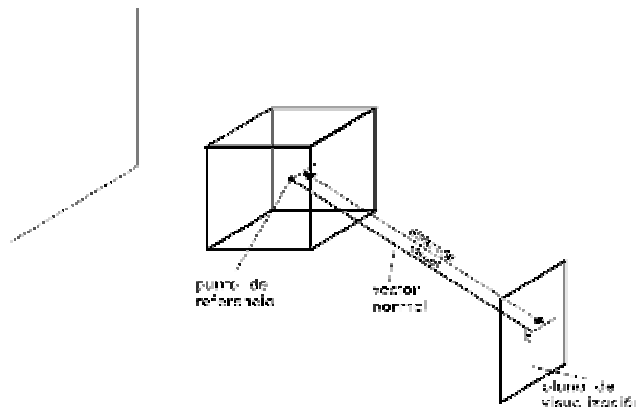


ILUSTRACIÓN III.10

Parámetros de visualización tridimensional

Algunos de los métodos para generar la vista de un objeto en tres dimensiones son los siguientes:

- a) **Proyección paralela:** Una proyección paralela se consigue trazando rectas paralelas desde cada uno de los vértices del objeto, hasta que intersectan al plano de la pantalla. El punto de intersección es la proyección de cada vértice. Los puntos de proyección se conectan mediante segmentos que corresponden a las conexiones existentes en el objeto original.

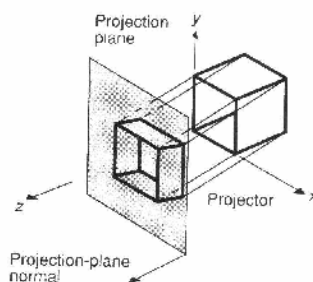


ILUSTRACIÓN III.11

Proyección paralela

- b) **Proyección en perspectiva:** En este tipo de proyección, cuanto más lejos está un objeto del observador, más pequeño parece. Esto permite al observador tener una idea de profundidad, una indicación de que porciones de la imagen corresponden a las partes del objeto que están más cerca o más lejos del observador. En la proyección en perspectiva las líneas de proyección no son paralelas, sino que convergen en un punto denominado centro de proyección. Estas podrían ser los rayos de luz que procedentes del objeto llegan a los ojos del observador. Será la intersección de estas rectas convergentes con el plano de la pantalla las que definan la imagen proyectada.

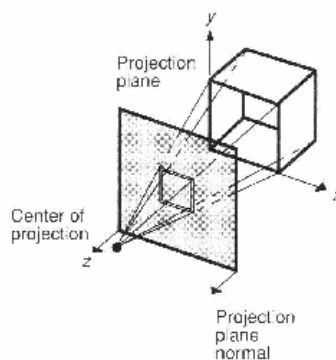


ILUSTRACIÓN III.12

Proyección Paralela

Adicionalmente a las proyecciones contamos con procedimientos que ayudan a visualizar de mejor manera un objeto tridimensional uno de ellos es el siguiente:

Eliminación de Líneas ocultas: Este método se utiliza para poder distinguir el frente y la parte de atrás de los objetos desplegados en pantalla.

Sin el ocultamiento de algunas líneas no podría distinguirse con claridad la parte del objeto que estamos visualizando, además de su posición respecto a otros.



ILUSTRACIÓN III.13
Eliminación de líneas ocultas

III.3 Texturas e Iluminación

Estos elementos son de gran importancia para incorporar realismo a nuestro objeto 3d, siendo necesario involucrar a la geometría dos elementos importantes: texturizado a sus superficies y elementos de iluminación.

¿Qué es un material?

Un material es un elemento agrupado en un conjunto, el cual es, usado con algún fin específico. En nuestro caso con la ayuda de 3D Studio Max 5 podemos crear materiales cuyo fin será, el de representar la apariencia de materia real dentro de nuestro programa, como una pared, la cual podría ser de diferentes materiales, de ladrillo, cemento, metal, etc.

Los tipos de materiales que se pueden crear desde 3D Max se muestran en la siguiente imagen:

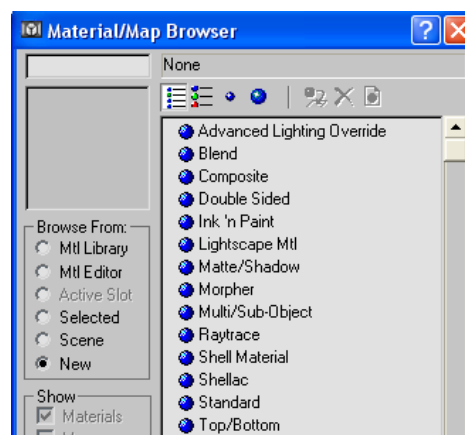


ILUSTRACIÓN III.14
Tipos de Materiales

De manera inicial se pueden asignar diferentes propiedades a los objetos ya construidos, y entre éstas propiedades tenemos la de agregar colores a los materiales creados anteriormente, esto se puede hacer por elemento o directamente a cada uno de sus polígonos, además de otras propiedades como lo son la intensidad luminosa, la cual ayuda a simular distintos efectos sobre el material como lo sería una apariencia metálica o cristalina.

¿Qué es una textura?

Es una imagen proyectada sobre parte o la totalidad de la superficie de un objeto para darle apariencia fotorealista. Existen texturas procedurales (calculadas) o texturas de mapa de bits.

La técnica utilizada para la incorporación de las texturas en modelos de tres dimensiones se conoce como texturizado, con esta podemos lograr mayor realismo, lo cual se obtiene con el uso de fotografías de materiales reales, que son los que se aplican sobre los polígonos del objeto tridimensional. El método de aplicación se maneja de manera similar a la proyección de mapas, comúnmente conocida como mapeo. Con este método se puede ajustar la imagen al polígono o conjunto de polígonos.

El trabajo de texturizado nos ayuda también a ahorrar geometría durante la construcción, esto quiere decir que puede disminuir la cantidad de polígonos, aumentando detalle al trabajo de texturizado, lo cual también nos lleva a un ahorro de procesamiento geométrico en objetos que no son tan importantes dentro de la escena del mundo virtual. Esto es, cuando tenemos elementos que se repiten dentro de nuestro universo, como lo pueden ser árboles dentro de la representación de un bosque, no tenemos que construir todos los árboles, simplemente con una imagen y hacer representaciones de la misma mediante planos o mediante una sola imagen bien elaborada de un conjunto de árboles. También para la representación del paisaje (background) para que este no sea modelado, se puede reemplazar por una imagen mapeada sobre una esfera o un cilindro.

La iluminación dentro de un espacio tridimensional se observa en la cantidad de luz que el objeto recibe y refleja, además que está en relación con la proximidad de éste a la fuente luminosa y con la intensidad de la misma.

Los encargados de mostrar el efecto de iluminación sobre los materiales son los conocidos como shaders.

¿Qué es un shader?

Shader es un trozo de código que puede estar escrito en diversos lenguajes, con el objetivo de representar la forma en la que la luz se aplica a un modelo 3D. Se orienta a sistemas en tiempo real como lo pueden ser videojuegos y programas de modelado 3D, en algunas ocasiones se requieren para ciertas tarjetas de video.⁴

⁴ <http://en.wikipedia.org/wiki/Shader>, 15/julio/2007

Tipos Básicos:

- Surface Shader. No se aplica la iluminación, solo aparecería como un color sólido o una textura sin sombras.
- Lambert. Se nombro así en honor al matemático que creo este algoritmo Johann Heinrich Lambert.
- Phong. Desarrollado por Bui Tuong Phong matemático. Aplicado a materiales de tipo plástico. Basicamente es la suma de un material tipo Lambert más funciones que crean brillos y reflejos.
- Blin. Algoritmo que define a un material metálico.

Ahora para ver esto desde otro punto de vista veamos el modelo de iluminación de Lambert, la iluminación tiene tres componentes principales las cuales son: Especular, Difusa y Ambiental.

- Componente Difusa

La iluminación de un objeto a partir de una fuente luminosa puntual, en el cual sus rayos son transmitidos de manera uniforme en todas direcciones. El brillo de un objeto es variable de una parte a otra, dependiendo de la dirección y la distancia respecto de la fuente de iluminación. A lo cual la ecuación de iluminación difusa es la siguiente y con los parámetros a considerar:

$$I = I_a \cdot K_a + I_L \cdot K_d \cdot \cos(\theta) = I_a \cdot K_a + I_L \cdot K_d \cdot (\vec{N} \cdot \vec{L})$$

I_L : Intensidad de la fuente
 θ : Angulo de incidencia [0°..90°]
 K_d : Coeficiente de reflexión difusa del objeto [0..1]
 N : Normal a la superficie (unitario)
 L : Vector de iluminación (unitario)

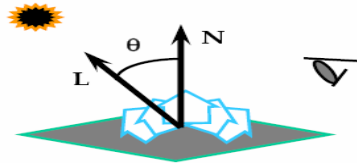


ILUSTRACIÓN III.15

Componente Difuso

El proceso anterior es el que vamos a aplicar a todos los materiales que creamos desde 3D Max. Cargando la textura desde el canal difuso del material en cuestión.

- Componente Ambiental

Ahora si tenemos a considerar que la luz ambiental afecta de la misma manera a todos los objetos desde todas las direcciones, la ecuación de esta componente es la siguiente:

$$I = I_a K_a$$

En donde I_a es la intensidad de la luz ambiental, que es constante para todos los objetos y K_a es la cantidad de luz ambiental reflejada por la superficie del objeto, el coeficiente

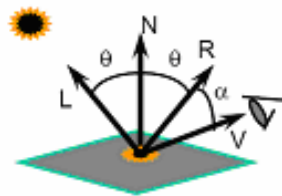
de reflexión ambiental, varía entre 0 y 1. Este coeficiente es directamente una propiedad del material.

- Componente Especular

En esta componente de la luz reflejada sobre una superficie brillante o pulida en una dirección preferente formando un brillo, depende de la posición del observador. En un espejo perfecto la dirección de observación V es la dirección de reflexión perfecta R . El brillo decae cuando el observador se aleja de R . Pero debido a que en la realidad no contamos con estos valores ideales esta componente obedece al modelo de Phong con la siguiente ecuación así como sus variables.

$$I_S = I_L \cdot K_S \cdot \cos^n(\alpha) = I_L \cdot K_S \cdot (\overline{R} \cdot \overline{V})^n$$

- K_S : Coeficiente de reflexión especular
- α : Angulo entre R y V
- n : Coeficiente de especularidad de concentración de brillo



ILUSTRACION III.16

Componente especular

III.4 Características del Software de RV.

El software de RV mantiene similitudes respecto a otros, como los de animación realista y los que ayudan al modelado 3D, como son, la obvia integración de objetos 3D, así como de animación de personajes y ambientes virtuales.

Pero presenta marcadas diferencias respecto a otros programas de software gráfico, la más significativa es que el software de RV le agrega interactividad, además de las capacidades de navegación en tiempo real y no a través de un video como lo haría otro tipo de software, además de que el audio también es en tiempo real. El software de RV también cuenta con sistemas de detección de colisiones y múltiples comportamientos programables dentro del mundo virtual.

Así dentro de los programas de RV podemos encontrar los siguientes elementos:

- Detección de colisiones
- Importación de Objetos
- Materiales y Texturizado
- Animación
- Iluminación
- Audio

- Nivel de Detalle
- Física ambiental
- Configuración de dispositivos
- Lenguajes de Programación
- Manipulación
- Librerías
- Conectividad
- Operaciones Geométricas

A continuación se explican los términos antes mencionados:

1.- Detección de colisiones: Es un sistema que nos permite simular el comportamiento de los objetos en la realidad, por ejemplo un objeto sólido no puede atravesar a otro con la misma característica.

2.- Importación de Objetos: La capacidad de incorporar modelos creados en múltiples formatos existentes en tres dimensiones, comúnmente provenientes de programas CAD.

3.- Materiales y Texturizado: Esta característica permite al software tratar con los materiales y texturas, ya aplicados dentro del programa que se utilizó para crear los modelos 3D, es decir podemos cambiar las propiedades de las texturas, así como la resolución de las mismas, según esto convenga al desarrollo y al procesamiento del proyecto.

4.- Animación: Aquí se guardan todos aquellos movimientos predeterminados que van a poder visualizarse dentro de nuestro mundo virtual, como lo podría ser los movimientos de algún personaje con el cual vamos a interactuar dentro del espacio virtual o con algún objeto en específico, estos movimientos se realizaron anteriormente en un programa CAD y se guardaron con un formato propio del mismo programa, para ser utilizados dentro del software de RV.

5.- Iluminación: Esta característica nos ayuda a darle realismo al entorno, ya que con esto podemos simular el día, la tarde, la noche, el clima, si nos encontramos en un lugar cerrado, dentro de una casa, un departamento, en una nave espacial, donde sea. Por lo que es de gran importancia tener una posición ideal de todas nuestras fuentes de luz, así como la intensidad de las mismas.

6.- Audio. Esta característica nos permite guardar y reproducir sonidos de todas clases, además es importante que soporte formatos diferentes. Esto al igual que la iluminación es un complemento ambiental importante, ya que dependiendo del lugar donde nos encontramos los sonidos cambian.

7.- Nivel de Detalle: Esta característica ayuda a que el procesamiento de los objetos sea menor, ya que solo va a dibujar al máximo detalle aquellos objetos que se encuentren cerca del punto de vista del observador, pero aquellos que no se encuentran en las cercanías del observador pueden ser dibujadas a un detalle menor.

8.- Física ambiental: Esta operación permite establecer reglas dentro de nuestro mundo virtual, específicamente sobre las reglas físicas como la gravedad, en el caso

de manejar saltos, disparo de proyectiles como podrían ser flechas, el comportamiento de una explosión o simplemente el establecer las zonas que serán navegables dentro del recorrido. También depende del medio sobre el cual vamos a navegar, el cual puede ir desde las profundidades del océano hasta un paseo por el espacio.

9.- Configuración de dispositivos: En este módulo veremos la forma de incorporar todos aquellos dispositivos de entrada/salida que podamos utilizar dentro de la navegación, como lo pueden ser joysticks, teclado, mouse, o hasta guantes y cascos.

10.- Lenguajes de Programación: Esto se refiere a que el Software disponga de comandos para el control de comportamientos de entrada y salida de datos.

11.- Manipulación: En este campo vamos a analizar todo lo referente a la interacción con los objetos dentro del espacio virtual. Esto abarca varias acciones entre las que están, el reconocer la posición del usuario o del personaje con el que vamos a interactuar, para así interpretar una programación y modificar la geometría, si así es necesario.

12.- Librerías: La mayoría del software de RV esta provisto de una buena cantidad de librerías 3D, que sirven para construir formas compuestas, así como de librerías con objetos complejos, además otras que actúan sobre el espacio virtual.

13.- Conectividad: Esta característica ha tomado importancia gracias al desarrollo de las redes de información, y esto es ampliamente aplicable tanto para redes locales como para Internet, siendo esta última muy utilizada dentro del EI permitiendo interactuar hasta 60 personas dentro de un espacio virtual.

14.- Operaciones Geométricas: Aquí se manipulan los objetos a través de las transformaciones geométricas para adecuarlas a las necesidades de nuestro ambiente virtual.

III.5 Estructura de un Software de RV

La estructura de un programa de RV tiene una mayor complejidad respecto a otros programas y esto se debe a que todos los elementos que la conforman se encuentran actualizándose a cada momento, es decir, en tiempo real.

Los elementos que conforman la estructura de un software de RV son:

- Entradas y Salidas
- Base de Datos
- Programación
- Procesamientos

En breve se explica en que consisten cada uno de los elementos anteriormente mencionados:

1.- Entradas y Salidas: Aquí es donde se definen los diferentes dispositivos de entrada y salida que van a ser utilizados dentro de la aplicación. Como habíamos mencionado anteriormente los dispositivos pueden ser muy variados, tanto en complejidad como en costo, pero siempre deben tener la opción más sencilla y común, el teclado y el mouse.

2.- Base de Datos: Tenemos una múltiple cantidad de datos dentro de nuestro mundo virtual, y estos datos van desde animaciones, personajes, edificios, texturas, sombras, luces, etc. Además de las características de dichos objetos como su posición, orientación, y la jerarquía de cada uno de los objetos dentro del espacio virtual.

3.- Programación: Este es el punto más fuerte de la estructura, ya que aquí se ve lo referente a la interacción. Aquí se definen las situaciones dentro del espacio virtual, ya que dependiendo de cada situación se van a tomar acciones que los objetos harán por ellos mismos o que necesiten de nuestra participación.

4.- Procesamientos: En el momento en el que estamos ejecutando el programa de RV se realiza una ardua tarea de procesamiento por parte de la computadora, ya que el procesamiento de nuestros datos se realiza en tiempo real, es decir se actualizan estos en una mínima fracción de tiempo. Esto es se procesan imágenes, pero al mismo tiempo se ejecutan cálculos geométricos, procesamiento del audio, el comportamiento de los dispositivos de entrada/salida e interpretaciones de la programación.

III.6 Hardware

Este es un elemento importante dentro de la RV, bueno, es obvio que necesitamos de una computadora para poder ejecutar un software de este tipo, pero podemos valernos de periféricos especiales que ayudan a la experiencia que queremos recrear sea más inmersiva. Este es el objetivo del hardware de RV.

Los programas de RV son identificados por la gente por contar con dispositivos como guantes y cascos, los cuales nos dan la sensación de “estar”, además de ayudarnos a interactuar con objetos virtuales.

Otro punto a analizar para los desarrolladores de sistemas de RV, es que este hardware es muy costoso especialmente los cascos, para lo cual se es necesario hacer un análisis de costos y establecer la viabilidad para hacer uso de estos dispositivos u optar por tecnologías más baratas y que logren un estado parecido de inmersión.

**ILUSTRACION III.17****Guantes y Casco de RV****III.6.1 Dispositivos de Entrada/Salida**

Gracias a estos dispositivos es como logramos la entrada y salida de datos, es decir estos dispositivos son los que nos ayudan a comunicarnos con la computadora, la cual solo nos sabe responder con si y no, 0 y 1 o falso y verdadero. Estos dispositivos nos traducen tanto lo que queremos decirle a la computadora como lo que ella nos quiere decir a nosotros. Con los dispositivos de entrada, el usuario transmite ordenes al sistema de RV indicando el desplazamiento sobre el espacio, cambiar la cámara a un punto de vista diferente o incluso contar con botones programados para acciones determinadas (como un gatillo en el caso que se simule el comportamiento de un arma). Y los dispositivos de Salida, que nos permiten visualizar todas las respuestas a estas entradas ya sea a través de una pantalla, casco o de un proyector.

Los dispositivos de entrada de RV se agrupan en los que necesitan sensores para los datos de posición, estos siendo ubicados en el cuerpo del usuario, y en los que no es necesario tomar en cuenta la posición como los guantes, joysticks, mouse, teclado, etc.

Otras instalaciones ligeramente más sofisticadas conocidas como cabinas de simulación, consisten en utilizar una habitación con pantallas en distintas posiciones alrededor del usuario. Muchos de estos sistemas se utilizan en la industria de los videojuegos, en algunos casos estos incluyen respuestas mecánicas o neumáticas que generan un efecto dinámico y realista como lo pueden ser los simuladores de coches, fútbol, esquiar, motociclismo, aviación, etc.



ILUSTRACION III.18

Dispositivos de Entrada



ILUSTRACION III.19

Dispositivos de Salida

III 6.2 Rastreadores de Posición y Movimientos

Para lograr una sensación integral de inmersión en el mundo virtual es importante tener una relación directa entre el movimiento del cuerpo y la acción en el ambiente computacional, lo que se denomina paralelismo cinético (Motion Parallax). Es especialmente importante para controlar la visión del entorno digital, al desplazar la cabeza o para la localización de las distintas partes del cuerpo. Para poder controlar el movimiento es necesario colocar rastreadores (trackers) en las distintas partes del cuerpo. Estos rastreadores pueden ser mecánicos, ultrasónicos, ópticos o magnéticos

que permiten conocer la posición tridimensional y la orientación (seis grados de libertad)⁵ definiendo exactamente la posición en el espacio.

Los más sencillos y utilizados son los denominados giroscopios. Se instalan en la parte posterior de los cascos y permiten reconocer giros de la cabeza para adecuar la imagen visualizada o bien en el casco en una pantalla.

Los rastreadores de Posición Absoluta tridimensional necesitan una referencia fija y tienen un alcance definido dependiendo de la tecnología empleada por los sensores de posición. Son especialmente utilizados en el mundo de la animación tridimensional para definir movimientos naturales. Uno de los mayores problemas que se presentan en las aplicaciones de RV es el tiempo de demora entre el movimiento y la consecución de dicho movimiento en un ambiente virtual.



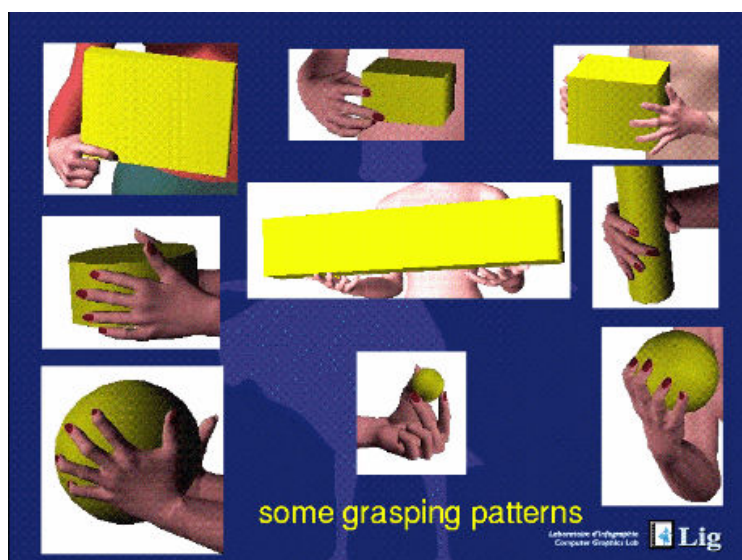
ILUSTRACION III.20

Dispositivos de Entrada

⁵ Un objeto libre en el espacio tiene seis grados de libertad diferentes: se puede trasladar en tres direcciones mutuamente perpendiculares (arriba, abajo, izquierda, derecha, atrás y adelante). Cualquier movimiento del objeto, no importa que tan complejo sea, se puede resolver en esos seis movimientos básicos

III 6.3 Manipulación de Objetos

Como ya vimos al tratar el tema de rastreadores de posición, estos dispositivos son muy útiles para capturar los movimientos de objetos reales. El movimiento capturado puede aplicarse a un objeto virtual para que la dinámica del mismo sea más realista. En algunos casos también se modelan los músculos, para aportar de una apariencia humana cuando se producen los movimientos. Y por último es esencialmente importante modelar el comportamiento de los pliegues y colisiones entre las distintas estructuras de nuestro objeto. Así como la programación de acciones e interacción entre objetos.



ILUSTRACION III.21

Manipulación

III.7 Sonido Virtual

El sonido es una sensación, en el órgano del oído, producida por el movimiento ondulatorio en un medio elástico (normalmente el aire), debido a rapidísimos cambios de presión, generados por el movimiento vibratorio de un cuerpo sonoro. Ahora en la RV se tienen dispositivos como los audífonos incorporados en los cascos o sistemas de sonido ambiental a través de bocinas puestas en posiciones estratégicas.

En este campo se enfoca a reproducir con exactitud todos los sonidos que puedan ser representados, tal cual pueden ser los sonidos ambientales, los producidos por los objetos o por comunicación oral en tiempo real de un usuario con otro, ya sea en la misma sala o en otro lugar, lo que se conoce como telepresencia.

III.8 Vida Artificial

La Vida Artificial trata de incorporar reglas de comportamiento, aprendizaje y evolución a modelos virtuales y robóticos. De esta manera los personajes pueden ser autónomos y evolucionar en su comportamiento, reproducirse, heredar y mejorar las

características de su especie. Para lo cual debemos entender el concepto de IA (Inteligencia Artificial).

Inteligencia Artificial: Es el estudio de cómo lograr que las computadoras hagan cosas que, por el momento, las personas hacen mejor⁶

Ahora entendiendo este concepto la RV tendrá que evolucionar a la par de esta rama para poder aprovechar los beneficios de esta aplicación, que proporcionará un mayor nivel de interactividad y realismo.

III.9 Aplicaciones

Actualmente las aplicaciones de la RV son múltiples. En los siguientes ejemplos que se exponen a continuación se tienen aplicaciones que actualmente existen y otras en vía de desarrollo o comercialización, que exponen los alcances de este software.

Medicina

Los usos de la RV en medicina pueden ser muchos, y de hecho es uno de los campos que mas ha aprovechado los avances esta tecnología. Actualmente los avances más importantes de aplicación son los simuladores para formación y entrenamiento en tareas sofisticadas, el tratamiento de enfermos de fobias y otros traumas (acrofobia, claustrofobia, etc.) y en la cirugía.

La RV se ha comenzado a utilizar poco a poco en la cirugía como herramienta de ayuda a tareas complicadas y que necesitan una gran habilidad. Existen aplicaciones que permiten realizar tareas de microcirugía a través de un robot trasladando y aumentando las sensaciones que el robot experimenta a un sistema de realimentación táctil, que maneja el cirujano. Otra aplicación es en la cirugía mínimamente invasiva ya que permite guiar las distintas herramientas que se utilizan para mejorar la intervención.

La evolución de las técnicas de imagen médica también ha constituido un nuevo tipo de (Image Guided Surgery) cirugía guiada mediante imágenes previamente obtenidas y en las que se identifican los objetivos de la intervención claramente.

Educación

La educación es uno de los ámbitos más beneficiados por herramientas de RV en todos los campos de la educación. Sin embargo hasta el momento estas tecnologías aun no están al alcance de los centros educativos de nuestro país, seguramente por el costo, y por la necesidad de que se implementen herramientas específicas de educación y teleeducación.

⁶ E. Rich, Inteligencia Artificial, Editorial Gustavo Hili, Barcelona 1988.

Comercio

La RV también se está aplicando al campo específico del comercio. Muchas empresas publicitarias empiezan a ofrecer entre sus productos la realización de tours virtuales para observarlos, navegar por las habitaciones de un hotel, o poder observar los acabados de un coche, es decir ver a detalle un producto antes de que este salga al mercado provocando mayor interés entre los consumidores.

Tecnología militar

Uno de los impulsos más fuertes al desarrollo de la RV ha sido su gran aplicación al área militar. Aunque no es muy fácil obtener información de las tecnologías y de las aplicaciones militares podemos distinguir fundamentalmente las siguientes aplicaciones, la mejora de los sistemas de simulación de vuelo, el entrenamiento militar y la incorporación de nuevas herramientas de realidad aumentada para los pilotos militares.

Los simuladores de vuelo constituyen hoy en día una herramienta fundamental para el entrenamiento de pilotos tanto militares como comerciales. Son sistemas muy sofisticados y costosos que incorporan todo tipo de interfaces para simular la situación real dentro de un avión, así como las distintas maniobras.

Y por último se han incorporado herramientas de realidad aumentada a los cascos de los pilotos militares para proveerlos de información adicional sobre el área que están sobrevolando, como indicadores de blancos, visión nocturna, mapeo de bases amigas y enemigas etc.

Arquitectura

La arquitectura es un área que rápidamente se ha adaptado a prototipos sencillos de RV. La navegación por futuros edificios para venta y análisis de los diseños ha derivado en la comercialización de herramientas específicas de diseño tridimensional para arquitectos. La RV permite en el caso de la arquitectura poder experimentar en la etapa de diseño los espacios creados y la adecuada adaptación al entorno.

Otra área creciente y de gran interés cultural es la reconstrucción virtual de monumentos históricos. Con dos finalidades, la visita virtual de monumentos, y las aplicaciones de realidad aumentada para las visitas de ruinas, tal y como lo es este trabajo de tesis. Estas aplicaciones permitirán la visita de monumentos arqueológicos reproduciendo sobre la visión real una reconstrucción del estado original del monumento.

Industria

El diseño de plantas industriales es un área relativamente complejo. La previsualización y examen de volúmenes de ruido y habitabilidad de las distintas áreas de la planta son algunas de las ventajas que aporta la RV a este campo. Por otra parte la RV constituye una herramienta de gran interés para diseño de prototipos industriales.

Entretenimiento Interactivo

Este es el campo de aplicación en el que la RV se ha desarrollado con mayor fuerza, ya que es el sistema de RV que se encuentra en estos momentos en la mayoría de los hogares del mundo. Siendo en sus principios un medio no inmersivo, actualmente la industria del EI se tienen dispositivos, como controles con sensores de movimiento incorporados, interacción mediante cascos y otros medios de visualización avanzados.

III.10 FUTURO DE LA RV

El objetivo de la RV ha sido la creación del ciberespacio, en la concepción que ha sido plasmada de manera más imaginativa a través de novelas.

Como tal, algunos de los requisitos fundamentales de este ciberespacio es que sea gráfico, masivo, distribuido e independiente de plataforma.

Para lograr la creación es necesario sobrepasar varios problemas actuales tales como el desempeño gráfico (especialmente en máquinas PC's ya que son la mayoría de la población), la latencia, la velocidad de red, y la creación de un modelo de interacción con miles de participantes.

El siguiente paso importante hoy en día para la RV es la creación de un marco que permita comportamientos, tales como la posibilidad para el usuario de causar cambios que afecten su entorno, como así mismo. Dichos cambios podrían ser activados por interacción, el tiempo, y por los objetos que lo rodean.

Dentro del campo de la educación y de la ciencia en general será una herramienta de gran valor y tal vez indispensable en el futuro. Imaginemos como será el aula en algunos años: nuestro asiento en el aula podrá ser nuestra propia sala o una terminal dentro de un campus universitario, complementada con un par de lentes o casco con audífonos integrados, así como un par de guantes especiales y traje ajustado de cuerpo completo. Con estos aditamentos podríamos dar ordenes verbales a nuestra computadora para que diera acceso a nuestro tema del día, por ejemplo, que tal un viaje al interior del cuerpo humano. Ante nuestros ojos aparecería una sala de cirugía con el paciente listo a ser explorado, con un comando virtual instruiríamos a la computadora a mostrar el sistema digestivo.

O sumergirnos en mundo de hace 150 millones de años reinado por dinosaurios y no solo veríamos los enormes animales, sino también la flora existente de esa era.

En áreas como la química, se vería beneficiada ya que los estudiantes serian capaces de abordar el interior mismo de la materia e ingresar al núcleo del átomo.

No cabe duda que la electrónica y las nuevas herramientas con las que cuenta la computación, harán de este siglo un mundo con esperanza, más humano, ya que el acceso al conocimiento sería más fácil y rápido, por ende una educación personalizada, eficiente, clara, efectiva y dinámica.

A medida de que las tecnologías de RV evolucionan, las aplicaciones se convierten literalmente en ilimitadas. Esto es asumiendo que la RV va a redefinir la interfaces entre las personas y la información, ofreciendo nuevas formas de comunicación.

Capítulo IV

Ingeniería del Software

IV.1.1 Software

El software de computadora se ha convertido en el “alma mater”. Es la máquina que conduce a la toma de decisiones comerciales. Sirve para la investigación científica moderna y solución a problemas de ingeniería. Es el factor clave que diferencia los productos y servicios modernos. Esta inmersa en todo tipo de de tareas en la vida moderna como lo es: el transporte, telecomunicaciones, entretenimiento, actividades militares, procesos industriales, etc.

Pero tenemos que respondernos antes una interrogante, y esta es ¿Qué es el software?

El software de la computadora es el producto que diseñan y construyen los ingenieros en computación. Esto abarca programas que se ejecutan dentro de una computadora de cualquier tamaño y arquitectura, documentos que comprenden formularios virtuales e impresos y datos que combinan números y texto y también incluyen representaciones de información de audio, video e imágenes.¹

Ahora el software es importante porque afecta muy de cerca cualquier aspecto de nuestra vida y esta extendido en nuestro comercio, cultura y en las actividades cotidianas.

¿Cómo se hace?

Construir software de computadora como construimos cualquier otro producto, aplicando un proceso que conduce a un resultado de alta calidad que satisface las necesidades de la gente que usará el producto. Desde el punto de vista de un ingeniero en computación, el producto obtenido son los programas, documentos y los datos que configuran el software de computadora. Pero desde el punto de vista de los usuarios el producto obtenido es la información resultante que hace de algún modo el mundo mejor a los usuarios.²

IV.1.2 Aplicaciones del Software

El software puede aplicarse en cualquier situación en la que se haya definido previamente un conjunto específico de datos y procesos (algoritmos). El determinismo de la información se refiere al tiempo de llegada de los datos así como su subsiguiente ejecución.

Algunas veces es difícil establecer categorías genéricas para las aplicaciones del software que sean significativas. Conforme aumenta la complejidad del software, es más difícil establecer comparaciones. Pero a grandes rasgos se consideran las siguientes áreas para la clasificación de las diferentes aplicaciones del software:

- Software de sistemas
- Software en tiempo real
- Software de gestión
- Software de Ingeniería y científico

¹ Ian Sommerville, Ingeniería del Software, Iberoamericana, México 1989.

² Ian Sommerville, Ingeniería del Software, Iberoamericana, México 1989.

- Software embebido
- Software basado en Web
- Software de inteligencia artificial

IV.2.1 Definición de Ingeniería del Software

La ingeniería del software es una disciplina o área de la Informática o de las Ciencias de la Computación, que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelvan problemas de todo tipo. Hoy en día la ingeniería del software trata con áreas muy diversas de la informática y de las ciencias de la computación, tales como construcción de compiladores, sistemas operativos o desarrollos en Intranet/Internet, abordando todas las fases del ciclo de vida del desarrollo de cualquier tipo de sistemas de información y aplicables a una infinidad de áreas tales como: negocios, investigación científica, medicina, producción, logística, banca, control de tráfico, meteorología, el mundo del derecho, la red de redes Internet, redes Intranet, EI, etc.

Definición del término “Ingeniería del software”

El término ingeniería se define en el Diccionario de la Real Academia Española de la Lengua como: Conjunto de conocimientos y técnicas que permiten aplicar el saber científico a la utilización de la materia y de las fuentes de energía. Otra definición de Ingeniería es: Conjunto de conocimientos y técnicas cuya aplicación permite la utilización racional de los materiales y de los recursos naturales, mediante invenciones, construcciones u otras realizaciones provechosas para el hombre.³

Si bien el término ingeniería del software no es introducido aún por el DRAE, esta debe mantener todas las características de la ingeniería como tal solo que guiada al propósito de la elaboración y mantenimiento del software, pero a continuación citaremos definiciones dadas por autores y conocedores del tema:

Definición 1

Ingeniería de Software es el estudio de los principios y metodologías para el desarrollo y mantenimiento de software [Zelkovitz 1978].

Definición 2

Ingeniería del Software es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación asociada requerida para desarrollar, operar y mantenerlos. Se conoce también como desarrollo de software o producción de software [Bohem, 1976].

³ Roger S. Pressman, Ingeniería del Software un enfoque práctico, Mc Graww Hill, México 2002.

Definición 3

Ingeniería del Software trata del establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable que sea fiable y trabaje en máquinas reales [Bauer, 1972].

Definición 4

La aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software; es decir, la aplicación de ingeniería al software. [IEEE, 1993]

IV.2.1 Fases de la Ingeniería del Software.

El trabajo que se asocia a la ingeniería del software se puede dividir en tres fases genéricas, con independencia del área de aplicación, tamaño o complejidad del proyecto. Cada una de estas fases es:

Definición o Análisis

Esta fase se concentra sobre el qué. Es decir, durante la definición, se intenta identificar que información ha de ser procesada, que función y rendimiento se desea, que comportamiento del sistema, que interfaces van a ser establecidas, que restricciones de diseño existen, y que criterios de validación se necesitan para definir un sistema correcto.

Desarrollo

La fase de desarrollo se centra en el cómo. Es decir, durante el desarrollo un ingeniero intenta definir como han de diseñarse las estructuras de datos, arquitectura del software, como han de implementarse los detalles del procedimiento, como han de caracterizarse las interfaces, como ha de traducirse el diseño en un lenguaje de programación y como ha de realizarse la prueba.

Mantenimiento

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas a medida que evoluciona el entorno del software y a cambios debidos a las mejoras producidas por los requisitos cambiantes del cliente. Durante la fase de mantenimiento se encuentran cuatro tipos de cambios:

1. Corrección
2. Adaptación
3. Mejora
4. Prevención

Además de las actividades de mantenimiento los usuarios de software requieren un mantenimiento continuo. Los asistentes técnicos a distancia, teléfonos de ayuda y sitios

web de aplicaciones específicas se implementan frecuentemente como parte de la fase de mantenimiento.

De estos integraremos a nuestro trabajo de tesis, el mantenimiento de adaptación y el de mejora o perfectivo, razones que explicaremos en el Capítulo VII.

IV.2.2 Modelos de proceso del software.

Para resolver los problemas reales de una industria, un ingeniero o equipo de ingenieros debe incorporar una estrategia de desarrollo que acompañe al proceso, métodos y capas de herramientas. Esta estrategia a menudo se llama modelo del proceso o paradigma de la ingeniería del software. Se selecciona un modelo de proceso para la ingeniería del software según la naturaleza del proyecto y de la aplicación, los métodos y las herramientas a utilizarse, y los controles y entregas que se requieren.

Algunos modelos de procesos para la ingeniería del software son los siguientes:

1. Modelo Lineal o Secuencial (ciclo de vida clásico)
2. Modelo de construcción de prototipos
3. Modelo DRA (Desarrollo Rápido de Aplicaciones)
4. Modelo de métodos formales
5. Modelo basado en componente
6. Modelos evolutivos de proceso del software
 - Modelo en espiral
 - Modelo de proceso adaptable
 - Modelo en espiral WINWIN
 - Modelo de desarrollo concurrente

De los modelos antes mencionados, elegimos trabajar con el Modelo lineal o secuencial, ya que es el que más se adecua a nuestras necesidades y también es el que permitió establecer los tiempos y metas a lo largo de cada una de las etapas del proyecto.

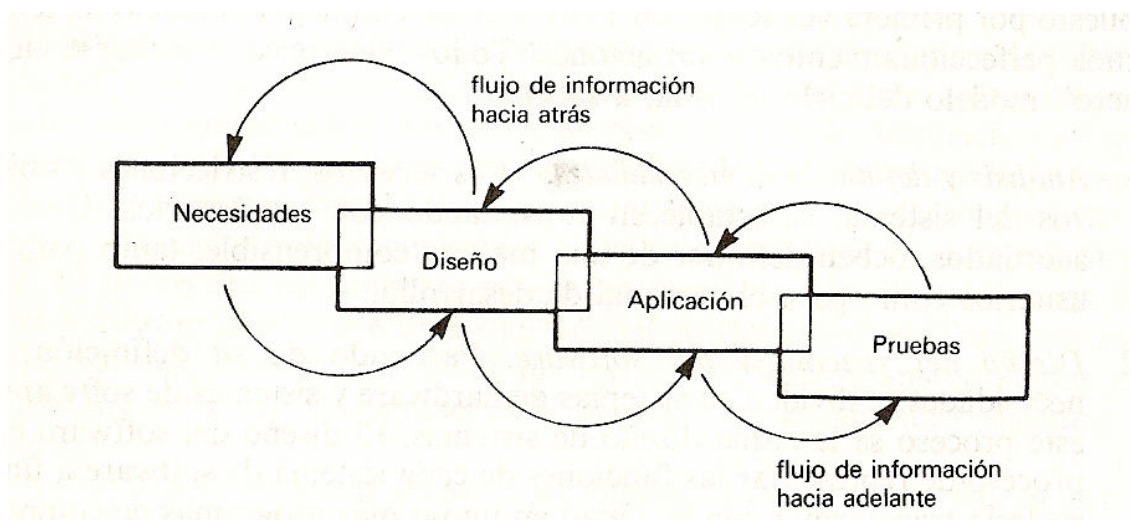
IV.2.3 Ciclo de vida del software

Como sabemos los grandes sistemas de software requieren un tiempo considerable para su desarrollo y permanecen en uso durante un tiempo mayor. En este periodo de desarrollo y uso pueden identificarse varias etapas, que juntas constituyen lo que se conoce como ciclo de vida del software.

