



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE DISEÑO GRÁFICO

**“Estudio Comparativo de herramientas de software libre y propietario para
modelado 3D. Caso práctico Modelado de Rostros Humanos”**

TESIS DE GRADO

**Previa a la obtención del título de
LICENCIADA EN DISEÑO GRÁFICO**

Presentado por:

KARINA VANESSA JARAMILLO VALDEZ

RIOBAMBA – ECUADOR

2011

Un profundo agradecimiento a mis padres y mi esposo por ser un apoyo fundamental en mi vida personal así como profesional, y al Director y Miembro de mi tesis ya que gracias a ellos ésta pudo ser realizada.

Esta tesis la dedico a mi hijo y a mi esposo por ser la razón más grande de mi vida para avanzar y a mis padres por ser quienes me educaron e inculcaron en mí la importancia del estudio.

| NOMBRE | FIRMA | FECHA |
|--|-------|-------|
| Ing. Iván Menes DECANO FACULTAD DE INFORMÁTICA Y ELECTRÓNICA | _____ | _____ |
| Ing. Milton Espinoza DIRECTOR ESCUELA DE DISEÑO GRÁFICO | _____ | _____ |
| Lcdo. Ramiro Santos DIRECTOR DE TESIS | _____ | _____ |
| Doc. Julio Santillán MIEMBRO DE TESIS | _____ | _____ |
| Lcdo. Carlos Rodríguez DIRECTOR DEL CENTRO DE DOCUMENTACIÓN NOTA DE TESIS | _____ | _____ |

RESPONSABILIDAD DEL AUTOR

“Yo KARINA VANESSA JARAMILLO VALDEZ, soy la responsable de las ideas, doctrinas y resultados expuestos en esta Tesis, y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo”.

Karina Jaramillo

INDICE GENERAL

PORTADA

AGRADECIMIENTO

DEDICATORIA

INDICE DE ABREVIATURAS

INDICE GENERAL

INDICE DE FIGURAS

INTRODUCCION

CAPITULO I: SOFTWARE

| | |
|--|-----------|
| 1.1 Definición Software..... | 17 |
| 1.2 Clasificación Del Software..... | 19 |
| 1.2.1 Software de sistema..... | 19 |
| 1.2.2. Software de programación..... | 19 |
| 1.2.3 Software de aplicación..... | 20 |
| 1.2.4 Sistema operativo..... | 20 |
| 1.3 Proceso de creación del software..... | 21 |
| 1.4 Definición de software propietario..... | 24 |
| 1.4.1 Software propietario..... | 24 |
| 1.4.2. Objeciones al término “propietario”..... | 25 |
| 1.4.3. Software privativo..... | 25 |
| 1.4.4. Objeciones al término “privativo”..... | 26 |
| 1.4.5. Software no libre..... | 26 |

| | |
|--|-----------|
| 1.4.6. Objeciones al término “no libre”..... | 26 |
| 1.4.7. Software privado..... | 27 |
| 1.5 Software Libre..... | 27 |
| 1.5.1 Historia Software Libre..... | 27 |
| 1.5.2 Libertades Del Software Libre | 29 |
| 1.5.3 TIPOS DE LICENCIAS..... | 31 |
| 1.5.3.1 Licencias GPL..... | 31 |
| 1.5.3.2 Licencias AGPL..... | 32 |
| 1.5.3.3 Licencias estilo BSD..... | 32 |
| 1.5.3.4 Licencias estilo MPL y derivadas..... | 33 |
| 1.5.3.5 Copyleft..... | 33 |
| 1.6 ESTUDIO DE OBSERVACIÓN | 35 |
| 1.6.1 Uso software libre y propietario en la ciudad de Riobamba..... | 35 |
| 1.6.2 Uso software libre o propietario en el área de diseño gráfico..... | 37 |
| 1.6.3 Uso legal Software..... | 38 |
| 1.7 Decreto 1014..... | 39 |
| | |
| CAPÍTULO II: MODELADO EN 3D..... | 40 |
| | |
| 2.1 Definición Modelado 3d..... | 40 |
| 2.2 Historia del Modelado en 3d..... | 42 |
| 2.2.1 LOS AÑOS 60 | 42 |
| 2.2.2 LOS AÑOS 70 | 42 |
| 2.2.3 LOS AÑOS 80..... | 43 |
| 2.3 Fundamentos Básicos..... | 45 |

| | |
|--|-----------|
| 2.3.1 Estructuras Predefinidas..... | 46 |
| 2.3.1.1 Primitivas..... | 46 |
| 2.3.1.2 Primitivas Extendidas..... | 46 |
| 2.3.1.3 Box Modeling..... | 47 |
| 2.3.1.4 NURBS Modeling..... | 47 |
| 2.3.2 Operaciones Booleanas..... | 48 |
| 2.3.3 Lathe..... | 49 |
| 2.3.4 Extrude..... | 50 |
| 2.3.5 Splines..... | 50 |
| 2.3.6 Sistema de Particulas..... | 51 |
| 2.3.7 Modelo por texturas..... | 52 |

CAPITULO III: APLICACIONES INFORMÁTICAS DE MODELADO Y ANIMACIÓN 3D

| | |
|--|-----------|
| 3.1 Aplicaciones 3D..... | 53 |
| 3.2 Elección de software para la realización del caso práctico..... | 54 |

CAPITULO IV: SOFTWARE PROPIETARIO: 3D STUDIO MAX

| | |
|---|-----------|
| 4.1 Autodesk 3d Studio Max..... | 55 |
| 4.2 Conocimiento de 3D Max..... | 57 |
| 4.2.1 Barras de menús..... | 57 |
| 4.2.2 Barra de herramientas..... | 58 |
| 4.2.3 Panel de solapas..... | 58 |
| 4.2.3.1 Herramientas del panel de solapas..... | 59 |
| 4.3 Creación De Geometrías..... | 61 |

| | |
|--|-----------|
| 4.3.1 Primitivas estándar..... | 62 |
| 4.3.2 Las primitivas geométricas..... | 62 |
| 4.3.3 Primitivas extendidas..... | 62 |
| 4.4 Creación De Formas..... | 62 |
| 4.4.1 Convert To Editable Mesh..... | 63 |
| 4.4.2 Set Id..... | 63 |
| 4.4.3 Element..... | 64 |
| 4.5 Creación De Objetos De Composición..... | 65 |
| 4.5.1 MORFISMO..... | 65 |
| 4.5.2 BOOLEANO..... | 65 |
| 4.5.3 SOLEVADO o LOFT..... | 66 |
| 4.5.4 FOILAGE..... | 66 |
| 4.5.5 DOORS..... | 67 |
| 4.5.6 WINDOWS..... | 67 |
| 4.5.7 STAIRS..... | 67 |
| 4.6 MODIFICADORES..... | 68 |
| 4.7 EDITOR DE MATERIALES..... | 84 |
| 4.8 MATERIALES DE COMPOSICIÓN..... | 84 |
| 4.9 TIPOS DE MAPAS..... | 85 |

CAPITULO V: SOFTWARE LIBRE: BLENDER

| | |
|---|------------|
| 5.1 SOFTWARE LIBRE: BLENDER..... | 87 |
| 5.1.1 Interfaz..... | 88 |
| 5.1.2 Conceptos de la Interfaz de Blender..... | 89 |
| 5.1.3 El teclado y el ratón..... | 89 |
| 5.2 El sistema de ventanas..... | 91 |
| 5.2.1 La escena por defecto de Blender..... | 91 |
| 5.2.2 Elementos de la Interface..... | 91 |
| 5.2.3 Tipos de ventanas..... | 92 |
| 5.2.4 Contextos, Paneles y Botones | 93 |
| 5.3 Menú de la Ventana de Botones..... | 95 |
| 5.3.1 Panel con fichas..... | 95 |
| 5.4 Caja de Herramientas (ToolBox)..... | 97 |
| 5.5 Pantallas (Screens)..... | 98 |
| 5.6 Escenas..... | 98 |
| 5.7 Navegando en el Entorno 3D..... | 99 |
| 5.8 Renderizando..... | 103 |

CAPITULO VI: COMPARATIVA AUTODESK 3D MAX VS BLENDER

| | |
|--|------------|
| 6.1 Comparativa blender vs 3d max (referencia informática)..... | 107 |
| 6.2 Detalles técnicos..... | 109 |
| 6.3 Herramientas de Modelado..... | 109 |
| 6.4 Viable Poligono Límite | 111 |

| | |
|---|------------|
| 6.5 UV Unwrap..... | 112 |
| 6.6 Texturizado y Pintura..... | 114 |
| 6.7 Animación y Simulación..... | 116 |
| 6.8 Prestación..... | 118 |
| 6.9 Interfaz y personalización..... | 119 |
| 6.10 Formatos de archivo admitidos importar..... | 120 |
| 6.11 Formatos de archivo admitidos exportar..... | 122 |

CONCLUSIONES

RECOMENDACIONES

RESUMEN

SUMMERY

GLOSARIO

BIBLIOGRAFÍA

INDICE DE TABLAS

| | |
|--|-----|
| Tabla I. Tabla porcentajes uso de software..... | 35 |
| Tabla II. Cuadro gráfico porcentajes uso de software..... | 35 |
| Tabla III. Tabla gráfico porcentajes uso de software propietario y libre..... | 37 |
| Tabla IV. Cuadro gráfico porcentajes uso de software propietario y libre..... | 37 |
| Tabla V. Tabla gráfico porcentajes uso de software propietario con licencia..... | 38 |
| Tabla VI. Cuadro gráfico porcentajes uso de software propietario con licencia..... | 38 |
| Tabla VII. Tabla explicación aplicaciones informáticas de modelado y animación 3d..... | 53 |
| Tabla VIII. Comparativa blender vs 3d max (informática)..... | 106 |
| Tabla IX. Comparativa detalles técnicos..... | 109 |
| Tabla X. Tabla herramientas de modelado..... | 109 |
| Tabla XI. Herramientas de modelado..... | 110 |
| Tabla XII. Tabla Limite viable del polígono..... | 111 |
| Tabla XIII. Limite viable del polígono..... | 112 |
| Tabla XIV. Tabla UV Unwrap | 112 |
| Tabla XV. UVUnwrap..... | 113 |
| Tabla XVI. Tabla texturizado y pintura..... | 114 |
| Tabla XVII. Texturizado y pintura..... | 115 |
| Tabla XVIII. Tabla animación y simulación..... | 116 |
| Tabla XIX. Animación y simulación..... | 117 |
| Tabla XX. Prestación..... | 118 |
| Tabla XXI. Tabla interfaz y Personalización..... | 119 |
| Tabla XXII. Interfaz y Personalización..... | 120 |
| Tabla XXIII. Tabla Archivos permitidos para importar..... | 120 |

| | |
|---|-----|
| Tabla XXIV. Archivos permitidos para importar..... | 122 |
| Tabla XXV. Tabla archivos permitidos para exportar..... | 122 |
| Tabla XXVI. Archivos permitidos para exportar..... | 124 |
| Tabla XXVII. Cuadro comparativo herramientas modelado 3d..... | 125 |

INDICE FIGURAS

| | |
|--|----|
| Fig.II.1 Primitivas Extendidas | 46 |
| Fig.II.2. Ejemplo de modelado a través de una caja | 47 |
| Fig.II.3 Ejemplo de modelado con NURBS | 48 |
| Fig.II.4 Ejemplo de booleanos..... | 49 |
| Fig.II.5 Ejemplo de Lathe..... | 49 |
| Fig.II.6 Ejemplo de extracción | 50 |
| Fig.II.7 Ejemplo de splines | 51 |
| Fig.II.8 Ejemplo manejo sistema de partículas..... | 51 |
| Fig.II.9 Modelo de texturas | 52 |
| Fig IV.10 Interfaz Autodesk 3d Max | 57 |
| Fig IV.11 Barras de Menú Autodesk 3d Max | 58 |
| Fig IV.12 Barras de Herramientas Autodesk 3d Max | 58 |
| Fig IV.13 Barras de Herramientas 2 Autodesk 3d Max | 58 |
| Fig IV.14 Panel de solapas Autodesk 3d Max | 59 |
| Fig IV.15 Primitivas estándar | 62 |
| Fig IV.16 Convertir en malla editable | 63 |

| | |
|--|----|
| Fig IV.17 Definir ID | 64 |
| Fig IV.18 Subdivisión de elementos | 64 |
| Fig IV.19 Morfismo | 65 |
| Fig IV.20 Loft | 66 |
| Fig IV.21 Follaje | 66 |
| Fig IV.22 Affect region | 69 |
| Fig IV.23 Afilar | 69 |
| Fig IV.24 Bevel | 70 |
| Fig IV.25 Bevel Profile | 71 |
| Fig IV.26 Cáscara | 72 |
| Fig IV.27 Lattice | 72 |
| Fig IV.28 Bend | 73 |
| Fig IV.29 Displace | 73 |
| Fig IV.30 Fillet Chanfer | 74 |
| Fig IV.31 Escalar mapa | 75 |
| Fig IV.32 Escalar mapa espacio universal | 75 |
| Fig IV.33 Esfericar | 76 |
| Fig IV.34 Mapa UVW | 77 |
| Fig. IV. 35 Wave | 78 |
| Fig. IV. 36 Mirror | 80 |
| Fig. IV. 37 Symmetry | 80 |
| Fig. IV. 38 Mesh Smooth | 81 |
| Fig. IV. 39 Cap holes | 82 |
| Fig. IV. 40 Twist | 82 |
| Fig. IV. 41 Lathe | 83 |

| | |
|--|-----|
| Fig. V.42 Interfaz Blender..... | 88 |
| Fig. VI.43 Interfaz 3d Max con el modelado del rostro | 108 |
| Fig. VI.44 Interfaz Blender con el modelado del rostro | 108 |
| Fig. VI.45 Malla editable 3d Max | 125 |
| Fig. VI.46 Modelado rostro 3ds Max II..... | 128 |
| Fig. VI.47 Modelado rostro 3ds Max III..... | 128 |
| Fig. VI.48 Modelado rostro 3ds Max IV..... | 129 |
| Fig. VI.49 Modelado rostro 3ds Max V..... | 129 |
| Fig. VI.50 Modelado rostro 3ds Max VI..... | 130 |
| Fig. VI.51 Render Modelado rostro 3ds Max VII..... | 130 |
| Fig. VI.51 Modelado rostro Blender I..... | 131 |
| Fig. VI.52 Modelado rostro Blender II..... | 131 |
| Fig. VI.53 Modelado rostro Blender III..... | 132 |
| Fig. VI.54 Modelado rostro Malla editable | 132 |
| Fig. VI.58 Renderizado Blender..... | 133 |

INTRODUCCIÓN

El campo del Diseño Gráfico, es afectado por la situación del monopolio con respecto al software pagado en ésta área, creando un mercado cautivo alrededor de sus usuarios (diseñadores), por lo que están muy poco motivados al uso o aprendizaje de otras alternativas en programas para diseño gráfico.

Un gran porcentaje de profesionales del diseño gráfico y aún instituciones educativas, no tienen la capacidad económica para adquirir las licencias de software de diseño para su utilización como herramienta de trabajo o de aprendizaje, incurriendo en el uso ilegal del mismo (piratería), esto debido al alto costo de las licencias que llegan a ser prohibitivas para quien quiere iniciar su carrera independientemente, por lo que se ha realizado un estudio comparativo de herramientas de software libre y propietario, que faciliten el crear rostros humanos en 3d.

Dado así para encontrar una solución de software que permita un trabajo impecable sin ir contra el tema legal, y así encontrar más opciones de software que permita una apertura de pensamiento y la libertad de elección con respecto al material que uno necesite o desee ocupar dentro de su profesión.

Para encontrar un resultado útil se ha realizado una comparativa de acuerdo a varios parámetros y así se podrá definir si esta tesis permitirá encontrar una alternativa de software más para el manejo del modelado y animación en 3d.

CAPITULO I

SOFTWARE

1.2 Definición Software

El software es una producción inmaterial del cerebro humano y tal vez una de las estructuras más complicadas que la humanidad conoce. De hecho, los expertos en computación aún no entienden del todo cómo funciona, su comportamiento, sus paradojas y sus límites. Básicamente, el software es un plan de funcionamiento para un tipo especial de máquina, una máquina ``virtual" o ``abstracta". Una vez escrito mediante algún lenguaje de programación, el software se hace funcionar en ordenadores, que temporalmente *se convierten* en esa máquina para la que el programa

sirve de plan. El software permite poner en relación al ser humano y a la máquina y también a las máquinas entre sí. Sin ese conjunto de instrucciones programadas, los ordenadores serían objetos inertes, como cajas de zapatos, sin capacidad siquiera para mostrar algo en la pantalla.

Los ordenadores sólo procesan lenguaje binario, pero para las personas este no es un modo válido de comunicarse (salvo a nivel sináptico. Si bien en los tiempos heroicos de los primeros ordenadores no les quedaba otro remedio que hacerlo, los programadores hacen mucho que no escriben su código en lenguaje binario (denominado técnicamente "código-máquina"), pues es terriblemente tedioso, improductivo y muy sujeto a errores. Hace tiempo que los programadores escriben las instrucciones que ha de ejecutar el procesador de la máquina mediante lenguajes formales, llamados "de alto nivel", bastante cercanos al inglés, si bien con rígidas reglas sintácticas que lo asemejan a los lenguajes lógico-formales. Esto facilita enormemente la tarea de escribir programas pero, para que esas instrucciones sean comprensibles para el procesador, deben ser convertidas antes a código-máquina. Esa *conversión* se realiza cómodamente con programas especiales, llamados compiladores. A lo que escribe el programador se le denomina "código-fuente". Al resultado de la "conversión" (compilación) en lenguaje-máquina, se le denomina "código-objeto", "binarios" o "ficheros ejecutables". En principio, al usuario común sólo le importa este último nivel, los "binarios", pero conviene tener clara la distinción entre fuentes y binarios pues es clave para entender el empeño de los partidarios del software libre en disponer de las fuentes.

Pero el software libre es mucho más que el derecho de los programadores y de los hackers³ a disponer de las fuentes del código: significa también la libertad de copiar y redistribuir esos programas. Esos derechos, o su ausencia, condicionan a cualquiera que use un ordenador y han configurado la industria del software y de la informática tal y como la conocemos hoy día. También ha dado lugar a un movimiento social --el del software libre-- cuya historia reconstruiremos brevemente en las próximas líneas.

1.2 CLASIFICACIÓN DEL SOFTWARE

Si bien esta distinción es, en cierto modo, arbitraria, y a veces confusa, se puede clasificar al software de la siguiente forma:

1.2.1 Software de sistema: Es aquel que permite que el hardware funcione. Su objetivo es desvincular adecuadamente al programador de los detalles del computador en particular que se use, aislándolo especialmente del procesamiento referido a las características internas de: memoria, discos, puertos y dispositivos de comunicaciones, impresoras, pantallas, teclados, etc. El software de sistema le procura al usuario y programador adecuadas interfaces de alto nivel y utilidades de apoyo que permiten su mantenimiento. Incluye entre otros:

Sistemas operativos

Controladores de dispositivo

Herramientas de diagnóstico

Herramientas de Corrección y Optimización

Servidores

Utilidades

1.2.2. Software de programación: Es el conjunto de herramientas que permiten al programador desarrollar programas informáticos, usando diferentes alternativas y lenguajes de programación, de una manera práctica. Incluye entre otros:

Editores de texto

Compiladores

Intérpretes

Enlazadores

Depuradores

Entornos de Desarrollo Integrados (IDE): Agrupan las anteriores herramientas, usualmente en un entorno visual, de forma que el programador no necesite introducir múltiples comandos para compilar, interpretar, depurar, etc.. Habitualmente cuentan con una avanzada interfaz gráfica de usuario (GUI).