1. Análisis y definición de necesidades. Los servicios, restricciones y objetivos del sistema, se establecen consultando con los usuarios. Una vez acordados, deben definirse de una manera comprensible, tanto para los usuarios como para el personal de desarrollo.
2. Diseño del sistema y del software. Partiendo de su definición, las necesidades se dividen en sistemas de hardware y sistemas de software. A este proceso se le llama diseño de sistemas. El diseño del software es el proceso de representar funciones de cada sistema de software a fin de poderlo transformar con facilidad en uno o mas programas de computadora.

3. Aplicación y pruebas de unidades. Durante esta etapa, el diseño del software se realiza como un conjunto de programas o unidades de programas escritos en algún lenguaje de programación ejecutable. Las pruebas de unidades implican la comprobación de que cada unidad cumple con su especificación.
4. Pruebas del sistema. Las unidades de programa individuales o los programas que se integran y prueban como un sistema completo para asegurar que se cubren las necesidades del software. Después de las pruebas, el sistema de software se envía al cliente.
5. Operación y mantenimiento. Esta fase suele ser la mas larga dentro del ciclo de vida. Se instala el sistema y se pone en uso práctico. La actividad de mantenimiento implica corregir errores que no se descubrieron en las primeras etapas del ciclo de vida, mejorar la aplicación de unidades del sistema y aumentar los servicios de este a medida que se perciben nuevas necesidades.⁴

Para fines administrativos, es útil diferenciar las fases del ciclo de vida del software, pero en la práctica, las etapas de desarrollo se superponen y proporcionan información unas a otras como se ve en la siguiente figura.



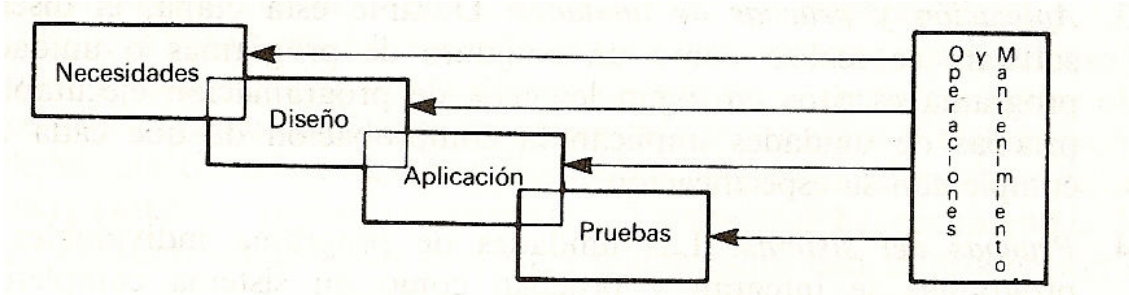
ILUSTRACION IV.1

El ciclo de desarrollo de software

La razón de que la fase de operación y mantenimiento no continúe después de la fase de pruebas del sistema es que la actividad de mantenimiento puede originar cambios en las necesidades, el diseño o la aplicación, o puede indicar la necesidad de pruebas adicionales del sistema. Lo cual es aplicable al 100% dentro del desarrollo de nuestro proyecto, directamente sobre la construcción de los diversos objetos, así como de la

⁴ Ian Sommerville, Ingeniería del Software, Iberoamericana, México 1989.

programación del comportamiento de los mismos. Ahora para incluir la operación y el mantenimiento, la figura debe modificarse como se muestra a continuación.



ILUSTRACION IV.2

El Ciclo de Vida del software

Capítulo V

Análisis y Diseño

A lo largo de este capítulo explicaremos las primeras etapas del Modelo lineal de la Ingeniería del software, el Análisis y Desarrollo, aplicadas a nuestro proyecto de tesis, para cada uno de los elementos que lo conforman.

V.1 Análisis de Requisitos

Este es el primer paso dentro del proceso de la ingeniería del software, ya que aquí se declaran objetivos y metas para un producto o sistema; como presentación gráfica, funciones específicas del sistema y niveles definidos de rendimiento. También tenemos otro tipo de requisitos por cumplir, como la facilidad de interacción hombre-maquina, buen funcionamiento, fiabilidad general, y facilidad en la ejecución del software.

V.1.1 Análisis de Requisitos del sistema

Aquí vamos a entender la función de la información de este proyecto. La finalidad es la de representar un sistema completo, juntando ambos componentes del mismo (hardware y software), mostrando la relación existente entre sus componentes, pero sin entrar en su estructura interna. También tenemos que identificar las funciones del sistema y asignarlas a algunos de sus componentes. Para esto se parte de los objetivos así como de las restricciones definidas.

Lo primero es delimitar el sistema, es decir vamos a establecer el funcionamiento así como el rendimiento deseado.

Después de realizada dicha delimitación, se debe proceder a la asignación de funciones. Tanto para el hardware como para el software. De acuerdo con nuestro sistema el desglose de dichas funciones sería el siguiente:

-Descripción

El nombre del proyecto **Reconstrucción Virtual de la Zona Arqueológica del Templo Mayor** comprende una parte del proyecto. El sistema en su totalidad va más allá de la simple reconstrucción tridimensional por computadora del sitio. Ya que se incluyen:

- Un recorrido virtual
- Un juego de recolección de piezas

La idea principal de este sistema, es simular un paseo por la zona del Templo Mayor, algo que nos sería imposible en nuestros días, ya que solo sabemos de esta zona por las maquetas y dibujos sobre la misma, pues físicamente solo se encuentran sus restos. Esto con la inclusión de un personaje, un caballero águila, lo cual es de gran ayuda para el usuario que este familiarizado con juegos de video, y para los que no, el programa cuenta con instrucciones fáciles sobre su uso.

La interacción se logra a través de diferentes tipos de dispositivos para poder movernos de un punto a otro dentro de la zona, ya sea mediante el uso de un joystick o del teclado y mouse.

- **Objetivo**

Desarrollar un paseo virtual y un juego dentro de la zona arqueológica del Templo Mayor, tomando importancia un fácil uso para el usuario, interacción y generación de los ambientes virtuales en tiempo real.

- **Interfaz**

Para facilitar el uso de este sistema se diseñaron distintas interfaces gráficas multimedia, las cuales apoyan las diferentes actividades que se tienen dentro del programa como: un juego, galería de imágenes, galería de sonidos, sitios web, controles y créditos.

- **Rendimiento**

Este dependerá del equipo de cómputo en el cual se ejecute el sistema. Es decir mientras el equipo tenga mayor velocidad y capacidad de memoria RAM y video, el rendimiento del sistema será mejor. En la siguiente lista se enuncian los requerimientos mínimos del hardware requerido para la buena ejecución del proyecto.

1. Sistema operativo Microsoft Windows XP
2. Procesador a 1 Ghz (recomendado 2Ghz)
3. 256 MB de memoria RAM (recomendado 512MB)
4. Unidad CD ROM y Bocinas
5. Tarjeta de video de 128 MB (recomendado 256 MB)
6. Monitor, Teclado, Mouse (recomendado joystick)

-**Restricciones**

El sistema funcionará correctamente si es ejecutado cumpliendo los siguientes puntos:

1. Equipo compatible (PC)
2. Sistema Operativo Windows Microsoft

V.1.2 Análisis de Requisitos del Software

En esta etapa se definen las funciones y el rendimiento del software, las interfaces con los demás componentes dentro del sistema, así como las restricciones que se deben cumplir.

Debemos centrarnos en problemas específicos: ¿Qué es lo que hay que hacer?, ¿Qué función se debe realizar?, ¿Qué datos utilizar?, ¿Qué interfaces?, y sobre todo ¿Qué restricciones tendrá?

Con la información recaudada de la especificación del sistema se debe definir los flujos así como la estructura de la información, la función del programa, y su protagonismo en el sistema, las características de la interfaz y restricciones del diseño. Con este conocimiento se deben sintetizar varias soluciones a diversos problemas que vayan surgiendo y que se ajustan al plan del proyecto.

El análisis de requisitos puede entenderse como un proceso en el que se realiza una síntesis de todo lo involucrado en el proyecto, esto es tanto a los problemas como a los objetivos, y partir del análisis de estos elementos sintetizar soluciones a los mismos.

- Especificación preliminar

En el desarrollo de un paseo virtual se hace necesario dividir sus elementos, dentro de los que tenemos: el modelado de los elementos geométricos para la ambientación y el de los objetos con los que se van a interactuar dentro de la navegación.

Para desarrollar el modelado de los objetos tridimensionales existe software con el cual nos podemos ayudar como 3D Studio Max, Maya, Lightwave, etc.

Para la navegación, existe en nuestros días software que tiene la capacidad de integrar geometría tridimensional con la programación, así como poder llevar a cabo una interacción dentro de un ambiente tridimensional, tal es el caso de algunos game engines (máquinas de juego) utilizados en el EI tales como Quake, Unreal Tournament, Half Life, etc.

Y tenemos otro tipo, los paquetes de desarrollo (kits) que integran la mayoría de las herramientas necesarias para el desarrollador, en las cuales se integran elementos importantes como iluminación, animación, controles, efectos especiales, física ambiental, inteligencia artificial, etc. Este tipo de software se utiliza mucho sobre todo en la industria del EI para el desarrollo de prototipos, así como productos finales y en la mayoría de las ocasiones son proporcionados por las grandes compañías directamente a los desarrolladores. Ejemplo de este tipo de software es Virtools.

En el caso de nuestro trabajo de tesis para el desarrollo del modelado, elegimos 3D Studio Max 5.0, debido a que en nuestra preparación académica se tomó el curso de Temas Especiales de Computación: Animación por computadora, en el que precisamente utilizamos este programa, además de contar con múltiples tutoriales en la red, libros especiales, cursos, etc. Otro aspecto importante sobre la preferencia en este software, es la popularidad por parte de desarrolladores especializados en la materia, lo cual significa que están disponibles “plug-ins” o programas externos que complementan al mismo 3D Studio Max.

Para la edición de imágenes como texturas y algunos mapas utilizados para la sombras se utilizaron dos editores de imágenes, primero el programa Paint de Microsoft, el cual daba solución a las operaciones sencillas de edición, de forma rápida. Pero si la textura necesitaba un tratamiento u efectos más complicados nos apoyamos con el programa Adobe Photoshop CS2 Versión 9.0.

En la edición del audio y video se utilizaron dos programas, el primero para el audio Goldwave y el segundo Movie Maker para editar las secciones de Introducción y Créditos.

Para el desarrollo de la navegación y la integración de medios elegimos Virtools, ya que es un software de fácil uso que permite reducir considerablemente los tiempos de desarrollo de aplicaciones 3D en tiempo real. Es muy utilizado en el campo de la RV y del EI.

Virtools es un conjunto de tecnologías para la visualización y la interacción en tiempo real, que cuenta con las siguientes características:

-Aplicación de creación.

Virtools es una aplicación que permite crear contenidos 3D interactivos. Es posible integrar la mayoría de los estándares multimedia como: objetos 3D, animaciones, imágenes y sonidos con el fin de hacerlos vivos gracias a su tecnología. Virtools no permite modelar objetos 3D, en cambio permite fácilmente crear cámaras cinematográficas, luces, curvas, elementos de interfaz así como frames 3D.

-Un motor de funciones

Una función simplemente es una descripción de una acción de cierto elemento dentro de un medio. Virtools ofrece una colección de funciones reutilizables que permiten crear más o menos todo tipo de contenido a partir de la interfaz gráfica del editor de scripts. Para la programación en su forma más estricta existe en complemento un lenguaje "Virtools Scripting Language" (VSL).

-Un motor de render

Virtools tiene un motor de render que permite dibujar las imágenes visibles en la interfaz 3D de Virtools Dev.

-Web player

El contenido Virtools puede ser hecho visible en Internet gracias al Web player gratuito que puede ser descargado por cualquiera.

-Un kit de desarrollo de software

Virtools Dev incluye un kit de desarrollo de software que permite acceder a ciertas partes de los procesos de comportamiento. Con SDK (Software Development Kit), podemos crear nuevos comportamientos (DLL), modificar las operaciones de los comportamientos existentes, escribir nuevos instrumentos (utensilios) con el fin de importar o de exportar nuevos tamaños, y poder modificar o reemplazar el motor de render.

Existen unos packs adicionales:

El VR Pack: permite administrar estereoscopia, cluster de PC y los periféricos de RV.

Physic Pack: permite crear la física para el juego. Es utilizado para crear juegos de coches, juegos de deslizamiento y otros juegos de deportes.

Server Pack: permite crear juegos en redes, aplicaciones con librerías de medios de comunicación de masa que hay que cargar.

Como vemos es un software muy completo para la integración de este tipo de proyectos, así como para juegos y otro tipo de software que involucre la simulación, es decir en la creación de medios totalmente interactivos.

-Ámbito de información

La función principal de cualquier software es el procesamiento de información sin importar su naturaleza, con esto se quiere decir que cualquier aplicación responde a una entrada de datos, procesamiento de los mismos, así como una visualización o salida de ellos. En nuestro caso la entrada se verá reflejada en movimiento de un stick en el joystick, al presionar una tecla o al mover el mouse; esta entrada será procesada y su resultado o salida se verá reflejada en el monitor para saber que es lo que sucede dentro de nuestro mundo virtual. Con esto vemos que los datos pueden contener diversos tipos de información, que puede ser numérica, lógica, imágenes, sonidos, cadenas de caracteres, etc.

A parte de los datos, también debemos considerar los eventos generados de la navegación, como la detección de colisiones, sincronización en las animaciones, reconocimiento de objetos, etc.

-Descomposición y Modelado del sistema de software

La descomposición es utilizada para dividir un problema en problemas más sencillos, para su fácil comprensión y solución, esto se hace la mayoría de las veces para abordar problemas demasiado grandes o con un nivel de complejidad elevado. Esta división se lleva a cabo de manera jerárquica ya que va de modelos generales a los más específicos en los que el nivel de detalle será mayor.

Los modelos del sistema de software quedarán representados en las tres acciones principales, la entrada y salida de información (modelo de datos), para las funciones dedicadas al tratamiento de la información (modelo de procesos) y en donde se define el comportamiento del sistema de acuerdo a las operaciones anteriores (modelo de control). Podemos representar la idea de estos modelos de manera gráfica, es decir con una serie de diagramas o de manera textual con el cual se logra una especificación más formal.

El diagrama general del sistema en donde se representan las relaciones entre sus elementos es el siguiente:

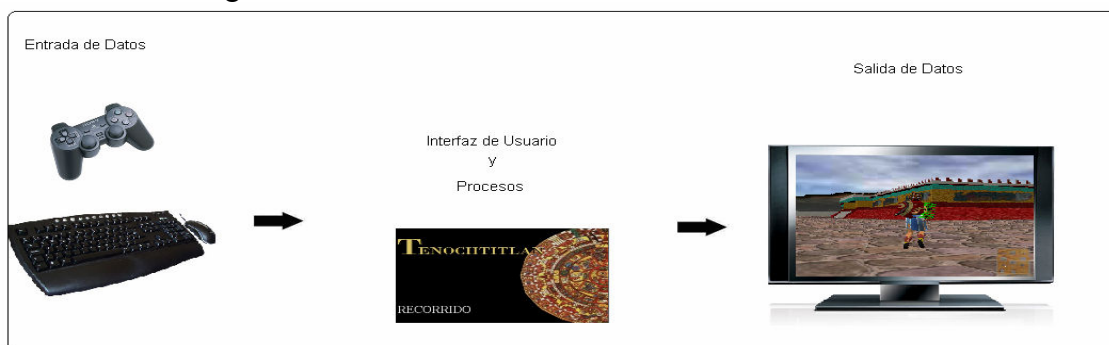


ILUSTRACIÓN V.1

Diagrama de representación general de la arquitectura

A continuación mostramos la descomposición del sistema en los tres diferentes modelos anteriormente mencionados, así como los elementos involucrados en cada uno de ellos.

Entrada y Salida de Datos (Modelo de Datos)

- Entrada de datos (teclado, mouse y joystick)
- Salida de datos desplegados en pantalla

Función del Sistema (Modelo de Procesos)

- Cambio de Cámaras (tercera persona, vista trasera y cámara libre)
- Controles
- Animación ambiental
- Detección de Colisiones
- Mapa

Control del Sistema e Interfaz de Usuario (Modelo de Control)

- Interfaz de Recorrido
- Interfaz de Juego

V.2 Planeación

La planeación se refiere a la etapa en la cual se desarrollan actividades relacionadas con la definición de objetivos, metas, métodos, tareas, tiempos, estándares (formas deseables en como debe darse un proceso), instrumentos, indicadores (formas de medición posterior de los resultados). También durante esta etapa se lleva a cabo un análisis de riesgos, además de definirse los recursos necesarios para el desarrollo del software.

Los indicadores antes mencionados son una herramienta importante ya que proporcionan datos sobre la viabilidad del proyecto tomando en cuenta los tiempos establecidos.

En resumen lo que se va a estudiar en esta etapa son los alcances del proyecto, para lo cual es importante tomar en cuenta cada una de las tareas a realizar para después descomponerlas en subconjuntos. La manera más sencilla de ayudar a que la planeación sea exitosa, es adecuar los tiempos en los que pueden realizar diversas tareas, esto dependiendo de su importancia dentro del proyecto, por ejemplo, no podemos realizar los procesos de colisiones si no tenemos completos los modelos 3D de la ciudad.

Por esto se optó por representar la planeación del proyecto mediante un cronograma en el que se representa el orden con el que se fueron efectuando las diferentes tareas que complementan al proyecto en su totalidad.

Cronograma

1. Análisis

- Análisis de Requisitos
- Análisis de Requisitos del Software
- Planeación

2. Diseño

- Diseño de Datos
 - Modelado
 - Navegación
- Diseño de la Arquitectura
- Diseño de Procedimientos
 - Diseño en diagramas
 - Pseudo código
- Diseño del audio
- Diseño de las interfaces
 - Menú Principal
 - Interfaz Recorrido
 - Interfaz Juego
 - Interfaz Pausa
 - Interfaz Galería

3. Desarrollo

- De las Estructuras Tridimensionales
 - Modelado
 - Texturizado
- Del personaje
 - Modelado
 - Texturizado
 - Animaciones
- Del Recorrido
 - Detección de Colisiones
 - Cámaras
 - Controles
 - Mapa
 - Iluminación
 - Ambientación
- Del Juego
 - Control del tiempo
 - Reconocimiento de objetos
 - Marcador en pantalla

4. Pruebas

- De las Estructuras Tridimensionales
- Texturizado eficiente
- Del recorrido
- Del juego

5. Mantenimiento

- De las Estructuras Tridimensionales
- Texturizado eficiente
- Del recorrido
- Del juego

V.3 Diseño

La etapa de Diseño es el proceso previo de configuración mental, en la búsqueda de una solución en cualquier campo o también lo podemos entender como la acción de plasmar el pensamiento de la solución mediante esbozos, dibujos, bocetos o esquemas trazados en cualquiera de los soportes, durante o posteriores a un proceso de observación de alternativas o investigación.

En esta etapa desarrollamos diferentes modelos de diseño para tareas específicas del proyecto y son las siguientes:

- Diseño de Datos. Fija las estructuras de la información o datos imprescindibles para el sistema.
- Diseño de la Arquitectura. Fija las relaciones entre los elementos estructurales del sistema.
- Diseño de Procedimientos. Indica las transformaciones de los elementos de la arquitectura del programa en una descripción referente al funcionamiento.
- Diseño de la Interfaz. Define la forma en la cual se comunica el software con el usuario.

A continuación se explica el desarrollo de los diseños antes mencionados explicando a detalle la intención de cada uno de ellos.

V.3.1 Diseño de Datos

Dentro del diseño de datos se crearon dos divisiones, los datos involucrados dentro de la etapa del modelado geométrico y la otra que obedece a la etapa de navegación.

Quedando ordenadas de la siguiente manera:

- Modelado Geométrico (Estructuras y personaje)
 1. Lista de Referencias
 2. Lista de Texturas
- Navegación
 1. Lista de sonidos
 2. Lista de iluminación
 3. Lista de objetos para colisiones
 4. Lista de modelos geométricos
 5. Lista de Cámaras

En lo referente a diseño de datos en el modelado, es importante mencionar que para la elaboración de los modelos geométricos, se necesitan medidas de las diferentes edificaciones a representar de manera tridimensional. Desgraciadamente como sabemos

no podemos obtener medidas reales de un lugar destruido en su totalidad, pero gracias a aportaciones arqueológicas así como investigaciones sobre la apariencia real de la zona arqueológica del Templo Mayor, contamos con una maquetas situadas en el Metro Zócalo, Museo de Antropología y en el Museo del Templo Mayor que es la más reciente y en la que nos apoyaremos para la reconstrucción virtual.

Una vez tomada como referencia la maqueta del sitio, se realizó la tarea, de tomar fotografías en todos los ángulos para cada uno de sus edificios por separado. Esto nos da la información necesaria para realizar un modelo completo, con dimensiones y proporciones, lo cual nos ayuda a aproximarnos a la idea que se tiene sobre la apariencia real del sitio.

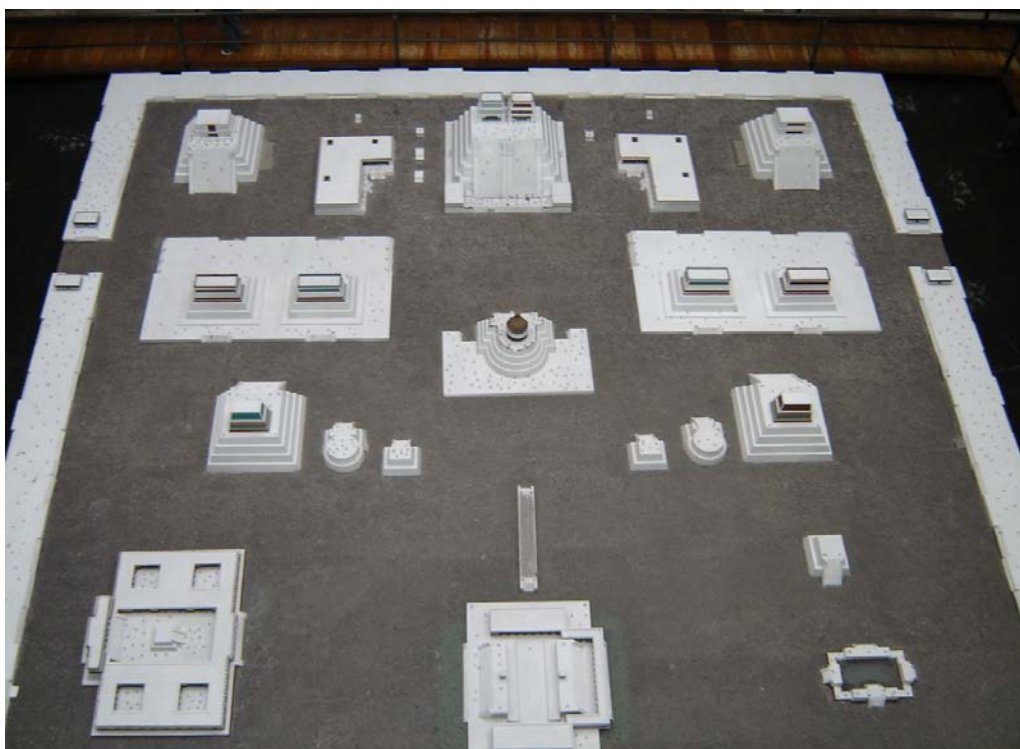


ILUSTRACIÓN V.2

Fotografía de la maqueta en su totalidad.

En la maqueta se cuenta con todos los recintos que se piensa guardaban esta ubicación, tomando como principal referencia el Templo Mayor, el cual era el templo principal de toda la zona, y se encuentra hacia el norte de la ilustración anterior.

Después de tomadas las fotografías procedimos a editar las imágenes que se necesitarían para el modelado 3D.

La edición se enfocaba prácticamente en obtener tres vistas básicas para el modelado tridimensional:

- Vista frontal
- Vista lateral
- Vista superior

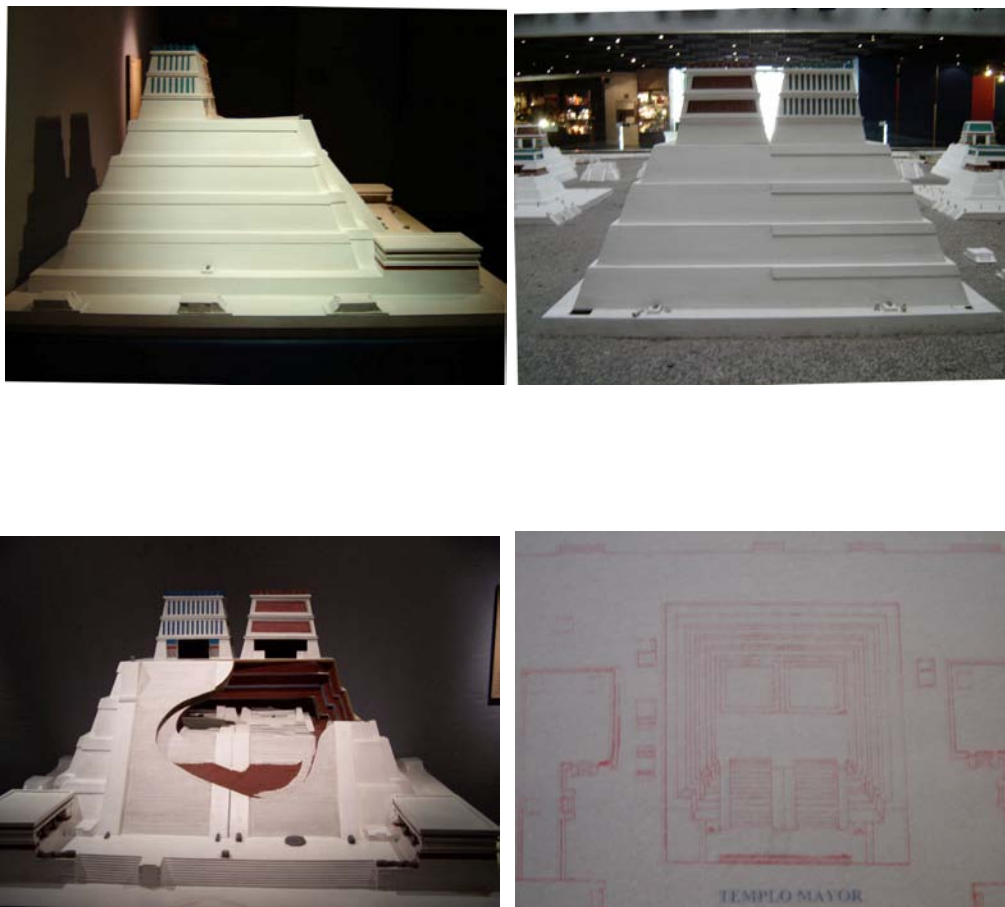


ILUSTRACIÓN V.3
Imágenes de cada una de las vistas
de las fotografías tomando como ejemplo las del templo mayor.

V.3.2 Lista de Imágenes de Referencias

Estas imágenes de referencia indican los resultados de la edición, ya que una vez modelados cada uno de los edificios, estos necesitan una ubicación dentro del espacio virtual, tomando como referencia principal la ubicación del Templo Mayor, que se encuentra al norte de zona de este espacio virtual¹, la cual se subdividió en sectores para una fácil identificación, mediante un sistema de referencia parecido a un tablero de ajedrez, en las posiciones horizontales vamos a identificar los sectores de la A a C y de manera vertical obedecerá a posiciones del 1 al 4, quedando cada uno de los conjuntos de la siguiente manera:

¹ En la realidad se encontraba orientado hacia el este

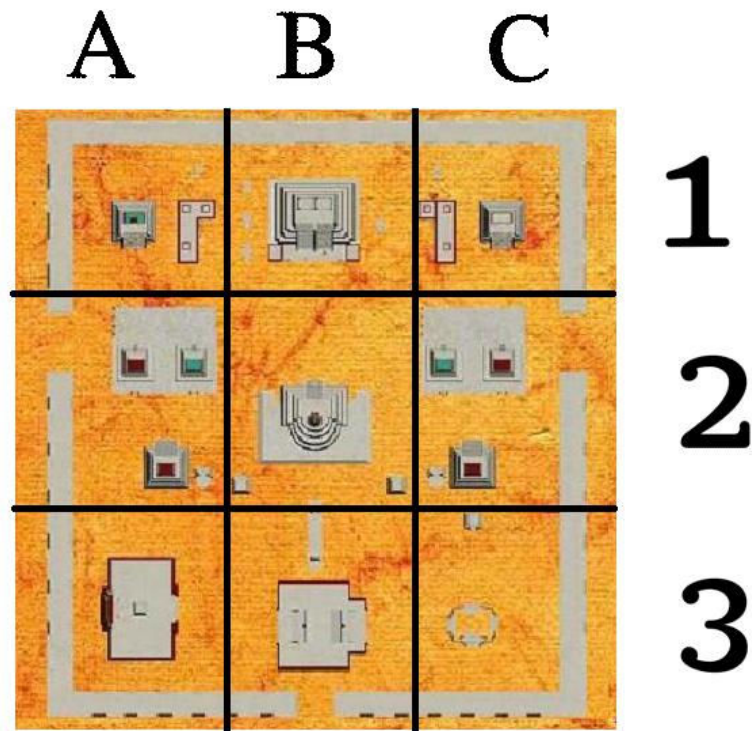


ILUSTRACIÓN V.4
Dibujo general de la zona con la división de ajedrez

Edificio	Locación
1. Tezcatipocla Pirámide Izquierda	A1
2. Casa de los Guerreros Jaguar	A1
3. Templo Mayor	B1
4. Tezcatipocla Pirámide Derecho	C1
5. Casa de los Guerreros Águila	C1
6. Coacalco	A2
7. Cihuacoatl	A2
8. Pirámide?	A2
9. Quetzalcoatl	B2
10. Chicomecoatl	C2
11. Xochiquetzal	C2
12. Tonatiuh	C2
13. Calmecac	A3
14. Tzompantli	B3
15. Juego de Pelota	B3
16. Tozpalatl	C3



ILUSTRACIÓN V.5

Dibujo con la orientación real del sitio

V.3.3 Lista de texturas

Para cada uno de los modelos terminados en su geometría, es necesario pasarlos a la siguiente fase, el texturizado. Esta es una de las etapas importantes del proyecto, ya que aquí es donde puede darse mayor realismo al paseo virtual y sin emplear mayor geometría que la necesaria.

Ahora cada modelo tendrá que utilizar no solamente una textura, para lo cual necesitamos una lista de materiales a utilizar, parecido a lo que sería la paleta de un pintor al óleo.

Este trabajo se realizó con lo recabado en las fotografías de la maqueta, en la cual se capturaron los diferentes materiales, así como las coloraciones de los mismos, teniendo entre ellos el estuco en múltiples tonalidades, otras como madera, piedras, pasto, paja y algunas otras imágenes editadas para dar mayor realismo y que forman parte de ésta lista.

Para la incorporación del personaje (Caballero Águila) se eligieron texturas de acuerdo con la vestimenta de esa época, también se incorporaron texturas para la piel, metal, plumas, tatuajes, etc.

Entonces, tendremos dos listas una para las texturas de la ciudad así como el ambiente y otra para el personaje con el cual vamos a interactuar dentro del espacio tridimensional.

La mayoría de estas imágenes utilizadas para el proceso de texturizado están en formato JPEG adecuadas a un tamaño pequeño, pero con el cual no se pierda en calidad y resolución, ya que gran parte de procesamiento se ve reflejado en el trabajo de texturas y si las trabajamos de esta manera se puede ahorrar bastante en cuanto a memoria RAM se refiere. Esto quiere decir que la mayoría cuenta con una profundidad de 16 bits y sin exceder los 256 píxeles tanto a lo largo como a lo ancho.

Existen algunas imágenes a las cuales es necesario realizar una transparencia, a estas imágenes se les puede tratar desde el programa para la navegación (Virttools), Utilizando el formato PNG el cual nos permite modificar el canal Alpha de la imagen, así obtuvimos el efecto deseado con mayor calidad que con una imagen JPEG.

A continuación mostramos como ejemplo las listas de materiales utilizados para un edificio en particular así como para el personaje.

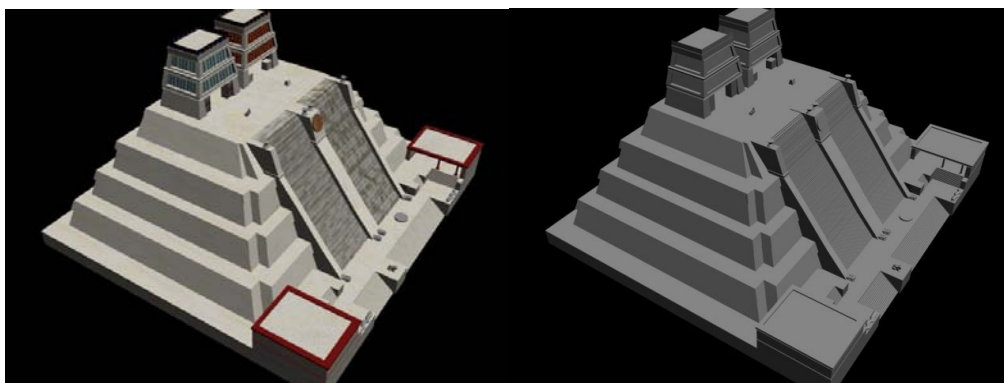


ILUSTRACIÓN V.6
Templo Mayor con y sin textura

Lista de Materiales para el templo mayor

1. Almena1
2. Almena2
3. GreekRoman021
4. GreekRoman021Rojo
5. Mural_Templo_Mayor
6. Relieve_Tlaloc
7. texturatemplomayorcalaveras
8. estucomugrosorelieve4

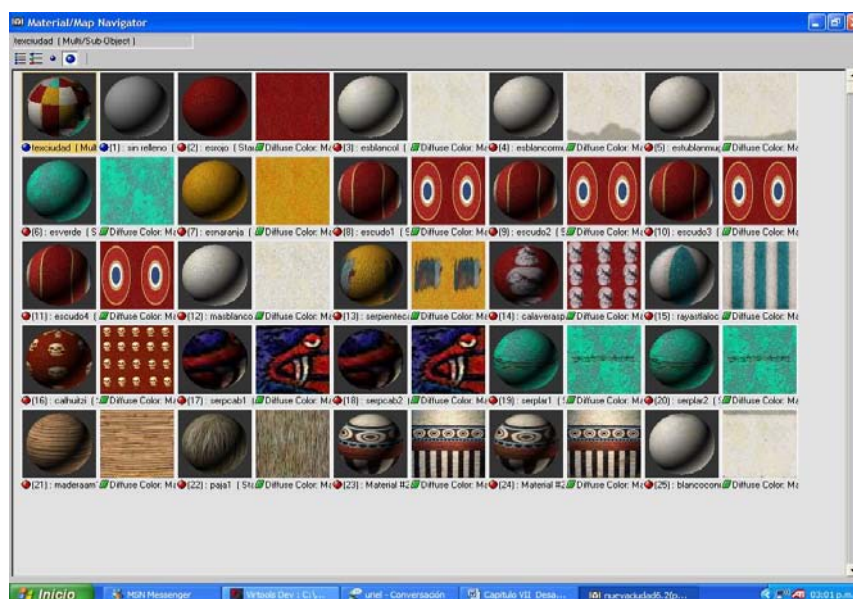


ILUSTRACIÓN V.7
Catálogo de Texturas y Materiales del Templo Mayor



ILUSTRACIÓN V.8
Personaje sin y con texturas

Lista de Materiales para el personaje

1. 3dsk012
2. african013
3. armadura dorada
4. armour001
5. armour001rojobrillante
6. armour001rojobrillante
7. armour005
8. celtic_001
9. ebilla
10. Emblema
11. metalpico
12. piel
13. pielcafe
14. pielfl
15. pielroja
16. plumascabeza
17. plumascabezanaranja3
18. tattoo2
19. telablanc

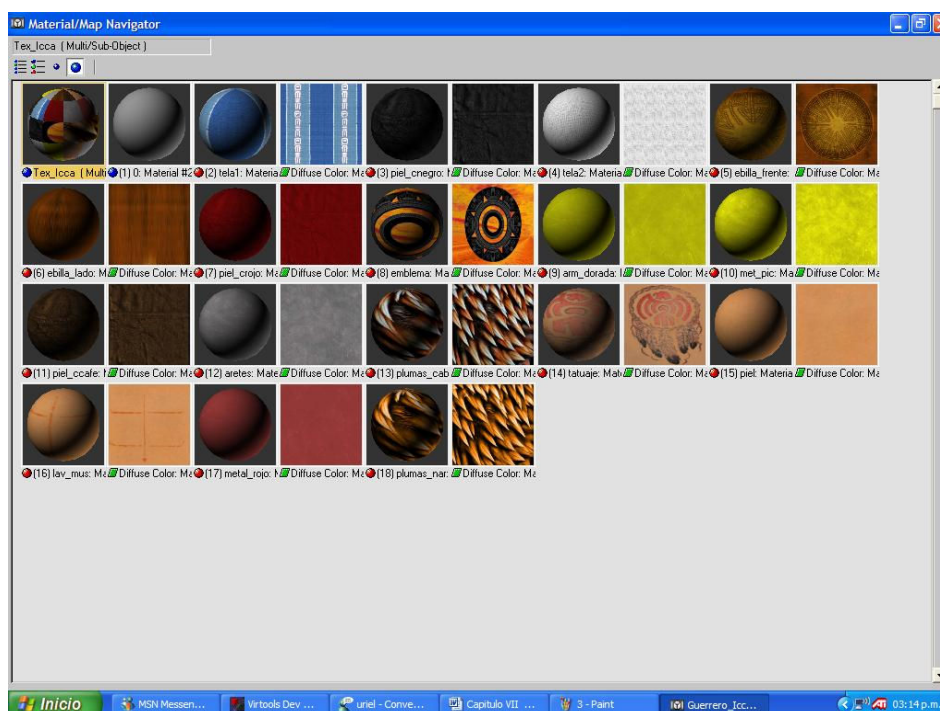


ILUSTRACIÓN V.9
Catálogo de texturas del Personaje

V.4 Diseño del Personaje

Para el diseño del personaje se tomó en cuenta el aspecto de nuestros antepasados desde el punto de vista morfológico, así como la piel y vestimentas, sin ser estrictamente parecidas a las reales o las que se pensaban que vestían en esos tiempos. La decisión de que se tratara de un guerrero águila, no fue difícil de tomar, porque como sabemos el pueblo azteca era dedicado en gran porcentaje de sus actividades a la guerra, y este tipo de guerreros eran un grupo de élite especializados en la exploración, el espionaje y en transportar mensajes especiales de carácter militar.

Sobre el diseño estructural del mismo nos basamos en imágenes y figuras de representaciones de diferentes caballeros águila, pero guardando las características importantes. Como se muestra en las siguientes ilustraciones:



ILUSTRACIÓN V.10
Imágenes de Referencia para el Caballero Águila



ILUSTRACIÓN V.11
Imagen de nuestro caballero águila
en varias vistas desde 3D Max

Basados en este tipo de imágenes diseñamos un guerrero que incluía su escudo y maquahuatl², los cuales se crearon a detalle y son fieles reproducciones de armas de la época.



ILUSTRACIÓN VI.12
Imagen del escudo y arma en 3D Max

Para la cabeza decidimos que no mostrara un rostro en particular, para que el usuario que lo utilice se adentre en el espacio virtual simulando que es el mismo, además de darle un toque de misticismo, característica propia de este pueblo.



ILUSTRACIÓN V.13
Imagen de la cabeza del guerrero águila.

V .4.1 Animaciones del Caballero Águila

El trabajo de darle vida al caballero águila fue uno de los más estrictos en tanto a su elaboración, ya que para que las animaciones guarden naturalidad es importante hacer diferentes ensayos y comparar una y otra vez con los movimientos en la realidad. El personaje guarda animaciones para distintas reacciones dentro del mundo virtual las más importantes se dan en cuanto a los movimientos básicos de una persona, permanecer de pie, caminar y correr, así como de animaciones secundarias que son utilizadas dentro del juego.

² Arma azteca parecida a un mazo

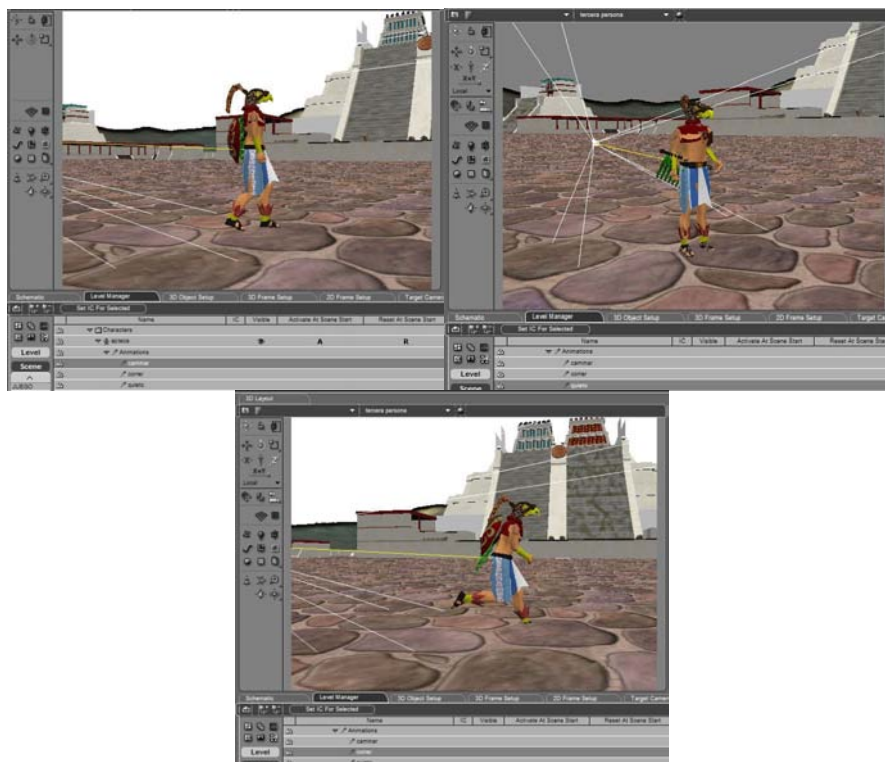


ILUSTRACIÓN V.14
Animaciones parado, caminar y correr

Para ejemplificar la elaboración de una animación tomemos por ejemplo la de correr, para trabajar las animaciones nos apoyamos en 3D Studio Max. Lo primero que se debe realizar después de terminar el modelado y texturizado del personaje, es asignar “huesos” al mismo y esto se logra con un plug-in de este programa llamado Character Studio el cual permite agregar un esqueleto a la medida para cualquier personaje bípedo.

Después de esto se procede a establecer valores de jerarquía a cada uno de los puntos que conforman, cada una de las partes del cuerpo del personaje, esto se hace para que al flexionar las extremidades no queden puntos rezagados y así tener un movimiento con mayor naturalidad.

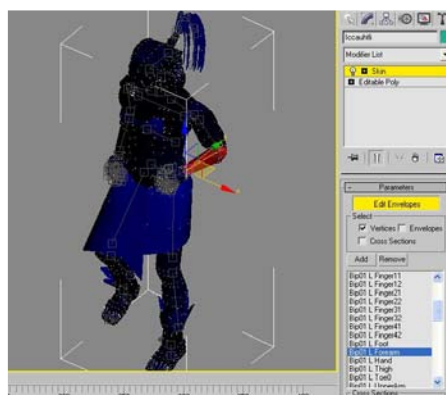


ILUSTRACIÓN V.15
Imágenes del Skin del personaje

Por último nos ayudamos con imágenes de cada una de las posiciones para el movimiento foto por foto tomando como referencia el personaje de un juego de video (Kratos de God of War II), con cada una de estas imágenes se logró conjuntar un movimiento que nos da la sensación de velocidad en comparación con la acción de caminar.

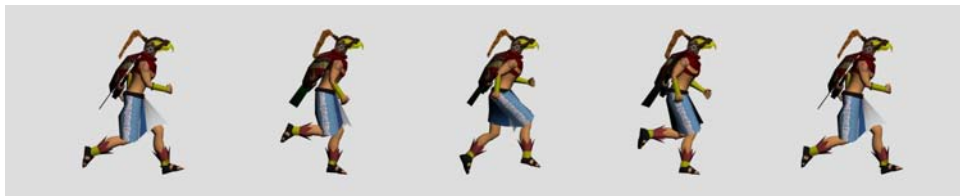


ILUSTRACIÓN V.16
Imágenes de la animación
de correr paso por paso

V.5 Diseño del Recorrido Virtual y Juego

Una vez que contamos con los elementos antes mencionados (estructuras y personaje texturizados además de las animaciones del mismo), dentro del diseño se establecen las reglas de nuestro mundo y dependiendo de este sabremos el nivel de interacción a utilizar. En otras palabras aquí es donde le vamos a dar vida al mundo virtual y donde se van a desarrollar elementos que lo hacen más rico. Para el recorrido los elementos a diseñar son los siguientes:

- Detección de Colisiones
- Cámaras
- Controles
- Mapa
- Iluminación
- Ambientación

En el juego vamos a contar con la mayoría de los elementos del recorrido virtual, con la diferencia de que aquí el recorrido no es libre, ya que se es necesario idear la mejor ruta posible para encontrar los objetos que sean requeridos, dependiendo de la dificultad seleccionada, en un tiempo determinado, para facilitarle al usuario esta tarea las piezas recogidas se presentaran en un marcador, así como en un mapa que nos da la ubicación de cada una de ellas.

Dentro de los elementos que se tiene que diseñar para la ejecución de esta tarea son los siguientes:

- Control del tiempo
- Reconocimiento de objetos
- Marcador en pantalla
- Contador de unidades
- Mapa con ubicación de las piezas

V.6 Diseño sobre el modulo de detección de colisiones

Dentro de nuestro mundo virtual nosotros podemos realizar cualquier tipo de tarea, como este dentro de nuestra imaginación, entonces lo que debemos tener en claro son “las reglas” o leyes que se van a cumplir, una de ellas son las colisiones.

Para generar un buen paseo virtual dentro de un espacio es necesario tomar en cuenta los “choques” o colisiones entre los objetos con los que se va a interactuar, los cuales en nuestro caso son piezas, cámaras, edificios y el personaje que manejamos. Esto sirve para dar mayor realismo, ya que sin esta facultad podríamos atravesar las paredes y en ocasiones quedar dentro de los edificios sin poder subir o bajar de ellos.

También dentro de nuestro espacio se construyeron paredes “invisibles”, es decir se encuentran dentro del espacio tridimensional pero durante la ejecución del programa permanecen escondidas, estas paredes nos sirven para delimitar la zona arqueológica así como para evitar que el personaje caiga precipitadamente cuando se encuentre en los límites de un piso en la parte alta de algún templo.

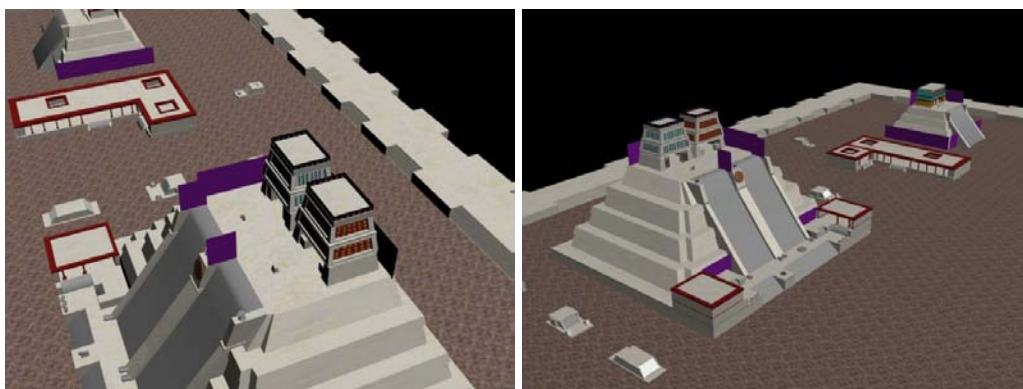


ILUSTRACIÓN V.17
Imagen de las paredes invisibles “visibles”

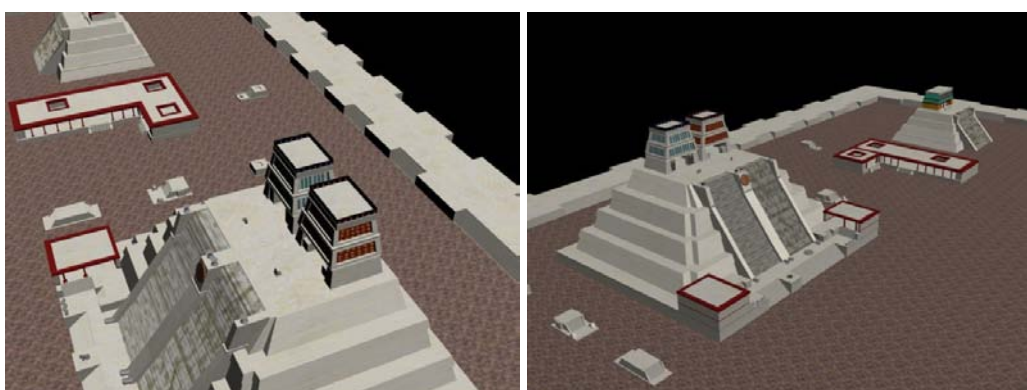


ILUSTRACIÓN V.18
Imagen de las paredes invisibles “invisibles”

Ahora para poder caminar así como para poder subir o bajar de un templo se diseñó la construcción de pisos, con la ayuda de planos en distintas posiciones, ya que a parte de los pisos normales tenemos rampas que permanecen ocultas detrás de cada una de las escaleras de los templos, esto para ayudar al personaje a subir con facilidad y con más naturalidad.

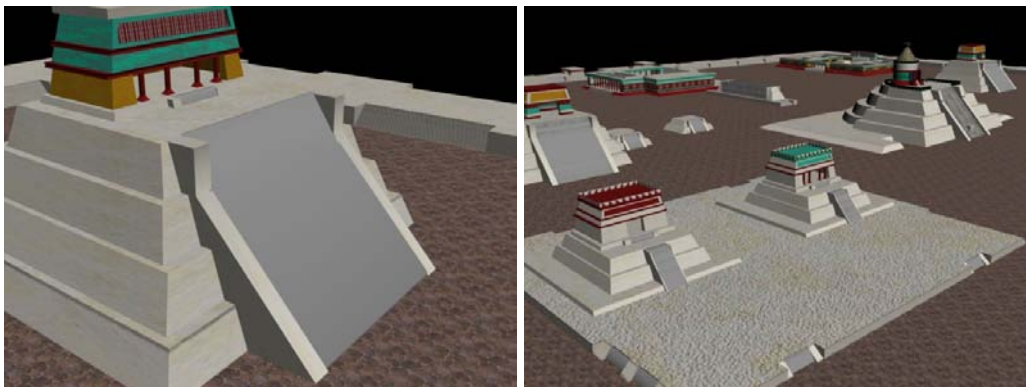


ILUSTRACIÓN V.19
Imagen de las rampas invisibles

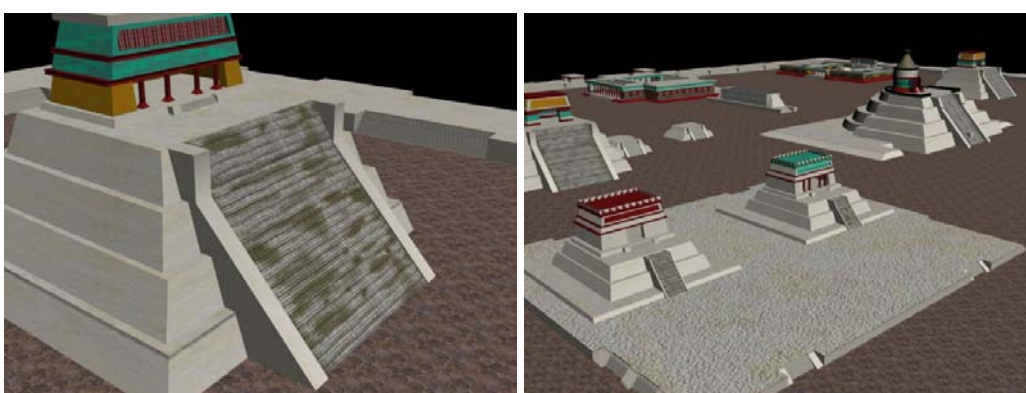


ILUSTRACIÓN V.20
Imagen sin las rampas invisibles

V.7 Diseño de la Iluminación

La iluminación es un elemento que ayuda a crear una atmósfera apropiada dentro del espacio virtual, ya que con ella podemos crear diferentes climas, porque que nuestro mundo se visite de día o de noche o con el simple hecho de entrar a un cuarto y notar la diferencia con el exterior.

En nuestro proyecto la iluminación, la trabajamos mediante la incorporación de luces, que pueden ser agregadas desde Virtools, aunque también se pudieron exportar desde 3D Studio Max, pero al poder trabajarlas en Virtools podíamos cambiarlas fácilmente para cualquier arreglo, sin la necesidad de exportarlas nuevamente desde el programa modelador utilizado.

En Virtools contamos con tres tipos diferentes de luces:

- Puntual
- Direccional
- Spot

Cada una de estas luces tienen parámetros particulares a cambiar, pero mantienen las principales dentro del espacio 3D, una de ellas es la posición dentro del sistema coordinado del mundo o en el caso de tenerse un sistema coordinado local se fija al mismo. Otras características importantes son la orientación expresada en grados y la intensidad de la luz.

Ahora cada una de ellas tiene características particulares.

Puntual: Esta luz se proyecta en todas las direcciones, con acción similar a un foco doméstico.

Spot: Se utiliza un cono de luz el cual solo ilumina a los objetos que se encuentren dentro.

Direccional: Esto es equivalente a usar una fuente de punto en una distancia infinita. Esto es similar al sol visto desde una escala humana.

A cerca de los parámetros que pueden ser cambiados tenemos, el color de la luz donde se definen los componentes rojos, verdes y azules. Cada componente puede llevar un valor que va de 0 a 255, el defecto es 255, 255, 255 (blanco).

La intensidad define el radio del área de iluminación y por último tenemos el factor de atenuación con lo cual la intensidad de luz baja más rápidamente con la distancia si se aumenta este.

Para que todos los ángulos de la ciudad quedaran iluminados, optamos por colocar luces direccionales desde cada uno de los lados, para iluminar cada una de las caras de los edificios, ninguna de estas luces proyectan sombra solamente es para dar mayor iluminación, la única luz que proyecta sombra es una luz direccional inclinada colocada en la parte de atrás del Templo Mayor la cual también nos ayuda a dar el efecto de la luz de sol al amanecer.

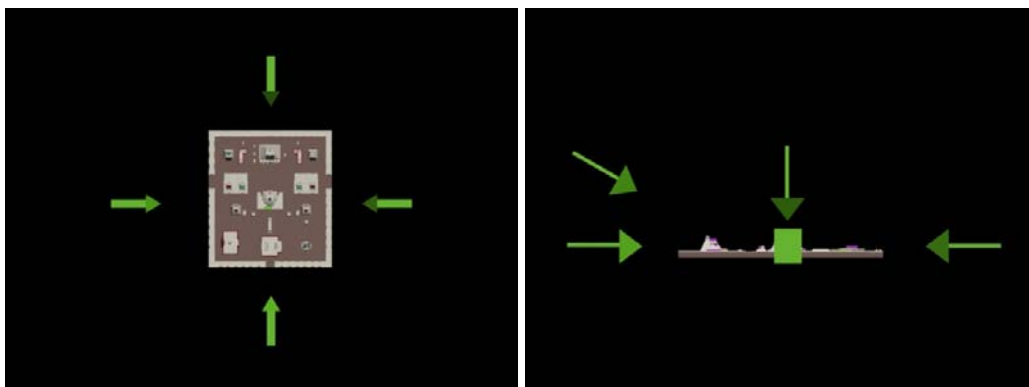


ILUSTRACIÓN V.21
Posición de las luces desde diferentes vistas

V.7.1 Diseño de sombras

En la realidad al postrarnos frente a una fuente luminosa, se proyecta una sombra de nuestra persona sobre el piso o cualquier otra superficie cercana a nosotros, de la misma manera se crearon sombras sobre cada una de las superficies, que así lo necesitaran dentro del espacio virtual. Principalmente de los edificios sobre el piso y sobre si mismos.

La construcción de estas sombras consiste en un mapeo sobre la superficie que recibe la sombra por parte de la superficie generadora, para lo cual tenemos tres factores involucrados:

1. Luz
2. Objetos que reciben
3. Objetos que proyectan

Ahora con estos tres elementos involucrados se dispone a separarlos para depositarlos cada uno dentro de grupos los cuales van a ser:

1. Luces
2. Reciben
3. Proyectan

La función que utilizamos para la generación de sombras dentro del espacio virtual es Static Light Map.

La utilidad de la función Static Light Map es la de crear como su nombre lo dice mapas o imágenes que se generan sobre la textura de cada una de las superficies que reciben sombra y sustituirlas por las actuales. Esto solo debe de hacerse una vez, ya que se genera nueva geometría así como materiales y texturas.

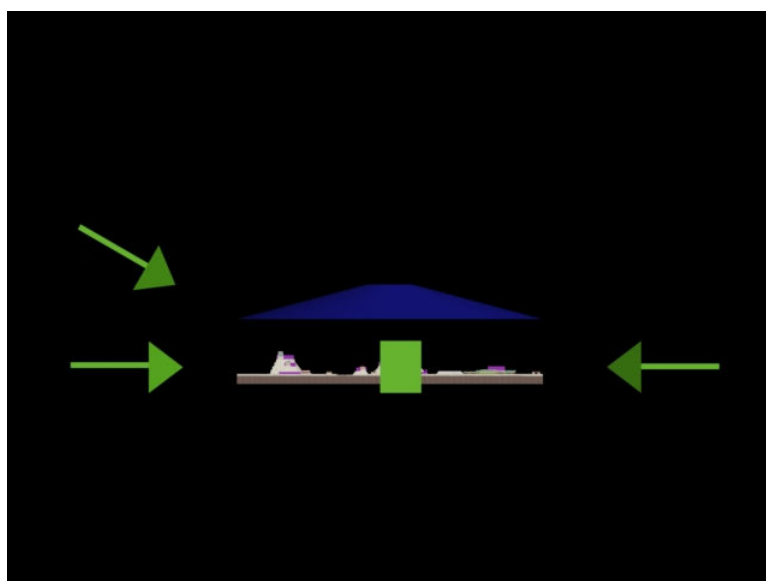


ILUSTRACIÓN V.22
Posición de las luces para las sombras

Las luces estaban orientadas hacia el norte para ser mas precisos por detrás del Templo Mayor, el objetivo de esto es para simular al sol y que las sombras quedaran con el mismo ángulo para todo el sitio.

Para la creación de las sombras se reemplazo la luz direccional que apuntaba perpendicular hacia el piso de la zona y se cambio por una luz del tipo spot la cual solo ilumina en un radio determinado, de esta manera logramos que las sombras fueran más tenues y menos oscuras.

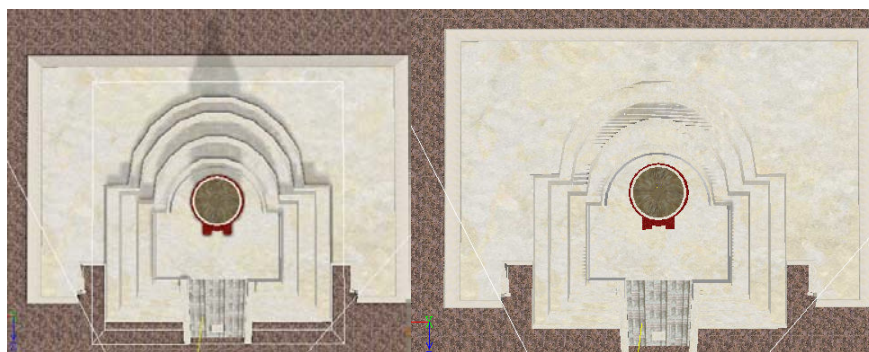


ILUSTRACIÓN V.23
Templo de Quetzalcoatl con y sin sombra

Por último si una sombra no convence del todo se editará con la ayuda del programa Photoshop, ya que en ocasiones se oscurece demasiado o no se lograba el tono adecuado para algún edificio en particular. Así después de haber sido editada una sombra se dispone a recargarla dentro del ambiente.



ILUSTRACIÓN V.24
Ejemplo LightMap

Para la sombra del personaje se implementó la función Simple Shadow, la cual utiliza una textura editada para crear la sensación de una sombra real, la cual se mueve cuando el personaje lo hace, a esta función simplemente se le incorpora el objeto generador de sombra así como el material que va a representar la sombra del personaje, el efecto funciona en todo aquello que hayamos declarado como piso.



ILUSTRACIÓN V.25

Textura de sombra, personaje sin sombra y personaje con sombra

V.8 Diseño del ambiente

A parte de la ciudad se dispuso a crear un limite dentro de nuestro mundo es decir lo que esta después del recinto. Esto ayuda al recorrido a verse mejor, incluyendo elementos como montañas y el cielo.

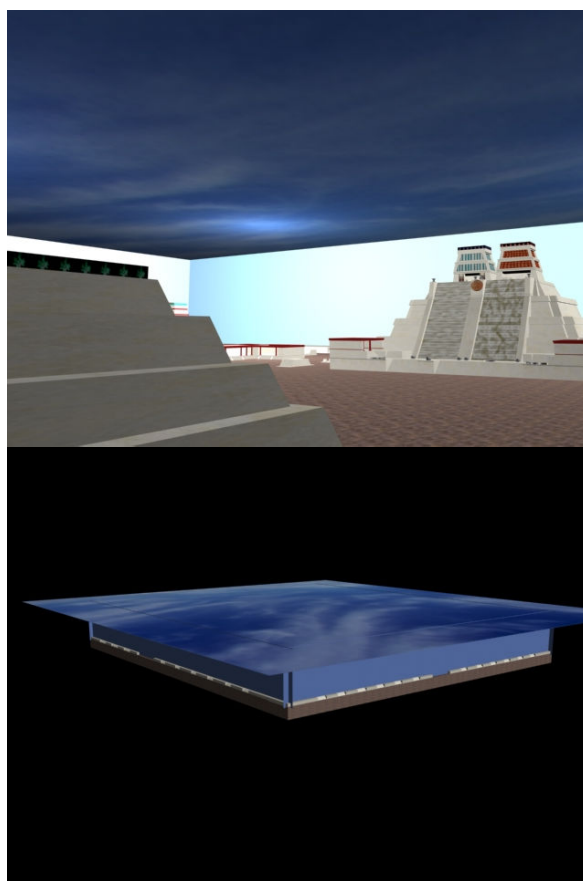


ILUSTRACIÓN V.26

Efecto del cielo

Se incluyo a los alrededores de la zona de los templos un pequeño lago, como sabemos en esos días, la ciudad de México estaba formada por lagos que fueron secados por los españoles. Ese efecto fue creado con una textura arreglada sobre un polígono a manera de tapa y cambiando sus parámetros entre ellos el color y el nivel de transparencia, para crear el efecto de agua, el cual también cuenta con movimiento gracias a una función

para darle animación a la textura conocida como Texture Sine. Para crear el efecto de profundidad se utilizó otra función conocida como Volumetric Fog, que en realidad sirve para crear niebla pero que arreglamos para este efecto.



ILUSTRACIÓN V.27
Vista superior de la ciudad
con efecto de agua a su alrededor

V.9 Diseño de la Arquitectura

La finalidad de este diseño es la de representar gráficamente como se van a relacionar los distintos módulos, así como el comportamiento de la información a lo largo del proyecto.

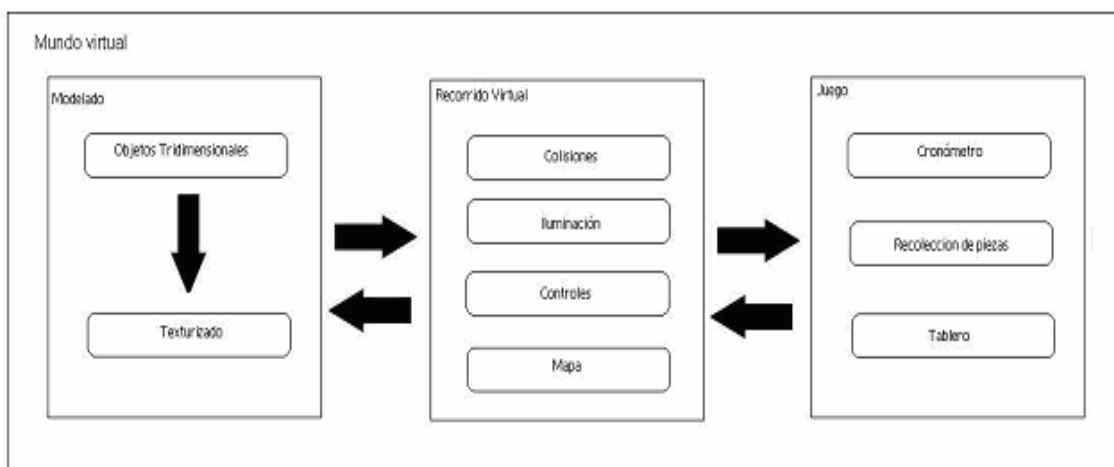


ILUSTRACIÓN V.28
Esquema del diseño arquitectónico

En el diagrama anterior se presentan tres grupos: modelado, el recorrido virtual y se incluye un tercero el cual es el juego. El modelado se subdivide en los Objetos Tridimensionales dentro de los que se encuentran incluidos: las piezas, edificios, ambiente, el personaje, y el texturizado de todos los objetos. El segundo grupo se divide en iluminación, colisiones, controles, cámaras, mapa.

Por último el tercer módulo, el juego que trabaja con los elementos del recorrido, pero su objetivo es diferente, ya que tiene agregados elementos extras como un cronómetro y un módulo para el reconocimiento de objetos.

Lo que se muestra en el esquema, es la manera en la que interactúan los módulos, en pocas palabras como se transmite la información de uno a otro. Pero también observamos que cuenta con flechas de regreso hacia cada uno de los módulos, lo cual nos ayuda en caso de que encontremos errores en etapas anteriores.

Como una analogía a un rompecabezas, el cual tiene como objetivo reunir todas las piezas que lo conforman en un solo objeto, en este caso, nuestro objeto es el programa que estamos realizando y para completarlo es importante analizar la relación de una pieza con otra, y que cada uno de los módulos se integren al proyecto final.

V.10 Diseño de Procedimientos

El diseño procedimientos ayuda a la programación del proyecto proporcionando información fundamental. Esto se logra con la incorporación de herramientas gráficas u escritas con las cuales podemos representar cualquier comportamiento o suceso que vaya a ocurrir dentro del espacio virtual, estas herramientas deben ayudar a que los objetivos queden perfectamente definidos, sin caer en lo ambiguo y con un número determinado de pasos, con la finalidad de obtener el mejor resultado a el problema planteado, parecido a lo que conocemos como diagrama de flujo.

A continuación a manera de ejemplo presentamos el diagrama de flujo a lo que se refiere a la función del mapa.

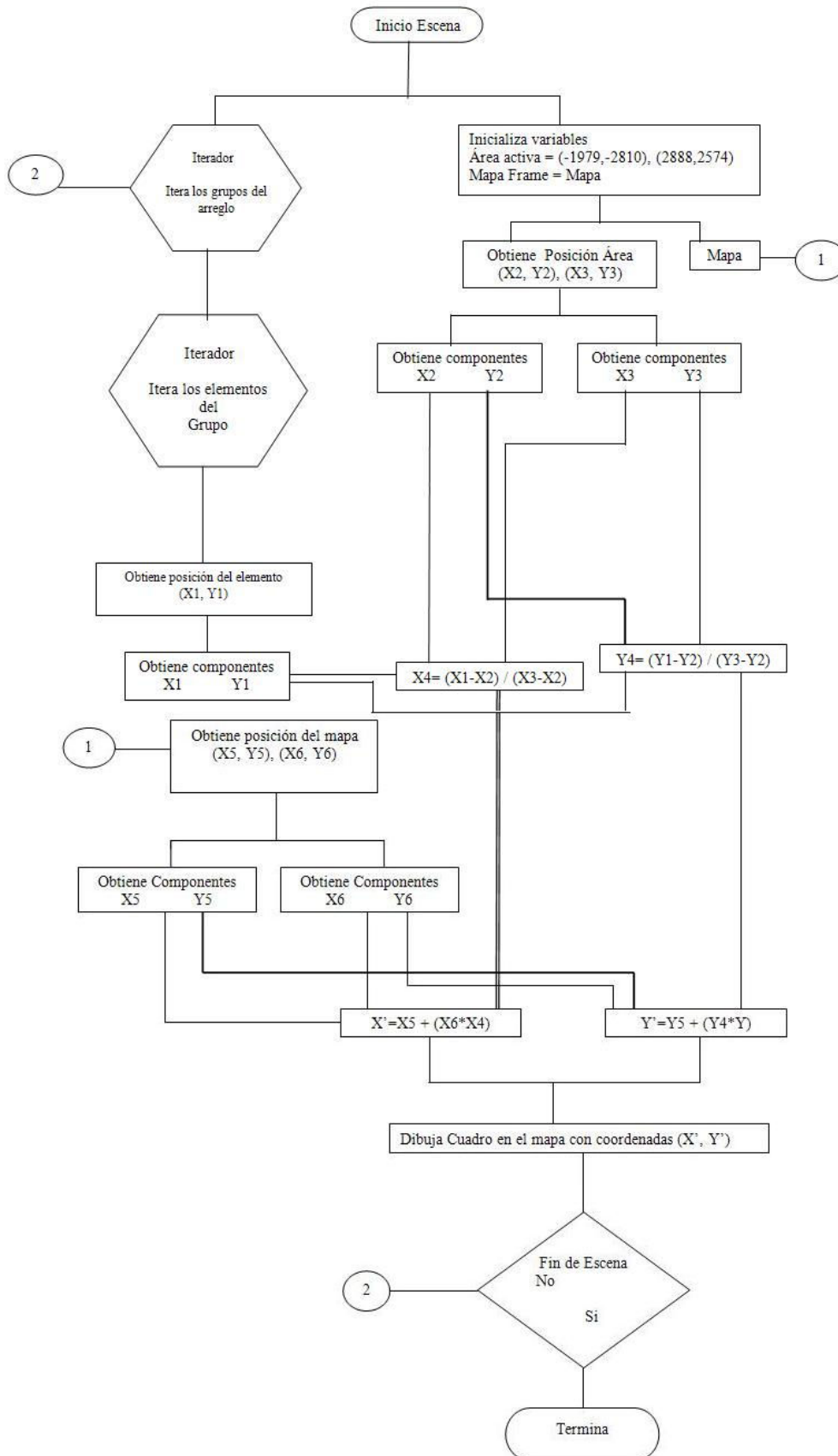


ILUSTRACIÓN V.29
Diagrama de Flujo para la función del Mapa



ILUSTRACIÓN V.30
Mapa de la Zona Arqueológica

Los diagramas no son la única herramienta con la que contamos para el diseño procedural, también un algoritmo es posible representarlo de forma escrita, esto es, realizando las famosas pruebas de escritorio, con las cuales de forma escrita representamos casos particulares que ocurren dentro del espacio virtual, lo cual fue de utilidad para el cálculo sobre la zona activa para el mapa.

V.11 Diseño del audio

Esta etapa es fundamental para el desarrollo del proyecto, ya que es aquí en donde integraremos un sentido más, el oído.

Lo primero que debemos hacer es encontrar el audio que permita ambientarnos en la época prehispánica, a lo cual debemos contar con los siguientes tipos de sonidos:

- **Música:** Esta debe ir de acuerdo con la época, ya que es lo que se va a oír a lo largo del recorrido virtual y del juego.
- **Sonidos de Objetivos:** Aquí se van a buscar aquellos sonidos que nos indiquen las acciones dentro del programa, en nuestro caso el de un tambor que se active cuando tomamos una pieza dentro del juego, así como el sonido de los pasos del personaje y el sonido sobre las opciones seleccionadas en el Menú.
- **Voces:** Estos son sonidos grabados por un actor, que ayudan al usuario a saber en que templo se encuentra, o también da el mensaje cuando pierdes o ganas dentro del juego.

Una vez que contamos con estos tres diferentes tipos de audio, lo siguiente es la edición del mismo, mediante el programa Gold Wave, en el cual se ajusta el tiempo de duración del audio requerido (caso específico del juego que necesita 7 minutos), se reduce el ruido para aquellos sonidos que fueron grabados, así como buscar el formato adecuado para Virtools (en este caso WMA).

V.13 Diseño de la Interfaz

Aquí es donde se realiza la separación de las diferentes actividades que se podrán realizar dentro de nuestro programa, las cuales son:

- Recorrido
- Juego
- Galería de Dioses
- Galería de Sonidos
- Sitios Web
- Controles
- Créditos

Ahora la interfaz gráfica está diseñada de forma que al usuario, le sea sencillo manejar el programa.

Lo primero que vamos a visualizar es el Menú de inicio, aquí es donde el usuario va a poder elegir entre cada una de las actividades antes mencionadas.



ILUSTRACIÓN V.31
Imagen del Menú principal

Ahora vamos a explicar cada una de las actividades con una breve descripción de sus objetivos.

Recorrido: Nos lleva directamente al recorrido virtual libre sobre la ciudad, con la interfaz del mapa, que nos ayuda a obtener nuestra ubicación dentro del espacio virtual, como lo presenta la pantalla siguiente.



ILUSTRACIÓN V.32
Imagen del recorrido

Juego: Aquí la interfaz cambia incluyendo un marcador de tiempo, así como la cantidad de piezas recolectadas y el mapa con la ubicación de cada una de las piezas.



ILUSTRACIÓN V.33
Imagen del juego

Galería de Dioses: Tiene la finalidad de mostrar imágenes de los diferentes dioses y guerreros que tenían representación dentro del recinto sagrado.



ILUSTRACIÓN V.34
 Imágenes de la interfase de la galería de dioses

Galería de Sonidos: Aquí es posible escuchar la música utilizada dentro del programa.



ILUSTRACIÓN V.35
 Imágenes de la interfase la galería de sonidos

Sitios Web: De aquí se puede acceder a sitios web de interés para el usuario.



ILUSTRACIÓN V.36

Imágenes de la interfase sioticsweb

Controles: En esta sección se visualiza una imagen con la configuración de los controles, tanto para el recorrido como para el juego.



ILUSTRACIÓN V.37

Imágenes de la interfase controles

Créditos y Agradecimientos: Aquí se reproduce un video con los nombres de las personas involucradas en este proyecto así como agradecimientos personales.

En cada una de estas pantallas se encuentra indicada en la parte inferior derecha la forma en la cual podemos entrar o salir de cualquiera de las opciones.

Capítulo VI

Desarrollo

Después de las etapas de análisis y diseño, seguimos con el desarrollo del software, en el cual generaremos e integraremos todos aquellos elementos que se diseñaron en la etapa anterior, para formar un solo proyecto. El desarrollo para mayor facilidad se dividió en dos etapas, las cuales se realizaron en el siguiente orden: la primera corresponde al desarrollo de las estructuras tridimensionales y del personaje, y como segunda parte tendremos el desarrollo del recorrido y juego.

VI.1 Desarrollo de las Estructuras Tridimensionales y del Personaje

En esta etapa vamos a separar las tareas necesarias para la creación de un ambiente tridimensional:

1. Modelado de cada uno de los elementos involucrados en el espacio virtual.
2. Texturizado de cada uno de los elementos involucrados.
3. Iluminación del espacio navegable.
4. Aplicación de las diferentes vistas (cámaras) a utilizarse en el espacio.
5. Rendering

1. Modelado

Para la elaboración de un escenario es importante modelar cada uno de los elementos presentes dentro del espacio virtual, esto se logra mediante la incorporación de cuerpos simples, como el paralelepípedo, el más utilizado y la estructura básica de cualquier objeto, con la incursión de otros elementos geométricos como planos (caso particular de paralelepípedo con altura igual a cero), conos, pirámides, esferas, etc. También se puede hacer uso de formas compuestas mediante operaciones booleanas y extrusiones de los diferentes planos de alguna de las formas antes mencionadas.

La geometría es construida de manera poligonal, la unidad mínima de un polígono es el triángulo, el modelo se va a medir en triángulos, y dependiendo del número que lo formen mayor será el detalle del mismo, pero con el agravante de la eficiencia del proyecto, debido a que se van a dibujar todos los objetos en escena y puede que exista alentamiento en el recorrido virtual. Esto incluye al modelado del personaje, que se realiza con un mayor detalle, pues cuenta con un número importante de polígonos, mucho mayor a la de cualquier templo.

Lo que tenemos que cuidar en esta etapa es encontrar un balance entre la estética del objeto y el número de polígonos que lo conforman, para que se vean lo mejor posible, sin que afecte al rendimiento del recorrido virtual. Después de la etapa de construcción nos pasamos a una etapa en la que podemos ahorrarnos polígonos para obtener mayor realismo, es decir, el texturizado.

2. Texturizado

El texturizado se utiliza de forma análoga a un pintor con su obra, aquí de igual manera contaremos con una paleta de colores, los cuales se escogieron previamente y editaron en la etapa de diseño. Esto tanto para el texturizado de la zona arqueológica del Templo Mayor así como para el personaje.

Con las texturas dentro del programa modelador 3D, se aplican polígono por polígono, para después cambiar alguna de sus propiedades (si así lo requiere el material), como los niveles de coloración RGB (rojo, verde, azul) u otros parámetros que involucran transparencia, para crear distintos efectos, como opacidad, objetos cristalinos, incluso niebla.

3. Iluminación

En el escenario virtual es de gran importancia contar con elementos de iluminación, ya que con ellos podemos aclimatar o lograr el escenario deseado. Utilizando los tres tipos de luces (direccional, spot y puntual) ya mencionados en el apartado V.7.

4. Cámaras

La incorporación de cámaras es importante, ya que van a ser nuestros ojos dentro del recorrido virtual. En nuestro caso la cámara principal será una en tercera persona por la incorporación del personaje, con la opción de cambiar a una vista aérea en primera persona y una vista trasera para no perder ningún ángulo del caballero durante el recorrido.

5. Rendering

Es un término técnico Inglés de difícil traducción. Podemos tomarlo como el proceso que tiene lugar en la computadora que tiene almacenados los datos para “dibujar” un gráfico en la pantalla, esta es la imagen tridimensional que incorpora la simulación de efectos de iluminación, tales como las sombras y también de las texturas que componen el objeto. Y que en conjunto las imágenes generadas producen el movimiento.¹

Ahora para que el movimiento se realice de una manera correcta y eficiente, se necesita realizar el cálculo de cada uno de los cuadros por segundo que van a componer la secuencia.

¹ http://es.wikipedia.org/wiki/Volume_rendering, 27 de Agosto 2007.

VI.1.1 Creación de los Objetos

Para ejemplificar la labor de la construcción de objetos y la texturización, se puso de ejemplo la creación tridimensional del Templo Mayor y del Personaje.

Para la creación de los objetos tridimensionales como se mencionó anteriormente utilizamos el software 3D Studio Max 5. El cual cuenta con la siguiente interfaz.

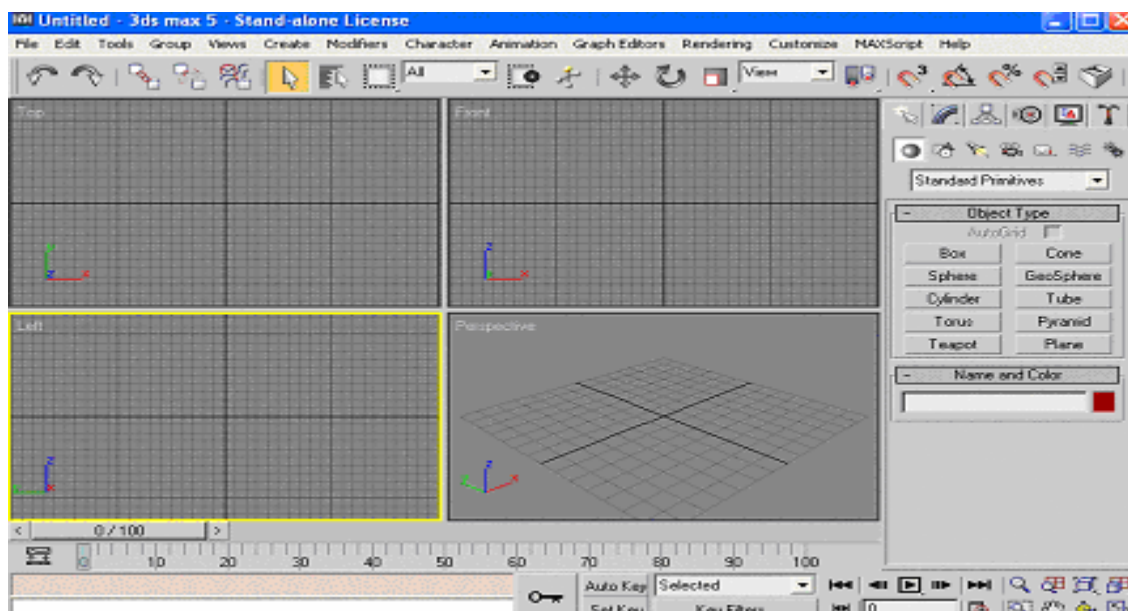


ILUSTRACIÓN VI.1
Interfaz del usuario 3D Max

Como muestra la imagen anterior el programa cuenta con una interfaz agradable y de fácil uso, la cual esta provista de herramientas para las distintas etapas para la construcción tridimensional. La primera de estas herramientas se encuentra a la derecha de la pantalla.