1.2.3 Software de aplicación: Aquel que permite a los usuarios llevar a cabo una o varias tareas específicas, en cualquier campo de actividad susceptible de ser automatizado o asistido, con especial énfasis en los negocios. Incluye entre otros:

Aplicaciones de control y automatización industrial

Aplicaciones ofimáticas

Software educativo

Software médico

Software de Cálculo Numérico

Software de Diseño Asistido (CAD)

Software de Control Numérico (CAM)

1.2.4 Sistema operativo

Un **sistema operativo** es un software de sistema, es decir, un conjunto de programas de computadora destinado a permitir una administración eficaz de sus recursos. Comienza a trabajar cuando se enciende el computador, y gestiona el hardware de la máquina desde los niveles más básicos, permitiendo también la interacción con el usuario.

Un sistema operativo se puede encontrar normalmente en la mayoría de los aparatos electrónicos que utilicen microprocesadores para funcionar, ya que gracias a éstos podemos entender la máquina

y que ésta cumpla con sus funciones (teléfonos móviles, reproductores de DVD, autoradios, computadoras, etc.).

Un sistema operativo desempeña 5 funciones básicas en la operación de un sistema informático: suministro de interfaz al usuario, administración de recursos, administración de archivos, administración de tareas y servicio de soporte y utilidades.

1.3 Proceso de creación del software

Se define como Proceso al conjunto ordenado de pasos a seguir para llegar a la solución de un problema u obtención de un producto, en este caso particular, para lograr la obtención de un producto software que resuelva un problema.

El proceso de creación de software puede llegar a ser muy complejo, dependiendo de su porte, características y criticidad del mismo. Por ejemplo la creación de un sistema operativo es una tarea que requiere proyecto, gestión, numerosos recursos y todo un equipo disciplinado de trabajo. En el otro extremo, si se trata de un sencillo programa (por ejemplo, la resolución de una ecuación de segundo orden), éste puede ser realizado por un solo programador (incluso aficionado) fácilmente. Es así que normalmente se dividen en tres categorías según su tamaño (líneas de código) o costo: de Pequeño, Mediano y Gran porte. Existen varias metodologías para estimarlo, una de las más populares es el sistema COCOMO que provee métodos y un software (programa) que calcula y provee una estimación de todos los costos de producción en un «proyecto software» (relación horas/hombre, costo monetario, cantidad de líneas fuente de acuerdo a lenguaje usado, etc.).

Considerando los de gran porte, es necesario realizar complejas tareas, tanto técnicas como de gerencia, una fuerte gestión y análisis diversos (entre otras cosas), por lo cual se ha desarrollado una ingeniería para su estudio y realización: es conocida como Ingeniería de Software.

En tanto que en los de mediano porte, pequeños equipos de trabajo (incluso un avezado analista-programador solitario) pueden realizar la tarea. Aunque, siempre en casos de mediano y gran porte (y a veces también en algunos de pequeño porte, según su complejidad), se deben seguir ciertas etapas que son necesarias para la construcción del software. Tales etapas, si bien deben existir, son flexibles en su forma de aplicación, de acuerdo a la metodología o Proceso de Desarrollo escogido y utilizado por el equipo de desarrollo o por el analista-programador solitario (si fuere el caso).

Los «procesos de desarrollo de software» poseen reglas preestablecidas, y deben ser aplicados en la creación del software de mediano y gran porte, ya que en caso contrario lo más seguro es que el proyecto o no logre concluir o termine sin cumplir los objetivos previstos, y con variedad de fallos inaceptables (fracasan, en pocas palabras). Entre tales «procesos» los hay ágiles o livianos (ejemplo XP), pesados y lentos (ejemplo RUP) y variantes intermedias; y normalmente se aplican de acuerdo al tipo, porte y tipología del software a desarrollar, a criterio del líder (si lo hay) del equipo de desarrollo. Algunos de esos procesos son Extreme Programming (XP), Rational Unified Process (RUP), Feature Driven Development (FDD), etc.

Cualquiera sea el «proceso» utilizado y aplicado al desarrollo del software (RUP, FDD, etc), y casi independientemente de él, siempre se debe aplicar un «modelo de ciclo de vida».

Se estima que, del total de proyectos software grandes emprendidos, un 28% fracasan, un 46% caen en severas modificaciones que lo retrasan y un 26% son totalmente exitosos.

Cuando un proyecto fracasa, rara vez es debido a fallas técnicas, la principal causa de fallos y fracasos es la falta de aplicación de una buena metodología o proceso de desarrollo. Entre otras, una fuerte tendencia, desde hace pocas décadas, es mejorar las metodologías o procesos de desarrollo, o crear nuevas y concientizar a los profesionales en su utilización adecuada. Normalmente los especialistas en el estudio y desarrollo de estas áreas (metodologías) y afines (tales como modelos y hasta la gestión misma de los proyectos) son los Ingenieros en Software, es su orientación. Los especialistas en cualquier otra área de desarrollo informático (analista, programador, Lic. en

Informática, Ingeniero en Informática, Ingeniero de Sistemas, etc.) normalmente aplican sus conocimientos especializados pero utilizando modelos, paradigmas y procesos ya elaborados.

Es común para el desarrollo de software de mediano porte que los equipos humanos involucrados apliquen sus propias metodologías, normalmente un híbrido de los procesos anteriores y a veces con criterios propios.

El proceso de desarrollo puede involucrar numerosas y variadas tareas , desde lo administrativo, pasando por lo técnico y hasta la gestión y el gerenciamiento. Pero casi rigurosamente siempre se cumplen ciertas etapas mínimas; las que se pueden resumir como sigue:

Captura, Especificación y Análisis de requisitos (ERS)

Diseño

Codificación

Pruebas (unitarias y de integración)

Instalación y paso a Producción

Mantenimiento

En las anteriores etapas pueden variar ligeramente sus nombres, o ser más globales, o contrariamente, ser más refinadas; por ejemplo indicar como una única fase (a los fines documentales e interpretativos) de «análisis y diseño»; o indicar como «implementación» lo que está dicho como «codificación»; pero en rigor, todas existen e incluyen, básicamente, las mismas tareas específicas.

En el apartado 4 del presente artículo se brindan mayores detalles de cada una de las listadas etapas.

1.4 DEFINICIÓN DE SOFTWARE PROPIETARIO.

El software no libre también es llamado software propietario, software privativo, software privado o software con propietario. Se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o que su código fuente no está disponible o el acceso a éste se encuentra restringido. En el software no libre una persona física o jurídica (por nombrar algunos: compañía, corporación, fundación) posee los derechos de autor sobre un software negando o no otorgando, al mismo tiempo, los derechos de usar el programa con cualquier propósito; de estudiar cómo funciona el programa y adaptarlo a las propias necesidades (donde el acceso al código fuente es una condición previa); de distribuir copias; o de mejorar el programa y hacer públicas las mejoras (para esto el acceso al código fuente es un requisito previo). De esta manera, un software sigue siendo no libre aún si el código fuente es hecho público, cuando se mantiene la reserva de derechos sobre el uso, modificación o distribución (por ejemplo, el programa de licencias shared source de Microsoft). No existe consenso sobre el término a utilizar para referirse al opuesto del software libre. Entre los términos más usados, en orden de frecuencia de uso (usando como fuente fidedigna una serie de consultas con el buscador Google en mayo de 2005) se encuentran:

1.4.1 Software propietario.

La expresión software propietario proviene del término en inglés “proprietary software”. En la lengua anglosajona, “proprietary” significa ((poseído o controlado privadamente)) (privately owned and controlled), que destaca la mantención de la reserva de derechos sobre el uso, modificación o redistribución del software. Inicialmente utilizado, pero con el inconveniente que la acepción proviene de una traducción literal del inglés, no correspondiendo su uso como adjetivo en

el español, de manera que puede ser considerado como un barbarismo. Sin embargo, todavía es el término preferido por cerca del 73 % de los sitios en Internet.

1.4.2. Objeciones al término “propietario”.

El término propietario en español resultaría inadecuado, pues significa que (tiene derecho de propiedad sobre una cosa), por lo que no podría calificarse de “propietario” al software, porque este no tiene propiedad sobre nada (es decir, no es dueño de nada) y, además, no podría serlo (porque es cosa y no persona). Asimismo, la expresión “software propietario” podría ser interpretada como “software sujeto a propiedad” (derechos o titularidad) y su opuesto, el software libre, está sujeto al derecho de autor.

1.4.3. Software privativo.

La expresión software privativo fue comenzada a ser utilizada por Richard Stallman, desde el año 2003, en sus conferencias sobre software libre, pues sería más adecuada que “software propietario” para definir en español al software que no es libre (“proprietary software” en inglés). Se realizó este cambio idiomático, básicamente por las siguientes razones: El término “privativo” ~~signi~~ que causa privación o restricción de derechos o libertades, es justamente lo que se pretende describir con el (privación a los usuarios de sus libertades en relación al software). Su uso ha ido en aumento constante y actualmente se puede constatar que cerca de 20 % de los sitios en Internet lo utilizan.

1.4.4. Objeciones al término “privativo”.

Este término haría referencia a una propiedad diferente a la de ser un software no libre, ya que la principal acepción de “privativo” es que (causa privación o la ¹⁵En este sentido, el contrario de “software privativo” sería “software no privativo”, es decir, sin limitaciones o como se le conoce: de dominio público. Así, software libre y su opuesto serían ambos conceptos diferentes a los sugeridos por “software privativo”. Asimismo, este término sería aplicable, de modo figurativo, al software cuyo coste es realmente elevado, no pudiendo ser soportado por un particular, caso muy común en el software diseñado para realizar labores muy específicas, con independencia a si este es libre o no.

1.4.5. Software no libre.

La expresión software no libre o software no-libre (en inglés “non-free software”), actualmente se utiliza en 5,5 % de los sitios Internet encontrados.

1.4.6. Objeciones al término “no libre”.

El término no libre es usado por la Free Software Foundation “FSF” (Fundación para el Software Libre), aunque para agrupar todo el software que no es libre, es decir, incluye al llamado en inglés “software semilibre” y al “software propietario”.

1.4.7. Software privado.

La expresión software privado es usada por la relación entre los conceptos de (tener) propietario y ser privado. Su utilización es poco difundida, cerca del 1 % de los sitios Internet utilizan este término.

1.5 SOFTWARE LIBRE

'*Software libre*' no significa '*no comercial*'. Un programa libre debe estar disponible para uso comercial, desarrollo comercial y distribución comercial. El desarrollo comercial del software libre ha dejado de ser inusual; el software comercial libre es muy importante.

Pero el software libre sin `copyleft' también existe. Creemos que hay razones importantes por las que es mejor usar 'copyleft', pero si tus programas son software libre sin ser 'copyleft', los podemos utilizar de todos modos.

Cuando se habla de software libre, es mejor evitar términos como: `regalar' o `gratis', porque esos términos implican que lo importante es el precio, y no la libertad.

1.5.1 HISTORIA SOFTWARE LIBRE

Entre los años 1960 y 1970, el software no era considerado un producto sino un añadido que los vendedores de las grandes computadoras de la época (las *mainframes*) aportaban a sus clientes para que éstos pudieran usarlos. En dicha cultura, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros. Este comportamiento era particularmente habitual en algunos de los mayores grupos de usuarios de la época, como DECUS (grupo de usuarios de computadoras DEC). A finales de la década de 1970, las

compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia.

En 1971, cuando la informática todavía no había sufrido su gran boom, las personas que hacían uso de ella, en ámbitos universitarios y empresariales, creaban y compartían el software sin ningún tipo de restricciones.

Con la llegada de los años 1980 la situación empezó a cambiar. Las computadoras más modernas comenzaban a utilizar sistemas operativos privativos, forzando a los usuarios a aceptar condiciones restrictivas que impedían realizar modificaciones a dicho software.

En caso de que algún usuario o programador encontrase algún error en la aplicación, lo único que podía hacer era darlo a conocer a la empresa desarrolladora para que ésta lo solucionara. Aunque el programador estuviese capacitado para solucionar el problema y lo deseara hacer sin pedir nada a cambio, el contrato le impedía que modificase el software.

El mismo Richard Stallman cuenta que por aquellos años, en el laboratorio donde trabajaba, habían recibido una impresora donada por una empresa externa. El dispositivo, que era utilizado en red por todos los trabajadores, parecía no funcionar a la perfección, dado que cada cierto tiempo el papel se atascaba. Como agravante, no se generaba ningún aviso que se enviase por red e informase a los usuarios de la situación.

La pérdida de tiempo era constante, ya que en ocasiones, los trabajadores enviaban por red sus trabajos a imprimir y al ir a buscarlos se encontraban la impresora atascada y una cola enorme de trabajos pendientes. Richard Stallman decidió arreglar el problema, e implementar el envío de un aviso por red cuando la impresora se bloqueara. Para ello necesitaba tener acceso al código fuente de los controladores de la impresora. Pidió a la empresa propietaria de la impresora lo que necesitaba, comentando, sin pedir nada a cambio, qué era lo que pretendía realizar. La empresa se negó a entregarle el código fuente. En ese preciso instante, Stallman se vio en una encrucijada: debía elegir entre aceptar el nuevo software propietario firmando acuerdos de no revelación y

acabar desarrollando más software propietario con licencias restrictivas, que a su vez deberían ser más adelante aceptadas por sus propios colegas.

Con este antecedente, en 1984, Richard Stallman comenzó a trabajar en el proyecto GNU, y un año más tarde fundó la Free Software Foundation (FSF). Stallman introdujo la definición de software libre y el concepto de "*copyleft*", que desarrolló para otorgar libertad a los usuarios y para restringir las posibilidades de apropiación del software.

1.5.2 LIBERTADES DEL SOFTWARE LIBRE

Software Libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

La libertad de usar el programa, con cualquier propósito (libertad 0).

La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.

La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2).

La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.

Un programa es software libre si los usuarios tienen todas estas libertades. Así pues, deberías tener la libertad de distribuir copias, sea con o sin modificaciones, sea gratis o cobrando una cantidad por la distribución, a cualquiera y a cualquier lugar. El ser libre de hacer esto significa (entre otras cosas) que no tienes que pedir o pagar permisos.

También deberías tener la libertad de hacer modificaciones y utilizarlas de manera privada en tu trabajo u ocio, sin ni siquiera tener que anunciar que dichas modificaciones existen. Si publicas tus cambios, no tienes por qué avisar a nadie en particular, ni de ninguna manera en particular.

La libertad para usar un programa significa la libertad para cualquier persona u organización de usarlo en cualquier tipo de sistema informático, para cualquier clase de trabajo, y sin tener obligación de comunicárselo al desarrollador o a alguna otra entidad específica.

La libertad de distribuir copias debe incluir tanto las formas binarias o ejecutables del programa como su código fuente, sean versiones modificadas o sin modificar (distribuir programas de modo ejecutable es necesario para que los sistemas operativos libres sean fáciles de instalar). Está bien si no hay manera de producir un binario o ejecutable de un programa concreto (ya que algunos lenguajes no tienen esta capacidad), pero debes tener la libertad de distribuir estos formatos si encontraras o desarrollaras la manera de crearlos.

Para que las libertades de hacer modificaciones y de publicar versiones mejoradas tengan sentido, debes tener acceso al código fuente del programa. Por lo tanto, la posibilidad de acceder al código fuente es una condición necesaria para el software libre.

Para que estas libertades sean reales, deben ser irrevocables mientras no hagas nada incorrecto; si el desarrollador del software tiene el poder de revocar la licencia aunque no le hayas dado motivos, el software no es libre.

Son aceptables, sin embargo, ciertos tipos de reglas sobre la manera de distribuir software libre, mientras no entren en conflicto con las libertades centrales. Por ejemplo, *copyleft* es la regla que implica que, cuando se redistribuya el programa, no se pueden agregar restricciones para denegar a otras personas las libertades centrales. Esta regla no entra en conflicto con las libertades centrales, sino que más bien las protege.

1.5.3 TIPOS DE LICENCIAS

Una licencia es aquella autorización formal con carácter contractual que un autor de un software da a un interesado para ejercer "actos de explotación legales". Pueden existir tantas licencias como acuerdos concretos se den entre el autor y el licenciatarario. Desde el punto de vista del software libre, existen distintas variantes del concepto o grupos de licencias:

1.5.3.1 Licencias GPL

Una de las más utilizadas es la *Licencia Pública General de GNU* (GNU GPL). El autor conserva los derechos de autor (copyright), y permite la redistribución y modificación bajo términos diseñados para asegurarse de que todas las versiones modificadas del software permanecen bajo los términos más restrictivos de la propia GNU GPL. Esto hace que sea imposible crear un producto con partes no licenciadas GPL: el conjunto tiene que ser GPL.

Es decir, la licencia GNU GPL posibilita la modificación y redistribución del software, pero únicamente bajo esa misma licencia. Y añade que si se reutiliza en un mismo programa código "A" licenciado bajo licencia GNU GPL y código "B" licenciado bajo otro tipo de licencia libre, el código final "C", independientemente de la cantidad y calidad de cada uno de los códigos "A" y "B", debe estar bajo la licencia GNU GPL.

En la práctica esto hace que las licencias de software libre se dividan en dos grandes grupos, aquellas que pueden ser mezcladas con código licenciado bajo GNU GPL (y que inevitablemente desaparecerán en el proceso, al ser el código resultante licenciado bajo GNU GPL) y las que no lo permiten al incluir mayores u otros requisitos que no contemplan ni admiten la GNU GPL y que por lo tanto no pueden ser enlazadas ni mezcladas con código gobernado por la licencia GNU GPL.

En el sitio web oficial de GNU hay una lista de licencias que cumplen las condiciones impuestas por la GNU GPL y otras que no.

Aproximadamente el 60% del software licenciado como software libre emplea una licencia GPL.

1.5.3.2 Licencias AGPL

La Licencia Pública General de Affero (en inglés Affero General Public License, también Affero GPL o AGPL) es una licencia copyleft derivada de la Licencia Pública General de GNU diseñada específicamente para asegurar la cooperación con la comunidad en el caso de software que corra en servidores de red.

La Affero GPL es íntegramente una GNU GPL con una cláusula nueva que añade la obligación de distribuir el software si éste se ejecuta para ofrecer servicios a través de una red de ordenadores.

La Free Software Foundation recomienda que el uso de la GNU AGPLv3 sea considerado para cualquier software que usualmente corra sobre una red.

1.5.3.3 Licencias estilo BSD

Llamadas así porque se utilizan en gran cantidad de software distribuido junto a los sistemas operativos BSD. El autor, bajo tales licencias, mantiene la protección de copyright únicamente para la renuncia de garantía y para requerir la adecuada atribución de la autoría en trabajos derivados, pero permite la libre redistribución y modificación, incluso si dichos trabajos tienen propietario. Son muy permisivas, tanto que son fácilmente absorbidas al ser mezcladas con la licencia GNU GPL con quienes son compatibles. Puede argumentarse que esta licencia asegura “verdadero” software libre, en el sentido que el usuario tiene libertad ilimitada con respecto al software, y que

puede decidir incluso redistribuirlo como no libre. Otras opiniones están orientadas a destacar que este tipo de licencia no contribuye al desarrollo de más software libre (normalmente utilizando la siguiente analogía: "una licencia BSD es más libre que una GPL si y sólo si se opina también que un país que permita la esclavitud es más libre que otro que no la permite").