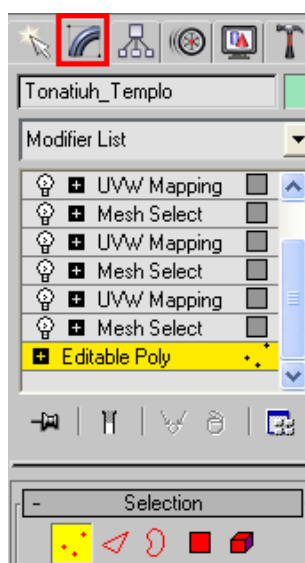


ILUSTRACIÓN VI.2
Panel de control

Para la elaboración de los modelos geométricos utilizaremos el panel de modificación de 3D MAX, el cual nos permite manipular nuestro modelo geométrico utilizando los diferentes elementos que lo conforman:

El vértice, el cual es el elemento más sencillo que conforma el modelo, son puntos que forman los polígonos.

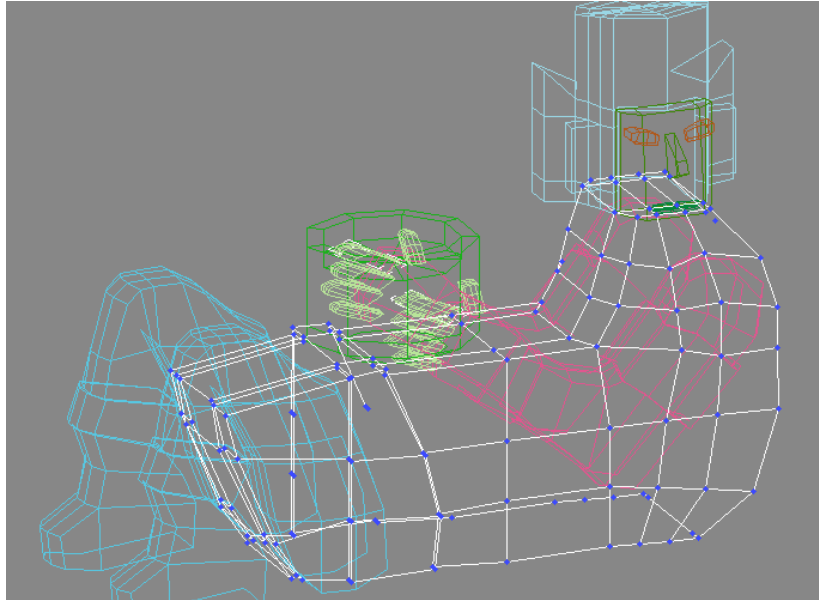


ILUSTRACIÓN VI.3
Selección por puntos

Los ejes, se forman de la unión entre vértices.

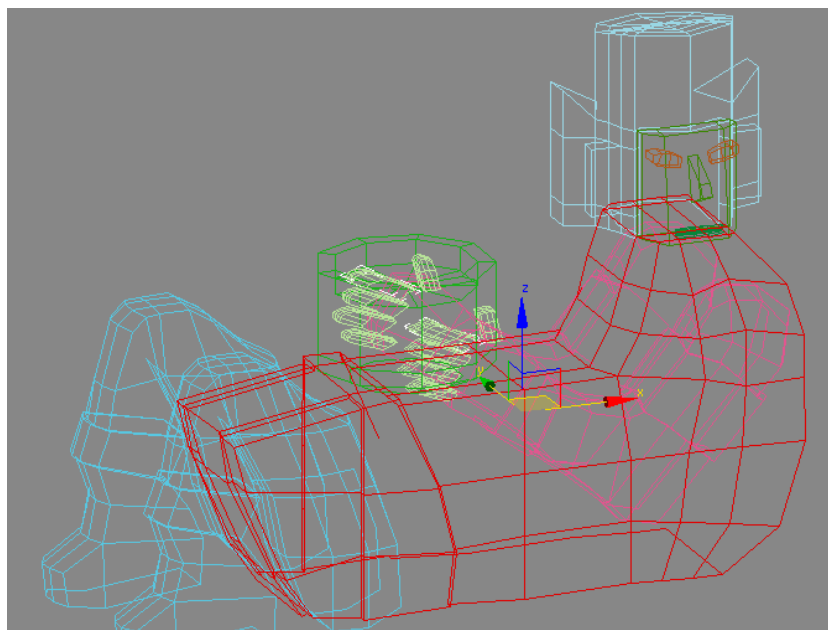


ILUSTRACIÓN VI.4
Ejes

Un caso especial de los ejes es el borde, opción que nos permite seleccionar una serie de ejes, los cuales forman el contorno del objeto.

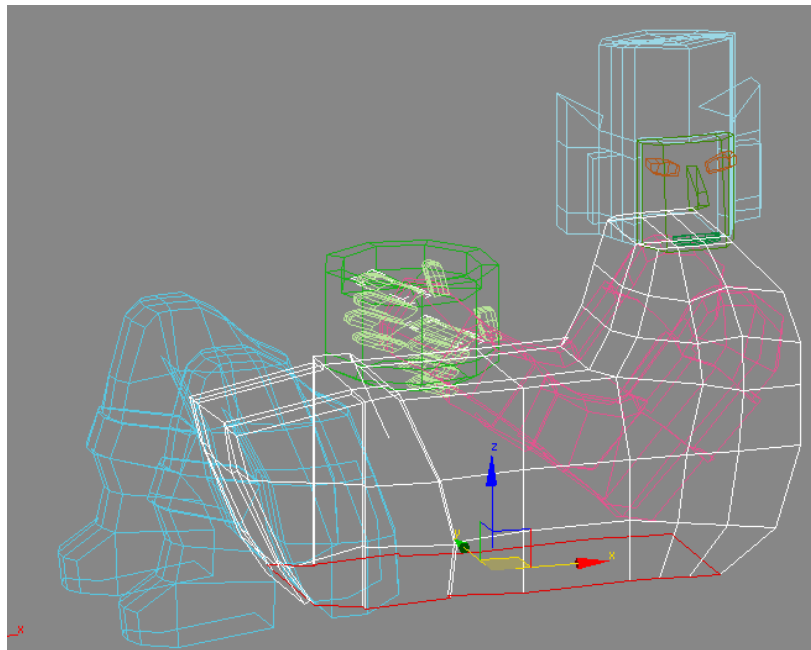


ILUSTRACIÓN VI.5
Selección por borde

También se puede modificar el modelo mediante las caras que forman la unión de los ejes.

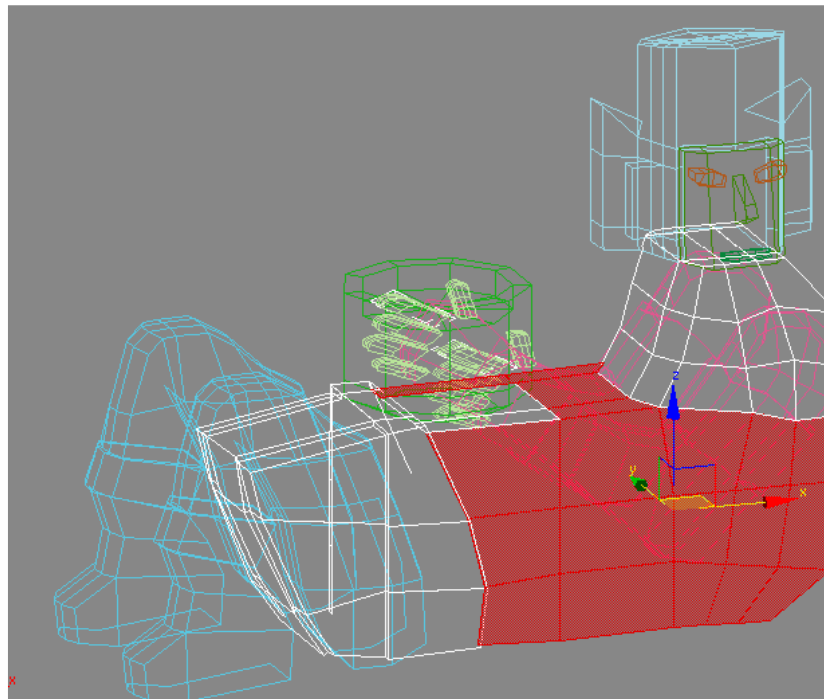


ILUSTRACIÓN VI.6
Selección por cara

Por último las caras se unen para formar un elemento poligonal, el cual también puede ser modificado.

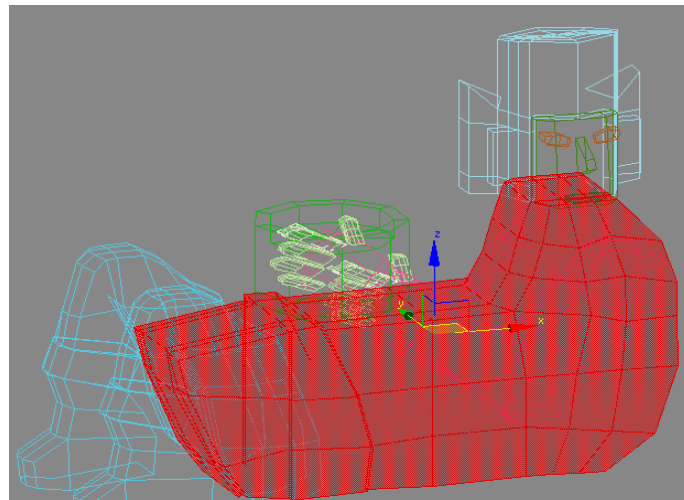


ILUSTRACIÓN VI.7
Selección poligonal

Como primer paso utilizamos la función Box para crear un objeto en el espacio de trabajo.

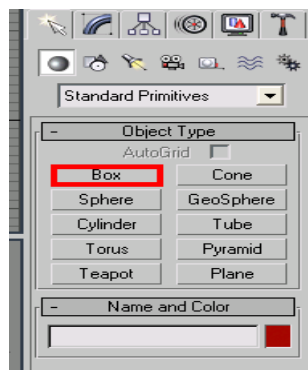


ILUSTRACIÓN VI.2
Botón Box

Este objeto será la base para crear la pirámide; para ello seleccionamos el objeto caja creado y utilizamos el comando Convert to Editable Poly, con el fin de poder manipular el objeto.

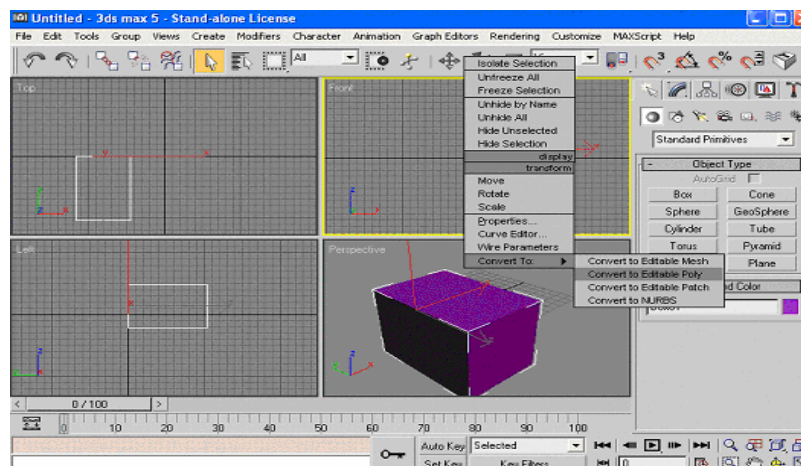


ILUSTRACIÓN VI.2
Opción Convert to Editable Poly

Con esto podemos realizar diferentes transformaciones de acuerdo a sus características, como lo son: vértices, aristas, caras y polígonos.

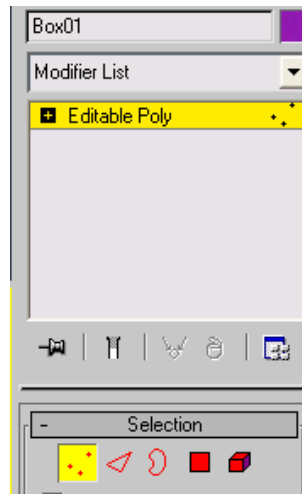


ILUSTRACIÓN VI.3
Transformaciones

Utilizando la vista frontal y seleccionando la caja por vértices, desplazamos hacia el centro de la figura los vértices superiores para formar un trapecio. Lo mismo hacemos en la vista izquierda.

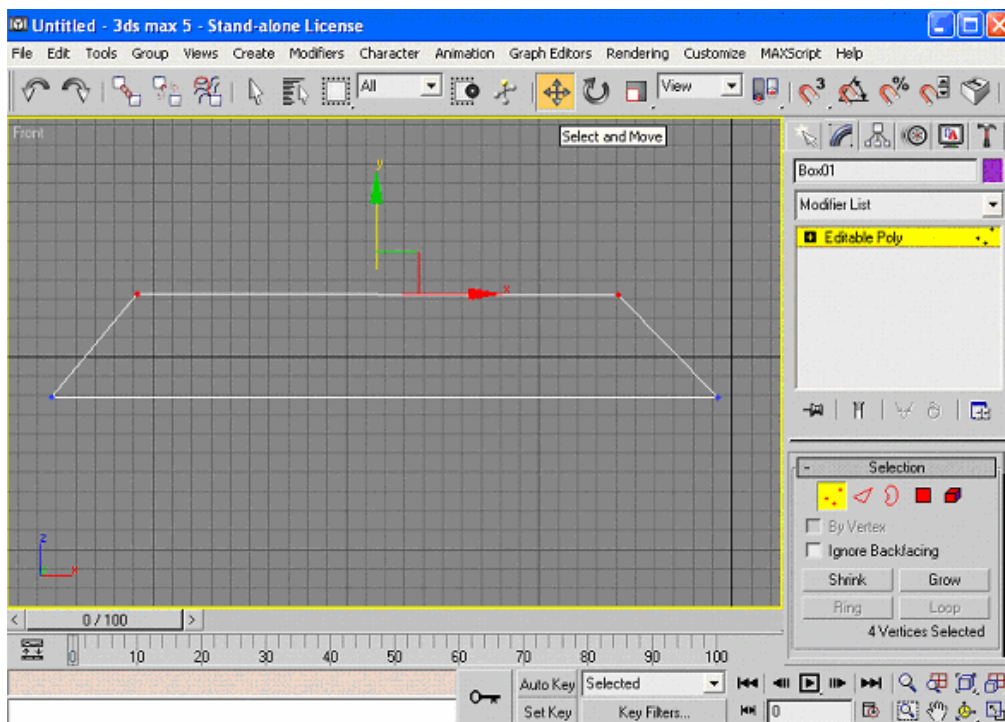


ILUSTRACIÓN VI.4
Trapezio vista Frontal

Cambiando la vista podemos darnos cuenta del resultado de las operaciones anteriores, con lo cual se tiene la base.

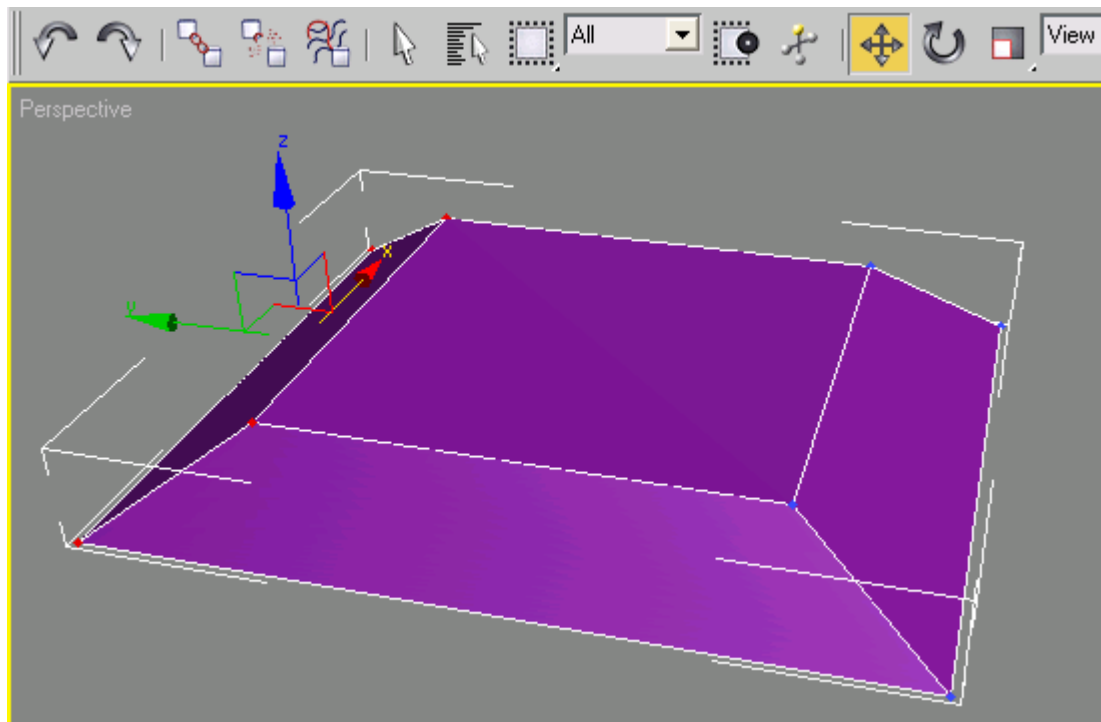


ILUSTRACIÓN VI.5
Trapezio en Perspectiva

Para construir las capas superiores hacemos una copia de la base de la pirámide utilizando la función clonar del menú Edit.

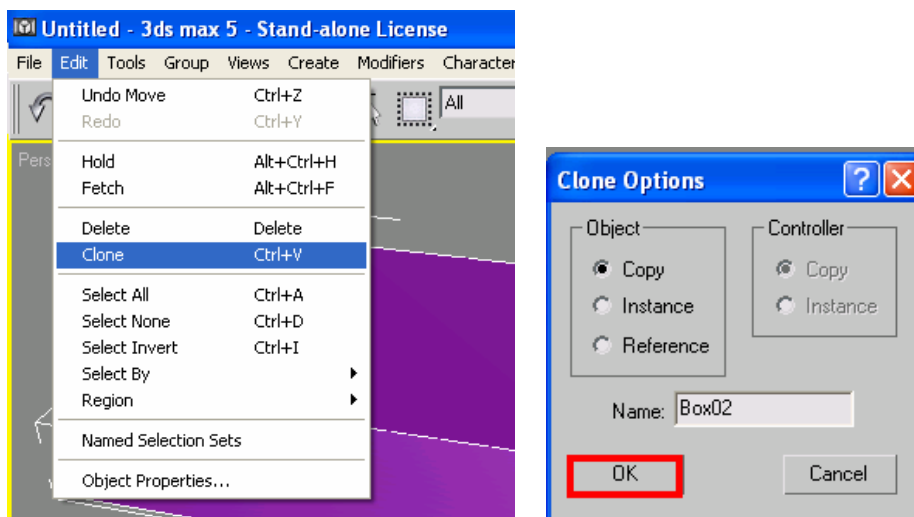


ILUSTRACIÓN VI.6
Función Clone

Este objeto es idéntico al anterior, posteriormente utilizamos la función de escalar uniforme, para modificar sus dimensiones en todas sus coordenadas para que sea de menor tamaño que el objeto original.

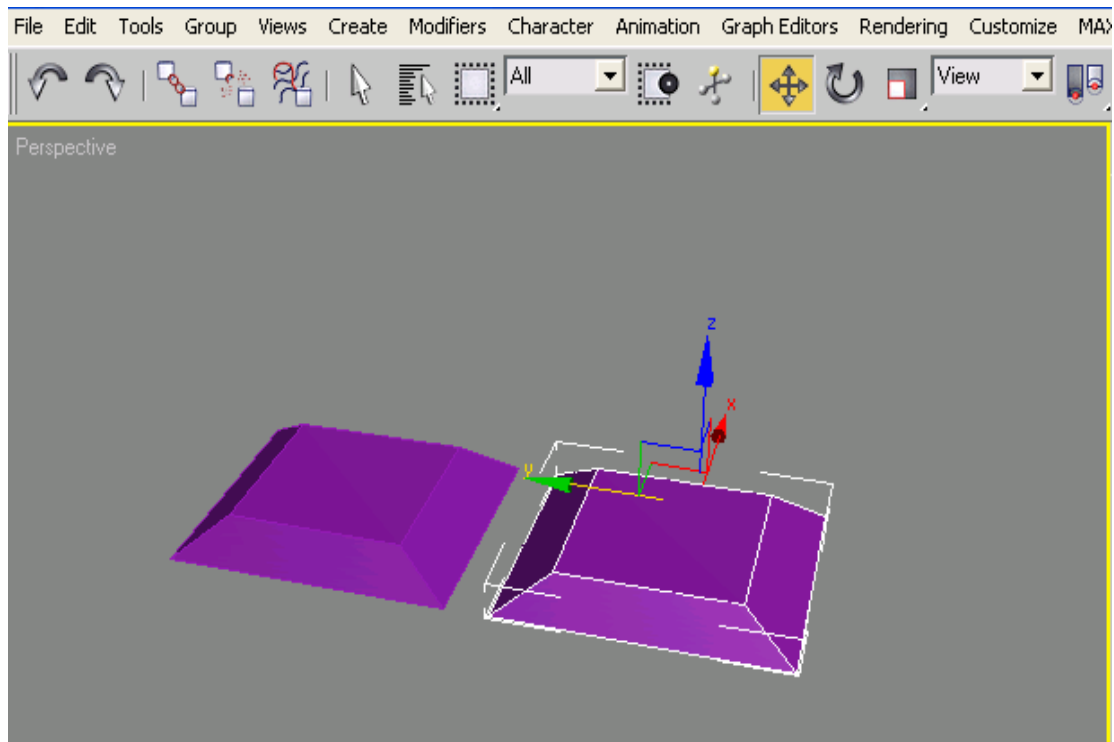


ILUSTRACIÓN VI.7
Resultado de función Clone

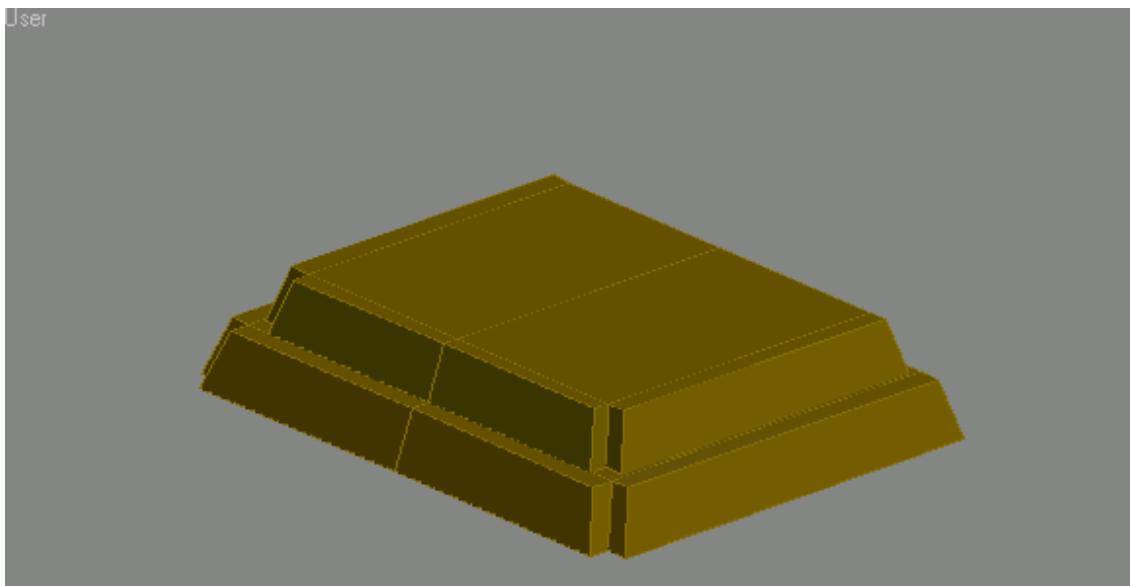


ILUSTRACIÓN VI.9
Trapezios en Conjunto

Este proceso se repite hasta obtener todos los niveles que la pirámide necesita.

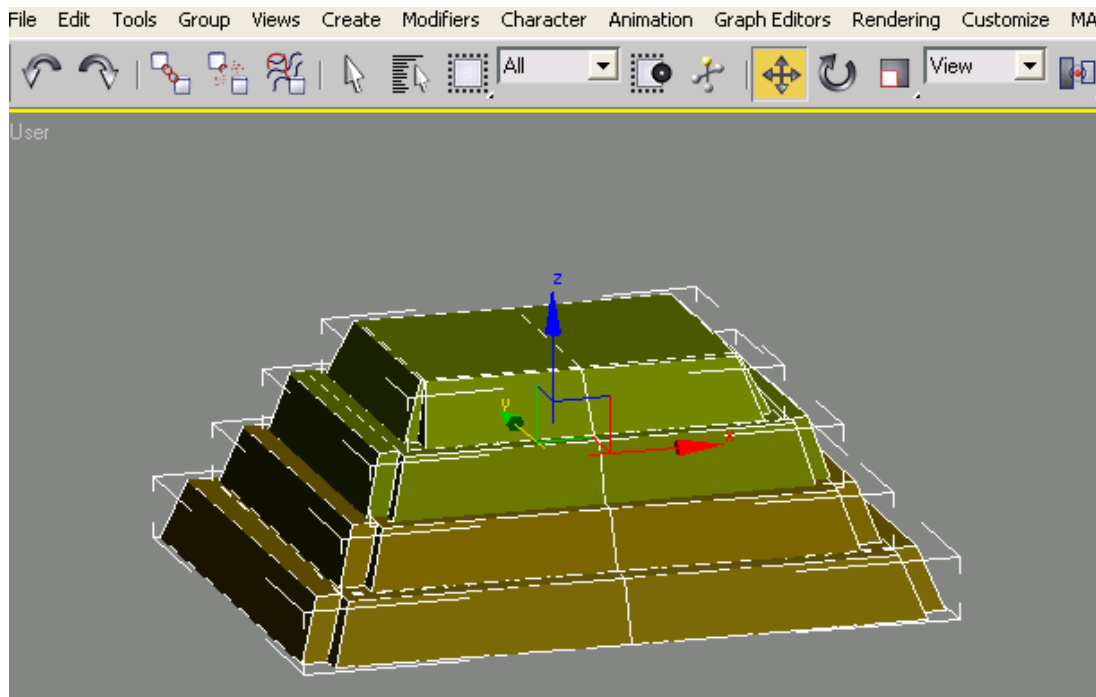


ILUSTRACIÓN VI.10
Pirámide de Cuatro Niveles

Para la construcción de los barandales colocados a lo largo de las escaleras generamos una caja como anteriormente se hizo.

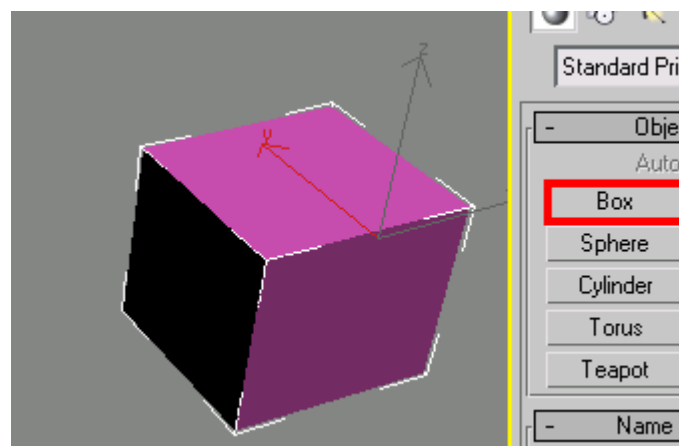


ILUSTRACIÓN VI.11
Función Box

Y se convierte en un polígono editable nuevamente. En la vista izquierda utilizamos la función Conect para agregar nuevas aristas a la caja.

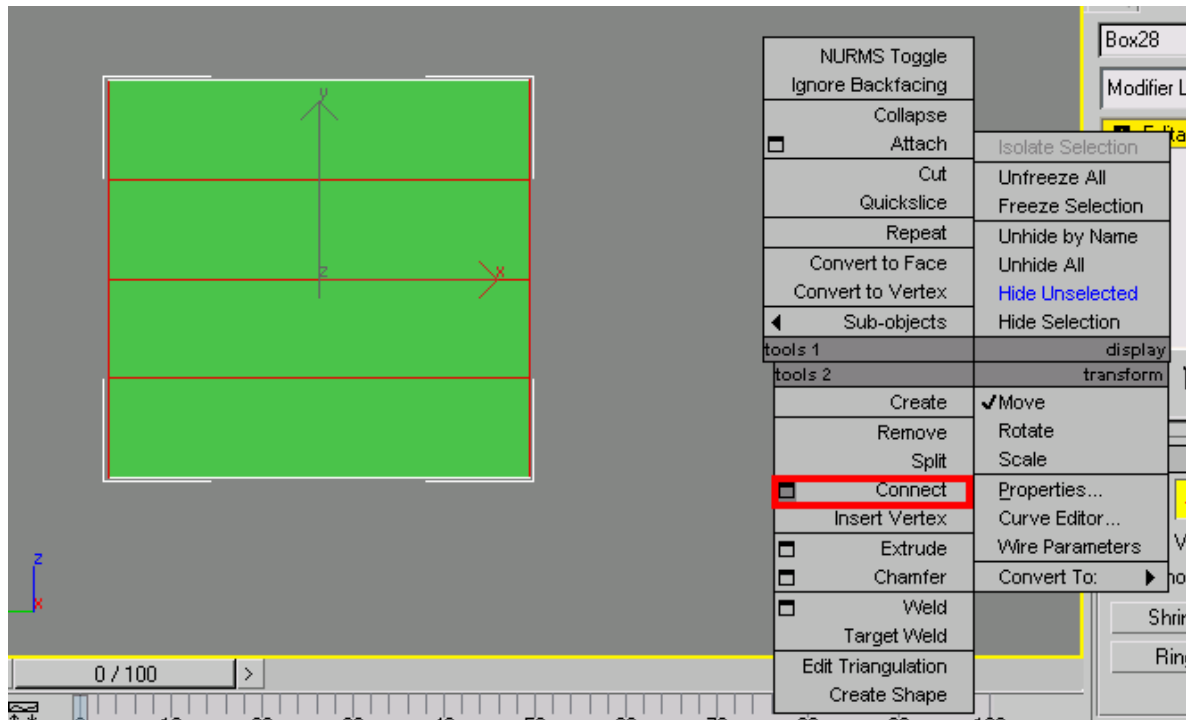


ILUSTRACIÓN VI.12
Función Connect

Para que al modificar sus vértices tenga la forma de pasamanos requerida.

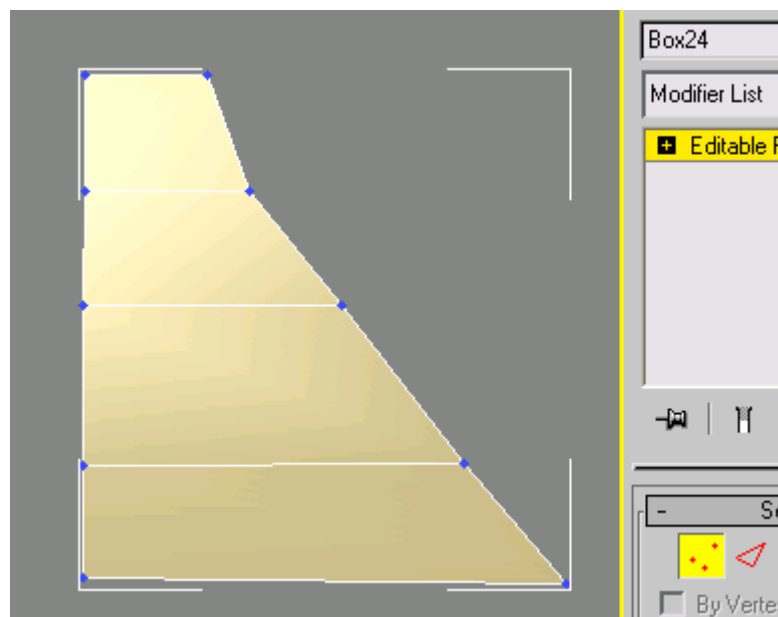


ILUSTRACIÓN VI.13
Pasamanos

Este objeto se clona para obtener los barandales.

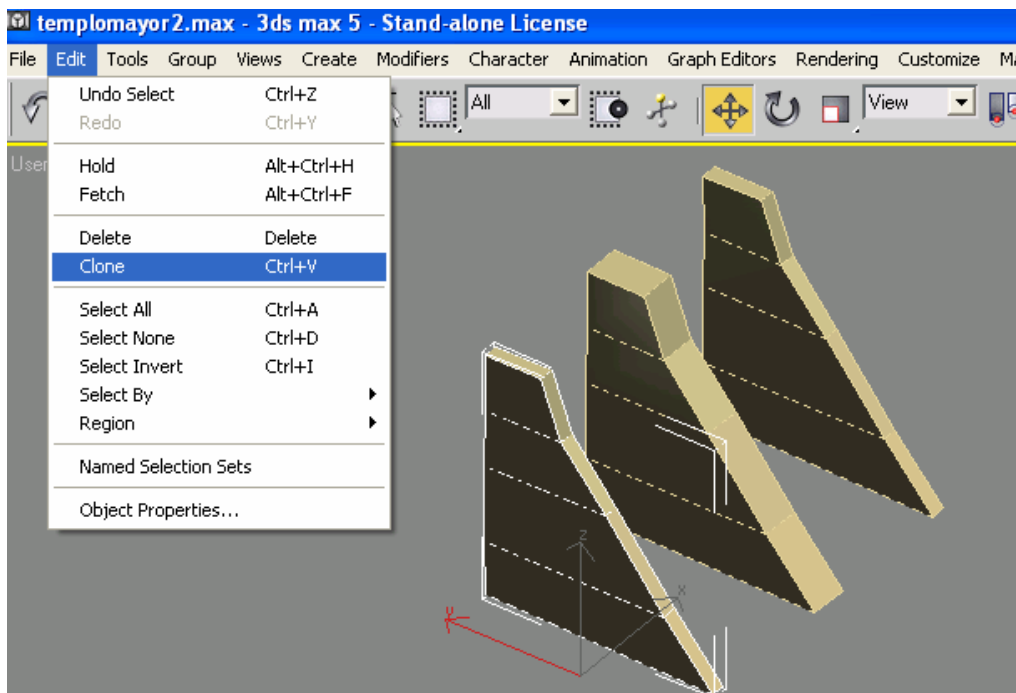


ILUSTRACIÓN VI.14
Función Clone

Para generar las escaleras utilizamos la herramienta Snap Toggle para que el trazo de las líneas que forman la escalera sean dibujadas sobre la cuadrícula de referencia.

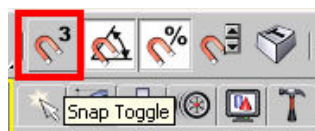


ILUSTRACIÓN VI.15
Botón Snap Toggle

Utilizamos la función crear línea.

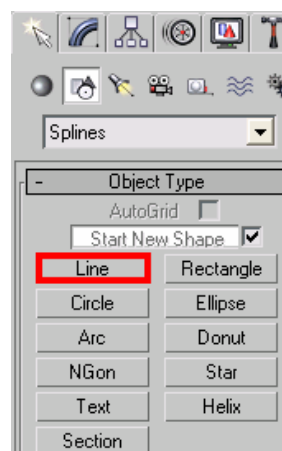


ILUSTRACIÓN VI.16
Función Crear Línea

La línea es creada con el mouse, al dar un click se inicia el segmento y al dar un segundo click el segmento es concluido, como se tiene encendido Snap Toggle la línea seguirá la cuadrícula.

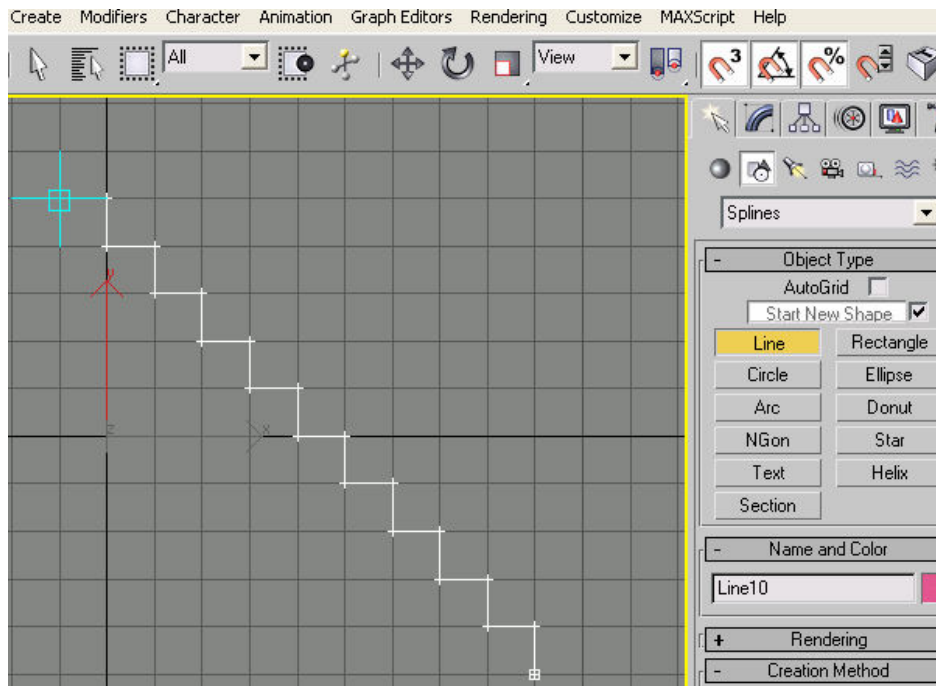


ILUSTRACIÓN VI.17

Construcción de Escalera

Una vez terminadas las escaleras se utiliza la función Extrude para darles volumen.

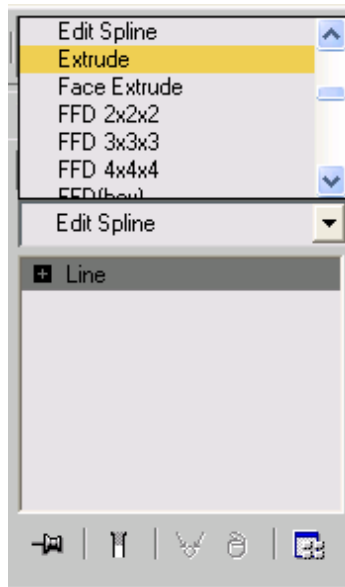


ILUSTRACIÓN VI.18

Función Extrude

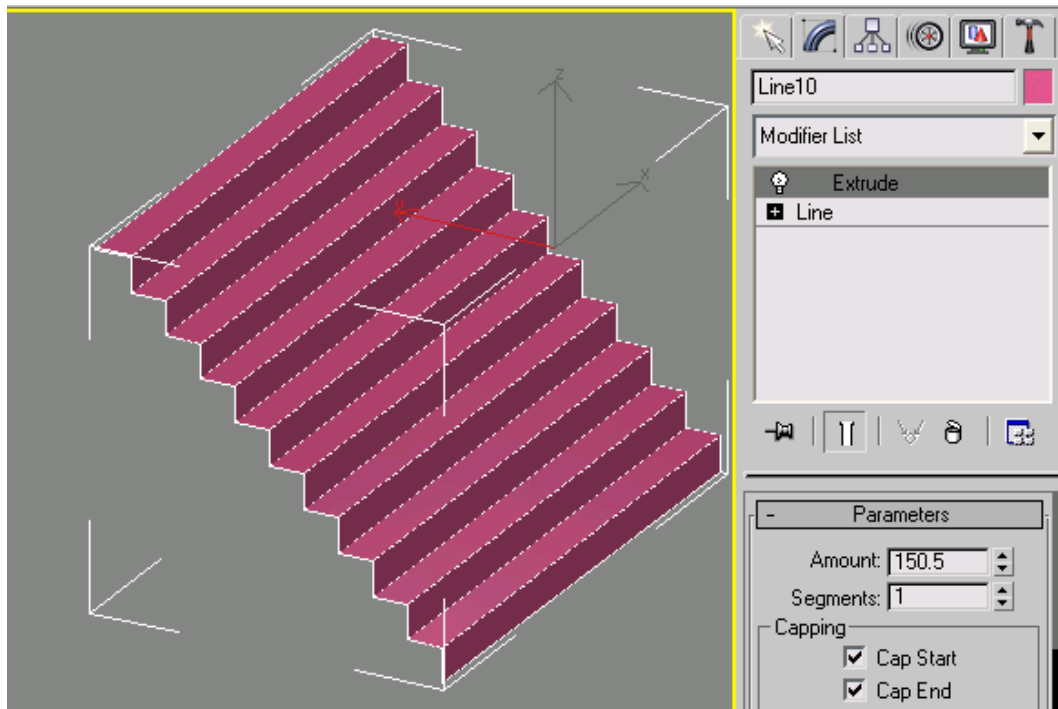


ILUSTRACIÓN VI.19
Resultado de Función Extrude

Las escaleras y los barandales se trasladan sobre el espacio de trabajo hasta colocarlos frente a la pirámide.

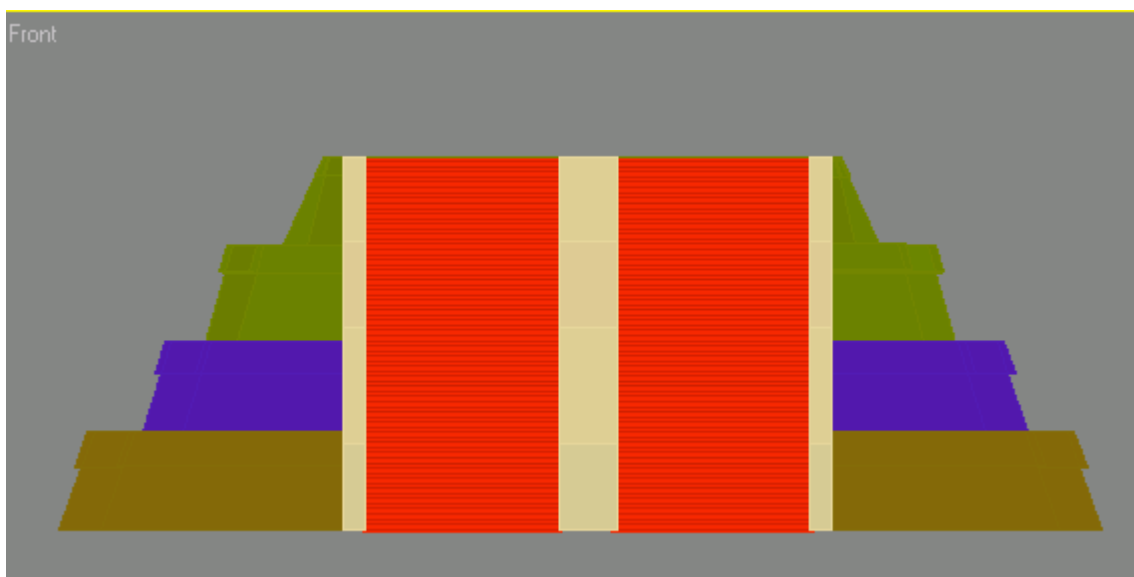


ILUSTRACIÓN VI.20
Pirámide con Escaleras
Vista Frontal

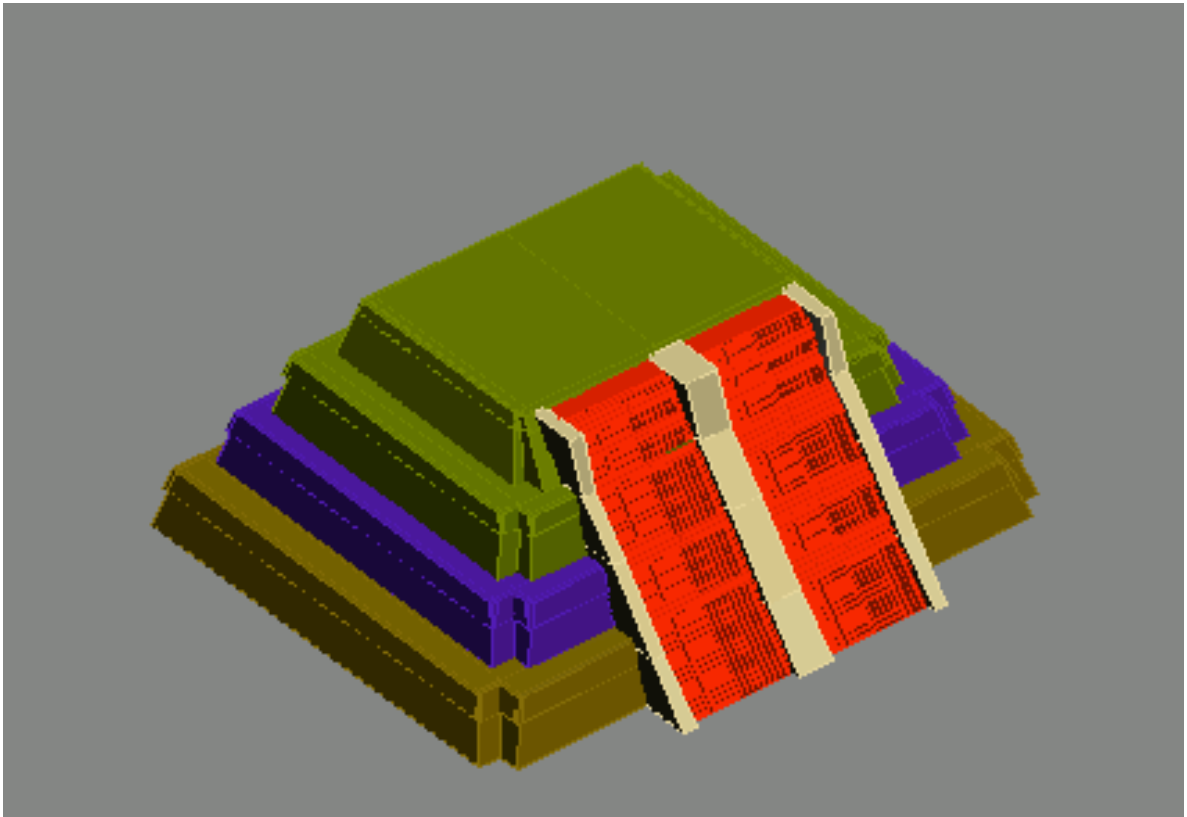


ILUSTRACIÓN VI.21
Pirámide con Escaleras
en Perspectiva

Para la construcción del templo sobre la pirámide utilizamos un objeto caja, el cual tiene el espesor de la pared y fue segmentado para formar las paredes del templo.

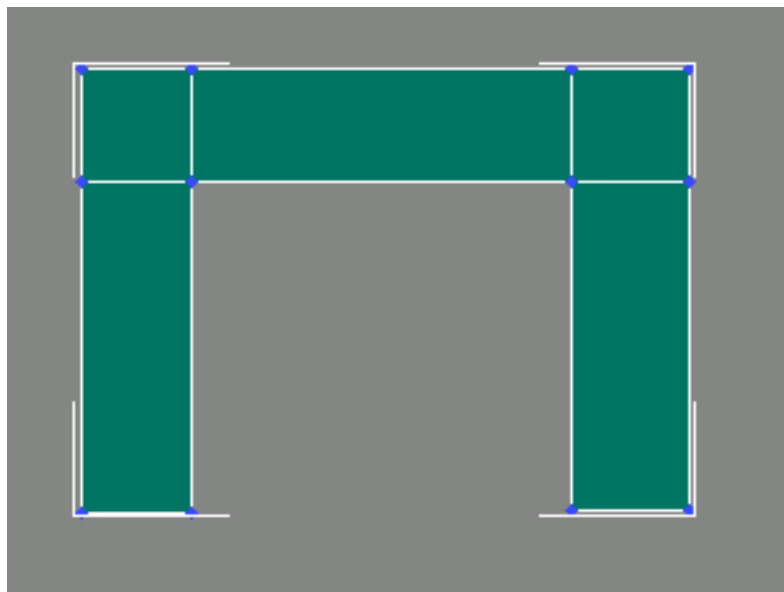


ILUSTRACIÓN VI.22
Construcción de los Templos

Posteriormente aplicamos la función Extrude para que las paredes tengan la altura requerida.

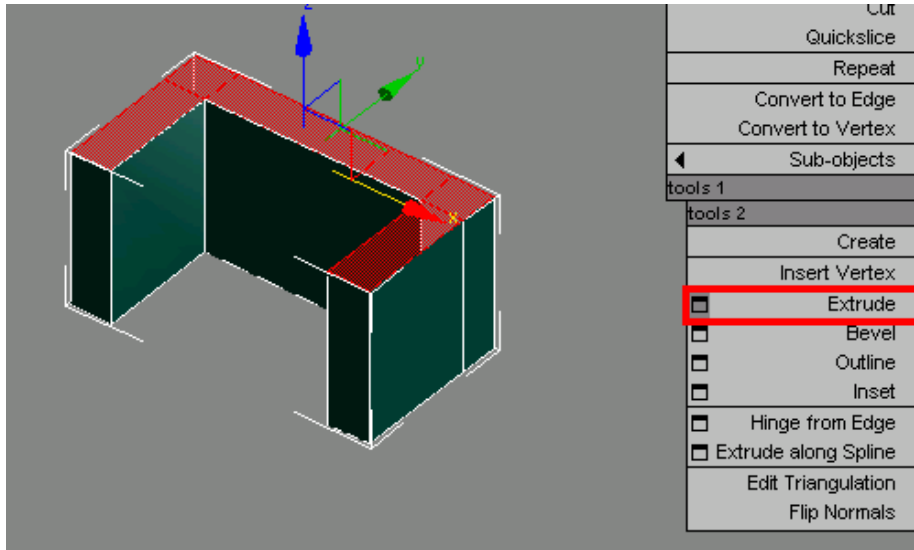


ILUSTRACIÓN VI.23
Función Extrude

El techo esta formado por diferentes cajas modificadas como Editable Poly.

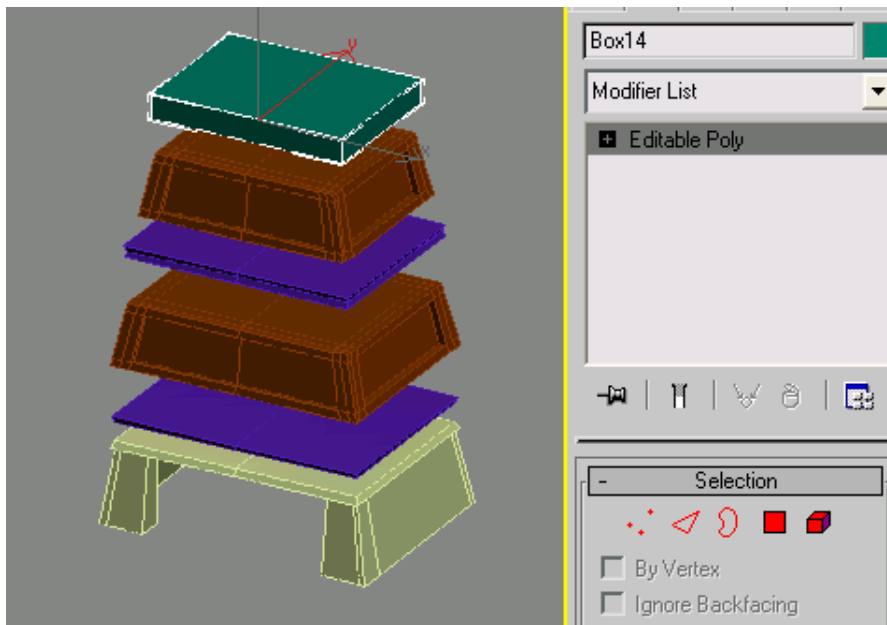


ILUSTRACIÓN VI.24
Niveles

De esta forma la pirámide queda completa.

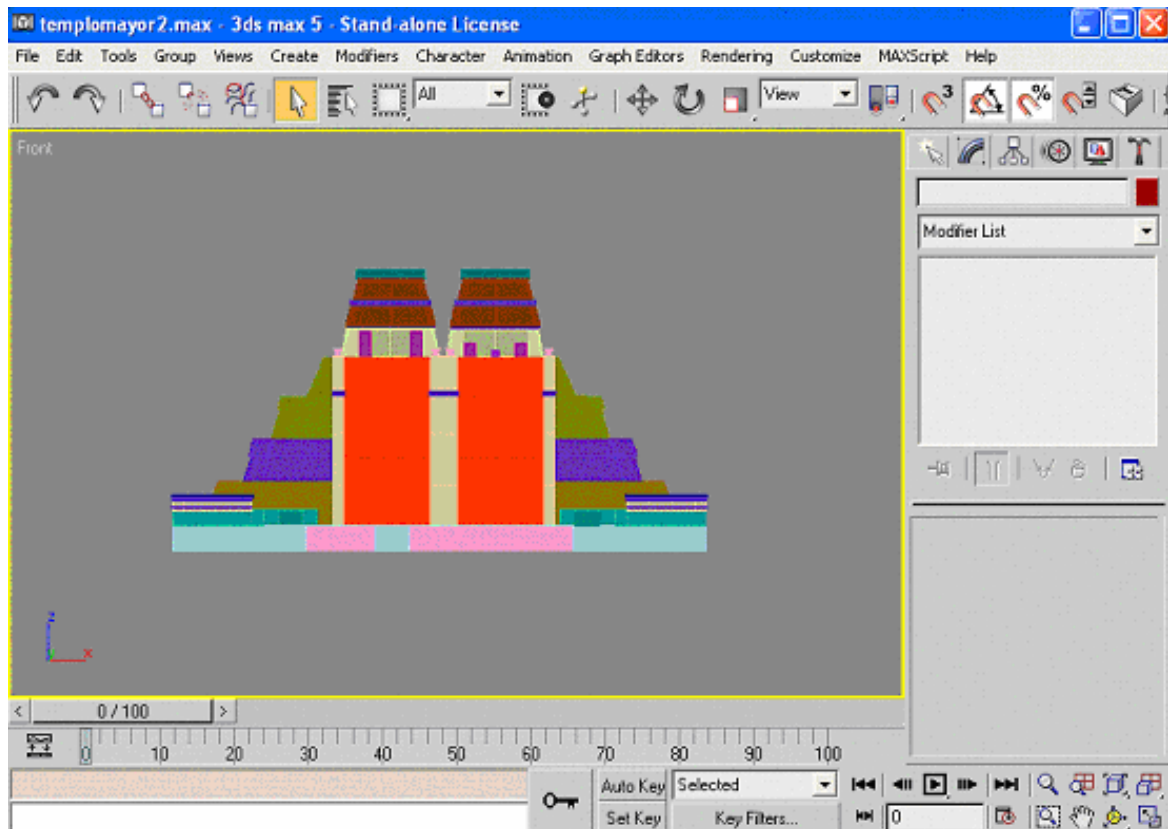


ILUSTRACIÓN VI.25
Pirámide Completa

Los templos laterales fueron hechos de forma similar, al igual que los demás templos de la ciudad.

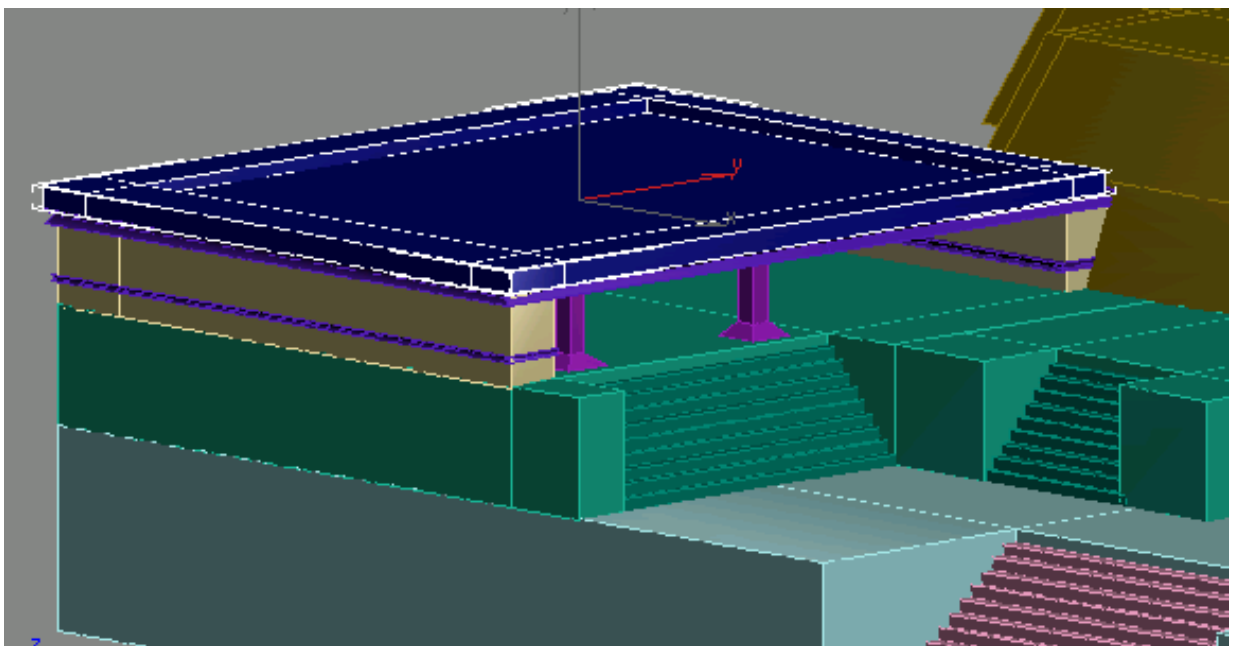


ILUSTRACIÓN VI.26
Templo Lateral

Para utilizar este modelo en el ambiente de Virtools, es necesario disminuir al máximo el número de polígonos que conforman el modelo, para que pueda ser procesado en tiempo real; por ello fueron borradas todas las caras internas de la pirámide ya que son innecesarias al momento de la ejecución del proyecto.

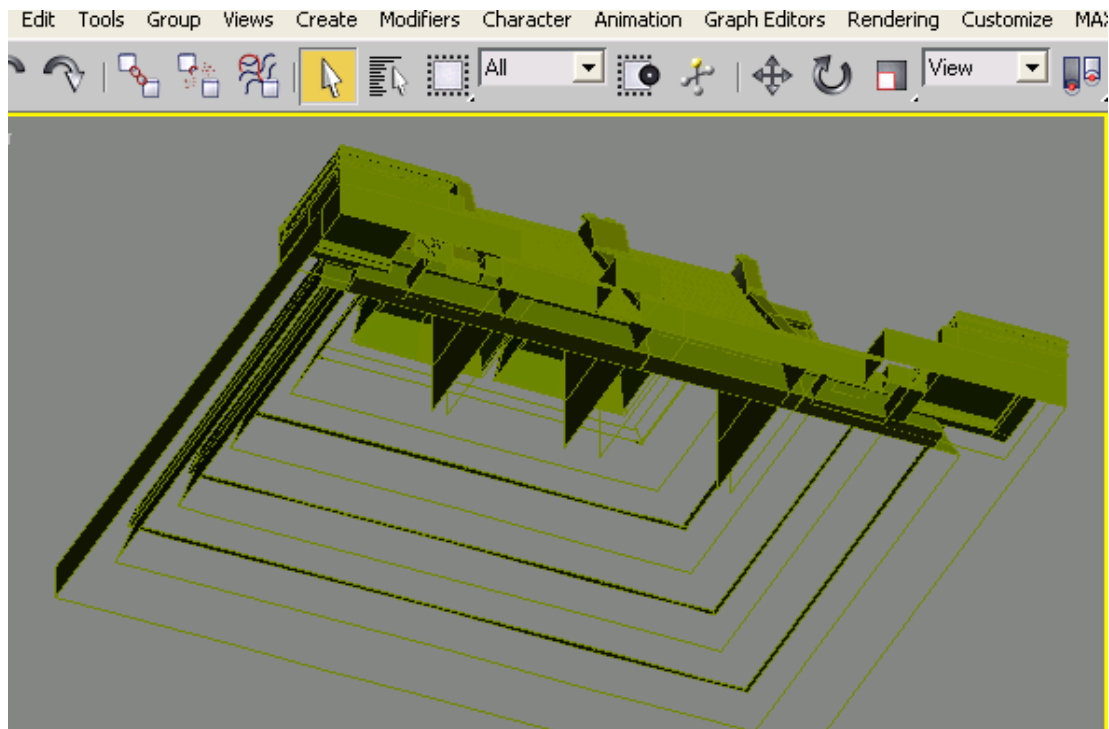


ILUSTRACIÓN VI.27
Templo Vista Inferior

Una vez hecho esto agrupamos los diferentes objetos que conforman la pirámide, para que sean reconocidos como uno solo, para ello seleccionamos uno de ellos y utilizamos la función Attach, la cual despliega un cuadro de dialogo donde seleccionamos los objetos a agrupar; una vez hecho esto, se selecciona la opción Attach la cual formará un solo objeto. Pero lo conveniente es separar al objeto en diferentes cortes lo cual ayudará tanto al procesamiento como al texturizado.

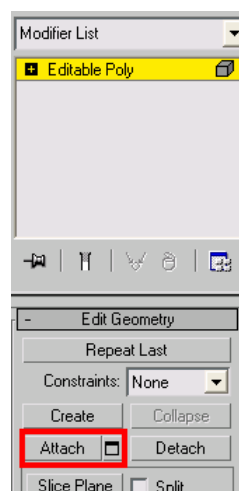


ILUSTRACIÓN VI.28
Función Attach

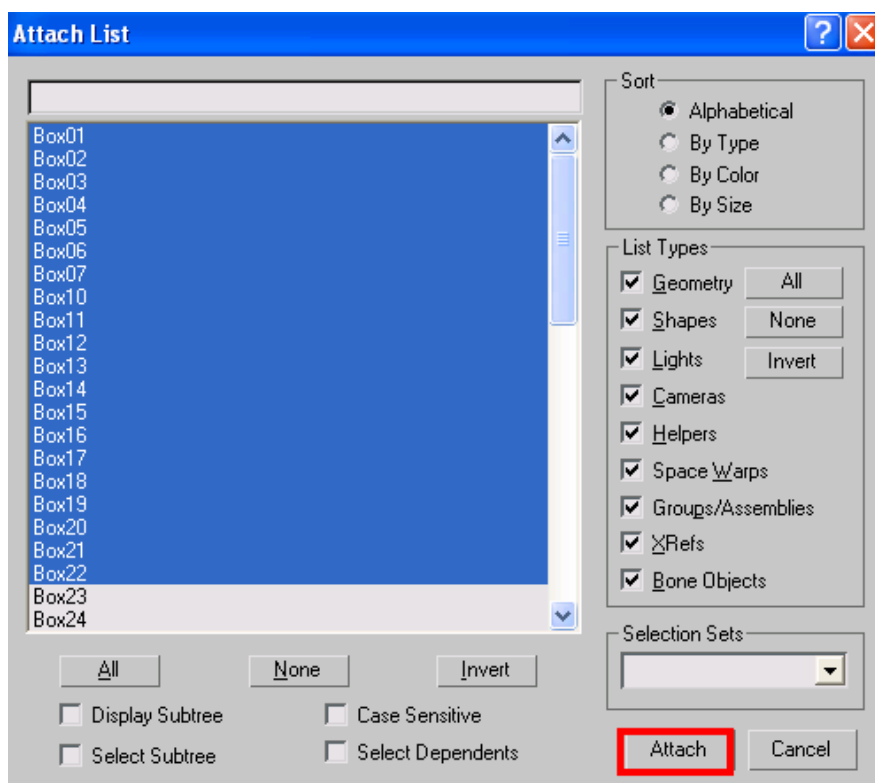


ILUSTRACIÓN VI.29

Lista de Elementos para la Función Attach

VI.1.1.2 Desarrollo del Personaje

Para el diseño del personaje utilizamos el método de modelado poligonal, siguiendo un proceso similar al utilizado para el modelado de las pirámides. El primer paso es colocar en diferentes planos una imagen de referencia (llamadas blue prints), estas imágenes corresponden a diferentes puntos de vista como lo son la vista frontal y lateral del personaje y para mayor precisión también se incluyó una vista superior.

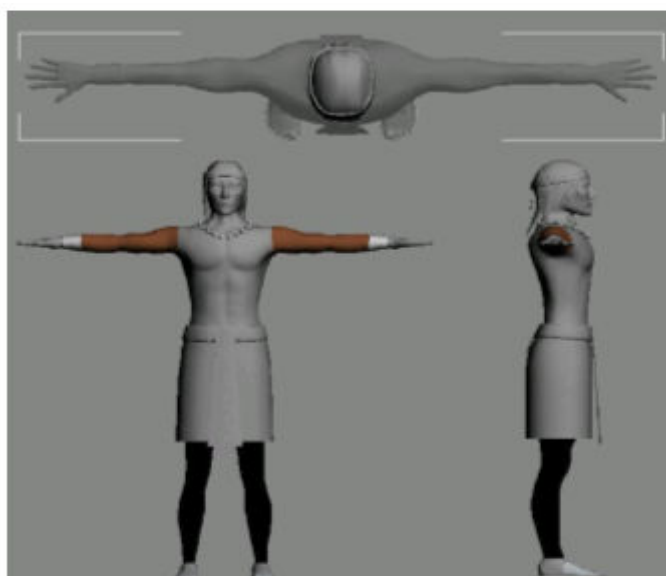


ILUSTRACIÓN VI.30

Blue Prints con las Vistas Superior, Lateral y Frontal

Las imágenes del personaje mostrado corresponden a uno modelado por el Ingeniero Francisco Salgado director de nuestra tesis, a partir de este personaje construiremos uno similar que pueda ser utilizado en el ambiente de Virtools (es decir con una menor cantidad de polígonos).

Las imágenes son enviadas a la ventana de trabajo donde son colocadas en su vista correspondiente.

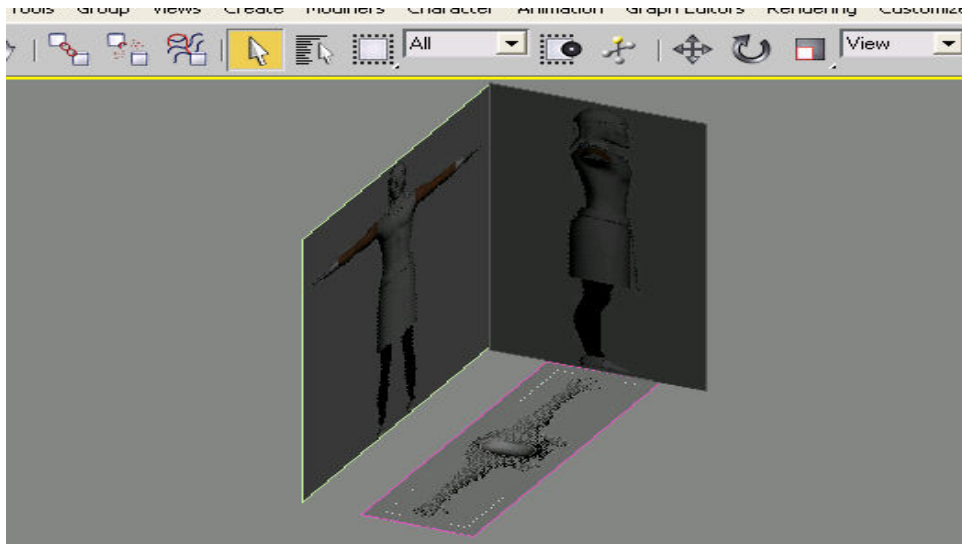


ILUSTRACIÓN VI.31
Colocación de Blue Prints

Iniciamos creando una caja para posteriormente convertirla en editable poly, y poder manipularla a través de operaciones de extrusión, subdivisión de caras y de desplazamiento de vértices, para que la caja empate con el contorno de la imagen en los diferentes planos para obtener el modelo tridimensional.

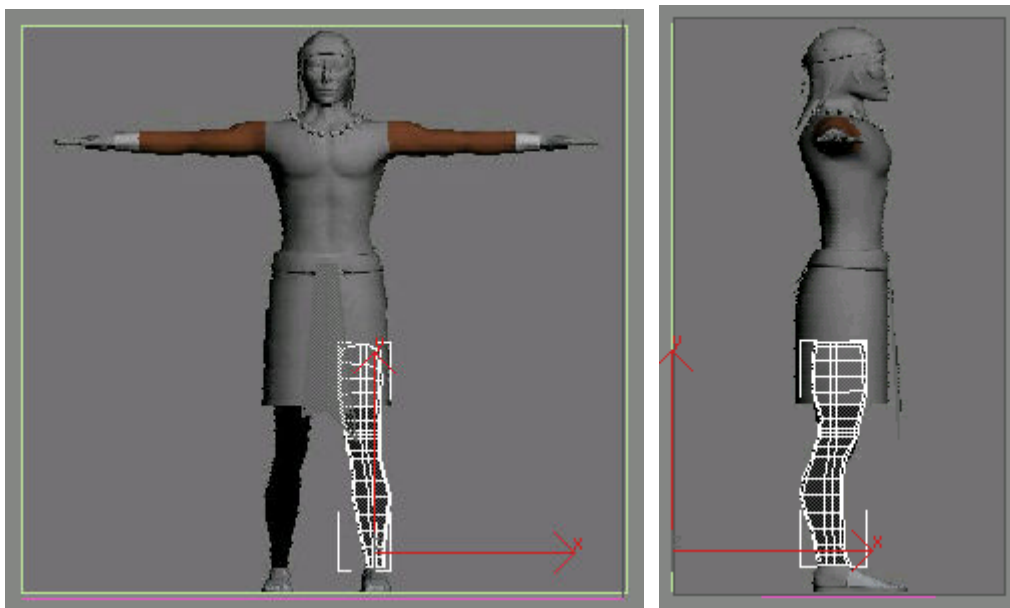


ILUSTRACIÓN VI.32
Construcción de Pierna

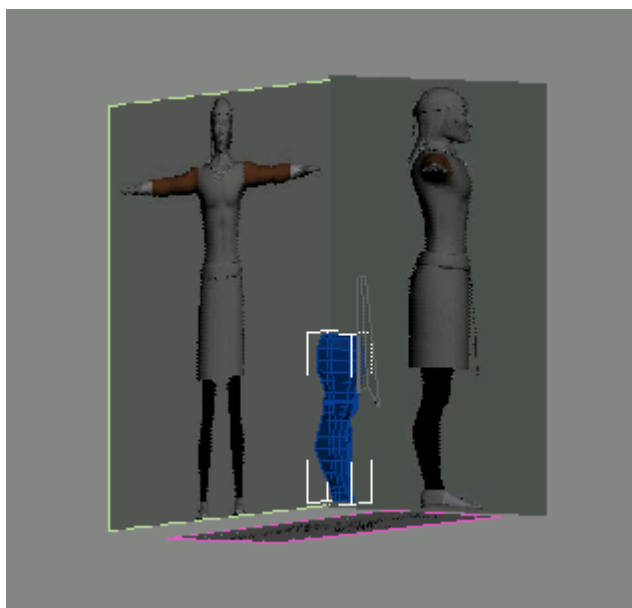


ILUSTRACIÓN VI.33
Pierna Terminada

El proceso continúa hasta modelar el cuerpo completo, en este caso decidimos modelar las diferentes partes del cuerpo por separado pero el modelo podría haberse hecho de una sola pieza, es una decisión que queda a consideración del modelador.

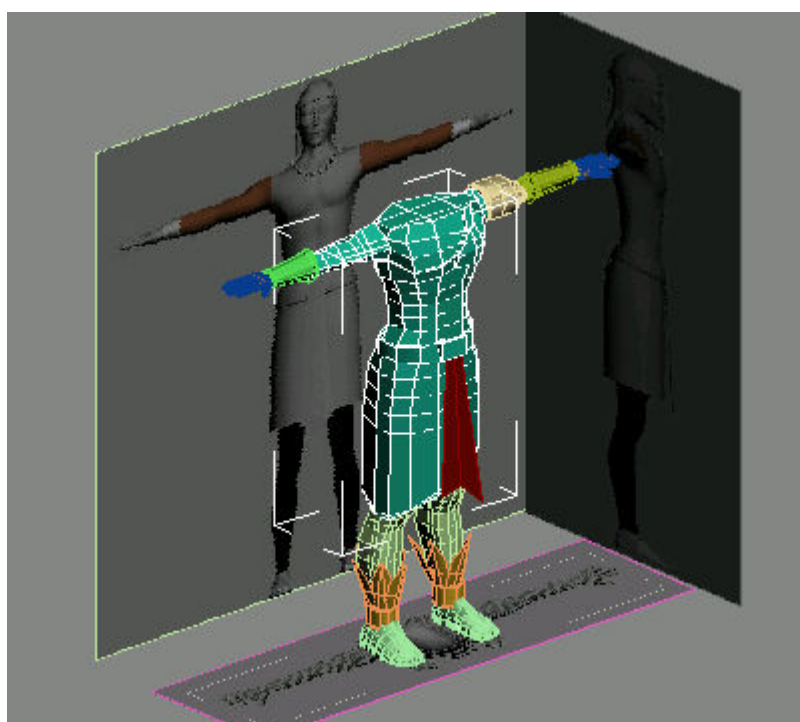


ILUSTRACIÓN VI.34
Piezas en Conjunto

Para la elaboración de los brazaletes y los adornos de las piernas utilizamos la primitiva Cylinder, la cual crea un cilindro en lugar de utilizar una caja como se había hecho anteriormente, esto es porque un cilindro se ajusta de mejor manera a la parte del cuerpo que se quiere modelar, como lo son los brazos.

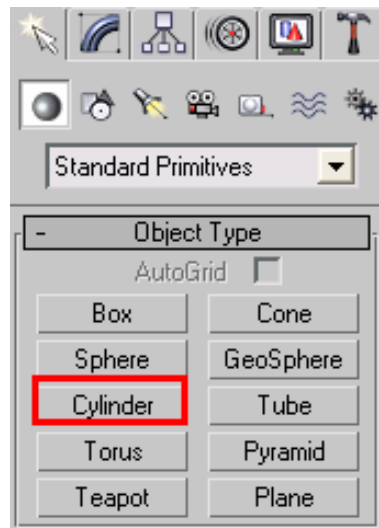


ILUSTRACIÓN VI.35
Primitiva Cylinder

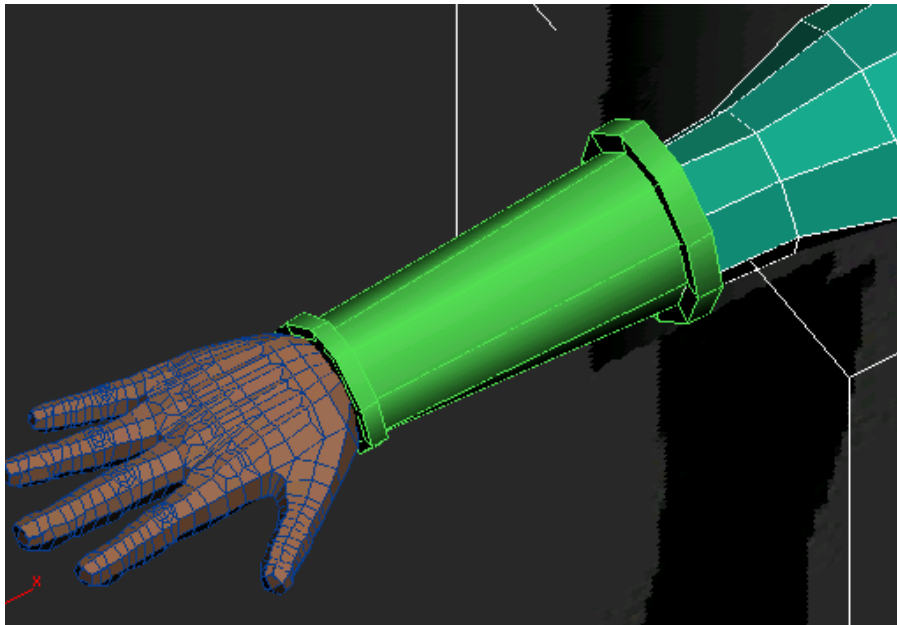


ILUSTRACIÓN VI.36
Brazalete Caballero Águila

VI.1.2 Aplicación de las Texturas

En esta etapa contamos con los objetos terminados en cuanto a su construcción y gracias a que contamos con que todos los polígonos son editables, a cada objeto se le aplicara más de una textura en la mayoría de los casos.

Vamos a tomar como ejemplo el Juego de Pelota de Tenochtitlan, lo primero es enumerar cada una de las texturas que vamos a utilizar para este edificio.

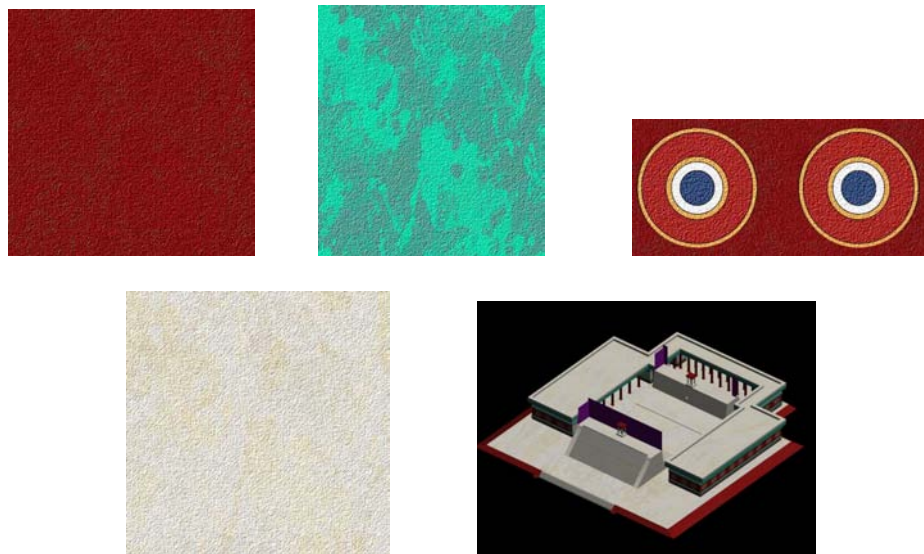


ILUSTRACIÓN VI.37
Algunas Texturas utilizadas
para el Juego de Pelota

Lo siguiente es crear una paleta de materiales en donde vamos a depositar cada una de las texturas, Por lo que tenemos que generar un material Multi/Subobjet, lo que quiere decir que dicho material albergará cada uno de los materiales en una lista, y con un índice único. Para almacenar una textura lo haremos desde el canal difuso dentro de las propiedades de cada uno de los materiales dentro de la lista. Se utiliza el canal difuso porque es en este en donde se refleja la luz en el material.