1.5.3.4 Licencias estilo MPL y derivadas

Esta licencia es de Software Libre y tiene un gran valor porque fue el instrumento que empleó Netscape Communications Corp. para liberar su Netscape Communicator 4.0 y empezar ese proyecto tan importante para el mundo del Software Libre: Mozilla. Se utilizan en gran cantidad de productos de software libre de uso cotidiano en todo tipo de sistemas operativos. La MPL es Software Libre y promueve eficazmente la colaboración evitando el efecto "viral" de la GPL (si usas código licenciado GPL, tu desarrollo final tiene que estar licenciado GPL). Desde un punto de vista del desarrollador la GPL presenta un inconveniente en este punto, y lamentablemente mucha gente se cierra en banda ante el uso de dicho código. No obstante la MPL no es tan excesivamente permisiva como las licencias tipo BSD. Estas licencias son denominadas de copyleft débil. La NPL (luego la MPL) fue la primera licencia nueva después de muchos años, que se encargaba de algunos puntos que no fueron tenidos en cuenta por las licencias BSD y GNU. En el espectro de las licencias de software libre se la puede considerar adyacente a la licencia estilo BSD, pero perfeccionada.

1.5.3.5 Copyleft

Hay que hacer constar que el titular de los derechos de autor (copyright) de un software bajo licencia copyleft puede también realizar una versión modificada bajo su copyright original, y

venderla bajo cualquier licencia que desee, además de distribuir la versión original como software libre. Esta técnica ha sido usada como un modelo de negocio por una serie de empresas que realizan software libre (por ejemplo MySQL); esta práctica *no* restringe ninguno de los derechos otorgados a los usuarios de la versión copyleft. También podría retirar todas las licencias de software libre anteriormente otorgadas, pero esto obligaría a una indemnización a los titulares de las licencias en uso. En España, toda obra derivada está tan protegida como una original, siempre que la obra derivada parta de una autorización contractual con el autor. En el caso genérico de que el autor retire las licencias "copyleft", no afectaría de ningún modo a los productos derivados anteriores a esa retirada, ya que no tiene efecto retroactivo. En términos legales, el autor no tiene derecho a retirar el permiso de una licencia en vigencia. Si así sucediera, el conflicto entre las partes se resolvería en un pleito convencional.

1.6 ESTUDIO DE OBSERVACIÓN

1.6.1 Uso software libre y propietario en la ciudad de Riobamba

En la ciudad de Riobamba fue realizado un estudio a base de la técnica de observación con respecto al uso de software libre en los trabajos de diseño gráfico y los datos encontrados son:

Porcentajes de uso de software libre o propietario en la ciudad de Riobamba tomando como universo un grupo de 20 lugares que realizan trabajos de diseño.

| | Software Propietario | Software Libre |
|----------------------|----------------------|----------------|
| Software Propietario | 100% | 0% |
| Software Libre | 0% | 100% |

Tabla I. Tabla porcentajes uso de software

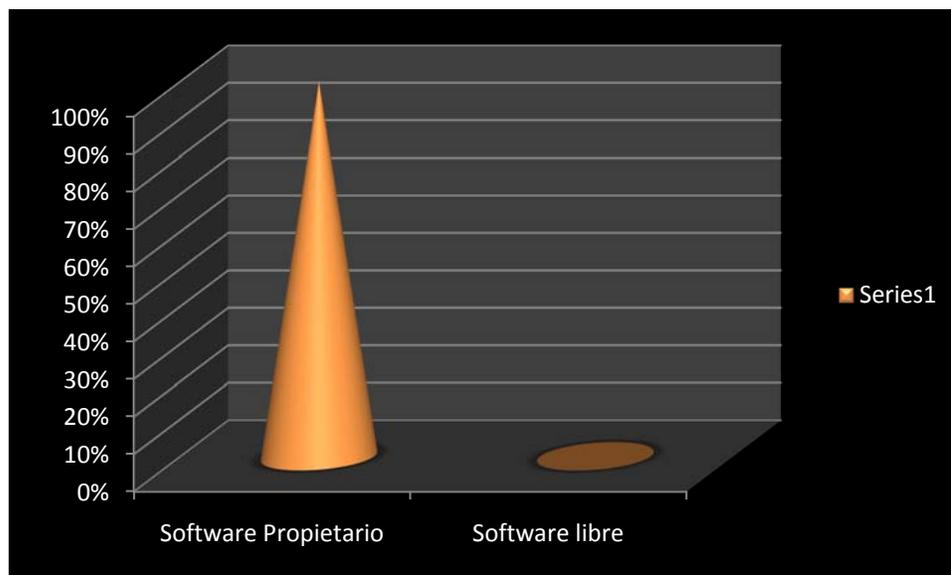


Tabla II. Cuadro gráfico porcentajes uso de software

Dando como resultado un desconocimiento total con respecto al software libre para diseño gráfico, a los lugares a los cuales se los observo fueron:

- Imprentas
- Estudios gráficos
- Boutiques creativas

Bajo consultas con respecto a la educación dada a los jóvenes en el tema informático se encontró que en un solo colegio de la ciudad de Riobamba, hay solo una institución educativa que enseña sobre software libre (linux) a sus alumnos de segundo y tercer año de bachillerado.

1.6.2 Uso software libre o propietario en el área de diseño gráfico

Con respecto al uso de software libre en diseño gráfico se encuentra el mismo panorama.

| | Software |
|----------------------|----------|
| Software Propietario | Libre |
| 100% | 0% |

Tabla III. Tablas gráfico porcentajes uso de software propietario y libre

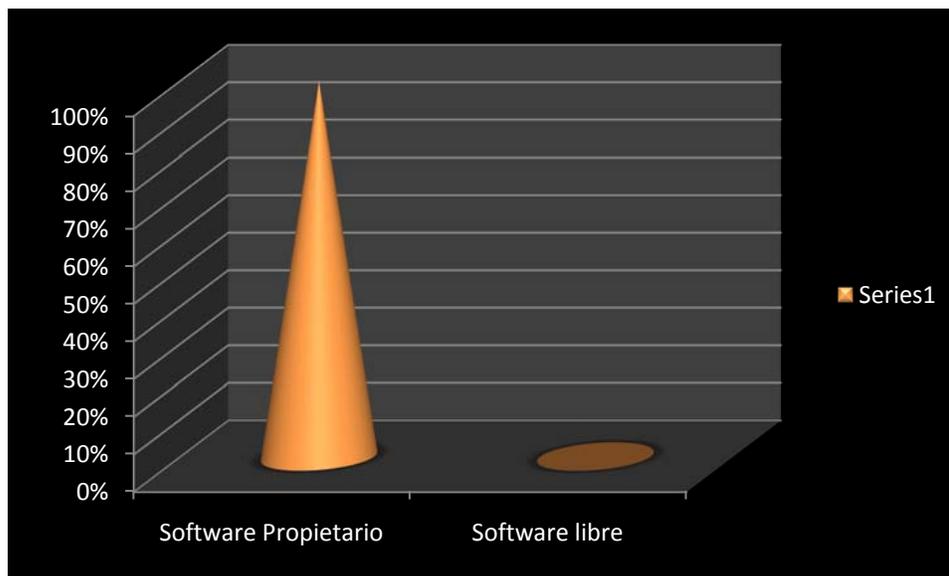


Tabla IV. Cuadro gráfico porcentajes uso de software propietario y libre

1.6.3 Uso legal Software libre

Con respecto al uso legal del software propietario en estos establecimientos se encuentra estos porcentajes.

| Licencia pagada | Uso ilegal |
|-----------------|------------|
| 3% | 97% |

Tabla V. Cuadro gráfico porcentajes uso de software propietario con licencia

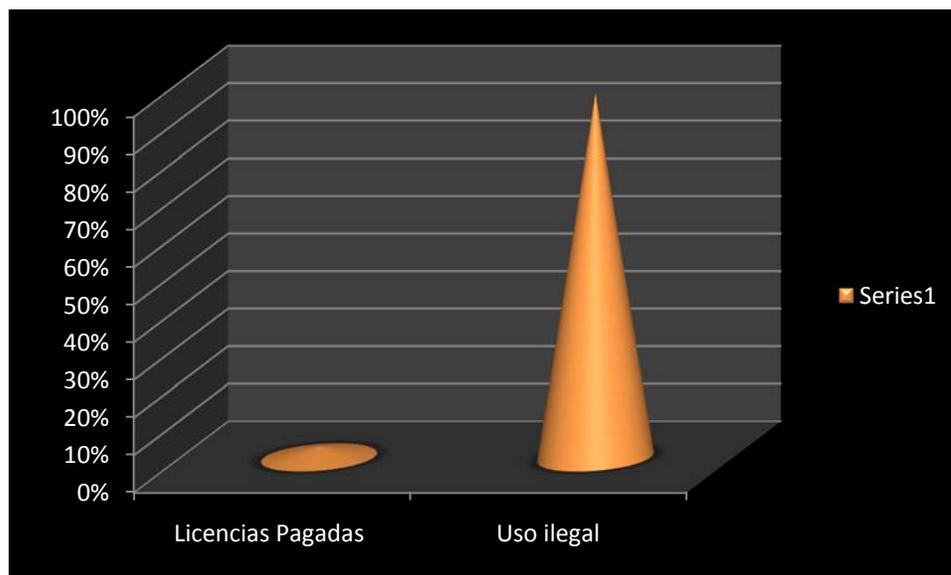


Tabla VI. Cuadro gráfico porcentajes uso de software propietario con licencia

Encontrando así, que para el medio en el que desarrollan estas empresas es difícil el pago de licencias debido a sus altos costos.

1.7 Decreto 1014

En la Actualidad con el Gobierno del Economista Rafael Correa se realizó decreto 1014 (incluido anexos) que en globa todo el tema de migrar a software libre obviamente en el tema informático, si en este hubiese programas que puedan suplantar al software propietario que ha sido usado por muchos años, siendo así que en las empresas públicas tiene la obligación de cambiar su software para cumplir con este decreto.

Por lo que en la empresas públicas buscan alternativas para poder cumplir con este decreto.

CAPÍTULO II

MODELADO EN 3D

2.1 Definición Modelado 3d

En computación, un modelo en 3D es un "mundo conceptual en tres dimensiones".

Un modelo 3D puede "verse" de dos formas distintas. Desde un punto de vista técnico, es un grupo de fórmulas matemáticas que describen un "mundo" en tres dimensiones.

Desde un punto de vista visual, valga la redundancia, un modelo en 3D es un representación

esquemática visible a través de un conjunto de objetos, elementos y propiedades que, una vez procesados (renderización), se convertirán en una imagen en 3D o una animación 3d.

Por lo general, el modelo visual suele ser el modelo 3d que las diseñadores manejan, dejando las fórmulas a procesos computacionales. Esto es así, porque lo que el modelo en 3d visual representa se acerca más a la imagen en 3D final que se mostrará al renderizarse.

Existen aplicaciones de modelado en 3D, que permiten una fácil creación y modificación de objetos en tres dimensiones. Estas herramientas suelen tener objetos básicos poligonales (esferas, triángulos, cuadrados, etc.) para ir armando el modelo. Además suelen contar con herramientas para la generación de efectos de iluminación, texturizado, animación, transparencias, etc. Algunas aplicaciones de modelado son 3D Studio Max, Alias, Blender, Cheetah3D, Cinema 4D, Generative Components, Houdini, LightWave, Maya, MilkShape 3D, modo Rhinoceros 3D, Softimage|XSI, trueSpace, ZBrush, etc.

El modelo en 3D describe un conjunto de características que, en conjunto, resultarán en una imagen en 3D. Este conjunto de características suele estar formado por objetos poligonales, tonalidades, texturas, sombras, reflejos, transparencias, translucidez, refracciones, iluminación (directa, indirecta y global), profundidad de campo, desenfocos por movimiento, ambiente, punto de vista, etc.

2.2 Historia del Modelado en 3d

2.2.1 LOS AÑOS 60

El primer sistema CAD fue desarrollado durante los años 60. El Dr. Hanratty es mundialmente conocido como 'el Padre del CAD/CAM' por su revolucionaria contribución en los campos de diseño y fabricación asistida por ordenador.

El Dr. Hanratty fue co-diseñador de DAC (Diseño Automatizado por Computadora) en la compañía General Motors. Este producto fue el primer sistema gráfico interactivo de fabricación. En 1971 el Dr. Hanratty fundó la compañía MCS (Manufacturing and Consulting Company). MCS era proveedor del código a compañías tales como Mc Donnell Douglas (Unigraphics), Computervision (CADDs),etc.

Analizando los sistemas CAD/CAM actuales se puede observar que el 70% de ellos encuentran sus raíces en el código original de MCS.

2.2.2 LOS AÑOS 70

Los años 70 se enfocaron en automatizar el diseño 2D. Se centraba en poner líneas y círculos en pantalla del ordenador y automatizar este proceso usando una Interface de programación de MACROS. En este período los operadores de CAD tenían que saber dibujar y además ser buenos programadores.

En estos años apareció el sistema Unigraphics. Éste fue un sistema CAD/CAM 3D desde sus versiones más tempranas, pero no alcanzó una precisión aceptable hasta 1979.

2.2.3 LOS AÑOS 80

En el principio de los años 80 aparecieron un conjunto de sistemas de diseño asistido por ordenador (CAM) para automatizar el proceso de mecanizado creando cintas digitales para manejar máquinas herramienta. También en esta época aparecieron las primeras herramientas de CAE para los análisis de diseño complejos.

En 1981 Unigraphics creó el primer sistema de modelado sólido: Uni-Solids. Se vendía como un producto separado.

Se produjeron mayores adelantos tecnológicos, incluyendo sistemas de modelado paramétrico, permitiendo al software CAD/CAM/CAE volverse una parte más integrada en el proceso de diseño de producto.

Uno de los avances más espectaculares que aparecieron en esta época fue la aparición del ordenador personal, y además la aparición de Autodesk. John Walker, programador y visionario, fundó Autodesk en 1982. Él y su equipo de programadores empezaron a desarrollar cinco aplicaciones de automatización. Hicieron esto pensando que una de estas cinco triunfaría. Entonces desarrollarían la que resultara triunfadora. Este producto resultó ser AutoCAD, un pequeño programa de gráficos 2D. Este programa permitía a los vendedores de Hardware de gráficos presumir de su Hardware de vídeo de alto-rendimiento. Esto resultó una ventaja muy importante para Autodesk porque tenía demostraciones de su programa AutoCAD sin necesidad de pagar. En 1987 Autodesk había vendido 100.000 copias de AutoCAD.

CADKEY llegó al mercado, pero se enfocó en el 3D. Por aquel tiempo trabajar en 3D era muy difícil en un PC. No fue hasta una versión posterior que CADKEY fue capaz de convertirse en un serio competidor de 3D.

El paseo hasta el diseño sólido en 3D sufrió un empujón muy importante en 1988 con las primeras versiones de Pro/ENGINEER. Pro/ENGINEER resultó ser un modelador muy robusto. En 1988 Unigraphics adquirió Shape Data Ltd. (diseñadores de Romulus, Romulus-D y Parasolid) y se comenzó a comercializar Parasolid como un producto autosuficiente. Parasolid es un Kernel de diseño sólido extremadamente poderoso que permite unir límites de superficies como un sólido.

A finales de los 80 Alvar Green tomó el mando de AutoDesk y en 1989 reunió a sus programadores para desarrollar nuevos productos.

2.3 FUNDAMENTOS BÁSICOS

El 3D es una mera representación de coordenadas, que conforman estructuras envueltas por una textura. Figúrenselo, como estructuras de alambre, recubiertas de papel de colores. El truco, es realizar la malla de manera simple, para luego crear el material por el cual le daremos sus características tales como metal, barro, agua, lo que sea.

Por tanto, primero se deben construir un modelo, para ello hay técnicas de modelo comunes, en las cuales se encuentran:

1. Estructuras Predefinidas
2. Box Modeling
3. NURBS Modeling
4. Operaciones booleanas
5. Extrude || Lathe
6. Loft
7. Sistema de Partículas
8. Modelo por texturas

Son empleadas, para casos particulares. NURBS para objetos orgánicos por ejemplo, lathe para objetos iguales en sus lados de un mismo eje, etc. Pero ahondemos en cada uno un poco:

2.3.1 Estructuras Predefinidas.

Me refiero con esto, aquellas estructuras ya armadas por el sistema (hablando de 3d Studio Max).

Existen 3 tipos elementales:

1. Primitivas
2. Primitivas extendidas
3. Librerías

2.3.1.1 Primitivas: *Caja, Cono, Esfera, Geo Esfera, Cilindro, Tubo, Anillo, Pirámide, Tetera, y Plano.*

2.3.1.2 Primitivas Extendidas: *Hedra, Nudo Toroide, Caja "redondeada", Cilindro "redondeado", Tanque de Aceite, Capsula, Sprindle, Forma L, Gengon, Forma C, Anillo ondulado, Hose, Prisma.*

Librerías: *Son formas armadas, disponibles en 3d Max 7; Puertas, Ventanas, Árboles, Escaleras.*

Todas estas estructuras nos sirven para poder modelar objetos o escenas más complejas a partir de ellas. Por ejemplo, con 3 cajas podríamos armar una escena para una habitación.

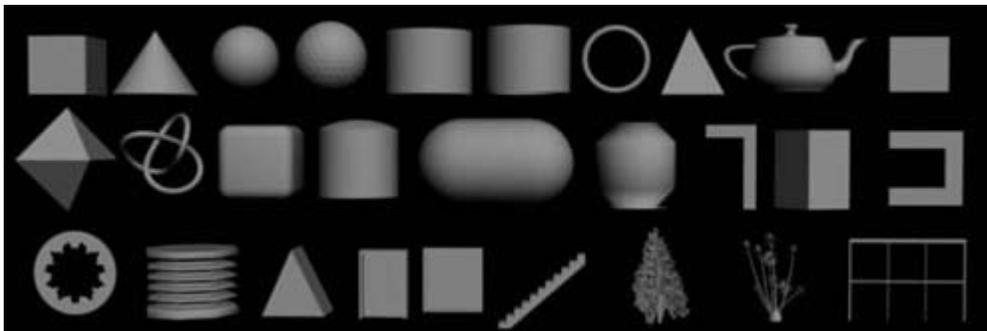


Fig.II.1 Primitivas Extendidas

2.3.1.3 Box Modeling.

Como su nombre lo indica, es el modelado de figuras complejas a través de una caja. Si una mera caja. Seguramente creerán es imposible realizarlo, pero empleando un modificador de mallas, *Edith Mesh*, podrán ir extendiendo la caja, convirtiéndola en otra cosa.

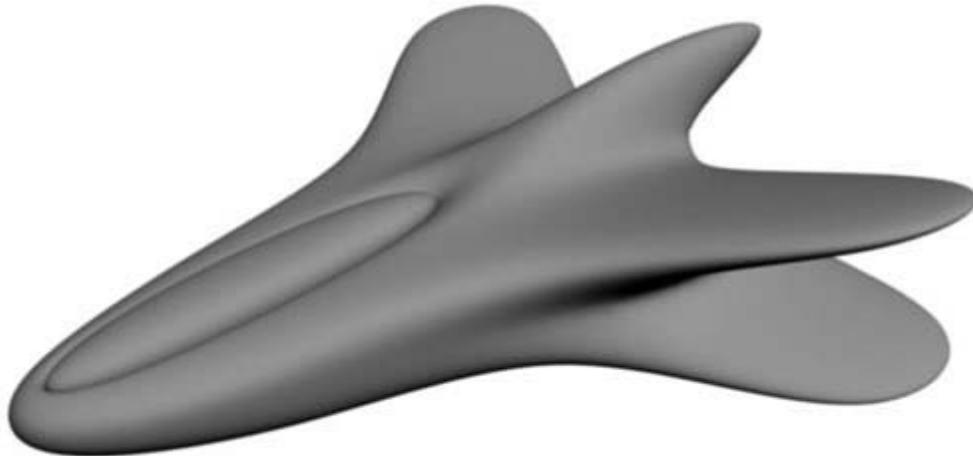


Fig.II.2 Ejemplo de modelado a través de una caja

2.3.1.4 NURBS Modeling.

Es una técnica para construir mallas de alta complejidad, de aspecto orgánico ó *curvado*, que

emplea como punto de partida splines (figuras 2d) para mediante diversos métodos, crear la malla 3d anidando los splines.