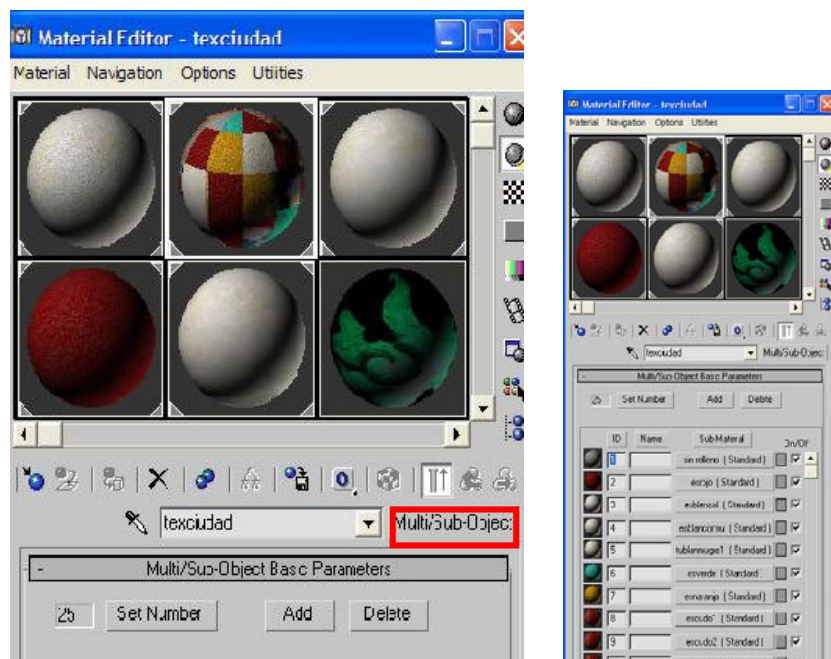


ILUSTRACIÓN VI.38
Función Multi/Subobjet

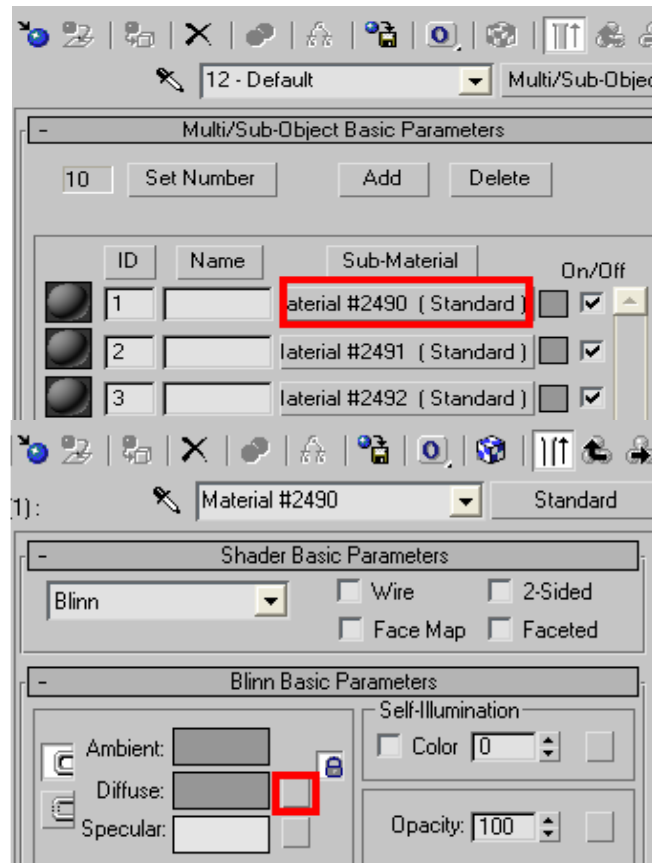


ILUSTRACIÓN VI.39
Función Multi/Subobject

Ahora respecto a la forma en que son guardadas estas imágenes son en bitmaps, o archivos de imágenes que son definidas por su número de píxeles, así como sus dos dimensiones ancho y altura.

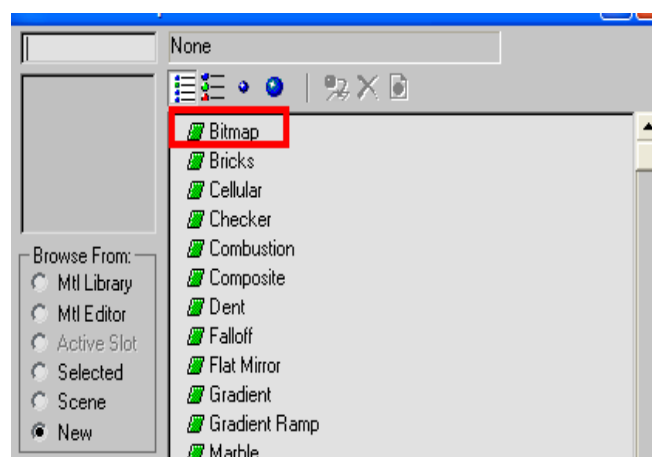


ILUSTRACIÓN VI.40
Asignación de un bitmap

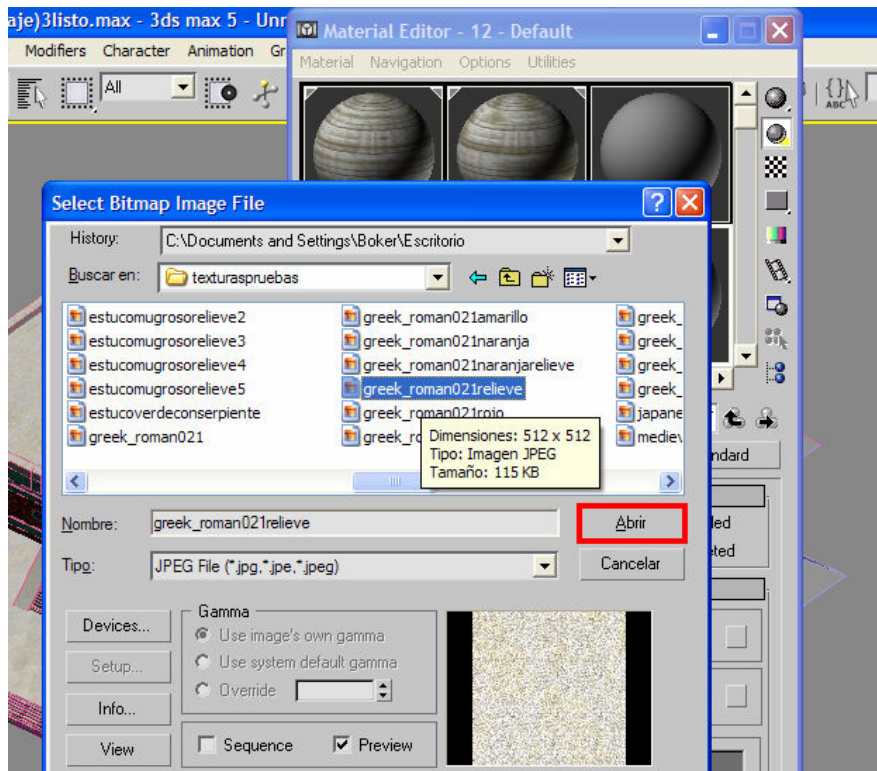


ILUSTRACIÓN VI.41
Asignación de la imagen en formato JPEG

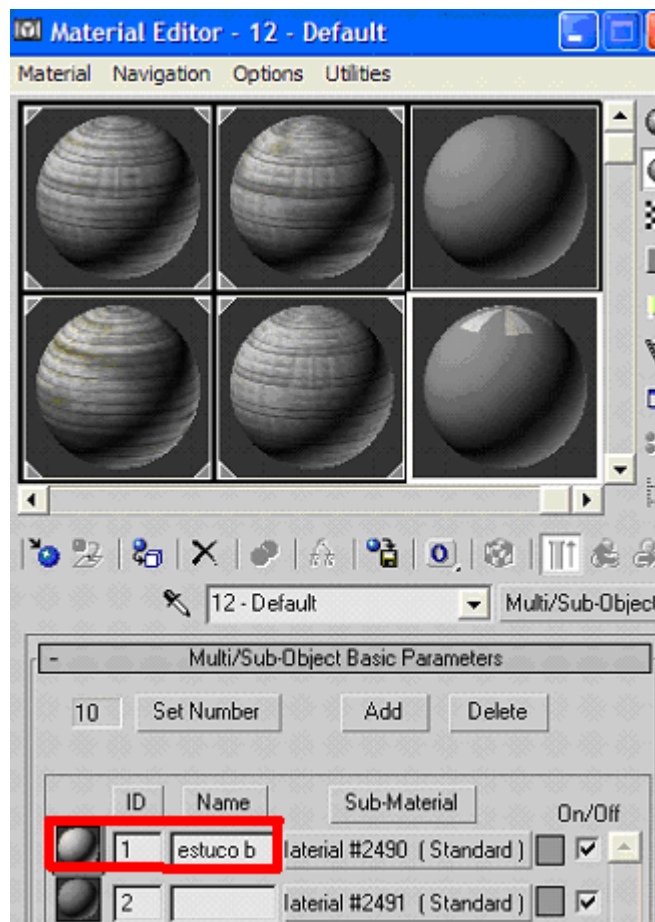


ILUSTRACIÓN VI.42
Material asignado a un ID

Después de almacenar las texturas dentro del Multi/Subobjet, se procede a la aplicación de cada una de ellas, para esto se utiliza el botón de asignación de material a la selección.

Antes dentro de las características del polígono editable, en polygon properties (propiedades del polígono) podemos asignar el material correspondiente dentro del Multi/Subobjet, el cual pide como entrada un ID, que debe coincidir con el del material correspondiente.

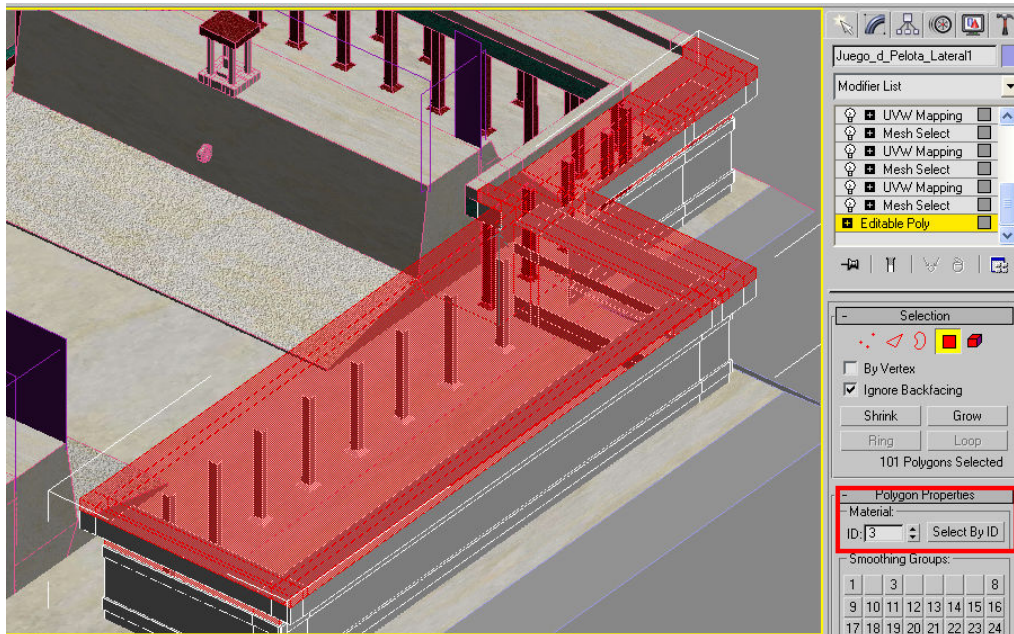


ILUSTRACIÓN VI.43
Asignación del ID para el estuco blanco

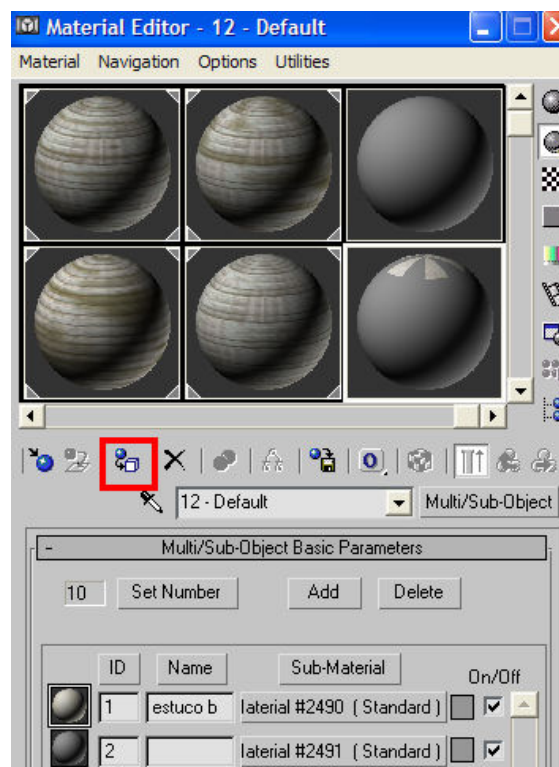


ILUSTRACIÓN VI.44
Botón de asignación de material

Debemos aclarar que para nuestro proyecto utilizamos imágenes JPEG, en su mayoría de baja resolución así como algunas de formato BMP y PNG (estas dos últimas para texturas con transparencia) para aplicar las texturas, siendo que 3D Studio Max utiliza diferentes tipos de formatos de imagen, pero se utilizaron los anteriores por su fácil uso en Virtools.

VI.1.2.1 Modificador UVW Mapping

Se necesita acomodar cada una de las texturas para crear el resultado deseado. Para este acomode necesitamos un modificador, el cual se encuentra dentro de la lista de modificadores y es UVW mapping.

Lo que realiza este modificador es aplicar las coordenadas de mapeo a un objeto, para controlar el orden de las imágenes que serán proyectadas sobre la superficie del objeto o de alguna de sus partes antes seleccionadas con un ID material.

Este modificador cuenta con distintos parámetros que sirven para controlar en cualquier momento la imagen entre los que contamos los siguientes:

- **Parámetros de mapeo:** Asigna el tipo de mapeo que se aplicará al objeto, el cual puede ser sobre un plano, un cilindro, una esfera, una caja o por cara es decir que la textura se repetirá en cada polígono del área seleccionada.
- **Parámetros dimensionales:** Indica el tamaño de cada uno de los elementos mencionados en el parámetro anterior para ajustar el tamaño de la imagen y así se ajuste a lo requerido para cada estructura en particular.
- **Parámetros de repetición:** Controla la repetición de la imagen dentro del gizmo², en sus coordenadas U, V, W (solo es otro sistema de referencia igual al X, Y, Z).

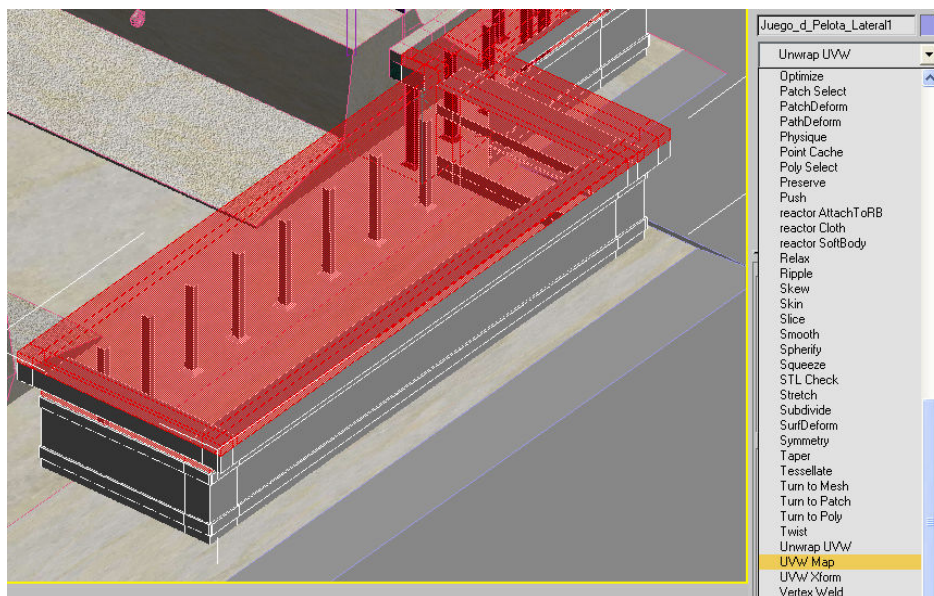


ILUSTRACIÓN VI.45
Modificador UVW Mapping

² Modificadores desplegados en los puertos de vista, se puede mover, escalar y rotar, alterando el efecto del modificador en el objeto.

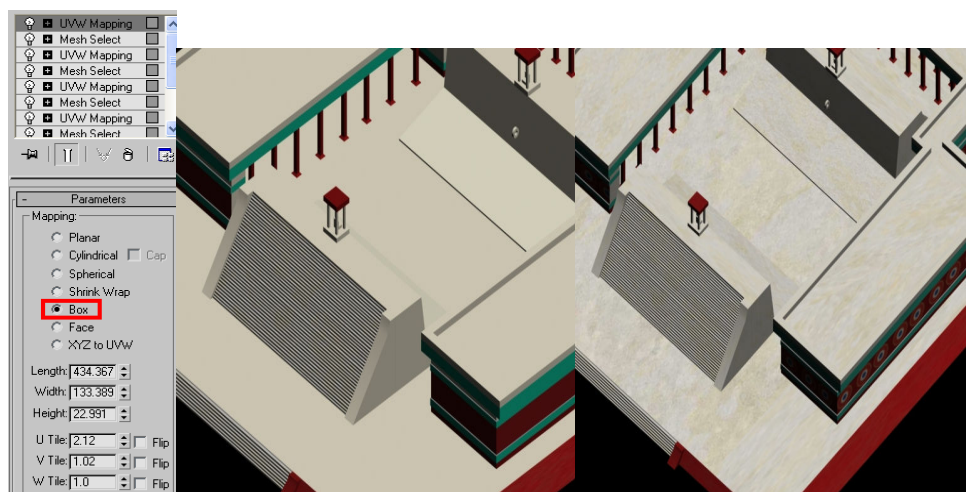


ILUSTRACIÓN VI.46
Mapeado en una caja

Con lo expuesto anteriormente englobamos al texturizado del personaje, simplemente que para este se utilizaron un mayor número de texturas, pero el procedimiento es exactamente el mismo. Ahora el personaje cuenta una característica que ninguno de nuestros objetos dentro del espacio virtual tiene, esta es la movilidad, para lo cual se desarrollaron animaciones para las distintas acciones programadas para este personaje.

A continuación explicaremos el proceso para el desarrollo de una animación.

VI.1.3 Desarrollo de Animaciones

Una vez texturizado el modelo el siguiente paso es el set up (en donde se crea un esqueleto que sustentará el movimiento del modelo) y el rig³ (donde se relaciona cada vértice del modelo con el movimiento de un hueso del esqueleto), esto permitirá animar al personaje.

Para crear el esqueleto utilizaremos la opción de Character Studio incluida en la versión de 3D Max. Para utilizarla vamos al panel de creación y seleccionamos el icono de Systems.

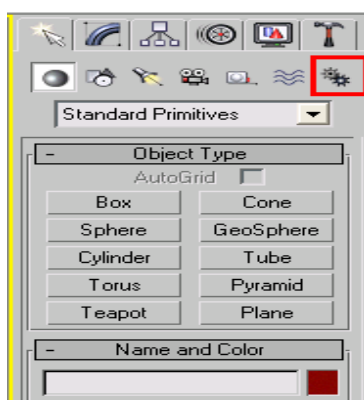


ILUSTRACIÓN VI.47
Ícono de Systems

³ Es una herramienta para los animadores que implementa una media sobre el objeto a animar.

Ahora seleccionamos la opción Biped para el esqueleto a utilizar, para hacerlo damos un click en el área de trabajo y arrastramos el mouse hasta que tenga el tamaño deseado.

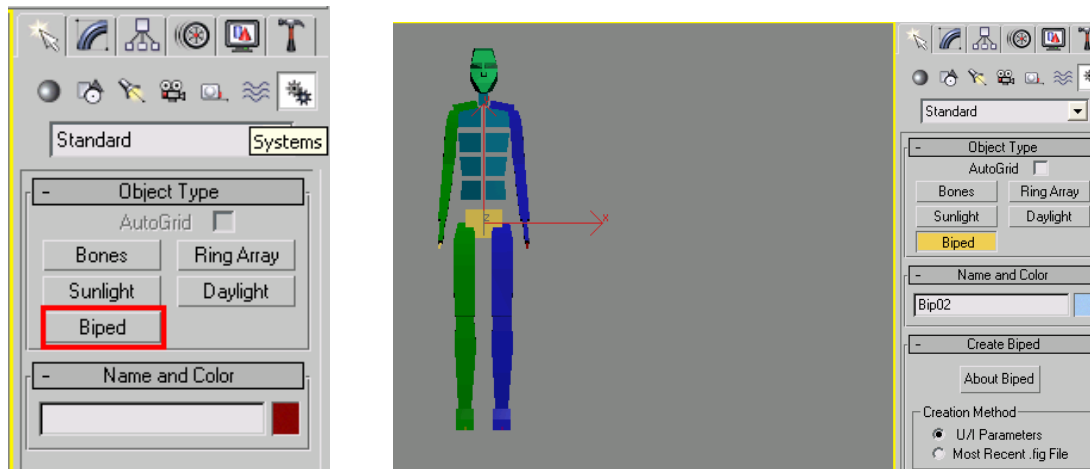


ILUSTRACIÓN VI.48
Opción Biped

El esqueleto es colocado dentro del modelo como un esqueleto real.

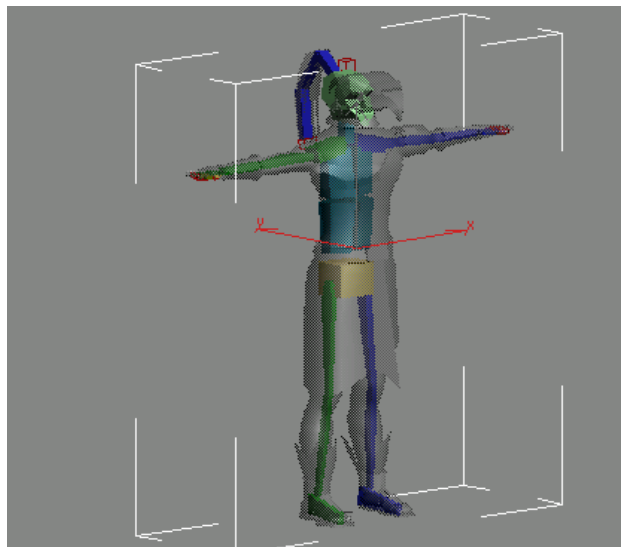


ILUSTRACIÓN VI.49
Esqueleto Caballero Águila

Para modificar los huesos del esqueleto y que estos queden contenidos dentro del modelo, vamos a la pestaña de Motion y seleccionamos la opción Figure Mode, de esta forma podemos aplicar transformaciones tales como escalar, rotar o trasladar a los diferentes elementos que conforman el esqueleto, así como el número de articulaciones que existen en las uniones entre los huesos. Al terminar el proceso es necesario desactivar la opción de Figure Mode.

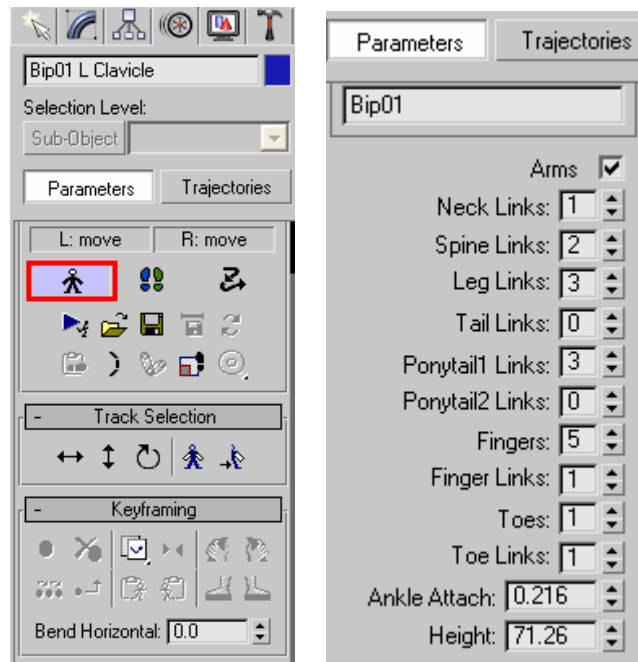


ILUSTRACIÓN VI.50
Opción Figure Mode

Para implementar el rig agregamos a la pila de modificadores la opción skin la cual unirá al esqueleto y al modelo.

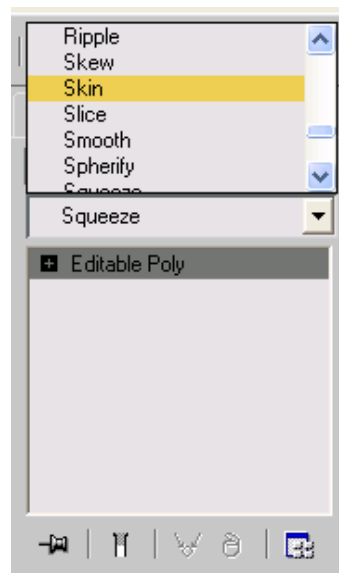


ILUSTRACIÓN VI.51
Opción Skin

En el panel de modificación seleccionamos Edit Envelopers y agregamos los huesos a la lista mediante la opción Add la cual despliega lo siguiente:

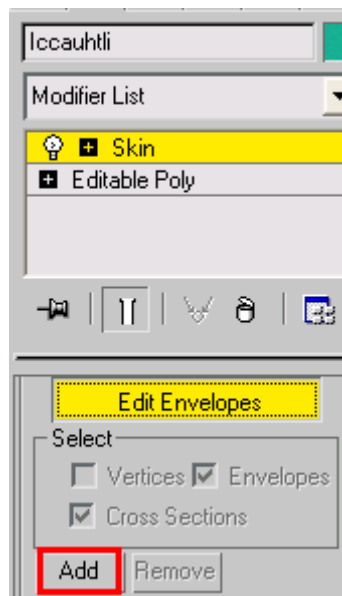


ILUSTRACIÓN VI.51
Opción Add

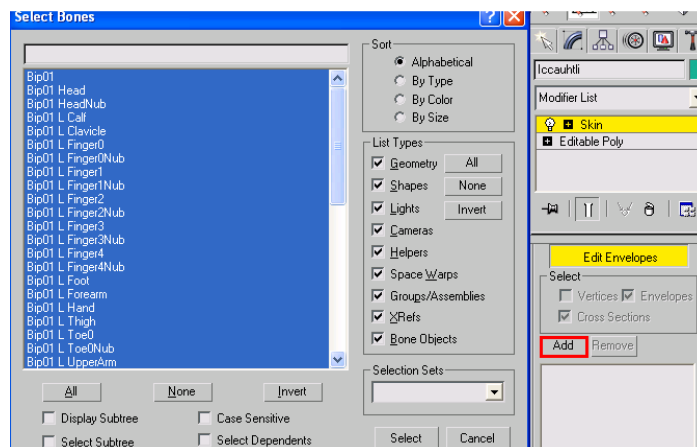


ILUSTRACIÓN VI.52
Pantalla para añadir elementos del Biped

Para una mejor visualización seleccionamos algunas opciones como Show no envelopers, Show color faces y show color vértices. En Advance parameters seleccionamos el numero de huesos que influirán en el movimiento de otros huesos en este caso seleccionamos 3 en la opción Bone Affect Limit, para simplificar.

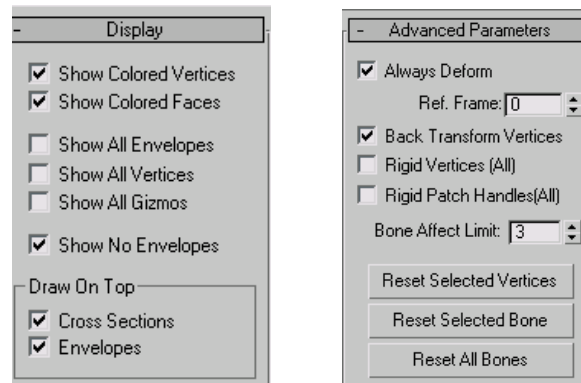


ILUSTRACIÓN VI.53
Opciones Character Studio

El siguiente paso es crear una animación sencilla de un movimiento utilizando la opción Auto Key.

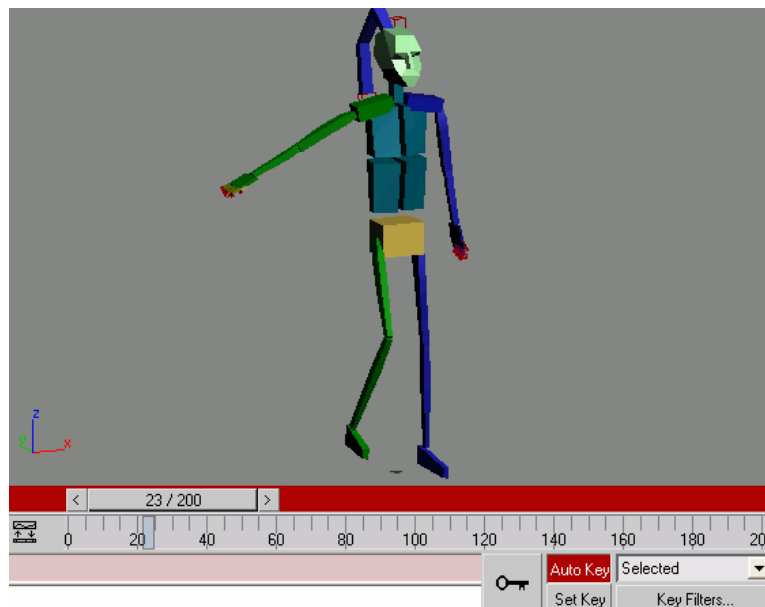


ILUSTRACIÓN VI.54
Opción Autokey

Después debemos asignar los valores a los diferentes vértices, seleccionando uno o varios, así como un valor entre cero y uno, lo cual sirve para mantener una jerarquía en cuanto al movimiento de cada uno de los puntos seleccionados, hasta que el movimiento luzca natural.

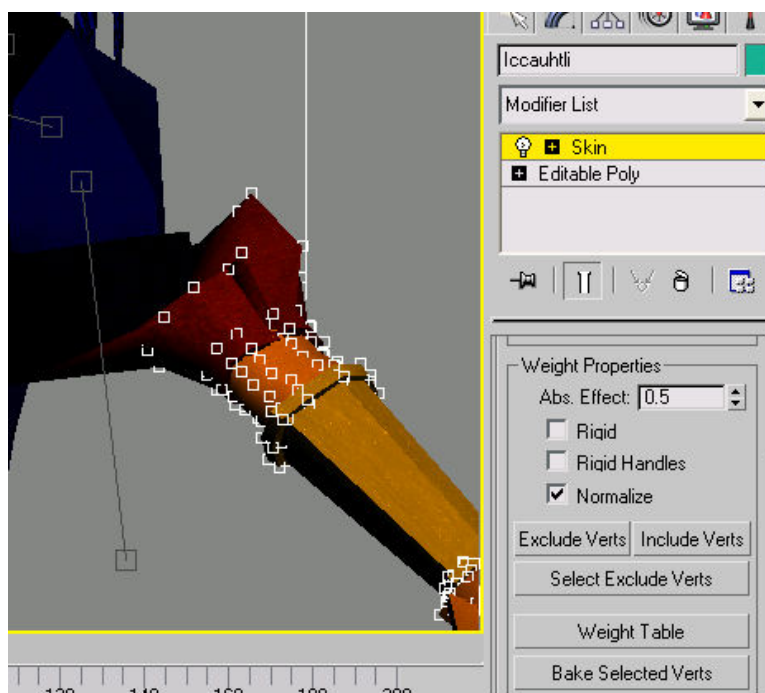


ILUSTRACIÓN VI.55
Puntos de Skin

Una vez que se tienen listas las acciones anteriores para cada una de las partes del personaje que van a contar con movimiento se procede a realizar cada animación.

Por último se coloca en su posición cada uno de los huesos del personaje de forma que cada una de esas posiciones quede grabada con ayuda del Auto Key, de la misma manera en como se anima una caricatura, en la que se tienen diferentes posiciones de un personaje, las cuales serán presentadas en un orden y tiempo determinado para simular movimiento. Solo que aquí el comportamiento será en tres dimensiones.

Después de concluidas las tareas tanto de modelado, texturizado y animación de los objetos, se procede a la exportación de cada uno de estos elementos a Virtools, para desarrollar el recorrido virtual y el juego de recolección.

VI.2 Exportación

Para exportar nuestro trabajo elaborado en 3D Studio Max, se utiliza el archivo especial * .NMO, una vez que guardamos nuestro modelo terminado (la malla completa, con sus texturas y un mapeado adecuado), podemos exportar eligiendo la opción Export (exporta) del menú de File (Archivo) de 3D Studio Max..



ILUSTRACIÓN VI.56
Función Export

Para después guardar el archivo de la siguiente manera

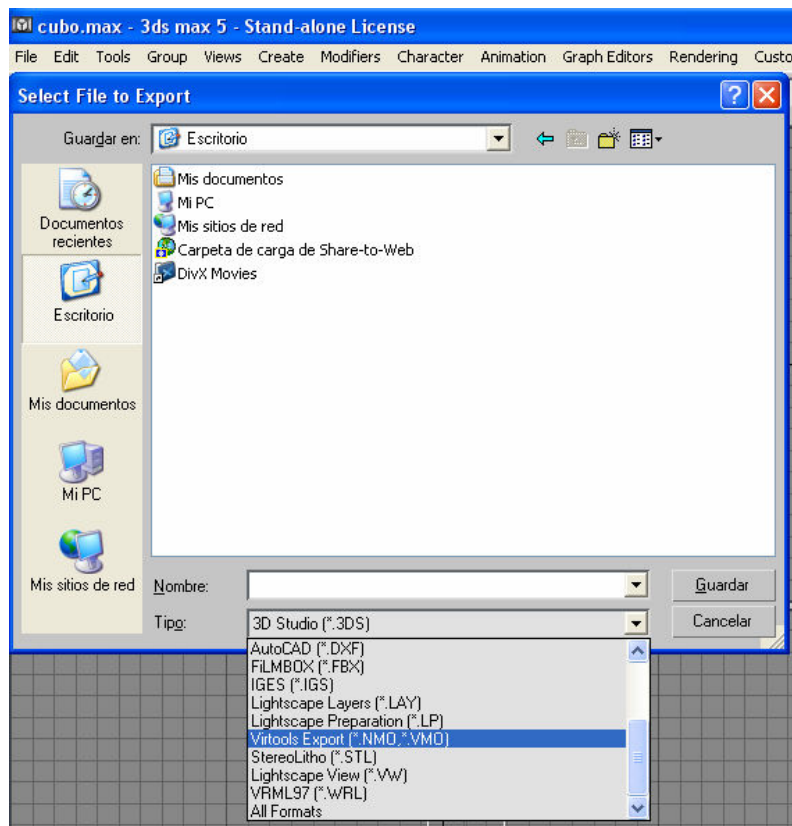


ILUSTRACIÓN VI.57
Pantalla de selección de Archivo

Esta pantalla sirve para introducir los parámetros de exportación.

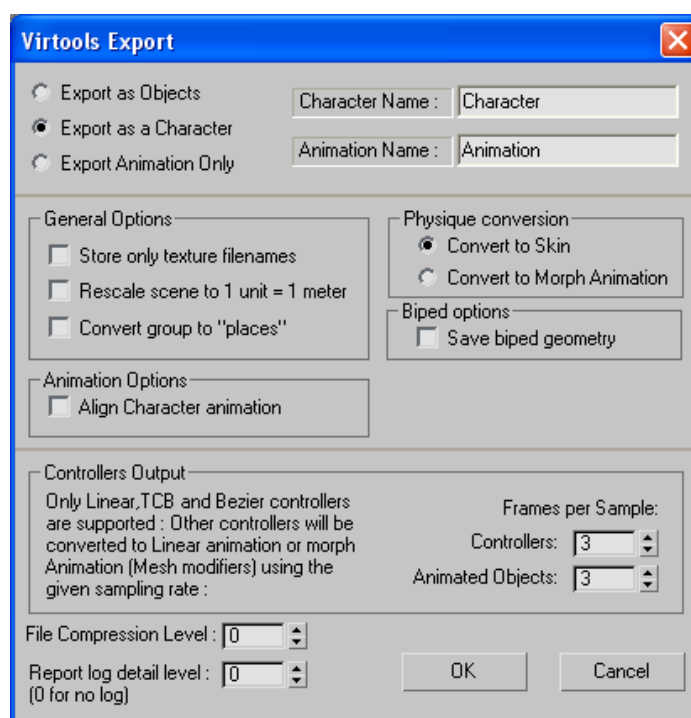


ILUSTRACIÓN VI.58
Pantalla de Virtools Export

Desde aquí podemos hacer tres tipos de exportaciones.

- Exportar Objetos
- Exportar Personajes
- Exportar Animaciones

Se puede exportar el proyecto completo o por partes, lo cual es mejor, ya que así podemos corregir errores sobre la arquitectura.

Primero exportamos todos los elementos ambientales, para después exportar al personaje y por último las animaciones del personaje. Desde la pantalla anterior, también se puede dar nombre, tanto al personaje como a cada una de sus animaciones.

VI.3 Desarrollo del Recorrido Virtual y Juego

Lo primero que tenemos que hacer en esta etapa es importar todos los elementos construidos y que se encuentran ahora guardados como archivos *.nmo. Lo siguiente es programar las acciones en tiempo real que van a realizarse dentro de nuestro mundo virtual, entre ellas la iluminación, las sombras, los efectos de sonido, efectos de agua, el cielo y otros efectos ambientales que pueden programarse desde Virtools.

Primero se programan los eventos del recorrido virtual y después los del juego, el cual incorpora los mismos elementos del recorrido, solo con la adición de otros que lo caracterizan y diferencian del primero.

Entre los elementos programables del recorrido virtual, que están separados en scripts⁴ tenemos:

- Detección de colisiones tanto para las cámaras como para el personaje.
- Control, posición y cambio de las diferentes cámaras.
- Mapa, el cual resulta muy útil para la navegación.
- Script para la descripción de cada edificio.

Para el juego se incorporaron elementos para cada una de sus funciones.

El objetivo del juego es el de recolectar objetos en siete minutos, aquí varía el número de objetos, así como el tipo dependiendo de la dificultad, la cual va desde fácil (cinco ofrendas), intermedio (ocho vasos ceremoniales) y avanzado (doce vasijas).

En el desarrollo del juego y con las características antes mencionadas, se requerían scripts para las siguientes acciones:

- Mostrar cada una de las piezas dentro del mapa, así como al personaje, con diferente color para evitar confusiones.
- Una función para recolectar cada una de las piezas y mandarlas a un contador, que muestre el marcador en pantalla.
- Una función que esconda las piezas después de haber sido recolectada.
- Una función que arranque un reloj, el cual parara cuando se cumpla con el objetivo.
- Si no se cumple con el objetivo se mande a una pantalla para intentarlo de nuevo o salir del juego.

Para simplificar ésta etapa del desarrollo, nos dedicaremos a explicar el funcionamiento de los principales scripts de este proyecto. Explicaremos cada una de las funciones comentadas, así como cada una de las operaciones que se necesitan para cada script.

La forma de programar en Virtools es gráfica a través de bloques llamados Building Blocks los cuales tienen funciones específicas que permiten realizar una tarea determinada, una operación matemática o ejecutar algún proceso, estos pueden ser interconectados entre sí para realizar procesos más complejos, de manera similar a como se arma un circuito.

Este programa también permite al usuario elaborar sus propios bloques a partir de operaciones matemáticas, variables locales y globales o utilizando bloques creados por otros usuarios o por uno mismo a través de su lenguaje VSL.

⁴ Conjunto de comandos escritos en un lenguaje interpretado para automatizar ciertas tareas de aplicación

VI.3.1 Script para el Mapa

Para la creación del mapa se requiere, un script que este sensando en todo momento donde se encuentran cada uno de los objetos en el espacio virtual a representarse dentro del mismo, al ser un script largo lo haremos en tres partes.

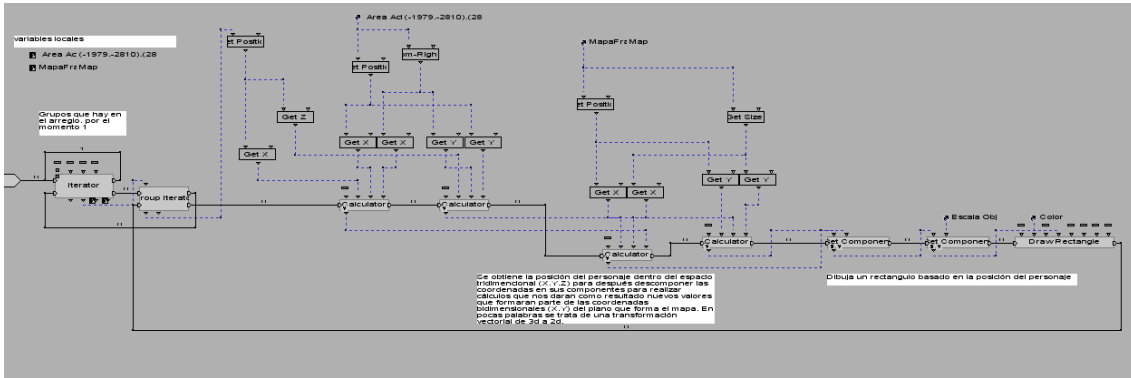


ILUSTRACIÓN VI.59
Script para el Mapa

Primero necesitamos calcular el Área Activa, esta será el área donde se estará sensando el movimiento de los objetos dentro del espacio virtual, después asignar el objeto en 2D sobre el que se dibujaran las posiciones, y esta corresponde a el área designada para desplegar el mapa en pantalla. También se necesita un arreglo, este arreglo de datos contendrá tres grupos uno con los objetos sensados, otro con el color a desplegarse en el mapa y el tercer grupo el tamaño que ocuparan en el mapa.

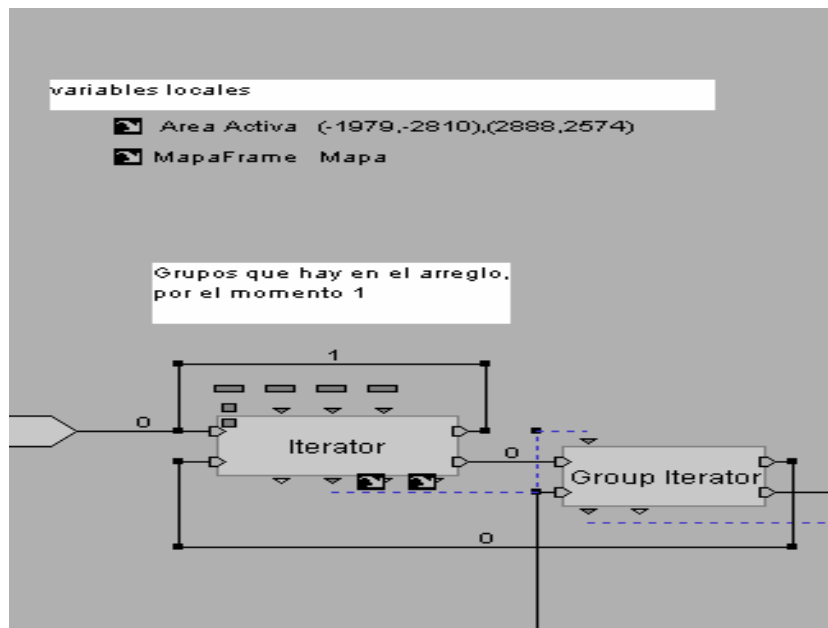


ILUSTRACIÓN VI.60
Script Mapa Primera Etapa

La segunda parte del script obtiene la posición de los objetos en el espacio virtual, y los descompone en sus componentes X, Z (no toma en cuenta la altura Y porque no es necesaria para este cálculo) en relación al mundo virtual, una vez obtenidos estos datos son enviados a un segundo proceso donde se realiza una transformación para obtener las componentes X', Y' que corresponde a una posición determinada dentro del área del mapa.

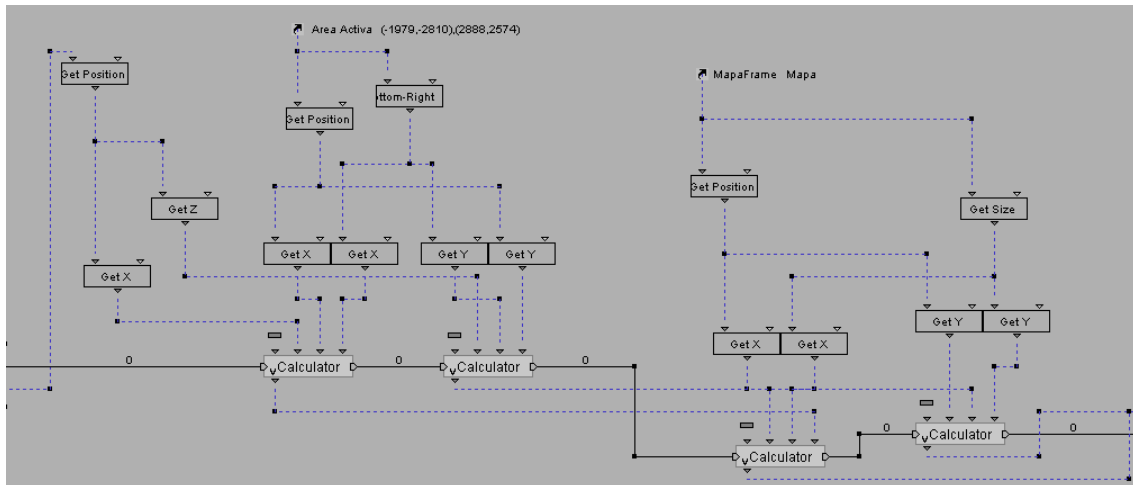


ILUSTRACIÓN VI.61
Script Mapa segunda Etapa

El último proceso simplemente dibuja un cuadrado en la posición X' Y' dentro del mapa que corresponde a un objeto ubicado en una posición similar en el mundo virtual.

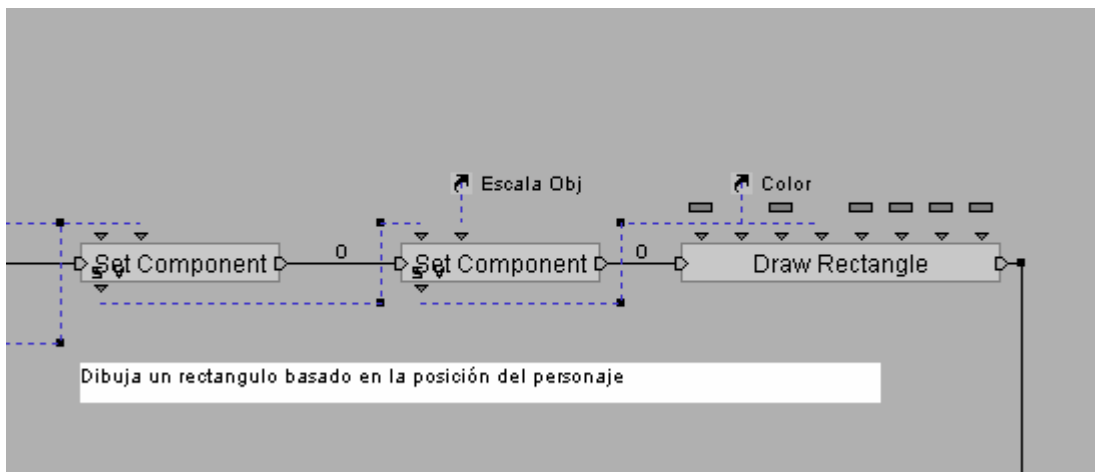


ILUSTRACIÓN VI.62
Script Mapa Tercera Etapa

Para al final obtener el siguiente resultado:

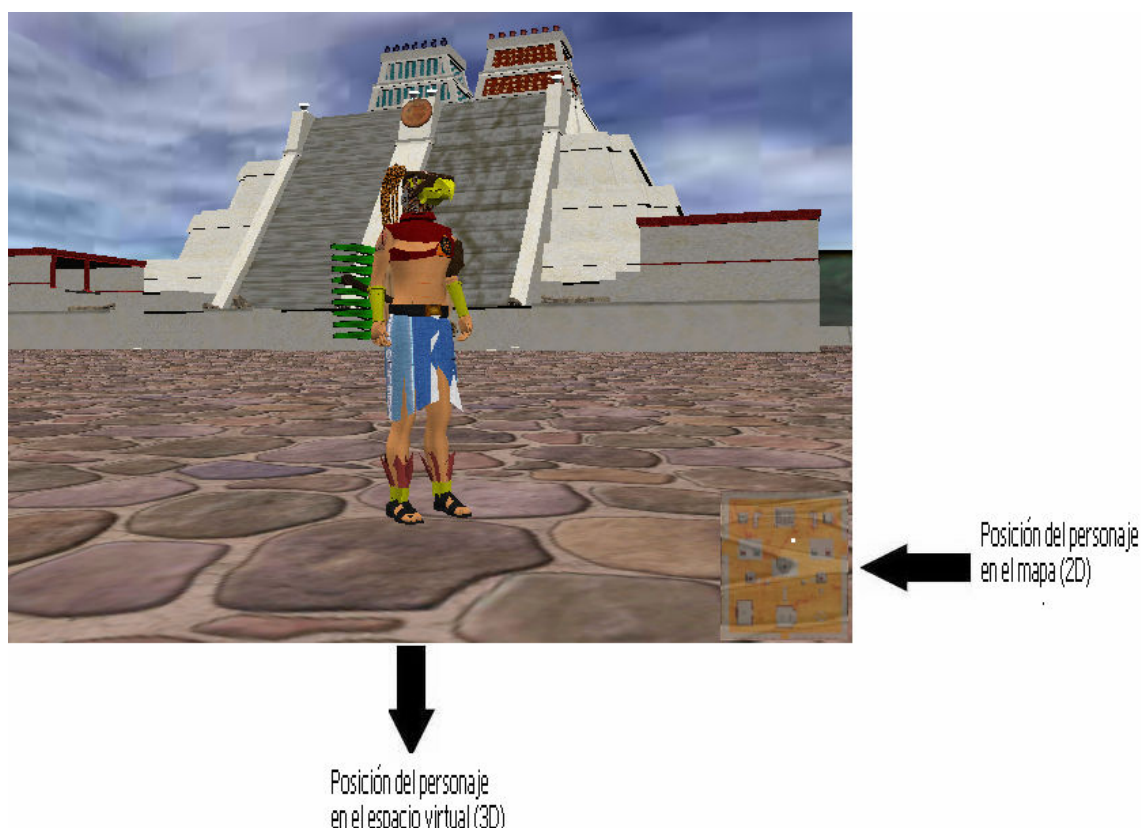


ILUSTRACIÓN VI.63
Posición dentro del mapa

VI.3.2 Script para los controles del personaje

Para el control del personaje se necesitan dos scripts diferentes uno de ellos para el control del teclado y otro para el joystick.

VI.3.2.1 Script para el Teclado

El teclado utiliza el bloque Character Controller, éste es el bloque en el cual se controlan las diferentes animaciones que tiene el personaje. Keyboard Mapper se utiliza para trasladar el movimiento que realiza el joystick al teclado, es aquí donde se elige la función que tendrá cada tecla. El bloque Character Keep On Floor hace que el personaje solo pueda desplazarse por los objetos declarados como piso.

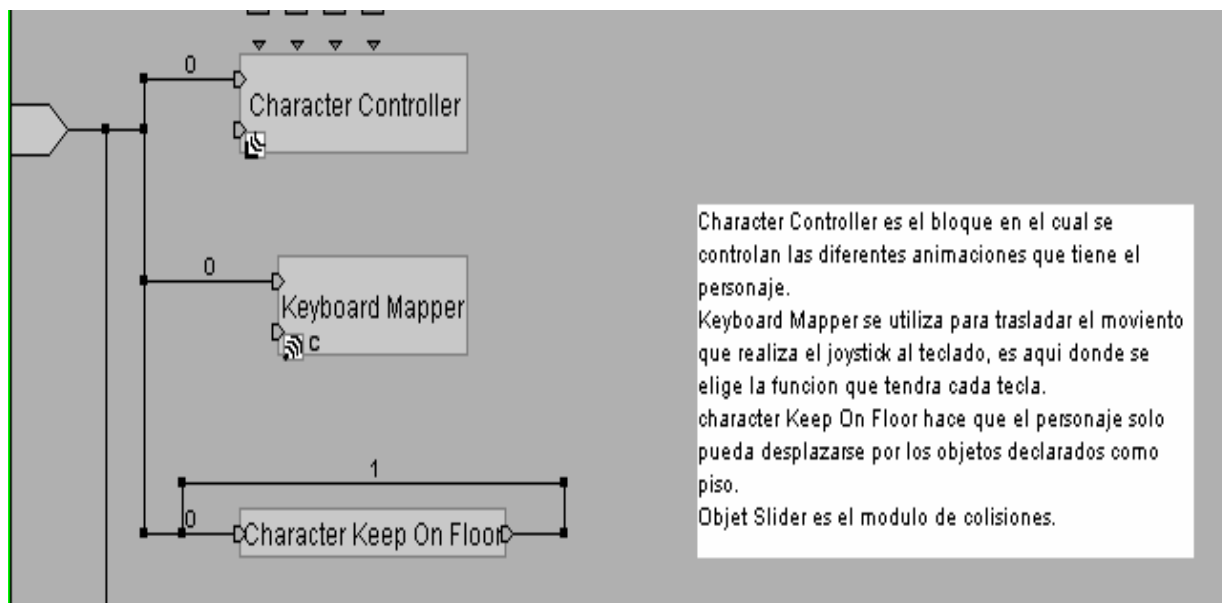


ILUSTRACIÓN VI.63
Script Control Teclado

VI.3.2.2 Script Joystick

Para la programación del joystick utilizamos el bloque llamado Joystick Waiter, el cual espera una señal desde el joystick para activar sus salidas. Este bloque tiene una variable llamada Analog Position, la cual es un vector con coordenadas en X,Y,Z, las cuales pueden tomar los valores de -1 a 1. X tomará el valor -1 cuando el joystick se encuentra presionado hacia arriba, 1 cuando se encuentra presionado hacia abajo y cero cuando no se encuentre presionado. De igual forma Y tomará el valor -1 cuando se encuentre presionado a la izquierda y 1 cuando se encuentre a la derecha, también tendrá el valor de cero cuando no este presionado.

Cabe aclarar que estos son los movimientos que se pueden realizar teniendo un solo stick, como el control que estamos utilizando esta provisto con un par, podemos incluir dos movimientos más, moviendo el segundo stick de arriba hacia abajo se obtiene la componente Z del vector Analog Position y sus valores también estarán entre -1 y 1. El movimiento de izquierda a derecha de este segundo stick controlan la componente Z de otro vector llamado Rotation el cual también esta dado en componentes X,Y,Z. Así que con el primer stick podemos controlar al personaje y con el segundo a la cámara que lo enfoca.

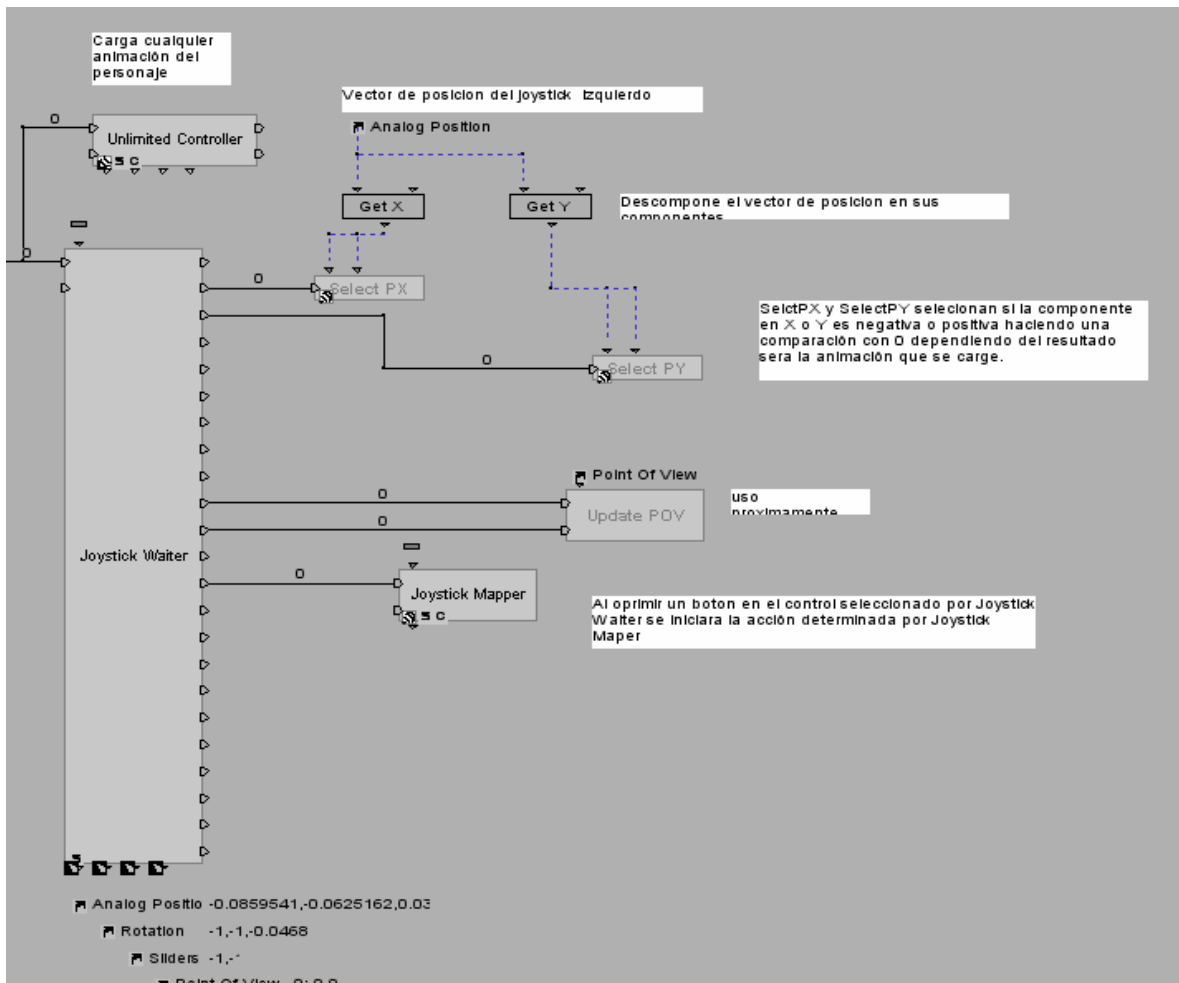


ILUSTRACIÓN VI.64
Script para el movimiento del personaje con Joystick



ILUSTRACIÓN VI.65
Palancas de Movimiento

Ahora bien dependiendo de los valores (-1 a 1) que arroje el bloque Joystick Waiter, entrarán a un segundo bloque, donde se determina si este número es mayor o menor que cero utilizando un bloque llamado Test, y la salida de este estará conectada al bloque Joystick Mapper, el cual determina que botón o stick se ha presionado para enviar una señal al bloque Unlimited Controller, que dependiendo del botón presionado llamará a la animación correspondiente.

VI.3.3 Scripts de cámaras

En el proyecto se tienen tres tipos de cámaras:

- **Tercera persona:** Sigue al personaje acierta distancia dando como resultado una vista en tercera persona.
- **Vista trasera:** Sigue al personaje desde atrás y el usuario puede modificar el punto de vista.
- **Cámara Libre:** Donde el movimiento es libre sin la necesidad de enfocar al personaje

Aquí también es necesario crear dos controles, uno para el teclado y otro para el control como lo mencionamos anteriormente, pero en este caso la cámara es un objeto que solo puede ser controlado por un solo elemento, el teclado o el joystick, así que la solución más practica fue duplicar la cámara y colocarla exactamente en la misma posición que la original así se tienen dos cámaras con el mismo punto de vista, controladas por diferentes scripts.

VI.3.3.1 Script Cámara Tercera Persona

Es el script más sencillo entre las cámaras. El Bloque Look At hace que la cámara enfoque un objeto predeterminado, en este caso la estructura del personaje Bip01.

Keep At Constant Distance, coloca la cámara a cierta distancia de un objeto dado (la misma estructura utilizada anteriormente Bip01), de esta forma la cámara seguirá al personaje a donde este se mueva. También se tiene un Object Slider para que la cámara colisione con los objetos.

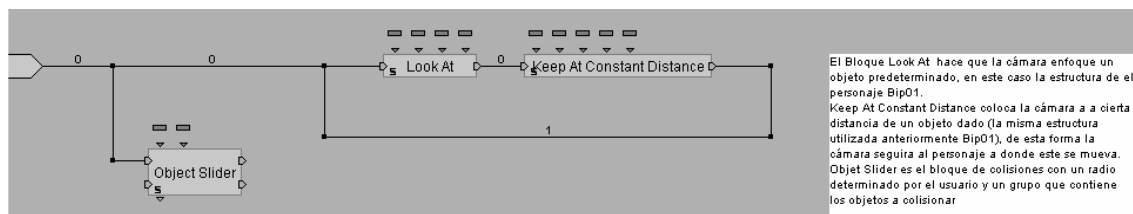


ILUSTRACIÓN VI.66
Script Cámara Tercera Persona

VI.3.3.2 Script Cámara Vista trasera

Para el teclado se tienen dos bloques Keep at Constant Distance, el primero fija la posición de la cámara y el segundo sigue al personaje donde este se mueva, Set Orientation es utilizado para que aunque el personaje se mueva el ángulo de visión que se tiene siga siendo el mismo. Keyboard Camera Orbit activa la rotación alrededor del personaje utilizando las flechas de dirección del teclado, así como el zoom utilizando las teclas Av.Pag y Reg.Pag

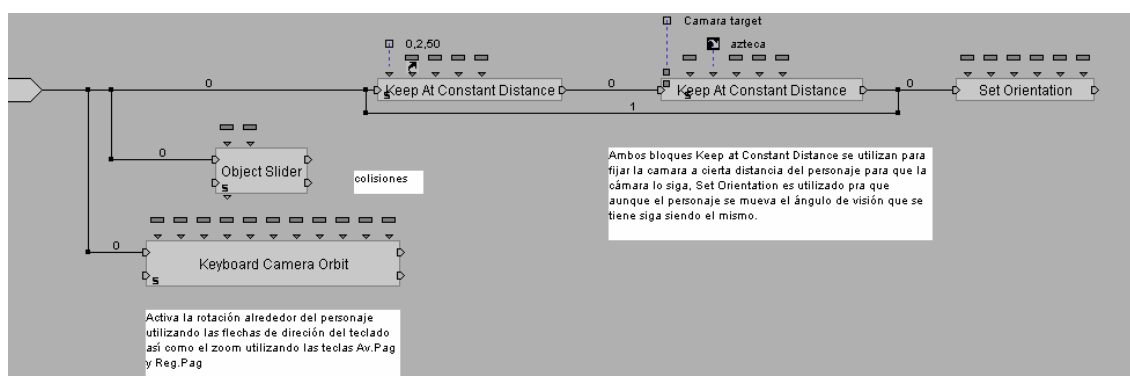


ILUSTRACIÓN VI.67
Script Cámara Sigue

Para el joystick el script que posiciona la cámara y la ancla al personaje es idéntico al script del teclado, la diferencia está en que la cámara orbite alrededor del personaje se utiliza un bloque Joystick Waiter, para enviar la señal al bloque Generic Camera Orbit que se activa cuando el segundo stick se ha movido y así realice el movimiento solicitado.

VI.3.3.3 Script Cámara Libre

En realidad son dos scripts el primero para la cámara el segundo para un frame 3D el cual servirá como base a la cámara, de igual forma la cámara se duplico para tener dos controles. Ambos la cámara y el frame se encuentran ligados, de forma jerárquica, la cámara está considerada como un subelemento del frame 3D, por lo que las transformaciones aplicadas al frame también se verán reflejadas en la cámara y no al contrario.

El script para el frame 3D contiene los controles para el joystick y el teclado ya que no tiene restricción alguna en su uso, para el teclado se utilizó el bloque Switch On Key el cual asigna una función determinada a una o varias teclas, la salida de este bloque está conectada a un par de bloques uno llamado Per Second donde colocamos las unidades que la cámara recorrerá en un segundo, el bloque Translate se encargará de realizar el movimiento de translación.

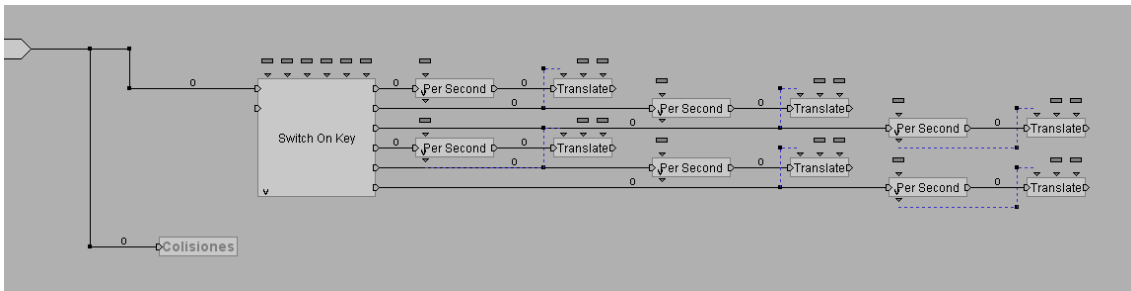


ILUSTRACIÓN VI.68
Script para Camara Primera Persona

Para el joystick una vez más se tiene Joystick Waiter y utilizando las componentes X,Y de Analog Position podemos hacer que el frame se desplace utilizando los mismos bloques Per Second y Translate, para que el frame 3D rote de lado a lado, se utiliza la componente Z del vector Rotate, de la misma forma utilizando un Per Second para determinar cuantos grados gire por segundo y un bloque Rotate (tiene el mismo nombre que el vector pero son diferentes) que realiza este movimiento.

Este script permite que la cámara gire únicamente, sin girar el frame 3D, como lo hace el joystick, de esta forma podemos asignar la rotación de la cámara de arriba hacia abajo, derecha a izquierda para el movimiento del mouse, para esto utilizamos el bloque Get Mouse Displacement, el cual recibe como entrada un vector 2D que representa el desplazamiento del mouse hacia arriba y hacia abajo y lo envía al bloque multiplicador para calibrar la velocidad de rotación mediante un ángulo.

El bloque Rotate realiza la rotación con el parámetro enviado por el bloque de multiplicación para calibrar la velocidad de movimiento. Get Mouse Position recibe como entrada el vector 2D con las coordenadas en (X,Y) del movimiento del mouse hecho de lado a lado, se adhiere un bloque de suma y multiplicación los cuales sirven para calibrar la velocidad de giro, la salida de este bloque es enviada a otro llamado Set Euler Orientation el cual convertirá la posición del Mouse en un ángulo y a su vez envía una señal al frame 3D para que rote en este ángulo, debido a que la cámara es una subestructura del frame3D no puede hacerlo directamente.

En el caso del Joystick únicamente utilizamos la componente Y del vector Analog Position para rotar la cámara de arriba hacia abajo, de nuevo utilizando un bloque Per Second, para determinar el ángulo a girar y Rotate para realizar el giro.

VI.3.4 Script de colisiones

Para las colisiones se utilizan varios bloques Object Slider, estos bloques funcionan utilizando dos parámetros, uno el grupo de objetos a colisionar y el otro es el radio de colisión. La razón de que existan varios Object Slider, es por que ya integrados los objetos del mundo virtual al colisionar con el personaje, no reaccionaban de la misma forma, en algunas ocasiones atravesaba paredes o no se podía pasar cerca de alguna estructura, por ello se decidió tener varios radios de colisión y para que estos funcionaran adecuadamente, los objetos se dividieron en grupos dependiendo del radio.

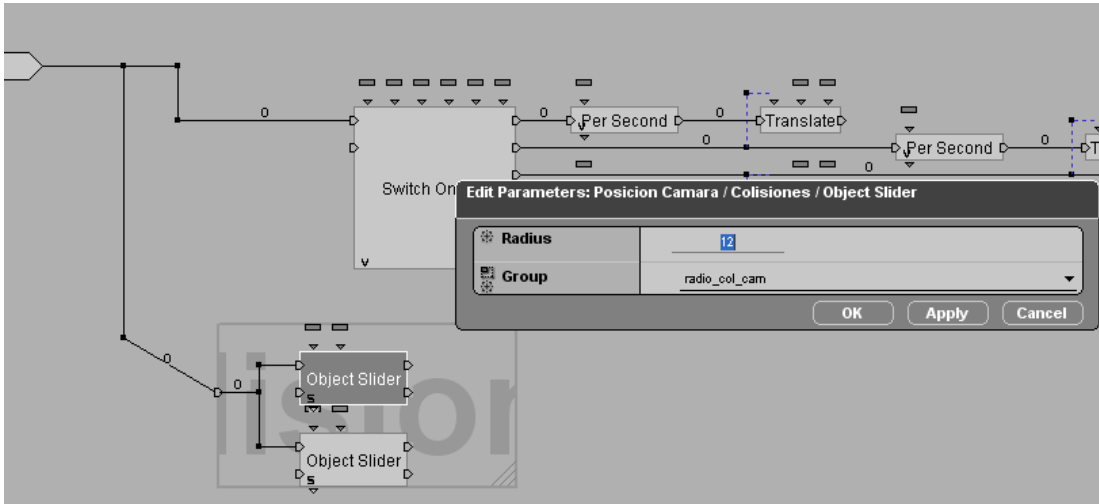


ILUSTRACIÓN VI.69
Script Colisiones

VI.3.5 Scripts para el juego

Para que el juego trabaje adecuadamente se tienen varios scripts, el más importante es el script General, el cual realiza las funciones más importantes y coordina los scripts secundarios.

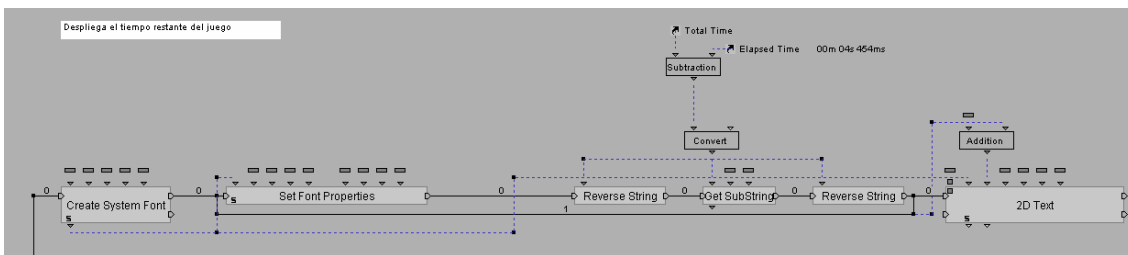


ILUSTRACIÓN VI.70
Script para el Cronómetro

Este script está formado en primer lugar por los bloques Create System Font y Set Font Properties los cuales seleccionan el tipo y formato de fuente para desplegar información del tiempo que se ha jugado, a la salida de estos bloques se realiza una operación de resta ya que el tiempo se mide en una cuenta regresiva y el resultado es enviado a un bloque 2D Text para ser colocado dentro de un frame 2D, esto con el fin de que se visualice en la pantalla.

En la siguiente sección del script se tiene el bloque Timer, que coloca el tiempo de juego, cuando este llega a cero se activa el bloque Send Message el cual envía un mensaje al script frame cielo para que se desactive (frame cielo coloca una textura de cielo alrededor de la ciudad y se tiene que desactivar para que no interfiera con otras escenas como el menú) y mande al menú de inicio del juego o salir. Si el usuario quiere salir del juego antes de terminar tendrá que presionar la tecla ESC del teclado.

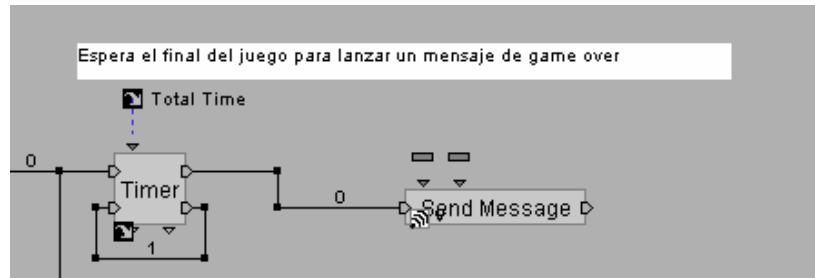


ILUSTRACIÓN VI.71
Bloque Timer y Send Message

La última sección de este script tiene la función de realizar la cuenta de los objetos que ha recolectado el usuario en el juego, este se encuentra formado por un bloque Identity que coloca la variable cuenta en cero cada vez que inicia el juego. Se tienen varios bloques Wait Message que esperan un mensaje de los objetos recolectados para activarse y enviar una señal al bloque Op, que realiza una suma, la cual se acumula en la variable cuenta, cuando la cuenta nos indica que se recogieron todas las piezas se lanza otra pantalla donde se le indica al usuario que es el ganador, donde puede seguir jugando o ir al menú de inicio.

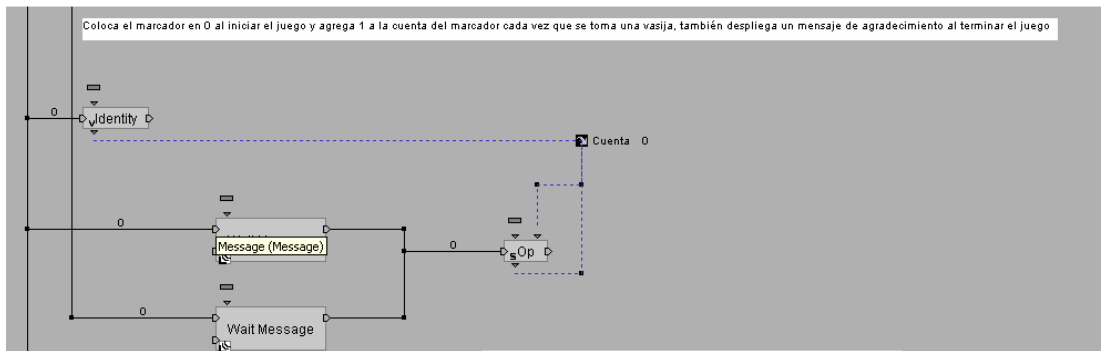


ILUSTRACIÓN VI.72
Script Activador del Contador de Piezas

Cada objeto a recolectarse tiene un script similar, esta formado por un bloque Show que muestra el elemento cada vez que inicie el juego, después se tiene un bloque Proximity, que sensa la distancia entre el personaje y el objeto, cuando el personaje se acerca a cierta distancia se activa un Key Event para el teclado y un Joystick Waiter para el joystick, cuando el usuario active la tecla o el botón indicado, manda el mensaje esperado por los bloques Wait Message en el script General, por último el objeto es escondido utilizando un bloque Hide y desactivado por el bloque Deactive Script para que su posición en el mapa sea borrada.

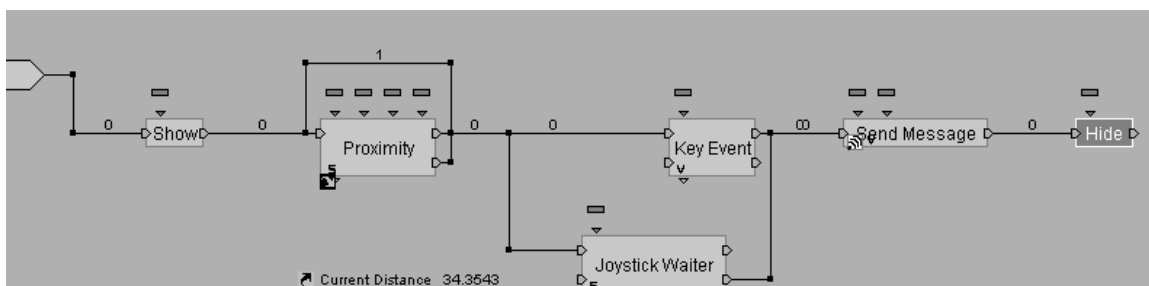


ILUSTRACIÓN VI.73
Script para Recolectar Piezas

Capítulo VII

Pruebas y Mantenimiento

VII.1.1 Pruebas sobre la construcción de los objetos tridimensionales

Las pruebas realizadas sobre el modelado fueron variadas, en primer lugar las correspondientes a las dimensiones de cada uno de los edificios, así como su ubicación dentro de la ciudad.

Posteriormente se hicieron pruebas sobre el texturizado, esto para conseguir mayor realismo respecto al la maqueta del museo del Templo Mayor en la que nos basamos para reproducir la ciudad, las cuales se hacían aplicando diferentes texturas y repetición de las mismas hasta obtener la imagen deseada.

La forma en la que se realizaron estas pruebas fue a través del render de los modelos uno por uno, obteniendo una imagen con texturas y con la iluminación ambiental que trae por default 3D Studio Max. Así nos podíamos dar cuenta si los resultados concordaban con lo planeado en etapas anteriores y de no ser así, hacer una corrección.

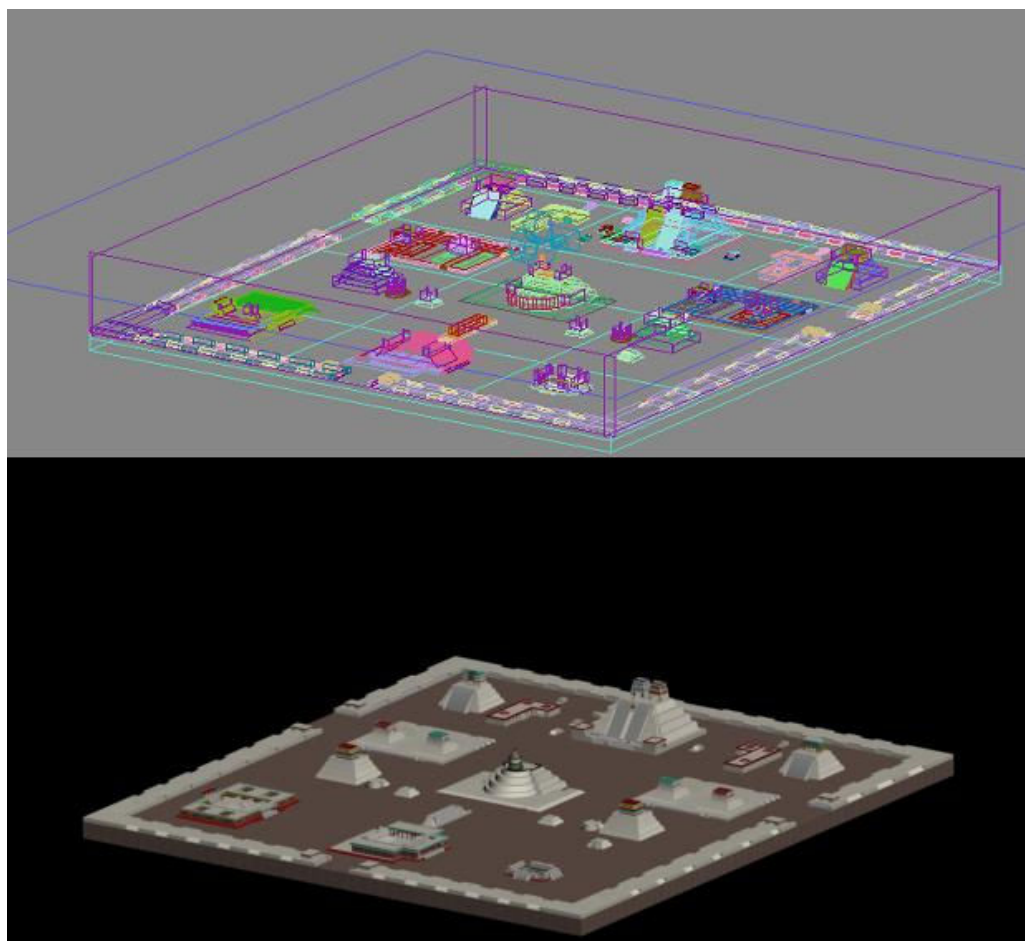


ILUSTRACIÓN VII.1
Render

VII.1.2 Pruebas para el Recorrido Virtual

En lo que se refiere al recorrido virtual, las pruebas consistían en la ejecución del programa cada vez que se agregaba un nuevo script, Vitoools tiene la ventaja de poder ejecutar en tiempo real el programa cada vez que el desarrollador lo requiera. Esto nos sirvió en un principio para probar si nuestra geometría se cargaba bien o para arreglar problemas con el modelado geométrico.

En nuestro caso sufrimos un problema, ya que al exportar la ciudad desde 3D Studio Max y después importarla nuevamente al proyecto nos dimos cuenta que la ciudad resultaba muy grande. A lo cual se decidió exportar e importar nuevamente la ciudad hasta obtener el resultado deseado.