Fig.II.3 Ejemplo de modelado con NURBS

2.3.2 Operaciones Booleanas.

Consiste, en tomar dos mallas y aplicarles una de tres operaciones booleanas disponibles:

1. Resta
2. Intersección
3. Unión

Resta: Resta dos figuras $A - B$ ó $B - A$.

Intersección: Da como resultado sólo lo que esta "tocándose" de ambas figuras.

Unión: Funde ambas figuras creando una única nueva.

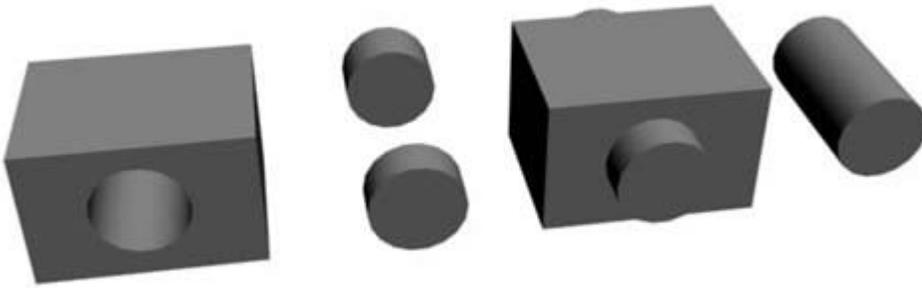


Fig.II.4 Ejemplo de booleanos

2.3.3 Lathe.

Es una técnica que a partir, de una figura 2d (spline) crea el volumen.



Fig.II.5 Ejemplo de Lathe

2.3.4 Extrude: Da profundidad a un objeto 2d. Extiende la profundidad.

Lathe: Tomando un spline, lo reproduce por un eje en toda su rotación. Ideal para botellas, copas, y demás objetos sin diferencia en sus costados. Aunque puede combinarse con otra técnica luego, y crear por ejemplo, una tasa.



Fig.II.6 Ejemplo de extrucción

2.3.5 SPLINES

Se deben emplear 2 ó más splines, para crear una malla 3d continua. El primer spline, funciona como path (camino) mientras que los demás, dan forma, extendiéndose, a través del *path*. Ideal para crear cables, botellas, etc.



Fig.II.7 Ejemplo de splines

2.3.6 Sistema de Partículas.

Es como su nombre lo indica, un sistema de partículas (proyección de formas geométricas, de forma controlada mediante parámetros varios tales como choque, fricción y demás). Es combinable, con efectos de dinámica y deformadores. Es ideal para crear humo, agua, ó cualquier cosa que sea muchos objetos y repetitivos.

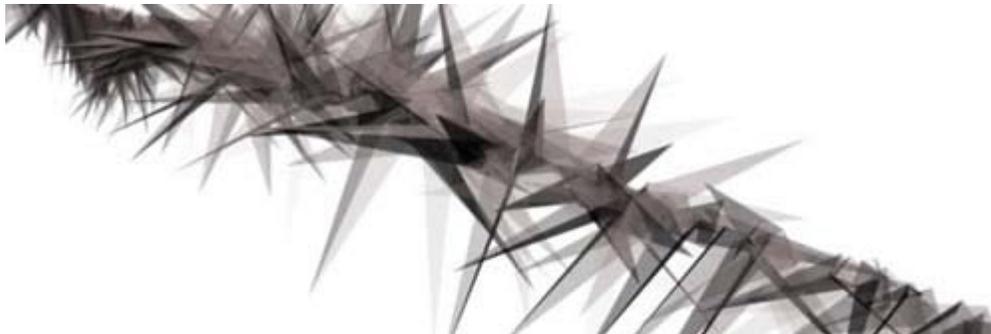


Fig.II.8 Ejemplo manejo sistema de partículas

2.3.7 Modelo por texturas.

Este tipo de modelado, si es que se lo puede denominar así, en vez de emplear deformadores en la malla, engañan la vista, con mapas del canal alpha (transparencia) para crear recortes, ó engaños directos de relieve (con un canal especial para esto independiente del de relieve) para crear terrenos por ejemplo.

Es un tipo de modelado, usado mucho para abstractos en 3d, y no es muy difícil de emplear, simplemente se deben manipular los canales para engañar la vista.



Fig.II.9 Modelo de texturas

CAPITULO III

APLICACIONES INFORMÁTICAS DE MODELADO Y ANIMACIÓN 3D

Dentro del área del diseño gráfico y más aún en el área del modelado y animación 3D encontramos gran variedad de software de acuerdo a los requerimientos del diseñador es así que a continuación se encuentra una lista de programas privativos y libres.

3.3 Aplicaciones 3D

| Nombre | Compañía | Enlace | Versión |
|---------------------------|---|---|----------------|
| Maya | Autodesk (antes alias wavefront) | http://www.autodesk.com/maya | Maya 2011 |
| SOFTIMAGE XS I | Autodesk (antes propiedad de AVID y antes de Microsoft) | http://www.softimage.com | Softimage 2011 |

| | | | |
|---------------------|----------------------|---|-----------------|
| 3DStudio MAX | Autodesk | http://www.autodesk.com/ 3dsmax | Max 2011 |
| LightWave | Newtek | http://www.newtek.com/ | LightWave 9.6 |
| Blender | Blender (OpenSource) | http://www.blender.org/ | 2.49 - 2.50 |
| Cinema 4D | Maxon | http://www.maxon.net | 11.5 |
| Houdini | Side Effects | http://www.sidefx.com/ | 11 |
| Rhinoceros | Rhino | http://www.rhino3d.com/ | 4 |
| Pov-ray | Povray | http://www.povray.org/ | 4 |
| Cheetah 3D | Cheetah 3D | http://www.cheetah3d.com/ | 4 |

Tabla VII. Tabla explicación aplicaciones informáticas de modelado y animación 3d

3.2 Elección de software para la realización del caso práctico

Debido que es un estudio comparativo se toma como referencia un programa para animación y modelado en 3d de software libre y uno de software propietario:

Software Propietario: 3D Studio Max

Software libre: Blender

Estos se han escogido debido a que son el software más conocido y utilizado en software libre y propietario.

CAPITULO IV

SOFTWARE PROPIETARIO: 3D STUDIO MAX

4.1 AUTODESK 3D STUDIO MAX

Autodesk 3ds Max (anteriormente **3D Studio Max**) es un programa de creación de gráficos y animación 3D desarrollado por Autodesk, en concreto la división Autodesk Media & Entertainment (anteriormente Discreet).

Fue desarrollado originalmente por Kinetix como sucesor para sistemas operativos Win32 del 3D Studio creado para DOS. Más tarde esta compañía fue fusionada con la última adquisición de Autodesk, Discreet Logic.

3ds Max es uno de los programas de animación 3D más utilizados. Dispone de una sólida capacidad de edición, una omnipresente arquitectura de plugins y una larga tradición en plataformas Microsoft Windows. 3ds Max es utilizado en mayor medida por los desarrolladores de videojuegos, aunque también en el desarrollo de proyectos de animación como películas o anuncios de televisión, efectos especiales y en arquitectura.

Desde la primera versión 1.0 hasta la 4.0 el programa pertenecía a Autodesk con el nombre de 3d Studio. Más tarde, Kinetix compró los derechos del programa y lanzó 3 versiones desde la 1.0 hasta la 3.0 bajo el nombre de 3d Studio MAX. Más tarde, la empresa Discreet compró los derechos, retomando la familia empezada por Autodesk desde la 4.0 hasta 7.0 esta vez bajo el nombre de 3dsmax. Finalmente, Autodesk retomó el programa desarrollándolo desde la versión 8.0 hasta la actualidad bajo el nombre de Autodesk 3ds Max.

Este programa es uno de los más reconocidos modeladores de 3d masivo, habitualmente orientado al desarrollo de videojuegos, con el que se han hecho enteramente títulos como las sagas 'Tomb Raider', 'Splinter Cell' y una larga lista de títulos de la empresa Ubisoft.

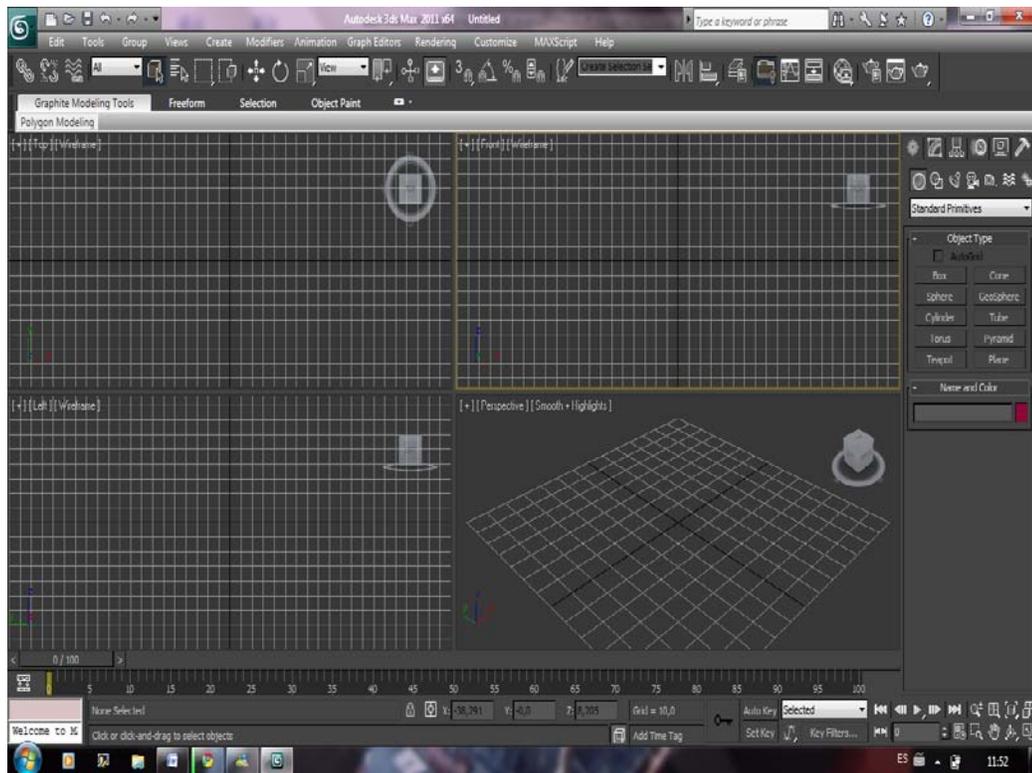


Fig IV.10 Interfaz Autodesk 3d Max

4.2 Conocimiento de 3D Max

Es un programa que nos permitirá la creación de animaciones, logotipos, diseños 3d, imágenes impactantes, etc. El programa contiene muchas herramientas que facilitan la creación de los mismos, así como vistas de diseño para la creación.

Entorno. Es la pantalla de inicio del programa en el cual trabajaremos.

4.2.1 Barras de menus (file, edit, tool, group, views...) En ella se muestran los menús que se utilizan en 3d, cada menú despliega más opciones como vemos en estas 2 imágenes:

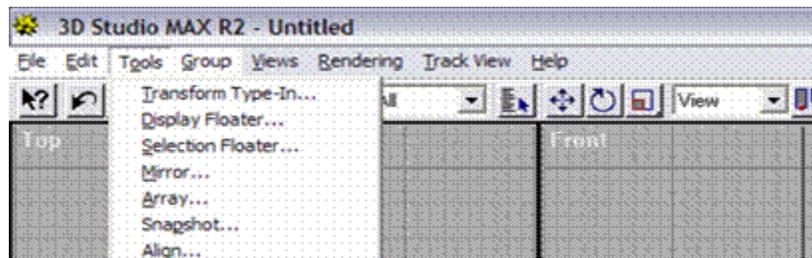


Fig IV.11 Barras de Menú Autodesk 3d Max

4.2.2 Barra de herramientas. Ofrece acceso rápido a herramientas y cuadros de diálogo para muchas tareas comunes.



Fig IV.12 Barras de Herramientas Autodesk 3d Max

Si no se visualizan los demás botones de esta barra de herramientas, coloca tu cursor en una orilla de esta barra y aparecerá una mano, cuando aparezca, muévela hacia la izquierda o derecha y verás estas nuevas herramientas que son: **material editor**, **render scene**, etc. Observa la imagen y ve la manita y las nuevas herramientas a la derecha:



Fig IV.13 Barras de Herramientas 2 Autodesk 3d Max

4.2.3 Panel de solapas. Proporciona comandos, controles y parámetros para crear, modificar, vincular, animar y presentar diseños de objetos.



Fig IV.14 Panel de solapas Autodesk 3d Max

4.2.3.1 Herramientas del panel de solapas

Create. Dando un clic en este icono aparecen otras herramientas que son:

Geometry. Geometría. Da acceso a las primitivas de 3D STUDIO MAX en 3d, que pueden ser: esferas, cajas, cilindros, etc. Su icono es:

Shapes. Formas planas. Da acceso a las primitivas planas del programa, son líneas polígonas, rectángulos arcos, etc.

Lights. Luces. Este es el comando para crear las luces con las que iluminaremos nuestra escena. En el momento que creamos una luz, la luz por defecto del programa desaparecerá. Estas luces pueden ser: Omnidireccionales, Focos, etc.

Cameras. Cámaras. Sirve para crear las cámaras de la escena. Una vez que hayamos creado una cámara, seleccionamos una viewport y pulsamos la tecla C y la viewport se transformará en la vista que se ve en la cámara.

Helpers. Ayudantes. Da acceso a los ayudantes: Cinta métrica, rejilla, etc.

Space warps. Espacios de modificación. Este botón da acceso a los comandos de efectos especiales como: Bomba, honda, Viento, etc.

Systems. Sistemas. Esto sirve para los huesos de animación, matriz tridimensional, etc.

Hay que decir que si tenemos la resolución de pantalla a una resolución menor a 800X600, la barra de herramientas y los menús no se verán por completo y tendremos que arrastrar hacia arriba y abajo y de izquierda a derecha para visualizar el resto, te recomiendo que tengas presente que si no observas algunos botones tienes que arrastrar los paneles cuando te aparezca la "manita" ya sea hacia arriba o hacia abajo; derecha o izquierda.

Modify. Es de los más importantes porque con este vamos a modificar todo lo que se puede crear en este programa.

Al dar clic en este botón cambia la solapa de abajo y aparecen botones los cuales se llaman modificadores.

Los demás botones los conocerás conforme las prácticas que hagamos.

Control de visores. Proporciona una manera de movimiento o de percepción de objetos dentro de las vistas en las que trabajaremos.

Control de tiempo. Permite controlar los movimientos, animar una película y moverte por los fotogramas que constituyen una película.

4.3 CREACIÓN DE GEOMETRÍAS

Todos los objetos del mundo real están constituidos mediante formas básicas, cubos, cilindros, esferas, las cuales son modificadas para conseguir la forma deseada, pasa lo mismo en 3ds MAX, en donde se comienza con formas básicas, a la cuales se les pueden ir aplicando distintos modificadores, que cambian la apariencia de los mismos, se puede comenzar con un sólo objeto al cual se le pueden ir añadiendo otros objetos, o creándolos, desde el objeto base. El modelado es algo muy parecido al proceso de escultura donde se comienza con una piedra y a esta se le va esculpiendo y detallando, hasta obtener el resultado deseado.

Antes de crear la geometría es de suma importancia saber que todos los objetos se crean en base a una cuadrícula inicial que corresponde a las coordenadas universales y por consiguiente esta no se puede alterar, lo que únicamente se puede modificar es el espaciado entre las cuadrículas, pero está cuadrícula inicial se puede desactivar, para esto se tiene que crear un objeto de cuadrícula desde el panel comandos, crear, ayudantes, GRIDS, esta cuadrícula se puede rotar y ajustar a la necesidad de cada usuario y definir el espaciado entre cada cuadrícula, se pueden crear tantas como se requiera.

En 3ds MAX por lo general se comienza con dos tipos de geometría básica que son:

4.3.1. Primitivas estándar.- Son las formas más básicas como:

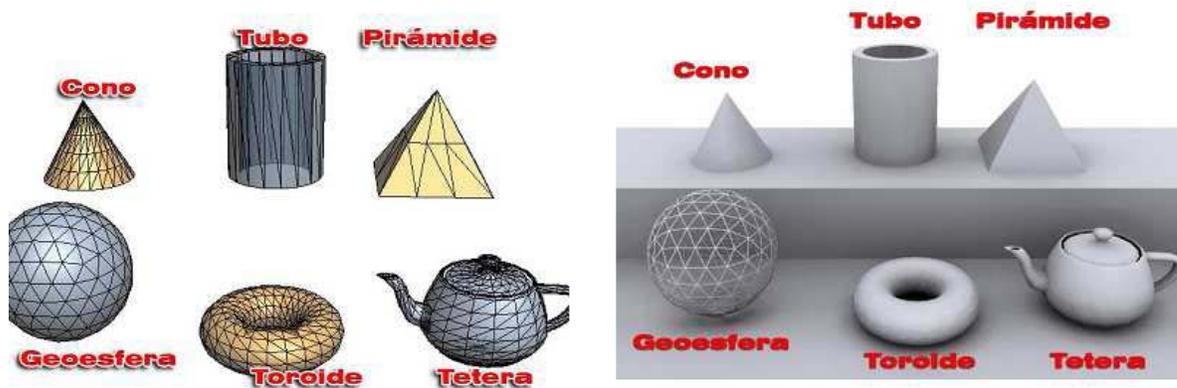


Fig IV.15 Primitivas estándar

4.3.2 Las primitivas geométricas.- Guardan relación con objetos del mundo real, como balones de playa, conductos, cajas, arandelas y barquillos de helado. En **3ds Max**, la mayoría de estos objetos pueden modelarse utilizando una sola primitiva. Las primitivas también pueden combinarse para formar objetos más complejos, que posteriormente pueden refinarse con modificadores. El programa incluye un conjunto de 10 primitivas básicas: caja, cono, esfera, geoesfera, cilindro, tubo, toroide, pirámide, tetera y plano

4.3.3. Primitivas extendidas.- Son una recopilación de primitivas complejas para el programa. Las Primitivas extendidas son: Poliedro, Nudo toroide, ChafCaja, ChafCil, Bidón, Cápsula, Huso, Ext-L, Gengon, Ext-C, OndAnillo, Manguera, Prisma

4.4 CREACIÓN DE FORMAS

Son objetos geométricos bidimensionales conformados ya sea por líneas o por curvas, entre las más importantes están: Splines (LINEAS), Curvas NURBS y Mallas editables.

Estos objetos geométricos, pueden ser modificados y adaptarse a las formas que nosotros necesitamos a través de:

4.4.1 CONVERT TO EDITABLE MESH (Convertir a malla editable).- El modificador Editar malla convierte los objetos en mallas triangulares. Ofrece herramientas de edición específicas para cada uno de los tres componentes de malla: Vértice, Cara y Arista

· También se puede transformar y aplicar Mayús-Clonar a componentes de malla seleccionados, así como aplicarles modificadores estándar. Puede trabajar en cualquier escala, desde una gran superficie de terreno hasta un vértice, una cara o una arista individuales.

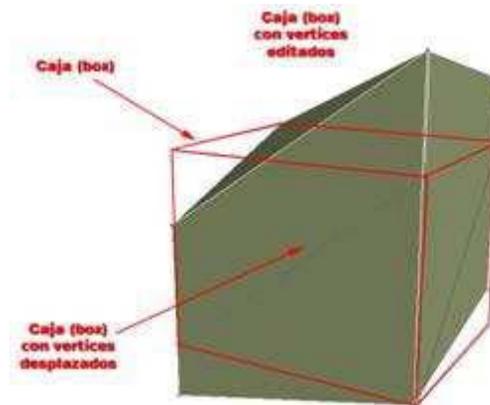


Fig IV.16 Convertir en malla editable

4.4.2 SET ID (Definir ID o identificación).- Cuando una primitiva es transformada en un objeto mallado los polígonos toman un ID, principalmente los ID son usados con el material subobjeto, el cual permite usar distintos materiales para cada ID determinado.

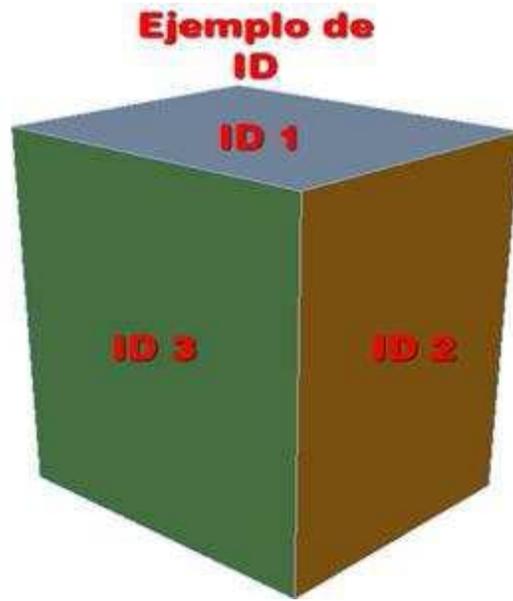


Fig IV.17 Definir ID

4.4.3 ELEMENT.- Es la subdivisión de los elementos de un objeto, estos elementos se puede extruir y biselar.



Fig IV.18 Subdivisión de elementos

4.5 CREACIÓN DE OBJETOS DE COMPOSICIÓN

4.5.1 MORFISMO.- Permiten distribuir un objeto mallado dentro de una superficie, los objetos se pueden distribuir aleatoriamente, en los distintos vértices a lo largo de la superficie vértices.

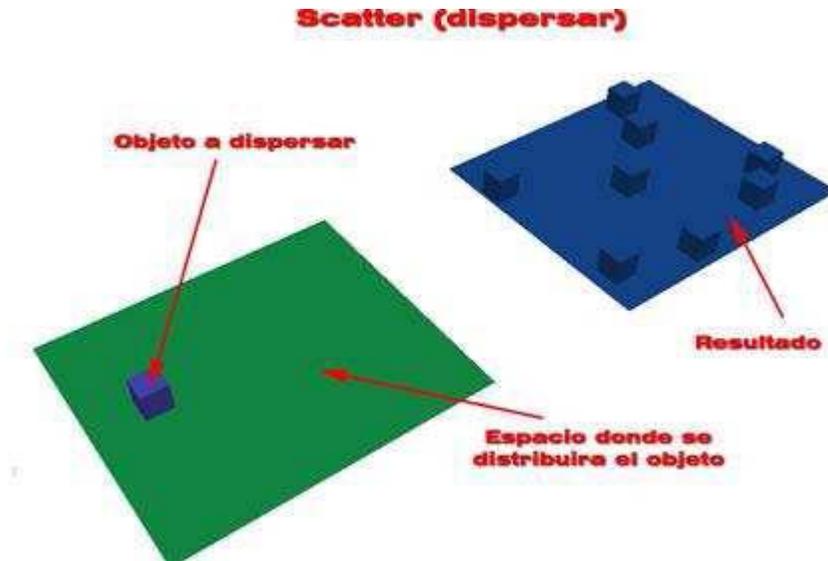


Fig IV.19 Morfismo

4.5.2 BOOLEANO

o **COMPOUND OBJET (Objetos de Composición).**- Son la composición de dos objetos o más ya sean splines u objetos mallados.

o **BOOLEAN (Operaciones Booleanas).**- Es la operación de de dos o más objetos, la operación puede ser una unión, sustracción, intersección o cortar.

4.5.3 SOLEVADO o LOFT (Solevado o extrucción con trayectoria).-Este tipo de objeto los constituyen dos o más formas splines bidimensionales, una forma actúa como modelo de sección y la otra como trayectoria.

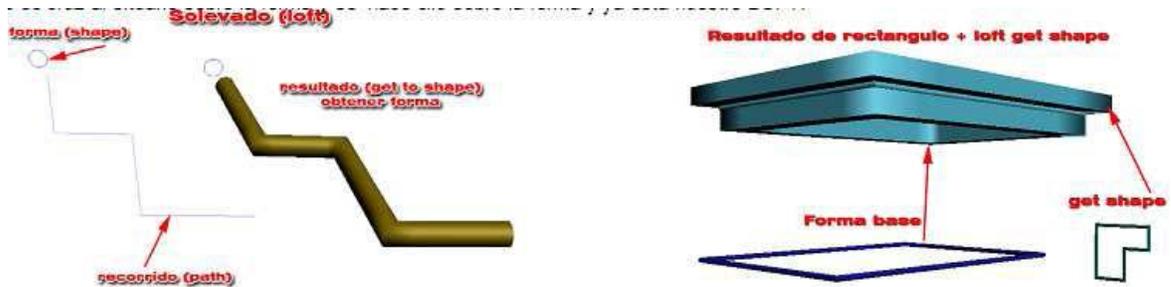


Fig IV.20 Loft

4.5.4 FOILAGE (Follaje).- Son principalmente árboles a los cuales se les puede ir modificando los distintos elementos de él, hay 3 niveles de resolución que son bajo, medio y alto, esto sirve para el detallado de cada elemento al momento del render, claro que a mayor resolución mayor tiempo de render.



Fig IV.21 Follaje

.5.5 DOORS (Puertas).- Las puertas son elementos prediseñados y se clasifican en 3 tipos distintos: PIVOT (puerta pivotante), SLIDING (puerta corrediza) y BIFOLD (desplegable). Estos elementos se pueden modificar sus características según se requiera.

4.5.6 WINDOWS (Ventanas).- Al igual que las puertas son elementos prediseñados y se pueden crear 6 tipos distintos tipo: FIXED (fija), AWNING (marquesina), CASEMENT (oscilante), PIVOTED (batiente), PROYECTED (proyectada) y SLIDING (corrediza), en cualquier tipo de ventana se tienen que dimensionar los marcos.

4.5.7 STAIRS (Escaleras).- Son 4 tipos de escaleras: STRAIGHT STAIR (escalera recta), ESPIRAL STAIR (espiral), UTYPESTAIR (en forma de U) y LTYPE STAIR (en forma de “L”). Los 3 tipos de escaleras nos permiten modificar sus características según las necesidades.

4.6 MODIFICADORES

Una gran parte del modelado corresponde a los modificadores, un modificador, como su nombre lo dice es un componente que modifica algún aspecto ya sea la apariencia de un objeto o las propiedades del mismo, los modificadores se van almacenando en un catálogo, en el cual se pueden ir añadiendo “N” número de modificadores, cualquier modificador aplicado a un objeto puede ser modificado posteriormente.

El catálogo de modificadores es una lista donde se almacenan los modificadores aplicados a un objeto, se pueden aplicar “N” número de modificadores a un objeto, el orden de los modificadores dentro de esta lista es importante, ya que estos se almacenan de tal manera que el último modificador aplicado se sitúa en la parte superior del catálogo de modificadores, y afectará a todos los modificadores que estén por debajo del él, se pueden seleccionar y editar los modificadores después de ser aplicados, o en dado caso hasta ser borrados. Los diferentes modificadores también pueden ser apagados, lo cual hace que los resultados del modificador no sean visibles, hasta que este se vuelva a encender.

Existen modificadores que tienen subobjetos dentro del mismo modificador, los cuales al lado izquierdo tienen un signo de más, cada modificador tiene sus características, pero en general contienen 2 subobjetos los cuales son: Modificadores de espacio universal y modificadores de espacio objeto. Entre modificadores que existen, podemos mencionar.

- **AFFECT REGION (Afectar región).**- Este modificador afecta sólo un aparte de un objeto mallado teniendo como base los vértices, haciendo una curvatura.

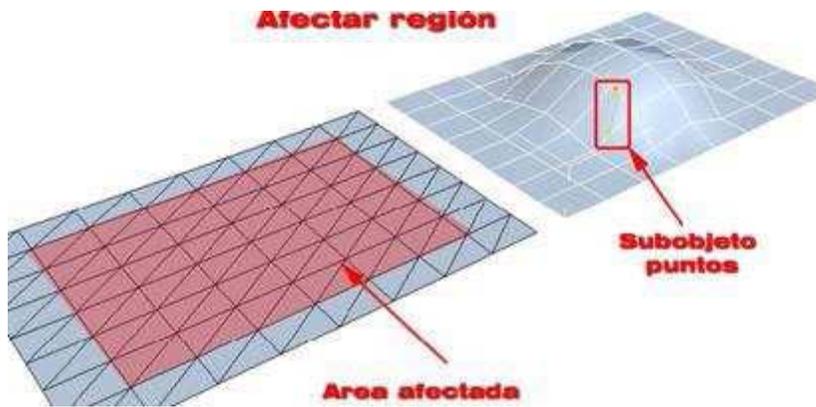


Fig IV.22 Affect region

- **TAPER (Afilar).**-Permite manipular la parte inferior y la parte superior de un objeto mallado, para crear un contorno de pirámide.

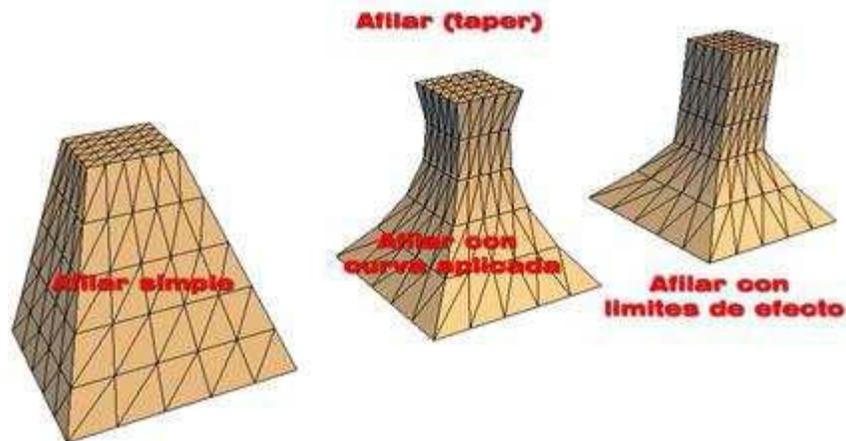


Fig IV.23 Afilar

- **BEVEL (Biselar).**-Este modificador sólo está activo cuando se seleccionan splines, lo que hace es una extrucción en las splines dándoles profundidad, al mismo tiempo de crea un chaflán en la cara

superior, pudiéndose aplicar hasta 3 ocasiones el extruir más el chaflán, generalmente es usado con textos.

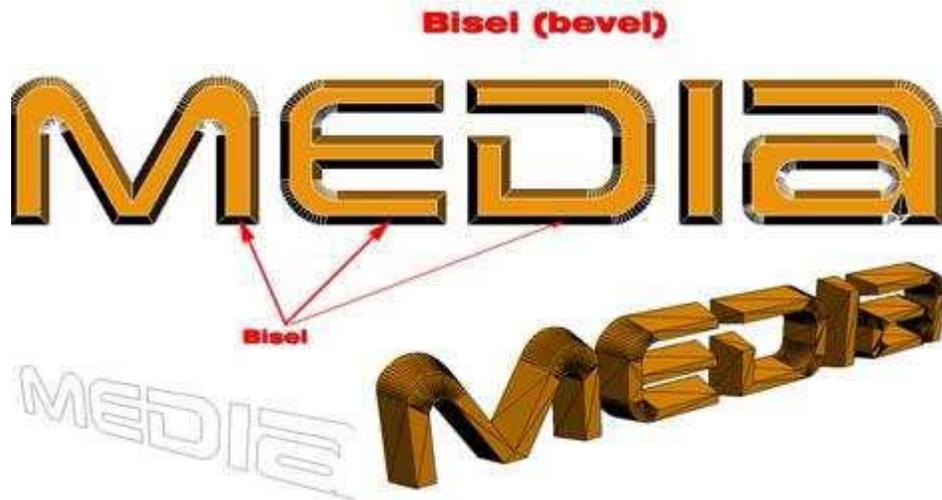


Fig IV.24 Bevel

· **BEVEL PROFILE (Biselar perfil).**- Funciona de manera similar al de biselar con la gran diferencia que en este tipo, se tiene que definir PICK PROFILE (perfil), el cual dará la altura a la spline.

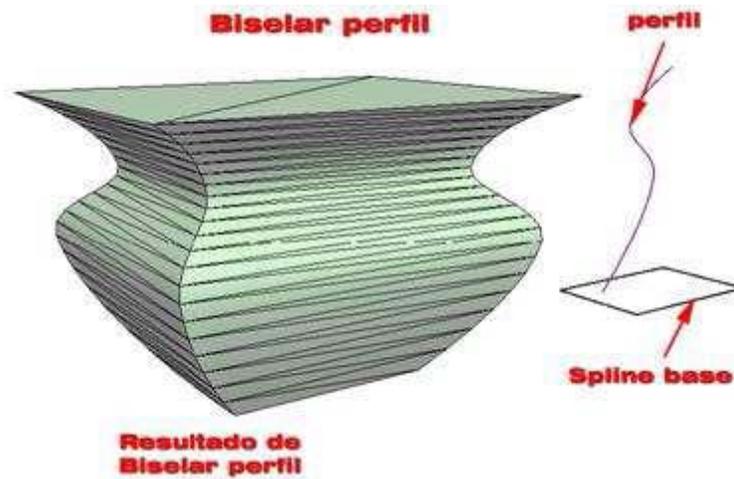


Fig IV.25 Bevel Profile

· **SHELL (Cáscara).**- Es un modificador muy útil, sirve para dar profundidad o altura a elementos que no la tienen en forma de extrucción, por ejemplo, un plano, el cual no tiene grosor y aplicándole este modificador se le puede dar un grosor, creando de esta manera una cáscara al plano.

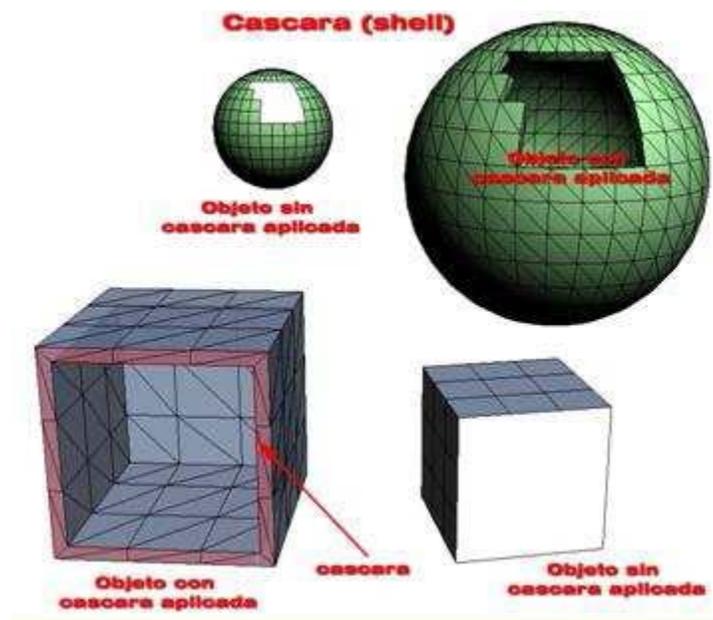


Fig IV.26 Cáscara

- **LATTICE (Celosía).**- Este modificador transforma las aristas en objetos cilíndricos de 3 dimensiones (travesaños), y los vértices en geoesferas (articulaciones), es muy útil para crear marcos de ventanas cuando no se cuenta con el tiempo suficiente para realizarlos.

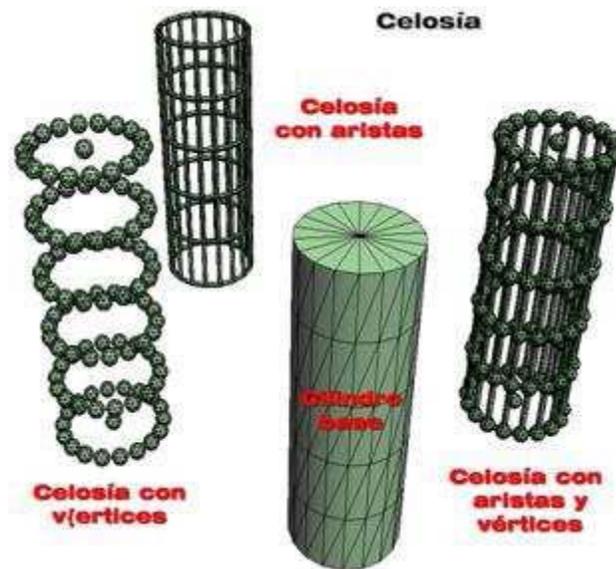


Fig IV.27 Lattice

- **BEND(Curvar).**- Permite afectar el objeto seleccionado de tal forma que ocasiona una curva, con un ángulo determinado, el cual puede ir de 0 hasta 360°, la curva se origina entre los 2 segmentos que componen el ángulo.

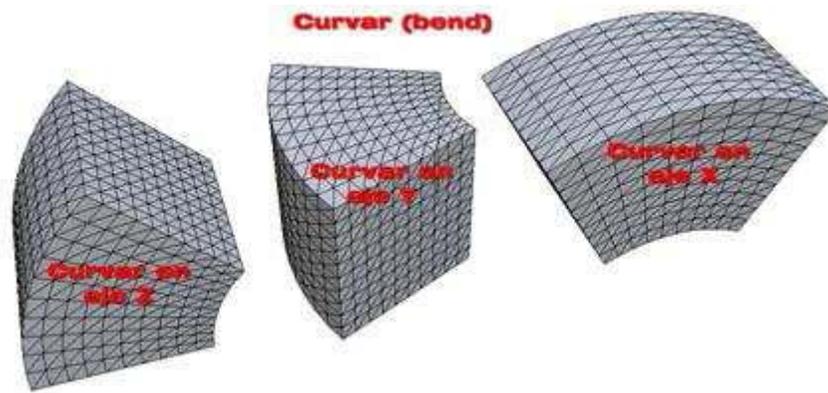


Fig IV.28 Bend

· **DISPLACE (Desplazar).**- Este modificador desliza los vértices verticalmente, de acuerdo a las características del mapa o de la imagen que se utiliza, tomando el color más claro con el mayor desplazamiento y los colores oscuros con menor desplazamiento, por lo regular se utiliza con planos.



Fig IV.29 Displace

- **EDIT MESH (Editar malla).**- Permite transformar temporalmente, el objeto seleccionado a una malla editable y tener las mismas opciones como si se hubiese convertido a malla editable.
- **EDIT POLY (Editar malla poligonal).**- Permite transformar temporalmente, el objeto seleccionado a una malla poligonal y tener las mismas opciones de la malla poligonal, con una gran diferencia, permite manejar la malla para MODEL (modelado) o ANIMATION (animación).
- **EDIT SPILNE (Editar spline).**- Permite transformar temporalmente el objeto seleccionado a una spline editable, sin tener que convertirla a editable spline.
- **FILLET/CHANFER (Fillet/chaflán).**- Este modificador funciona únicamente con splines, lo que produce es redondear (FILLET) los vértices de las splines.