Sobre el personaje se realizaron pruebas de navegación y de colisiones, las cuales consistieron en ver al personaje caminar por todos los sitios que podía recorrer libremente y en verificar que las colisiones con los demás objetos funcionaran correctamente, incluyendo las paredes y rampas invisibles dentro de la ciudad, las cuales ayudaban a delimitar el espacio para el personaje.

Otras pruebas que realizamos se referían a la incorporación del joystick como medio de interacción, lo cual consistía en regular la sensibilidad del mismo a diferentes movimientos del personaje.

Sobre el mapa, las pruebas ayudaron a ajustar el tamaño del frame 2D utilizado, así como para comprobar la posición del personaje en todo momento durante el recorrido.

Las pruebas sobre el juego incorporado al proyecto se realizaron respecto a las reglas del mismo, para analizar la dificultad y así ajustar un nivel adecuado para el usuario promedio, mediante la exposición al juego de a diferentes personas ajenas al proyecto (alrededor de 25 usuarios), con diferentes capacidades, es decir, gente de todas las edades con o sin experiencia en cuanto a juegos de video se refiere.

VII.2 Mantenimiento

Dentro de los tipos de mantenimiento los que mejor se adaptan a este proyecto son los siguientes:

- Mantenimiento de Adaptación
- Mantenimiento Perfectivo

Respecto al primero podemos mencionar que con el paso del tiempo, es probable que cambie el entorno original (CPU, el sistema operativo, características externas) para el desarrollo del proyecto en pocas palabras, el mantenimiento adaptativo produce modificación en el software para acomodarlo a los cambios de su entorno externo (hardware).

Por ejemplo si en el futuro nos encontramos con un engine más poderoso que el de Vitoools, es decir que permita mejorar tanto gráficamente como en procesamiento y podamos adaptar nuestro proyecto a esa nueva tecnología, así como a mejores maquinas en un futuro.

Ahora el mantenimiento perfectivo se refiere a encontrar mejoras o funciones nuevas que ayuden a enriquecer al programa. El mantenimiento perfectivo lleva al software más allá de sus requisitos funcionales originales.

En nuestro caso esto serviría para agregar nuevos módulos, como por ejemplo una mejor vista, o la incorporación de nuevas funciones o diferentes, así como incluir nuevas interfaces de interacción, etc.

Capítulo VIII

Conclusiones

El desarrollo de la tecnología computacional en las últimas décadas, ha permitido que la RV también crezca a pasos agigantados, brindándonos un gran número de aplicaciones que van desde la investigación científica y la enseñanza, hasta el uso recreativo.

En nuestro país se desarrolla software, pero desgraciadamente nos caracterizamos más en ser consumidores del mismo. Consideramos que crear tecnología en México no es muy difícil, sobre todo considerando que hay gente muy inteligente, capaz, creativa y totalmente comprometida. Lo difícil es lograr que esa tecnología sea aprovechada por más mexicanos cada día. Sobretudo a lo que se refiere al EI, el cual es un campo muy poco explorado por los desarrolladores mexicanos. Ya que no hay muchas compañías que se dediquen a esta actividad, aunque si eventos como el Creanimax en Guadalajara o el EGS (Electronic Game Show) en México, que intentan cambiar la perspectiva que se tiene sobre el EI, el cual no solo entretiene o divierte, si no también educa y mantiene interés por parte del usuario por descubrir nuevas formas de aprendizaje.

Un ejemplo claro de cómo la RV puede ayudarnos en el aprendizaje de nuestra historia es la reconstrucción virtual de sitios históricos, ya que a través de ellos podremos visitar lugares que con el tiempo han desaparecido para dar lugar a nuestras modernas ciudades, tal es el caso de la gran ciudad de Tenochtitlan, la cual con la llegada de los españoles fue transformándose para dar lugar a la Ciudad de México.

Tal vez se piense que los proyectos de RV son demasiado caros, por todos los aditamentos que se utilizan, y en parte es cierto, si la experiencia que se desea tener es completamente inmersiva, sin embargo la RV no está restringida a usar grandes aparatos de simulación; simplemente con una computadora, como la que ahora les es posible tener a la mayoría de las personas en sus hogares, es suficiente para lograr desarrollar mundos virtuales que nos permitan explorar diferentes lugares a través del mouse o del teclado.

Nuestra tesis estuvo enfocada a una zona arqueológica, pero una de las grandes ventajas de este proyecto, es que podemos expandir la idea para aplicarla a diferentes actividades, ayudando a visualizar lugares y conceptos de una forma diferente y divertida, es decir, la aplicación se puede expandir, no solo a usarse en la reconstrucción de sitios históricos, sino utilizarse en el estudio de las matemáticas, visualizando los diferentes cuerpos geométricos, en la física observando a los cuerpos que intervienen en un sistema, en la química para ver las moléculas y átomos, etc.

La labor de la Ingeniería en Computación debe estar concentrada en la construcción y desarrollo de proyectos que nos permitan aplicar métodos y técnicas para resolver los problemas, a través de herramientas y procedimientos apoyados en los siguientes objetivos:

- Mejorar la calidad de los productos de software
- Aumentar la productividad y trabajo de los ingenieros del software.
- Facilitar el control del proceso de desarrollo de software.
- Suministrar a los desarrolladores las bases para construir software de alta calidad en una forma eficiente.

- Definir una disciplina que garantice la producción y el mantenimiento de los productos de software desarrollados en el plazo fijado y dentro del costo estimado.

Todo lo anterior se sintetiza en que la ingeniería está para desarrollar soluciones a diferentes problemas que beneficien y faciliten el desarrollo de la sociedad. En nuestro caso el problema que tratamos es el de la educación, ya que lo que intentamos como tal en este proyecto, es ayudar a la difusión cultural, despertando el interés del usuario de este programa por conocer más sobre nuestra cultura y antepasados.

También nuestra intención con este trabajo de tesis es la de crear un sistema de RV, que utilice la incursión de un personaje, el cual nos represente dentro del espacio virtual y resulte así familiar al usuario que tiene experiencia con el EI (juegos de video), logrando simular a un videojuego de tercera persona, los más comunes en la actualidad, pero también está diseñado para cualquier otro usuario que no haya tenido experiencia alguna con el software antes mencionado (algo difícil en nuestros días), recalcando que es el fácil uso para los que no estén familiarizados con este tipo de software.

Podemos resumir el proyecto en cuatro grandes tareas:

La primera parte consistió en recopilar información para lo cual acudimos a diversas fuentes como libros, Internet, visitar la zona y el museo. Con todos los datos obtenidos comenzamos la reconstrucción 3D de la zona, una vez terminada editamos y aplicamos los distintos materiales y texturas que dan color a los edificios y piezas, terminando con el modelo y listo para exportarse al software que utilizamos para programar.

En la segunda fase nos enfocamos al diseño de cada uno de los elementos interactivos del mundo virtual como la iluminación, las colisiones, la navegación, el desarrollo del juego y el recorrido.

Nuestra tercera tarea consistió en llevar todo lo que se diseñó y desarrolló al mundo virtual, y empezar la programación, en primer lugar, la navegación con una cámara posteriormente para después introducir al personaje, teniendo estos dos elementos continuamos con el sistema de colisiones e iluminación para generar sombras y dar un aspecto real, después desarrollamos el juego; ésta parte en especial nos llevó un poco más de tiempo, y aunque se trata de un juego sencillo, requirió del uso de varios módulos para que tuviera un buen funcionamiento, con el juego terminado nos dedicamos a colocar pequeños detalles, que se harían para lograr un proyecto más vistoso, como sonidos, efectos ambientales y cuadros con información.

Por último recopilamos los elementos completamente desarrollados y los integramos en uno solo a través de una interfaz que fuera amigable para el usuario así como llamativa, con esto quedó terminado nuestro proyecto.

Los conocimientos adquiridos a lo largo de la carrera así como nuestra experiencia como “gamers” (jugadores) con el EI nos ayudaron a desarrollar de forma práctica un recorrido virtual por la zona Arqueológica del Templo Mayor, con lo cual se alcanzó el objetivo planteado en este trabajo de tesis, agregando a estos objetivos un “extra”, el cual fue la programación del juego de recolección de piezas, el cual también significó trabajo y una ardua labor de investigación.

Una de las ventajas con las que cuenta nuestro software es poder portar cualquier otra geometría y cualquier otro personaje o incluso incluir más personajes para la elaboración de un producto diferente, lo cual podría ser no solo un recorrido virtual, si no un juego de video completo, o el poder explorar cualquier otro espacio virtual real o cualquiera, esto al incorporar los elementos desarrollados en este trabajo de tesis.

ANEXO I

Manual de Usuario

Control del Menú

Dentro de esta pantalla solo hay que situarse con el apuntador del mouse y hacer click sobre la opción de interés.



ILUSTRACIÓN VIII.1

Imagen del la interfaz Menú Principal

Recorrido

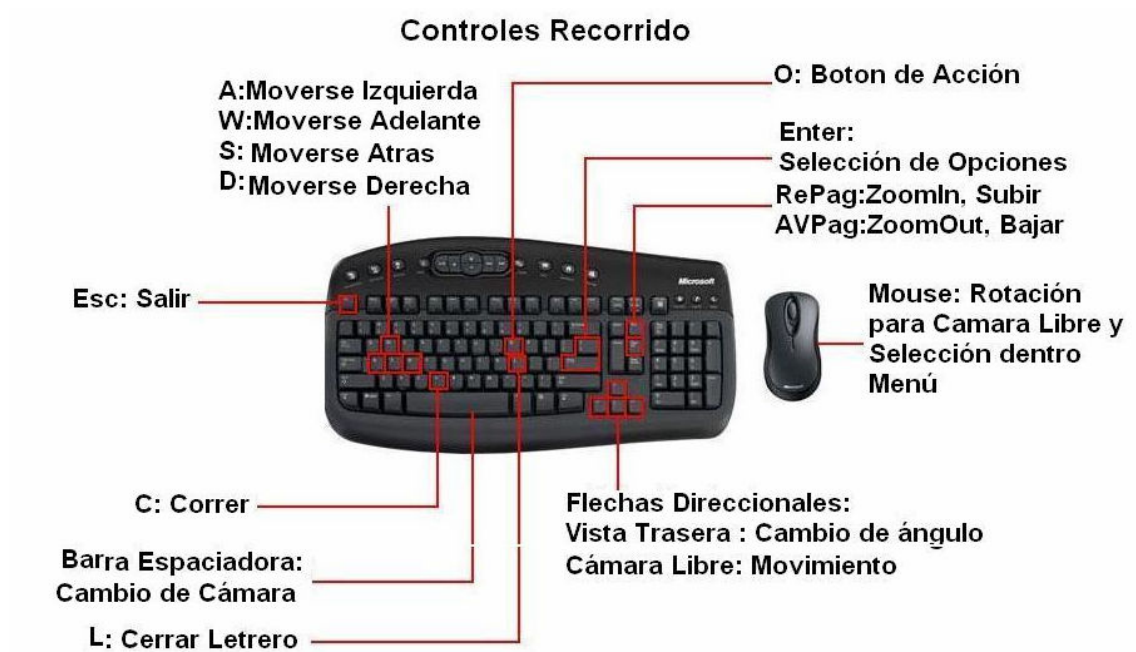
En esta opción puedes recorrer la Zona Arqueológica del Templo Mayor de forma libre, con la ayuda de tres cámaras, esto para no perder de vista ningún detalle de esta enigmática ciudad.



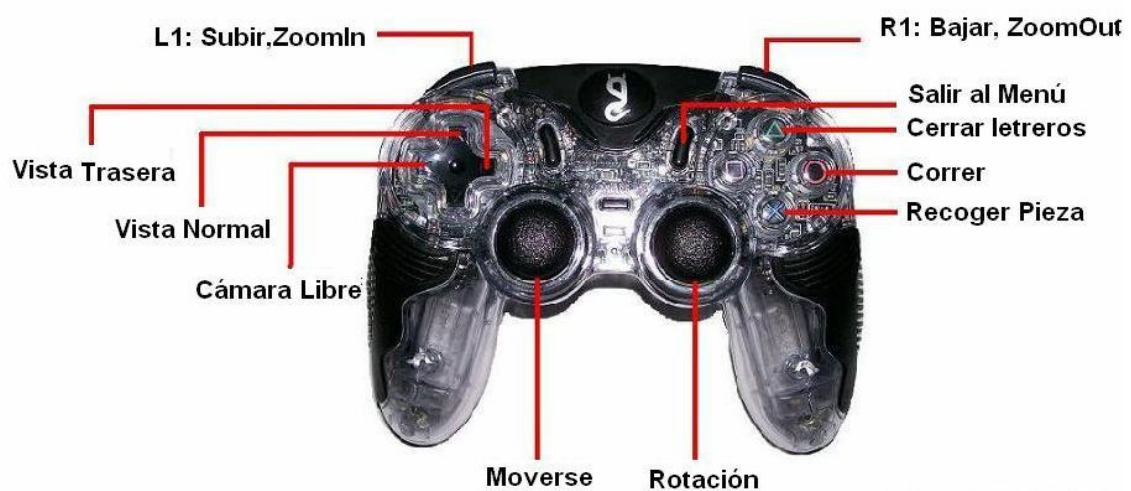
ILUSTRACIÓN VIII.2

Imagen del recorrido explicando sus partes

Controles Recorrido teclado



Controles Recorrido Joystick



Juego

Aquí podrás entrar en un emocionante juego de destreza, en el que tu mejor estrategia, será planear la mejor ruta, con el fin de obtener todos los objetos marcados en el mapa en un tiempo límite.

Después de seleccionar esta opción se te presentará la siguiente pantalla, en la cual debes seleccionar la dificultad del juego.



ILUSTRACIÓN VIII.3
Imagen de la dificultad

Ya que elegiste el nivel de dificultad se te presenta el objetivo de cada uno de los niveles con cada una de las siguientes pantallas.

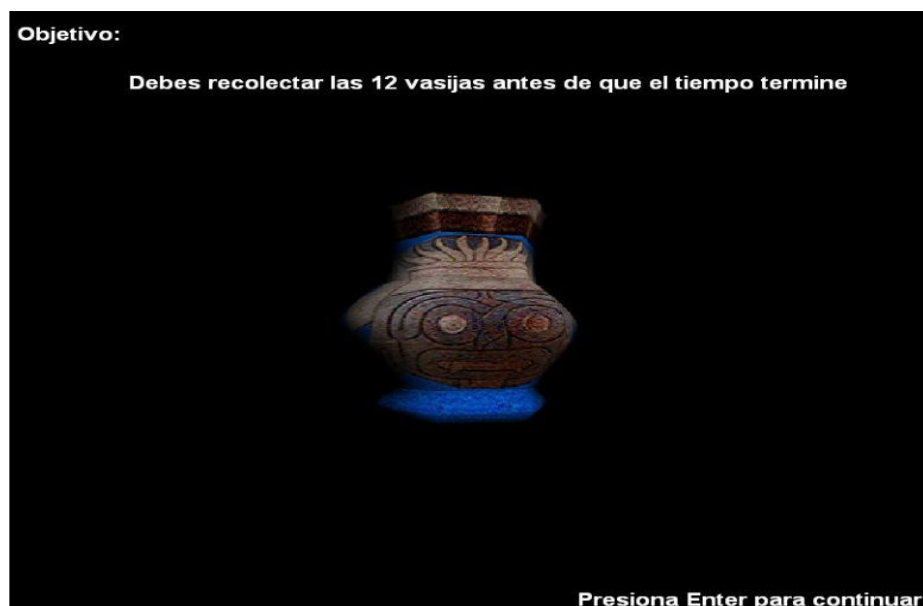


ILUSTRACIÓN VIII.4
Imagen de la pantalla avanzado

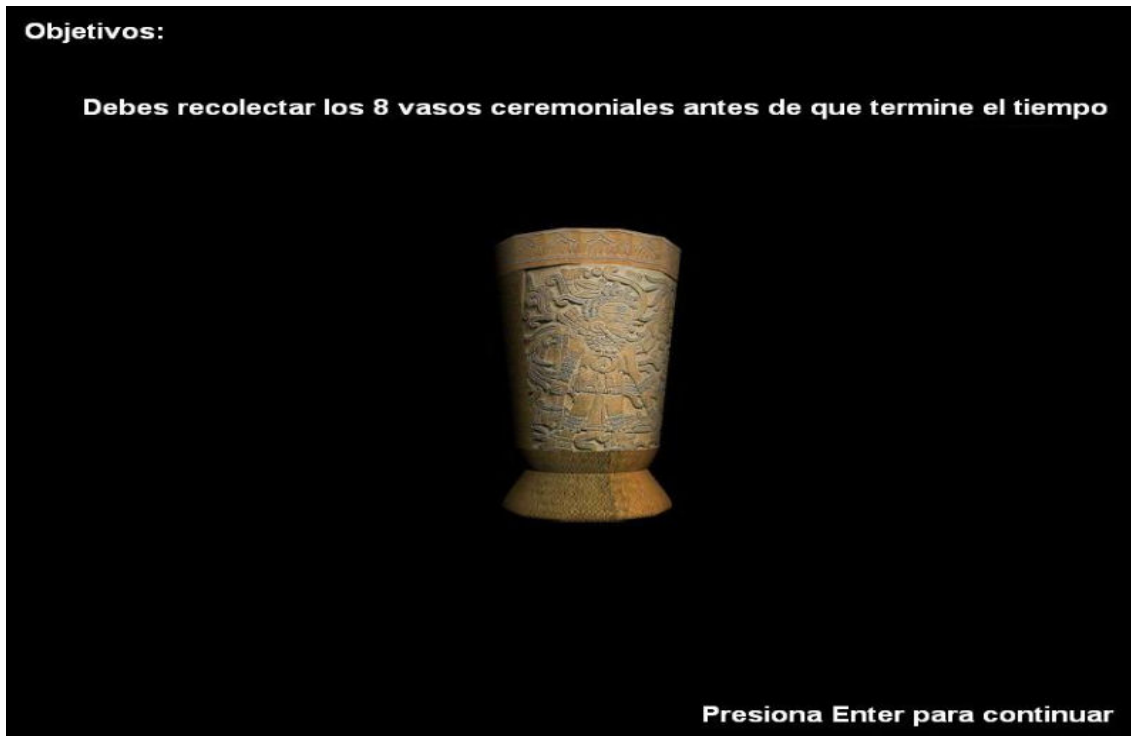


ILUSTRACIÓN VIII.5
Imagen de la pantalla intermedio

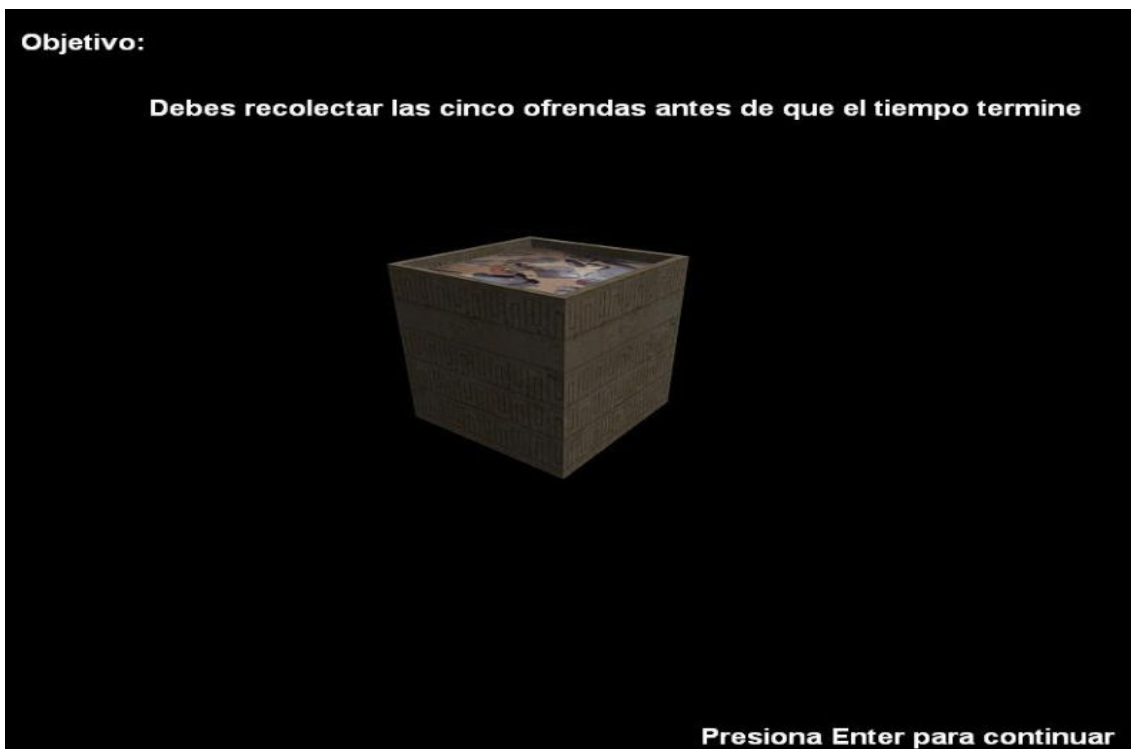


ILUSTRACIÓN VIII.6
Imagen de la pantalla fácil

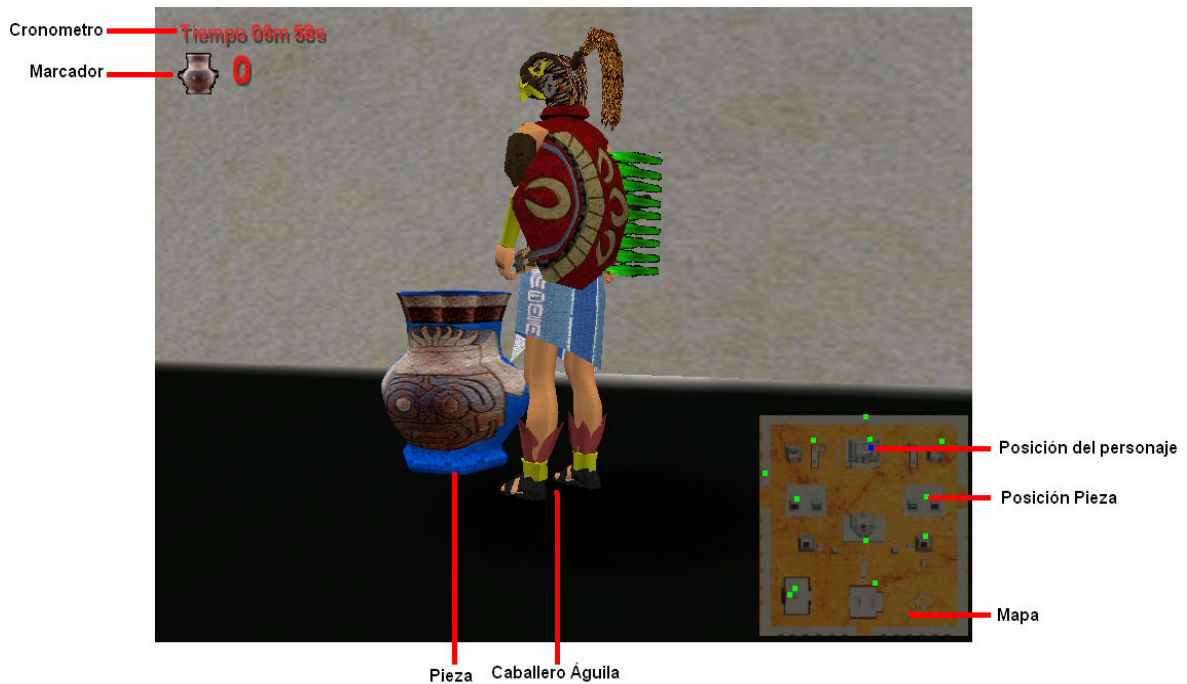


ILUSTRACIÓN VIII.7

Imagen de la pantalla del juego con cada una de sus funciones

Al terminar el tiempo y si no cumpliste con el objetivo, es decir recuperar todas las piezas, te encontraras con la pantalla Perdiste, en la que muestran las acciones para continuar o salir del juego.

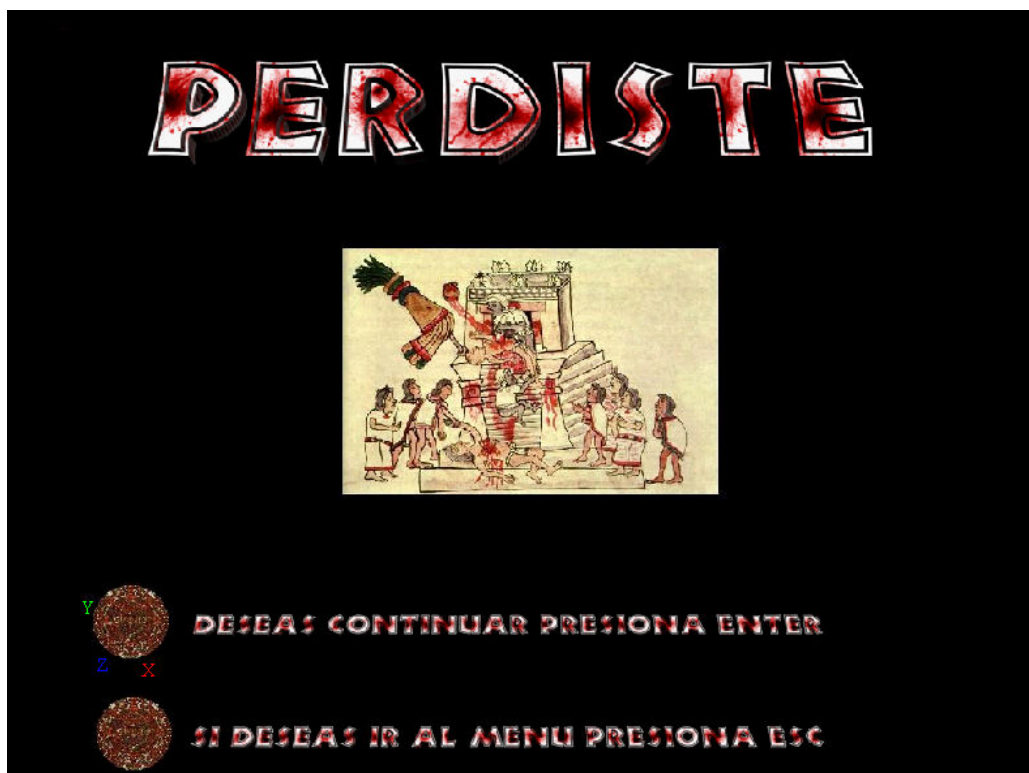


ILUSTRACIÓN VIII.8

Imagen perdiste

Si por el contrario logras obtener todas las piezas el resultado será la siguiente pantalla. Y para salir de ella simplemente se hace presionando los botones representados en pantalla.



ILUSTRACIÓN VIII.9
Imagen Ganaste

Controles Juego Teclado

Controles Juego



Controles Juego Joystick



Galería de Dioses

Aquí podrás ver las imágenes de los dioses más importantes de este recinto, simplemente con un click sobre el nombre de la deidad.



ILUSTRACIÓN VIII.10
Imagen galería de dioses

Galería de Sonidos

En esta sección podrás escuchar las melodías y los sonidos que utilizamos en la realización de este proyecto con los nombres originales de los artistas, así como sus títulos, para escucharlos simplemente selecciona con el indicador del mouse.

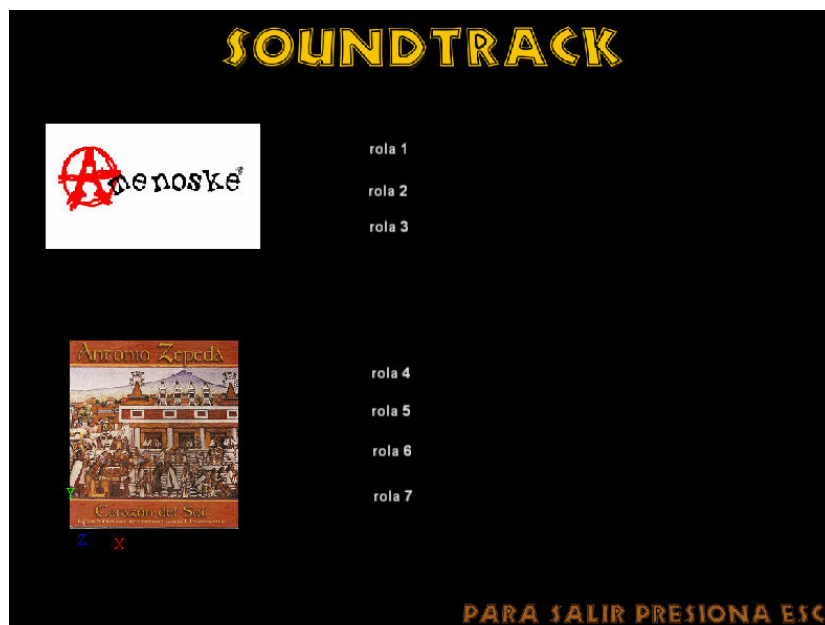


ILUSTRACIÓN VIII.11

Imagen galería de dioses

Sitios Web

Aquí se muestra una pantalla desde la cual se puede acceder a Sitios Web, con más información sobre la zona arqueológica, así como de los instrumentos utilizados para la realización de este proyecto, siempre y cuando cuenten con una conexión a Internet.



ILUSTRACIÓN VIII.12

Imagen Sitios Web

Controles

En esta pantalla se encuentra la explicación gráfica sobre los controles, para el recorrido virtual y para el juego, con joystick o teclado.



ILUSTRACIÓN VIII.13
Imagen controles

Créditos y Agradecimientos

Aquí simplemente encontraras un video con los créditos y agradecimientos de este trabajo de tesis.

Nota: Para salir de Cualquiera de las pantallas anteriores y regresar al menú, solo con presionar el botón indicado para cada pantalla.

ANEXO II

Glosario

Atach. Operación utilizada para formar un solo elemento a partir de la unión de varios.

Bitmap. Un bitmap es una imagen producida por una matriz fija de píxeles coloreados, como un mosaico. Se pueden usar los bitmaps como las texturas para los materiales.

Blue Print. Imágenes colocadas en las vistas frontal, lateral y superior para ayudar en la construcción de un moelo.

Booleano. Un objeto Booleano combina dos objetos realizando una operación Booleana sobre ellos. Un objeto Booleano es hecho de dos objetos interceptados. Los dos objetos originales son los operando (A y B) Y el objeto Booleano es el resultado de la operación.

Cámara. Es uno de los objetos básicos del entorno en 3D. La cámara se considera como la vista usuario en una escena en 3D. Sin una cámara, el usuario no puede ver el mundo en 3D.

Canal Alfa. Hace referencia a la información almacenada dentro de un mapa de bits cuyo objetivo es controlar la cantidad de transparencia por píxel. A los canales alfa se hace referencia utilizando su profundidad de bits, que describe cuantos niveles de transparencia son capaces de admitir.

Colisión. Un proceso que exige mucha memoria que comprueba si los modelos están o no colisionando entre si.

EI. Abreviatura de Entretenimiento Interactivo.

Editable Poly. Un polígono editable es un tipo de objeto deformable que polígonos triangulares. Los polígonos editables son útiles para crear objetos simples y mallas de control para MeshSmooth. Se pueden convertir NURBS y superficies patch a un polígono editable.

Elemento. Un elemento es uno de dos o más poligonales individuales (es decir, grupos de caras inmediatas) agrupados en un objeto más grande. Por ejemplo, si se liga una esfera a otra, se crea un objeto poligonal de dos esferas. Cada esfera es ahora un elemento del objeto. Cualquier función que se realice en ese objeto afectara a ambos elementos. Sin embargo, se pueden manipular los elementos independientemente.

Escalamiento. Una de las principales de transformaciones, se utiliza para aumentar o disminuir el tamaño de los modelos.

Formato de Archivo BMP. Formato de Microsoft Windows, tanto para 8 bits como para 24 bits. Admite compresión propia.

Formato de Archivo GIF. Formato gráfico desarrollado por Compuserve y destinado en un principio a imágenes de 8 bits (256 colores).

Formato de Archivo JPG. Formato utilizado principalmente para Internet por no ocupar mucho espacio en el disco duro del servidor, lo que permite cargarlas con mayor velocidad.

Formato de Archivo MAX. Es un formato de archivo nativo de 3D Studio Max, el cuál sirve para almacenar modelos tridimensionales, luces, cámaras, ayudantes, animaciones entre otros.

Formato de Archivo PNG. Portable Network Graphics, es un formato de archivo de imagen desarrollado para el uso de Internet. Con la característica principal que cuenta con 8 bits del canal alfa para poder asignarle un canal de transparencia a la respectiva imagen.

Gizmo. Modificador desplegado en los puertos de vista como una caja que inicialmente rodea al objeto seleccionado. Un gizmo actúa como un tipo de recipiente que transfiere la modificación hacia el objeto al cual está ligado.

Grupo. Es uno de los nodos principales de un entorno en 3D, sirve para conjuntar elementos del mismo o diferente tipo bajo el nombre asignado al conjunto o grupo.

Luz Ambiental. Se utiliza para aproximarse a un nivel general de la luz en un entorno. Concretamente, el uso de la luz ambiental es un intento de aproximarse a la cantidad de luz dentro de una escena.

Material. Es un conjunto de datos que se asignan a la superficie o caras de un objeto para que este aparente estar hecho de algún material en específico (piedra, madera, metal, agua, etc.) cuando es renderizado. Los materiales afectan el color de los objetos, su brillo y su opacidad. Un material estándar contiene una componente ambiental, difusa, y especular.

Material Compuesto. Los materiales compuestos permiten crear un material integrado por dos o más submateriales. El material compuesto Multi/subobjeto permite asignar diferentes materiales a subobjetos diferentes, al nivel del subobjeto de su geometría.

Mip Mapping. Proceso que crea varias versiones sucesivamente más pequeñas de una textura. Permite al render en 3D elegir una versión de la textura que tenga el tamaño adecuado, dado que el tamaño de la textura varía dependiendo de su ubicación en el entorno en 3D.

Modelo. Es uno de los elementos principales de un entorno en 3D. Los modelos se utilizan para representar los objetos en una escena.

Modificador. Cambia la estructura geométrica de un objeto, deformándolo de alguna manera. Los modificadores hacen los cambios en la geometría y permanecen en efecto hasta que se ajuste o anule el modificador.

Objeto. Son elementos tales como la geometría primitiva (cajas, esferas, etc), geometría más compleja (booleanos, toroides, tubos, etc.) y otros (cámaras, luces, etc.). Existen objetos renderizables (geometría) y no renderizables (luces, cámaras, y ayudantes).

Píxel. Un píxel (el elemento más pequeño de una imagen) es un solo punto en una imagen gráfica. Los monitores despliegan imágenes dividiendo la pantalla en miles (o millones) de píxeles, arreglados en filas y columnas.

Plug- In. Es una característica o funcionalidad proporcionada para un programa independiente o componente, es decir, extiende la funcionalidad de un programa. Estos pueden ser proporcionados por distintos medios, por ejemplo por diseñadores de software independientes.

Primitivas. Son los recursos básicos incorporados a programas de modelado 3D, estos son: cajas, cilindros, esferas, planos, etc.

Render. Es un proceso de representación de los entornos en 3D, donde se observan los modelos geométricos texturizados, con iluminación, animación, sombras, etc; vistos desde el punto de vista de la cámara.

Rotación. Es uno de los tres tipos de transformación. Es el proceso de girar los puntos de un elemento geométrico respecto a otro dentro del entorno 3D.

RV. Abreviatura de Realidad Virtual.

Shader. Es un elemento de los entornos en 3D que se utiliza para definir las cualidades de una superficie, como su color o su brillo

SubObject. El subobjeto es un subconjunto de la geometría de un objeto. Muchos objetos tienen varios tipos de subobjetos que se pueden trabajar independientemente. Por ejemplo, un subobjeto de un polígono editable son vértices, bordes, caras, polígonos, y elementos.

Textura. Elemento de los entornos 3D que se utiliza en combinación con un shader para definir las cualidades materiales de un modelo.

RECONSTRUCCIÓN VIRTUAL DE LA ZONA ARQUEOLÓGICA DEL TEMPLO MAYOR

Transformación. Término que se utiliza para hacer referencia al proceso que orienta, coloca y escala los elementos en el entorno en 3D.

Traslación. Es uno de los tres tipos de transformación. Es el proceso de posicionar los elementos dentro del entorno 3D.

UVW Mapping (Mapeo UVW). Es el proceso de aplicar un shader a un modelo. Los shaders se aplican a los modelos en el espacio en 3D relativo a estos. Para evitar la confusión entre las coordenadas de la aplicación de mapas y las coordenadas normales, se utilizan las letras UVW, aunque técnicamente se trata de las coordenadas XYZ. La mayoría de los mapas de materiales son planos 2D asignados a una superficie 3D. Por consiguiente, el sistema de coordenadas usado para describir la colocación y transformación de los mapas es diferente del los ejes X, Y, Z coordenadas usadas en el espacio 3D.

Bibliografía

LIBROS

- Ingeniería del Software Un enfoque práctico, Roger S. Presman, Mc Graw Hill, México 2002.
- Ingeniería de Software, Ian Sommerville, Addison Wesley Iberoamericana, México 1988.
- Geometría Analítica, Lehmann, Uteha, 1982
- 3D Studio Versión 5.0, Jim Lamers y Michael Tood Peterson, New Riders, 1998.
- Historia del nombre y de la fundación de México, Gutierre Tibon, Fondo de Cultura Económica, México, 1993
- Historia Universal Moderna y Contemporanea, Ida Appendini y Silvio Savala, Porrúa, 1986.
- Ser Digital, Nicolás Negroponte, Atlantida Océano, 1995.
- Miles de Millones, Carl Sagan, Sinequanon, 1998.
- Inteligencia Artificial, E. Rich, Colección Ciencia Informática, 1988 Barcelona.

TESIS

- Avendaño Martínez, Jose Carlos, Salgado Rodríguez, José Francisco, Valiente Gómez, Santiago Igor (asesor), “Reconstrucción de sitios arqueológicos: Teotihuacan”, Facultad de Ingeniería, UNAM, México, 2004.

SITIOS WEB**Tutoriales 3D-StudioMax y Virtools**

- <http://www.3dtotal.com/>
- <http://www.3dtotal.com/team/Tutorials>
- <http://fr.wikipedia.org/wiki/Virtools>
- <http://www.virtools.com/>
- <http://accad.osu.edu/~aprice/courses/BVE/SimpleSceneMayaTutorial.pdf>

i

Información histórica

- <http://www.conaculta.gob.mx/templomayor/>
- <http://members.fortunecity.es/kaildoc/tenochtitlan/cronologia.htm>
- http://es.wikipedia.org/wiki/Templo_Mayor
- <http://www.arqueomex.com/>
- <http://www.conaculta.gob.mx/templomayor/>

Información para la elaboración del trabajo escrito

- www.monografias.com/trabajos4/realvirtual/realvirtual.shtml
- www.activamente.com.mx/vrml/
- <http://www.monografias.com/trabajos/vr/vr.shtml>
- <http://www.iaa.upf.es/~msoler/docencia/rv/index.html>

Imágenes para el Software

- <http://telematica.cicese.mx/computo/super/cicese2000/realvirtual/>
- www.monografias.com/trabajos4/realvirtual/realvirtual.shtml
- www.activamente.com.mx/vrml/
- http://es.wikipedia.org/wiki/Templo_Mayor
- http://www.sep.gob.mx/wb2/sep/sep_572_ubicacion_del_edific
- Biblioteca de consulta encarta.
- <http://www.arqueomex.com/>
- <http://www.conaculta.gob.mx/templomayor/>
- <http://members.fortunecity.es/kaildoc/tenochtitlan/cronologia.htm>

REVISTAS

- Revista Arqueología Mexicana VOLUMEN I, #4, Tenochtitlan (octubre-noviembre, 1993)
- Revista Arqueología Mexicana VOLUMEN VI, #31, Templo mayor (mayo-junio, 1998)
- Revista Atomix #82 “Creanimax”
- Revista Atomix #66 “Motor Gráfico del Unreal 3”

OTRAS REFERENCIAS

- Guía Arqueológica del Templo Mayor para niños y jóvenes, Federico Navarrete, Ediciones Tecolote.
- Metal Gear Solid III, Snake Ester, User Manual, USA 2005.