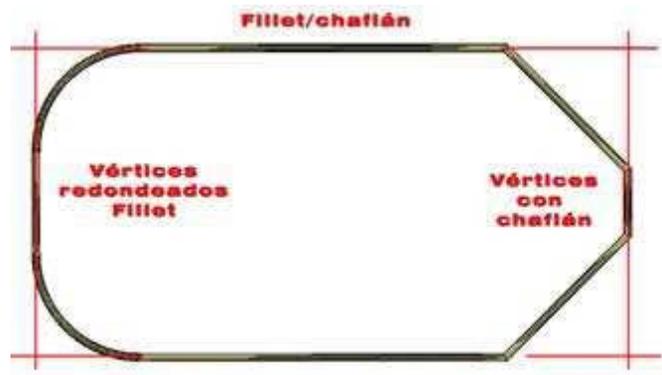


Fig IV.30 Fillet Chanfer

MAP SCALER OSM (Escalar mapa).- Ajusta el mapa aplicado al objeto, de tal manera que la dimensión del mapa es proporcional al tamaño del objeto.



Fig IV.31 Escalar mapa

· **MAP SCALE WSM (Escalar mapa espacio universal).**- Ajusta el tamaño del mapa aplicado de tal forma de que este no varía, aun y cuando el objeto sea cambiado de dimensiones, el mapa no cambia de dimensiones a menos de que se cambien en escala.



Fig IV.32 Escalar mapa espacio universal

· **SPHERIFY (Esfericar).**- Ocasiona un efecto de contraer los vértices, lo cual ocasiona un efecto esférico en el objeto.

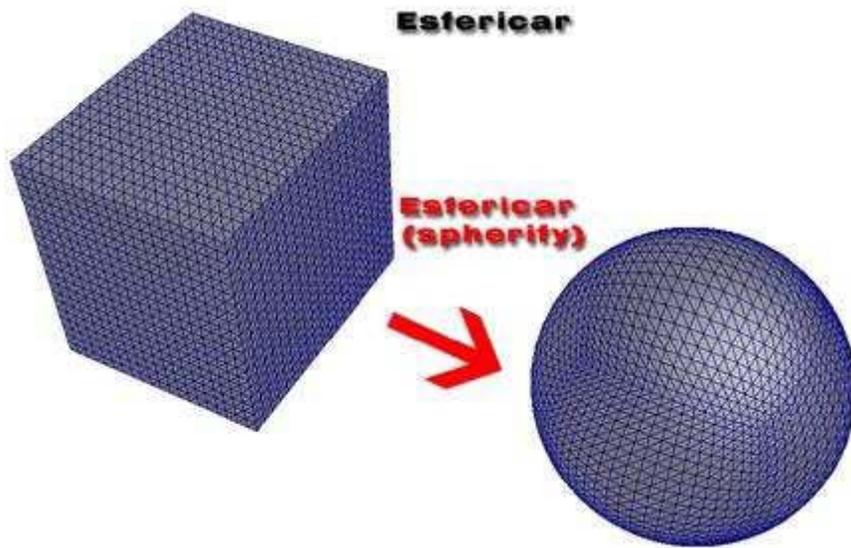


Fig IV.33 Esfericar

- **STRECH (Estirar).**- Desplaza el objeto, partiendo desde el pivote del objeto, produciendo un alargamiento del objeto.
- **FACE EXTRUDE (Extruir caras).**- Extruye las caras seleccionadas, para aplicar este modificador, antes se tiene que aplicar un modificador de selección a nivel de caras, o aplicar un modificador que permita acceder a la selección de subobjetos de caras o polígonos.
- **FFD (FFD).**- Es la abreviación de FREE FORM DEFORMATION (deformación de forma libre), como su nombre lo indica al aplicar este modificador a un objeto este se puede deformar de la manera que se desee, la forma de actuar de este modificador es la de crear una estructura en los 3 ejes de simetría.
- **MAP UVW (Mapa UVW).**- Permite ajustar las coordenadas del mapa o material sobre el objeto seleccionado, mediante una forma determinada, así como también cambiar el tamaño del mapa, y la

ubicación del mismo mediante el subobjeto de gizmo, además de generar coordenadas para aquellos objetos que carecen de coordenadas.



Fig IV.34 Mapa UVW

- **MATERIAL.**- Permite cambiar el ID de un material al objeto aplicado, se utiliza con un material tipo subobjeto, es ideal para animar un material.
- **WAVE (Onda).**- Produce ondulaciones en un objeto, es algo parecido cuando se arroja un objeto al agua, el objeto produce ondas a través del agua, es lo mismo que produce este modificador con la gran diferencia que las reproduce en un solo sentido.

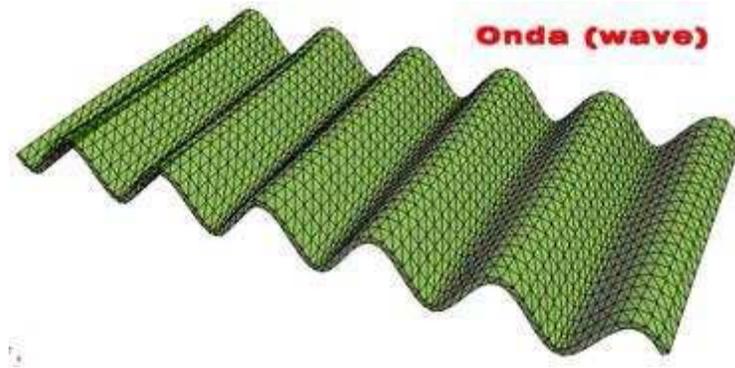


Fig. IV. 35 Wave

o **NOISE (Ruido)**.- Produce irregularidades desplazando los vértices en una forma aleatoria en los ejes del objeto seleccionado.

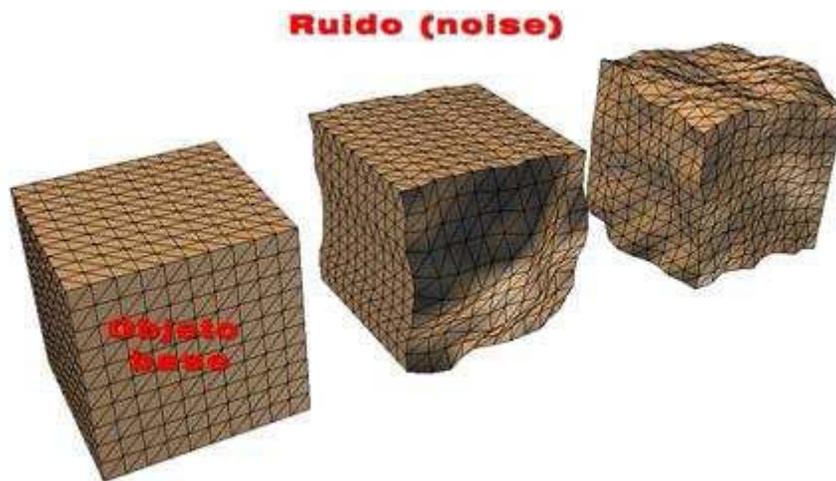


Fig. IV. 35 Noise

o **CROSS SECTION (Sección transversal)**.- Este modificador está activo para las splines, lo que produce es la creación de una sección transversal entre 2 o más splines, conectando los vértices de las splines.

o **SLICE (Segmentar)**.- Produce un corte en un objeto, creando un plano de referencia que sirve para realizar el corte, los objetos se pueden cortar desde REMOVE BOTTON (parte inferior) del plano de segmentación, o desde REMOVE TOP (parte superior) del plano.

o **SURFACE (Superficie)**.- Genera una superficie basándose en los contornos de las splines, lo más común es la utilización del modificador sección transversal antes de aplicar superficie, con este modificador se pueden realizar superficies de complejidad considerable, es una de las técnicas de mayor uso dentro del modelado, por la facilidad de edición de las splines.

o **SKEW (Inclinar)**.-Permite crea un inclinación uniforme en el objeto en cualquier eje. Este modificador tiene 2 subobjetos que son GIZMO y CENTER (centro) los cuales pueden ser manipulados.

o **MIRROR (Simetría)**.- Refleja el objeto seleccionado en un determinado eje, pudiendo únicamente cambiar la orientación del objeto, o produciendo una copia del mismo objeto.

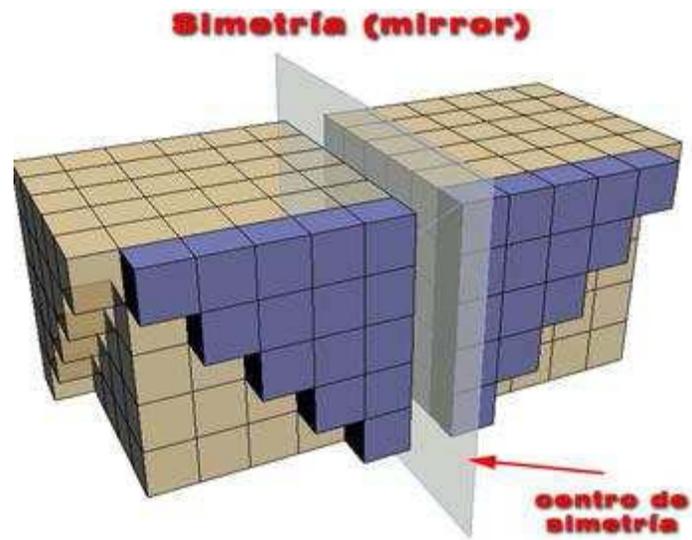


Fig. IV. 36 Mirror

o **SYMMETRY (Simetricidad)**.- Este modificador es una herramienta muy útil para el modelado, el efecto de este modificador es reproducir el objeto seleccionado (objeto mallado), creando la misma geometría con respecto al eje que se determine.

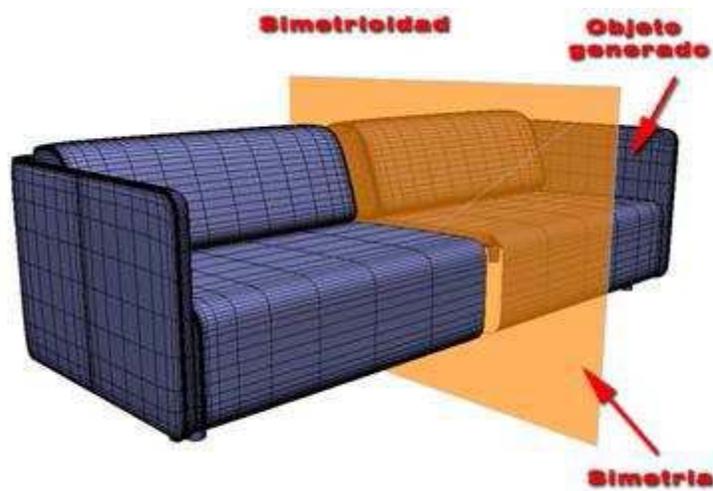


Fig. IV. 37 Symmetry

o **MESH SMOOTH (Suavizar malla).**- Permite evitar los vértices afilados, transformándolos en vértices redondeados, este modificador es más evidente con aquellos objetos, que tienen esquinas afiladas. Se pueden definir 3 tipos de suavizado los cuales son: NURBS, CLASIC (clásico) y QUAD. El suavizado dependerá de la segmentación de los distintos objetos, entre mayor sea el número de segmentos el suavizado será más uniforme.

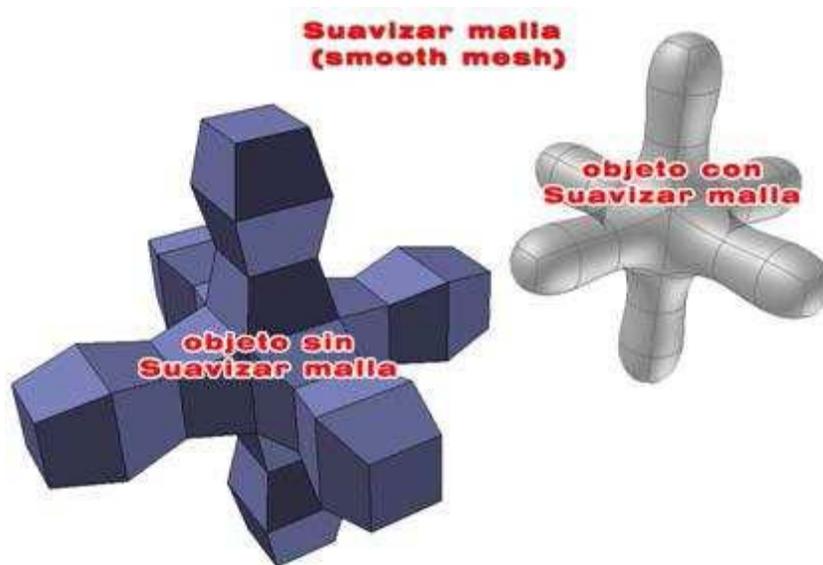


Fig. IV. 38 Mesh Smooth

o **CAP HOLES (Tapar agujeros).**- Mediante este modificador se puede rellenar los huecos en un objeto, generando las caras necesarias para cubrir los huecos en el objeto.

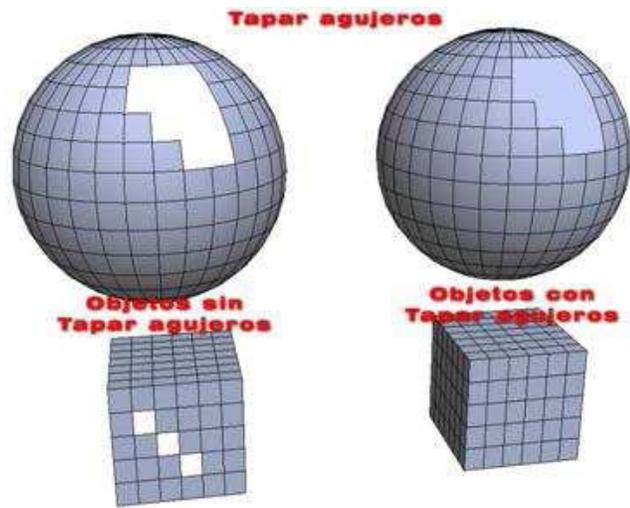


Fig. IV. 39 Cap holes

o **TWIST (Torcer)**.- Rota el objeto con respecto a un eje, haciéndolo girar alrededor del eje seleccionado. Este modificador tiene 2 subobjetos los cuales son: gizmo el cual se puede mover, rotar y escalar, a diferencia del centro que solamente se puede mover.

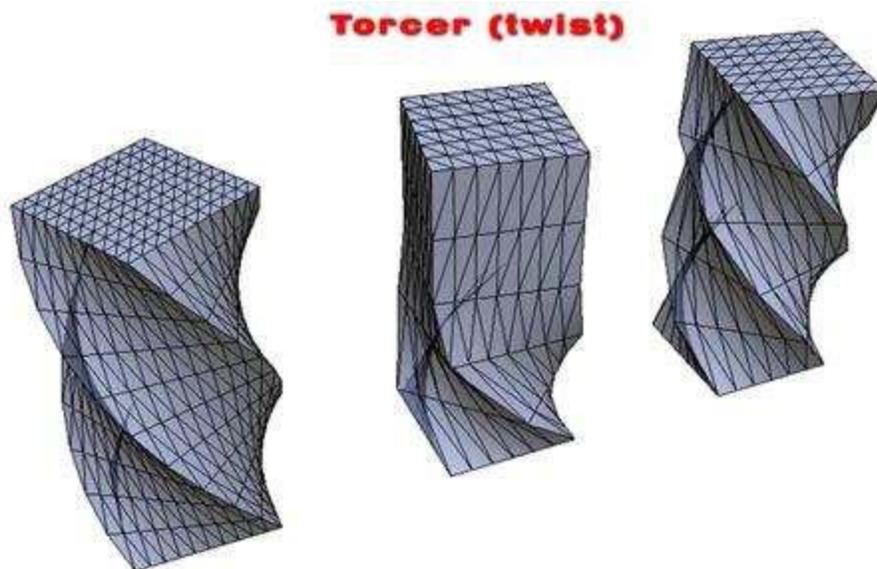


Fig. IV. 40 Twist

o **LATHE (Torno)**.- Genera una rotación de 360° a un spline, la cual sirve como perfil para conformar un objeto 3d, generando secciones transversales con la splines, y conectándolas entre sí, lo cual forma la estructura del objeto creado mediante torno.

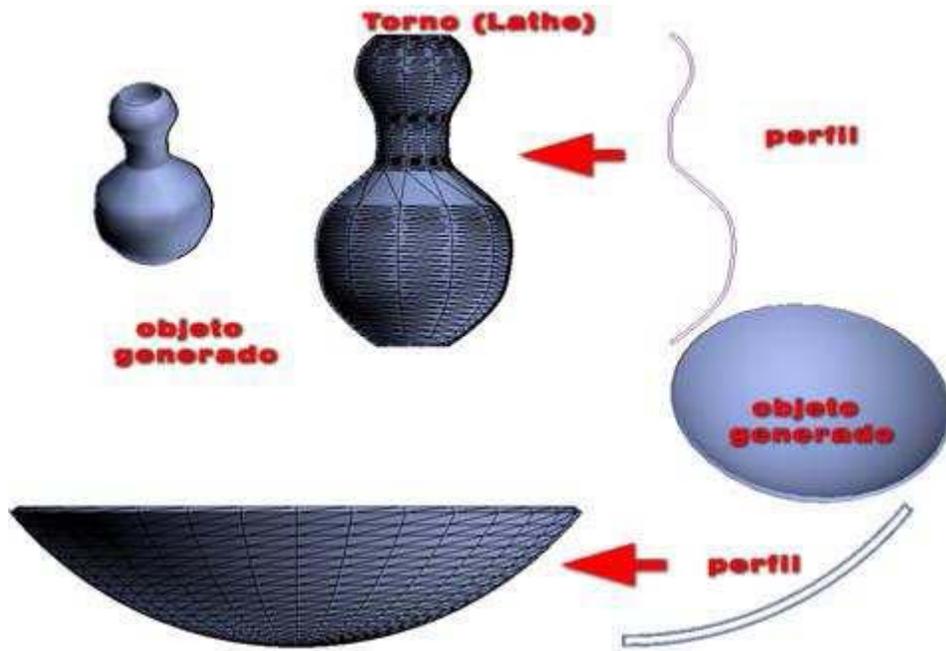


Fig. IV. 41 Lathe

4.7 EDITOR DE MATERIALES

Los materiales se asignan a los objetos o conjuntos de selección individuales; una misma escena puede contener muchos materiales diferentes.

- Estándar es el material predeterminado. Es un modelo de superficie versátil y con un gran número de opciones.
- Raytrace permite crear reflexiones y refracciones de Raytrace. Además, admite niebla, densidad de color, translucidez, fluorescencia y otros efectos especiales.
- Arquitectónico proporciona un material físicamente preciso. Está concebido especialmente para usarse con el renderizador detallado predeterminado y radiosidad.

4.8 MATERIALES DE COMPOSICIÓN

Mezcla 2 lados Multi/subobjeto

- Mezcla, es un material que combina dos materiales en un mismo lado de una superficie.
- 2 lados, permite asignar materiales distintos a las caras anteriores y posteriores de los objetos.
- Multi/Subobjeto, usa el nivel de subobjetos para asignar varios materiales a un mismo objeto, según los valores de ID de material.

4.9 TIPOS DE MAPAS

El uso más frecuente de los mapas es para mejorar el aspecto y realismo de los materiales. También puede utilizar mapas para crear entornos o proyecciones de luces. Cada tipo de mapa crea un efecto distinto y se comporta de una forma determinada. Un tipo de mapa puede combinar varios materiales.

· Los tipos de mapas 2D son bitmaps o patrones 2D generados por procedimiento. o **Bitmap:** Una imagen guardada como matriz de píxeles en un formato archivo de imagen fija, como .tga, .bmp, etc., Todos los tipos de archivos bitmap o de animación que admite 3ds max pueden utilizarse como bitmap en un material.

o **Ladrillos:** Crea ladrillos u otros materiales de mosaico con colores o mapeados de material. Incluye los patrones de ladrillos arquitectónicos habitualmente definidos.

o **Cuadros:** Combina dos colores en un patrón de damero. Ambos colores pueden sustituirse por un mapa.

o **Degradado:** Crea una amortiguación lineal o radial de tres colores.

· Tipos de mapas 3D, se incluyen:

o **Celular:** Genera un patrón celular, útil para diversos efectos visuales, como mosaico, superficies de guijarros e incluso superficies oceánicas.

o **Cavidad:** Genera protuberancias tridimensionales sobre una superficie.

o **Mármol:** Simula el vetado del mármol con dos colores explícitos y un tercer color intermedio.

o **Ruido:** Crea un patrón de turbulencia en tres dimensiones. Como Cuadros en 2D, se basa en dos colores que pueden mapearse.

o Humo: Genera patrones de turbulencia basados en fractales para simular los efectos del humo en un rayo de luz, u otros efectos de mapeado fluido nubosos.

o Estuco: Genera un patrón fractal similar al estuco.

o Agua: Crea efectos acuosos u ondulados generando varios centros de ondas esféricas y distribuyéndolos aleatoriamente.

o Madera: Crea un patrón de vetas de madera 3D.

- Tipos de mapas cajistas, consiste en superponer dos o más imágenes para combinarlas.

o Compuesto: Compone varios mapas. Al contrario que Mixto, Compuesto no dispone de controles específicos para la cantidad de mezcla, sino que basa la cantidad de mezcla en el canal alpha de los mapas.

o Máscara: Mapa que controla la posición donde se aplica un segundo mapa.

o Mixto: Mezcla dos colores o dos mapas. La cantidad de mezcla puede ajustarse especificando un nivel de mezcla. El nivel de mezcla se puede mapear.

o Multiplicar RGB: Combina dos mapas multiplicando sus valores RGB y alpha.

CAPITULO V

SOFTWARE LIBRE: BLENDER

5.1 SOFTWARE LIBRE: BLENDER

Blender es un programa informático multiplataforma, dedicado especialmente al modelado, animación y creación de gráficos tridimensionales.

El programa fue inicialmente distribuido de forma gratuita pero sin el código fuente, con un manual disponible para la venta, aunque posteriormente pasó a ser software libre. Actualmente es compatible con todas las versiones de Windows, Mac OS X, Linux, Solaris, FreeBSD e IRIX.

Tiene una muy peculiar interfaz gráfica de usuario, que se critica como poco intuitiva, pues no se basa en el sistema clásico de ventanas; pero tiene a su vez ventajas importantes sobre éstas, como la configuración personalizada de la distribución de los menús y vistas de cámara.

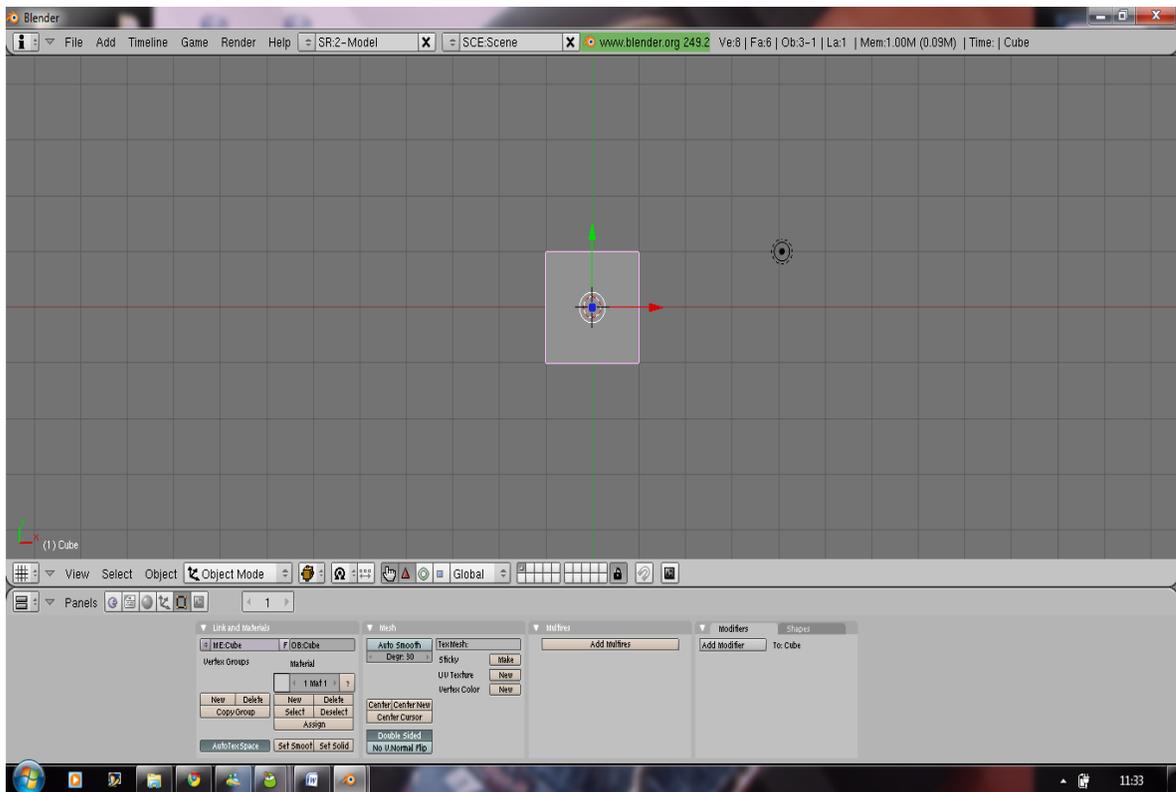


Fig. V.42 Interfaz Blender

5.1.1 Interfaz

Los conceptos detrás de la interfaz de Blender hacen que esta no sea muy estándar, sino que sea diferente de los otros paquetes 3D. Especialmente los usuarios de Windows necesitarán coger el truco a la manera en la que Blender maneja los controles, tales como botones o movimientos de ratón. Pero esta diferencia es de hecho la gran fuerza de blender: una vez comprenda la manera en la que funciona Blender, encontrará que puede hacer su trabajo extraordinariamente rápido y de manera muy productiva.

Por otro lado, la interfaz de Blender ha cambiado sustanciosamente desde la versión 2.28 a la versión 2.3, así que incluso los usuarios más experimentados podrán sacar partido a este capítulo.

5.1.2 Conceptos de la Interfaz de Blender

La interfaz de usuario es el mecanismo de interacción mutua entre el usuario y el programa. El usuario se comunica con el programa mediante el teclado y el ratón, el programa responde por medio de lo que muestra en pantalla.

5.1.3 El teclado y el ratón

La interfaz de Blender saca provecho de los ratones de tres botones y una amplia gama de atajos de teclado (en el Volumen II se encuentra una descripción detallada). Si el ratón sólo tiene dos botones, es posible emular el botón central (*interface_functions_settings* describe como). Es posible usar un ratón con rueda, pero no es obligatorio, ya que también existen atajos de teclado que cumplen la misma función. Este libro asume las siguientes convenciones para describir la entrada de usuario:

- Los botones del ratón se abrevian como BIR (botón izquierdo del ratón), BMR (botón medio del ratón) y BDR (botón derecho del ratón).
- Si el ratón tiene una rueda, BMR se refiere a hacer click con la rueda como si ésta fuera un botón, mientras que RR significa girar la rueda.

- Las letras de los atajos se nombran añadiendo *TECLA* a la letra, es decir *TECLAG* se refiere a la letra g en el teclado. Las teclas pueden ser combinadas con los modificadores *SHIFT*, *CTRL* y/o *ALT*. Generalmente para las teclas modificadas el sufijo *TECLA* se descarta, por ejemplo *CTRL-W* o *SHIFT-ALT-A*.
- *NUM0* a *NUM9*, *NUM+* y así sucesivamente, se refiere a las teclas ubicadas en el teclado numérico. Generalmente *NumLock* debería ser activada.
- En el caso de otras teclas, se les refiere usando sus nombres, tal como *ESC*, *TAB*, *F1* a *F12*.
- Otras teclas especiales para tener en cuenta son las teclas de dirección, *ARRIBA*, *ABAJO*, *IZQUIERDA* y *DERECHA*.

Dado que Blender hace un uso tan extensivo del ratón y el teclado, entre los usuarios se ha hecho popular una "regla de oro": mantener una mano en el ratón y la otra en el teclado! Si normalmente usa un teclado que es considerablemente diferente de la distribución de teclas inglesa, podría ir pensando en cambiar a uno con ésta o la distribución americana para el trabajo con Blender. Las teclas de mayor uso están agrupadas de tal manera que pueden ser alcanzadas con la mano izquierda en posición estándar (dedo índice sobre *TECLAF*) en la distribución de teclado inglesa. Se asume que se usa el ratón con la mano derecha.

5.2 El sistema de ventanas

5.2.1 La escena por defecto de Blender.

La escena por defecto de Blender. muestra la vista de pantalla que debe obtenerse al haber iniciado Blender(excepto por las flechas y el texto añadido). Por defecto está separada en tres ventanas: el menú principal en la parte alta, la vista 3D grande y la Ventana de Botones abajo.

La mayoría de las ventanas tiene un encabezado (la franja con un fondo gris claro que contiene botones con iconos - por esto se le llamará, también, *Barra de Herramientas*); si está presente, el encabezado puede ubicarse en la parte alta (como en la Ventana de Botones) o en la parte baja (como en la vista 3D) del área de una ventana. Si se mueve el ratón sobre una ventana, es posible notar que su encabezado cambia a un gris más claro. Esto significa que ésta está "enfocada"; todos los atajos de teclado que se presionen afectarán ahora el contenido de esta ventana. Es posible personalizar el sistema de ventanas de Blender para satisfacer las necesidades y gustos del usuario. Una ventana nueva puede ser creada fraccionando una existente. Esto se hace "enfocando" la ventana que se quiere fraccionar (poner el puntero del ratón dentro de esta), hacer click en el borde con BCR o BDR y seleccionar *Split Area* (*El menú de Split para crear ventanas nuevas.*). Ahora es posible ubicar el nuevo borde haciendo click con BIR, o cancelar el fraccionamiento presionando la tecla ESC. La nueva ventana será un clon de la ventana que fue fraccionada, pero ahora puede ser convertida a una ventana de otro tipo, o hacer que muestre la escena desde otro punto de vista.

5.2.2 Elementos de la Interface

Las etiquetas en los botones de la interface, los elementos de menú y, en general, cualquier texto mostrado en la pantalla se resalta en este libro de esta manera.

El menú de Split para crear ventanas nuevas.

Un nuevo borde vertical es creado al escoger *Split Area* en un borde horizontal y viceversa.

Cada ventana puede ser redimensionada arrastrando un borde con el BIR. Para reducir el número de ventanas, se hace click en un borde entre dos ventanas con BCR o BDR y se escoge *Join Areas*. La ventana resultante recibe las propiedades de la ventana enfocada previamente. Para ajustar la posición de un encabezado se hace click con el BDR sobre el encabezado y se escoge *Top* o *Bottom*. También es posible ocultar el encabezado seleccionando *No Header*, pero esta opción sólo se recomienda si todos los atajos de teclado relativos se conocen. Un encabezado oculto se puede volver a mostrar haciendo click en el borde de la ventana con BCR o BDR y seleccionando *Add Header*.

5.2.3 Tipos de ventanas

El marco de cada ventana puede contener diferentes tipos y conjuntos de datos, dependiendo de lo que se está haciendo. Estos pueden incluir modelos 3D, animación, materiales de la superficie, scripts Python, y así sucesivamente. Se puede seleccionar el tipo de cada ventana haciendo click en el botón de más a la izquierda en su encabezado con el BIR (*El menú de selección del tipo de ventana.*).

El menú de selección del tipo de ventana. Se explicarán las funciones y el uso de cada tipo de ventana a lo largo del libro. Por ahora sólo se prestará atención a los tres tipos de ventana que se encuentran en la escena por defecto de Blender:

· *Vista 3D*

Provee una vista gráfica de la escena en la cual se está trabajando. Es posible ver la escena desde cualquier ángulo con una variedad de opciones; para información detallada, véase *interface_3d*.

Tener varias vistas en la misma pantalla es útil si se desea observar el efecto de los cambios desde distintas perspectivas al mismo tiempo.

· *Buttons Window*(*Ventana de Botones*)

Contiene la mayoría de las herramientas para editar objetos, superficies, texturas, luces y mucho más. Esta ventana se usa constantemente si los atajos de teclado no se saben de memoria. Por supuesto, es posible tener más de una ventana de estas, cada una con un conjunto de herramientas diferente.

5.2.4 Contextos, Paneles y Botones

Los botones de Blender son mucho más emocionantes que los de la mayoría de otras interfaces de usuario y se volvieron aún mejores en la versión 2.30. Esto es mayormente debido al hecho de que son basados en vectores y dibujados en Opongo, lo que los hace elegantes y permite hacer zoom en ellos. Los botones se hallan agrupados mayormente en la Ventana de Botones. Desde Blender 2.3 la Ventana de Botones muestra *seis* contextos principales, los cuales se pueden seleccionar por medio de alguno de los botones en el primer grupo en el encabezado (*Contextos y Sub-Contextos*), a su vez cada uno de estos puede ser subdividido en un número variable de sub-contextos, a los cuales se puede acceder a través de los botones del segundo grupo en el encabezado (*Contextos y Sub-Contextos*):

Contextos y Sub-Contextos

- *Lógica* - atajo F4
- *Scripts* - sin atajo
- *Shading* - atajo F5

- *Lámpara* - sin atajo
- *Material* - sin atajo
- *Textura* - atajo F6
- *Radiosidad* - sin atajo
- *Mundo* - atajo F8
- *Objeto* - atajo F7
- *Edición* - atajo F9
- *Escena* - atajo F10
- *Renderizado* - sin atajo
- *Animación/Reproducción* - sin atajo
- *Sonido* - sin atajo

Usualmente, una vez el contexto ha sido seleccionado por el usuario, el subcontexto es determinado por Blender basándose en el objeto activo. Por ejemplo, en el contexto "Shading", si una lámpara está seleccionada, el subcontexto muestra los Botones de Lámpara, si una malla u otro objeto renderizable está seleccionado, entonces el subcontexto activo vendrá a ser el de Botones de Material y si una cámara está seleccionada el subcontexto activo es Mundo. La novedad más notable en la interface es probablemente la presencia de *Paneles* para agrupar lógicamente los botones. Cada panel es del mismo tamaño. Pueden ser movidos haciendo click con el BIR en su encabezado y arrastrando. Los paneles pueden ser alineados haciendo click con el BDR en la ventana de botones y escogiendo la distribución deseada en el menú que aparece (*Menu de la Ventana de Botones.*).

5.3 Menú de la Ventana de Botones.

La RR desplaza los paneles en su dirección de alineamiento, CTRL-RR y CTRL-BCR hacen zoom de acercamiento y alejamiento. Paneles individuales pueden ser colapsados/expandidos haciendo click con el BIR en el triángulo ubicado a la izquierda de su encabezado. Los paneles particularmente complejos se organizan en *Fichas*. Haciendo click con el BIR en una ficha en el encabezado del panel cambian los botones mostrados (*Paneles con fichas*). Las fichas se pueden "desligar" de un panel para formar paneles independientes haciendo click sostenido con el BIR en su encabezado y arrastrándolas fuera. De manera similar, paneles separados se pueden convertir en un solo panel con fichas soltando el encabezado de un panel sobre otro.

5.3.1 Panel con fichas.

Como un último elemento de interface, hay varios tipos de botones que se disponen en las fichas de los paneles:

- **Botones de Operación.**

Estos son botones que efectúan una operación cuando se les hace click (con el BIR, como todos los botones). Pueden ser identificados por el color pardusco que tienen en el esquema visual por defecto de Blender. (*Un botón de operación*).

- **Botón Alternante.**

Los botones alternantes vienen en varios tamaños y colores (*Botones alternantes*). Los colores verde, violeta y gris no modifican la función, sólo ayudan a facilitar la agrupación visual y reconocer el contenido de la interface más rápidamente. Hacer click en un botón de estos no genera acción alguna, sólo alterna un estado entre "on" y "off".

Botones alternantes

• **Botones de Radio.**

Los botones de radio son grupos particulares de botones alternantes mutuamente excluyentes. Tan sólo uno de estos botones puede estar "on" en un momento dado.

• **Botones Numéricos.**

Los botones numéricos (*Botones numéricos*) pueden ser identificados por sus títulos, que contienen ":" seguido por un número. Los botones numéricos son manipulados de varias formas:

Para incrementar el valor, se hace click con el BIR en la parte derecha del botón, donde se observa un triángulo pequeño; para decrementarlo, se hace click en la parte izquierda del botón, donde se encuentra otro triángulo.

Botones numéricos

Es posible ingresar un valor usando el teclado si se sostiene la tecla SHIFT y se hace click con el BIR. Presionando la combinación SHIFT-RETROCESO se elimina el valor contenido en el botón; con SHIFT-IZQUIERDA se mueve el cursor al inicio; y SHIFT-DERECHA mueve el cursor al final. Presionando ESC se restaura el valor original. Algunos botones de número contienen un deslizador y no sólo un número con triángulos laterales. El mismo método de operación se aplica para estos, excepto que el click con el BIR se debe efectuar en la parte izquierda o derecha del deslizador, en tanto que hacer click en la etiqueta o el número automáticamente iniciará el modo de entrada por teclado.

• **Botones de Menú.**

Los botones de menú se usan para escoger de una lista de elementos creada dinámicamente. Los botones de menú son usados principalmente para enlazar "DataBlocks" entre sí. (DataBlocks son estructuras como Mallas, Objetos, Materiales, Texturas y otros; al enlazar un material a un objeto, éste queda asignado). Un ejemplo de este tipo de bloque de botones se muestra en *Botones de enlace de DataBlocks*. El primer botón (con los diminutos triángulos apuntando arriba y abajo) abre un menú que permite seleccionar el DataBlock al cual efectuar el enlace, esto se hace sosteniendo oprimido el BIR y liberándolo cuando el puntero se halle sobre el elemento deseado. El segundo botón muestra el tipo y nombre del DataBlock enlazado y permite editar su nombre después de hacer click con el BIR. El botón con la "X" elimina el enlace, el botón con el "carro" genera un nombre automáticamente para el DataBlock y el botón con la "F" especifica que el DataBlock deberá ser almacenado en el archivo aún si no está siendo usado (no tiene enlace alguno).

Objetos sin enlace alguno: Los datos sin enlace sólo se pierden hasta que se cierra Blender. Esta es una poderosa característica que permite deshacer errores. Si un objeto es eliminado el material que tiene asignado pasa a estar sin enlace, pero aún existe! Solo se debe re-enlazar a otro objeto o presionar el botón "F".

Botones de enlace de DataBlocks

5.4 Caja de Herramientas (ToolBox)

Al presionar la tecla ESPACIO en la Vista 3D, o sostener el BIR o el BDR con el ratón quieto durante más de medio segundo se abre la Caja de Herramientas. Esta contiene 6 contextos principales dispuestos en dos líneas, cada uno de los cuales abre menús y submenús. Tres de estos contextos abren los mismos tres menús presentes en el encabezado de la vista 3D; los otros tres, *Add* permite añadir objetos nuevos a la escena mientras *Edit* y *Transform* muestran todas las

operaciones que se pueden efectuar sobre el(los) objetos seleccionado(s). (*La Caja de Herramientas*).

La Caja de Herramientas

5.5 Pantallas (Screens)

La flexibilidad de las ventanas de Blender permite crear entornos de trabajo personalizados para diferentes tareas, tales como modelado, animación y creación de scripts. A menudo resulta útil intercambiar rápidamente entre diferentes entornos dentro del mismo archivo. Esto es posible al crear varias Pantallas: Todos los cambios a las ventanas que se describen en *El sistema de ventanas* y *Tipos de ventanas* se almacenan en una pantalla, de modo que si las ventanas de una pantalla son modificadas, las otras no resultarán afectadas en lo absoluto.

5.6 Escenas

También es posible tener varias escenas dentro del mismo archivo de Blender. Las escenas pueden usar objetos de otras o ser completamente independientes entre sí. Con el botón de menú *SCE*, ubicado en el encabezado de la Ventana de Preferencias de Usuario, se puede crear o seleccionar una escena (Selectores de Pantalla y Escena). *Cuando se crea una escena nueva, es posible escoger entre cuatro opciones para controlar su contenido:*

- *Empty* Crea una escena vacía.
- *Link Objects* crea la escena nueva con el mismo contenido de la escena seleccionada actualmente.

Los cambios en una escena afectarán igualmente a la otra.

- *Link ObData* crea la nueva escena basada en la escena seleccionada actualmente, con enlaces a las mismas mallas, materiales, etc. Esto significa que la posición de los objetos puede ser cambiada,

junto con otras propiedades, pero modificaciones de las mallas, materiales, etc. afectarán igualmente la otra escena, a menos que se indique manualmente la creación de copias con usuario único (single-user copies).

· *Full Copy* crea una escena completamente independiente con copias del contenido de la escena seleccionada actualmente.

5.7 Navegando en el Entorno 3D

Blender permite trabajar en un espacio tridimensional, pero las pantallas de nuestros monitores son sólo bi-dimensionales. Para poder trabajar en tres dimensiones, debe ser capaz de cambiar tanto tu punto de vista como la dirección en la que ves la escena. Esto es posible en todas las vistas 3D. Aunque vamos a describir la ventana de vista en 3D, la mayoría de las ventanas no-3D usan una serie de funciones equivalentes, por ejemplo es incluso posible arrastrar y escalar una Ventana de Botones y sus Paneles.

La dirección de la vista (rotando)

Blender ofrece tres direcciones de vista por defecto: Lateral, Frontal, and Superior. Como Blender usa un sistema de coordenadas "de mano derecha" con el eje Z apuntando hacia arriba, la vista "Lateral" corresponde a una mirada desde el eje X axis, en la dirección negativa; la vista "Frontal" es desde el eje Y; y la "Superior" desde el eje Z. Puedes seleccionar la dirección de la vista para una ventana 3D con las entradas de Menú *Vista* (*El menú de vistas de una ventana 3D.*) o presionando los atajos de teclado NUM3 para "Lateral", NUM1 para "Frontal", y NUM7 para "Superior".

El menú de vistas de una ventana 3D.

Atajos de teclado: Recuerde que la mayoría de los atajos de teclado afectan a la ventana activa, así que comprueba que el cursor del ratón está en el área donde quieres trabajar antes de usar los atajos de teclado.

Aparte de estas tres direcciones por defecto, la vista puede ser girada con cualquier ángulo que necesites. Haz clic y arrastre MMB en el área de la ventana: si comienzas en el medio de la ventana y te mueves hacia arriba y hacia abajo o hacia la derecha y la izquierda, la vista gira alrededor del centro de la ventana. Si comienzas en el borde y no te mueves hacia el centro, puede girar alrededor de tu eje de vista. Juega un poco con esta función hasta que controles su manejo. Para cambiar el ángulo de vista en pasos sucesivos, usa NUM8 y NUM2, que corresponden al arrastre vertical. MMB o usa NUM4 y NUM6, que corresponden al arrastre horizontal MMB.

Trasladando y Escalando la Vista

Para trasladar la vista, pulsa SHIFT y arrastra MMB en la Ventana 3D. Para hacerlo en pasos sucesivos, usa las combinaciones de teclas CTRL-NUM8, CTRL-NUM2, CTRL-NUM4 y CTRL-NUM6 como con las rotaciones. Puede escalar la vista (acercarte y alejarte) pulsando CTRL y arrastrando MMB. Los atajos de teclado son NUM+ y NUM- . Los sub-menús *View>>Viewport Navigation* contienen también estas funciones.

Modo de Dibujado

Dependiendo de la velocidad de tu ordenador, de la complejidad de tu escena y del tipo de trabajo que esté haciendo, puedes alternar entre diferentes modos de dibujado:

- *Texturizado* - Intenta dibujarlo todo tan completamente como sea posible, aunque no es aún una alternativa al renderizado. Nota que si no hay ninguna luz en tu escena todo saldrá negro.

- *Sombreado* - Dibuja superficies sólidas incluyendo el cálculo de iluminación. Como en el modo Texturado, no verás nada sin luces.
- *Sólido* - Las Superficies se dibujan como Sólidos, pero también funciona sin luces.
- *Enrejado* - Los objetos sólo consisten en líneas que hacen reconocibles las formas. Este es el modo de dibujado por defecto.
- *Caja de Límites* - Los objetos no son dibujados en absoluto; en vez de eso, este modo enseña sólo las cajas rectangulares que corresponden al tamaño y a la forma de cada objeto.

El modo de dibujado puede se elegido con el botón de menú apropiado en la cabecera (*Un botón de modo de dibujado de una vista 3D.*) o con atajos de teclado: ZKEY cambia entre Enrejado y Sólido, SHIFT-Z cambia entre Enrejado y Sombreado.

Un botón de modo de dibujado de una vista 3D.

Vista Local

En vista local, sólo los objetos seleccionados son mostrados, lo que puede facilitar la edición en escenas complejas. Para entrar en vista local, primero selecciona los objetos que quiere (ver *mesh_modelling_objectmode_selecting*) y después usa la entrada de menú; *Vista>>Vista Local* usa la entrada de menú *Vista>>Vista Global* para volver a la vista global. (*El menú de vistas de una ventana 3D.*) El atajo es NUM/, que cambia entre vista Local y Global.

El sistema de Capas

Las escenas en 3D a menudo se hacen mucho más confusas a medida que aumenta su complejidad. Para mantenerlas bajo control, los objetos pueden ser agrupados en "capas", para que sólo las capas que elijas sean mostradas en cualquier momento. Las capas de 3D difieren de las capas que puede conocer de las aplicaciones gráficas en 2D: no tienen influencia en el orden de dibujado y están ahí (a excepción de algunas funciones especiales) únicamente para ofrecer al modelador un mejor

control de las vistas. Blender utiliza 20 capas; usted elige las que se muestran con los pequeños botones sin título de la cabecera (*Los botones de Capas de una vista 3D.*). Para seleccionar sólo una capa, haz clic en el botón apropiado con LMB; para seleccionar más de una, presiona MAYÚS mientras haces clic.

Los botones de Capas de una vista 3D.

Para seleccionar capas con el teclado, presiona desde la tecla 1KEY hasta la tecla 0KEY (en el área principal del teclado) para las capas de 1 a 10 (la fila superior de botones), y ALT-1 hasta ALT-0 para las capas 11 a 20 (la fila inferior). La tecla MAYÚS para selecciones múltiples también funciona con estos atajos. Por defecto, el botón de bloquear a la derecha del bloque de capas está activado; esto significa que los cambios a la visibilidad de las capas afecta a todas las ventanas 3D. Para seleccionar ciertas capas sólo en una ventana, desactiva primero el bloqueo. Para mover los objetos seleccionados a una capa diferente, presiona MKEY, seleccione la capa que quieres en el diálogo emergente, y finalmente haz clic en OK .

Guardando archivos

Guardar archivos es como cargarlos: Cuando presionas F2, la ventana activa cambia temporalmente a una ventana de selección de archivos, como se muestra en la *Ventana de Selección de Archivos - guardando..* Haga clic en la ventana de edición inferior para escribir un nombre de archivo. Si no lo acaba con ".blend," la extensión es añadida automáticamente.

Después presiona ENTER para guardar el archivo. Si ya existe un archivo con el mismo nombre, tendrá que confirmar que quieres guardar el archivo en la pregunta de sobre-escribir. Ventana de Selección de Archivos - guardando.

El diálogo de salvar tiene una pequeña cualidad añadida para ayudar a crear múltiples versiones de tu trabajo: Presionando NUM+ o NUM- incrementará o disminuirá un número contenido en el

nombre del archivo. Para simplemente salvar sobre el archivo actualmente cargado evitando el diálogo de salvar, presiona CTRL-W en vez de F2 y sólo tendrás que confirmar la pregunta.

5.8 Renderizando

Esta sección ofrece sólo un repaso rápido de lo que necesitarás para renderizar tu escena.

Encontrará una descripción detallada de todas las opciones en *chapter_rendering*. Las Opciones de Render están en el contexto *Escena* y en el sub-contexto *Botones de Renderizado (Opciones de Renderizado en los Botones de Render.)* al que se accede haciendo clic en , o presionando F10.

Opciones de Renderizado en los Botones de Render.

Por ahora sólo nos interesa el panel *Formato*. El tamaño (número de pixeles horizontales y verticales) y el formato de archivo de la imagen a crear se eligen aquí. Se puede fijar el tamaño usando los botones *SizeX* y *SizeY*. Haciendo clic en la caja de selección de debajo (en *Opciones de Renderizado en los Botones de Render.*, está seleccionado "Targa") abre un menú con todos los formatos de salida disponibles para imágenes y animaciones. Para imágenes fijas se puede elegir Jpeg, por ejemplo. Ahora que las opciones están completas, la escena puede ser renderizada presionando el botón *RENDER* en el panel *Render* o presionando F12. Dependiendo de la complejidad de la escena, esto puede tardar normalmente entre unos pocos segundos y varios minutos, y el progreso es mostrado en una ventana separada. Si la escena contiene una animación, sólo se renderiza el fotograma actual (Para renderizar la animación completa, ver *rendering_animation*). Si no se ve nada en la vista renderizada, asegúrese de que tu escena está construida convenientemente. ¿Tiene luz? Está la cámara situada correctamente, y apunta en la buena dirección? ¿Están visibles todas las capas que quieres renderizar?

Nota: Una imagen renderizada no se guarda automáticamente en el disco. Si está satisfecho con el renderizado, debe salvarlo presionando F3 y usando el diálogo de salvar tal como describimos en Section 3.3.2. La imagen es salvada en el formato que hayas seleccionado previamente en los botones de Display.

Extensiones de archivo: Blender no añade la extensión del formato automáticamente a los archivos de imagen por lo que se tiene que elegir el formato deseado.

Preferencias de usuario y Temas

Blender tiene unas pocas opciones que no son guardadas con cada archivo, sino que se aplican se aplican a todos los archivos de un usuario. Estas preferencias incluyen detalles de la interfaz del usuario, y propiedades del sistema como el ratón, las fuentes y los lenguajes.

Como las preferencias del usuario son raramente necesarias, están escondidas bajo el menú principal. Para hacerlas visibles, baja el borde de la ventana del menú (normalmente el borde superior en la pantalla). Las opciones están agrupadas en siete categorías que pueden ser seleccionadas con los botones violeta mostrados en la *Ventana de Preferencias del Usuario*. Ventana de Preferencias del Usuario.

Ya que la mayoría de los botones se explican por sí mismos o muestran una útil ayuda si mantienes el cursor parado sobre ellos, no vamos a describirlos en detalle aquí. En lugar de eso, te ofrecemos un repaso de las categorías de Preferencias:

- Vista & Controles

Las opciones concernientes a cómo debe reaccionar la entrefaz a las acciones del usuario, como qué método de rotación debe usarse en las vistas 3D. Aquí puedes activar también la emulación del ratón de 3 botones si el tuyo tiene sólo dos. MMB puede ser emulado con ALT-LMB.

- Métodos de Edición

Permite especificar los detalles del funcionamiento de ciertos comandos de edición como duplicar.

- Lenguaje y Fuentes

Seleccionar una fuente TrueType alternativa para mostrar en la interfaz, y elegir entre los lenguajes disponibles para la interfaz.

- Temas

Desde la versión 2.30 Blender permite la utilización de Temas para definir colores personalizados para la interfaz. Se puede crear y manejar temas desde aquí.

- Guardar Automáticamente

Puede configurar Guardar Automáticamente para tener una copia de emergencia en caso de que algo vaya mal. Estos archivos se llamarán Nombre.blend1, Nombre.blend2, etc.

- Sistema & OpenGL

Debe consultar esta sección si experimentas problemas con los gráficos o con la salida de audio, o si no dispones de un teclado numérico y quieres imitarlo (para portátiles). Además, aquí puede también configurar el esquema de luces para los modos de dibujado Solido y Sombreado.

- Directorios de Archivos

Escoger el directorio por defecto para varios diálogos de carga de archivos.

CAPITULO VI

COMPARATIVA AUTODESK 3D MAX VS BLENDER

6.1 Comparativa blender vs 3d max (referencia informática)

| Blender 3D | 3D MAX |
|--|---|
| <ul style="list-style-type: none">● Ocupa 9 Mb del disco rígido. | <ul style="list-style-type: none">● Ocupa 270 Mb del disco rígido. |
| <ul style="list-style-type: none">● Se distribuye como software libre desde el sitio oficial | <ul style="list-style-type: none">● Se vende por más de 3000 Dólares Norteamericanos. |

| | |
|--|---|
| <ul style="list-style-type: none">• Toda una comunidad de Desarrolladores lo modifica permanentemente, permitiendo que haya cambios cada semana, y cuando estos cambios son estables, se lanza una versión oficial. | <ul style="list-style-type: none">• Sólo LA EMPRESA PROPIETARIA puede modificar el código del programa y los lanzamientos de nuevas versiones, son decisiones de mercado, muchas veces dejan al soft inestable, o solo se le agregan plug-ins a la versión vieja con pocos cambios |
| <ul style="list-style-type: none">• Es sistema de Scripts es Python, al igual que muchas aplicaciones de LINUX, muchos desarrolladores ya conocen esta herramienta. | <ul style="list-style-type: none">• El sistema de MAXSCRIPT, es bastante simple, sin embargo se debe dedicarle tiempo para aprenderlo, y no sirve para otras aplicaciones. |
| <ul style="list-style-type: none">• Está disponible para casi todos los sistemas Operativos: LINUX, MS-WINDOWS, MACOS X, UNIX, FreeBSD, etc. | <ul style="list-style-type: none">• A pesar de que LA EMPRESA, posee productos para SGI (silicon graphics) basadas en UNIX, ya han manifestado que 3D MAX será un producto para PC y que correrá sobre plataformas de MICROSOFT. |

Tabla VIII. Comparativa blender vs 3d max (informática)

3D MAX

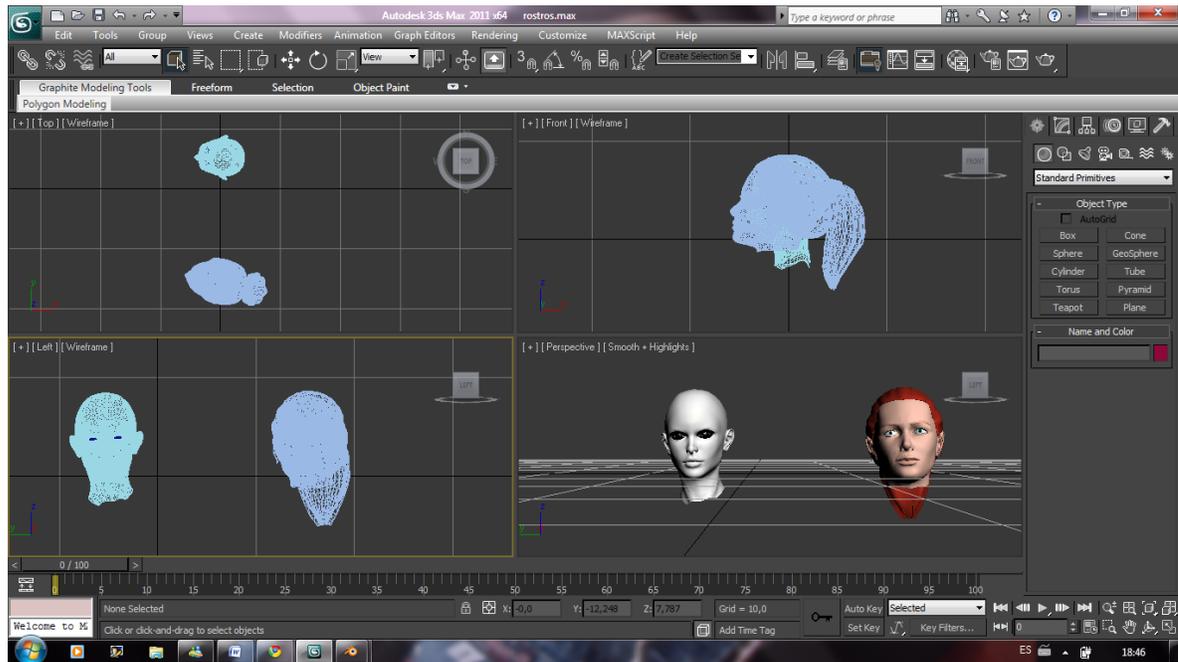


Fig. VI.43 Interfaz 3d Max con el modelado del rostro

Blender

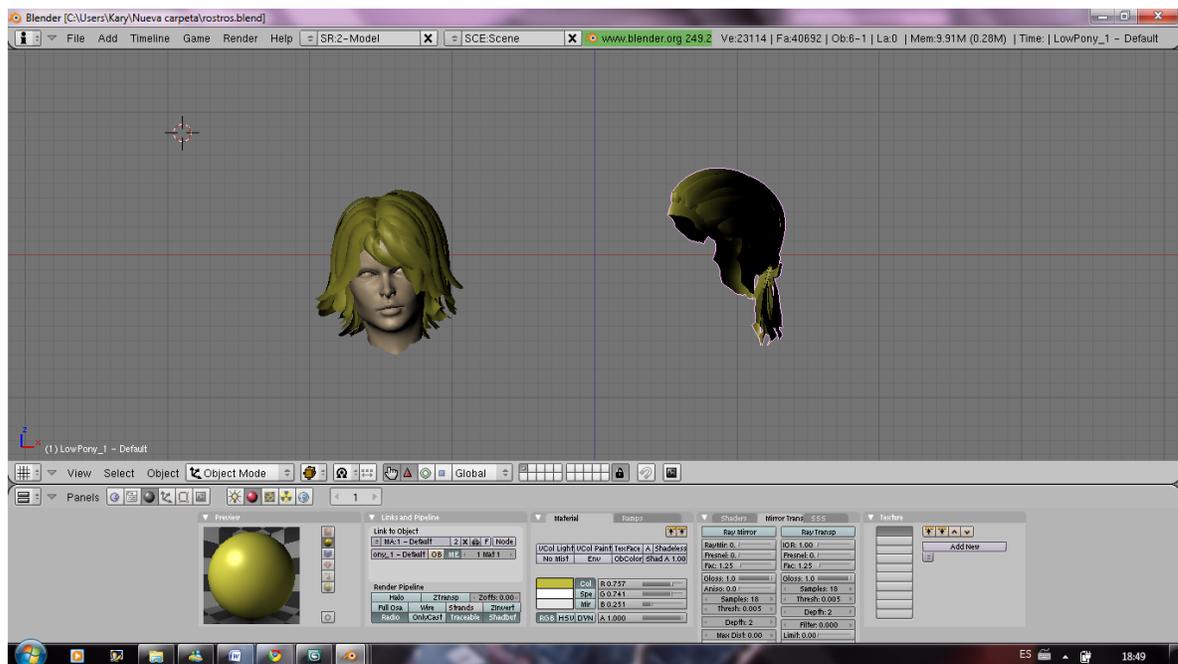


Fig. V.44 Interfaz Blender con el modelado del rostro

6.2 Detalles técnicos

| Paquete | Plataformas | De vídeo compatibles | Idioma de extensión | Precio con declaró Funcionalidad |
|----------------|--|---|------------------------|-------------------------------------|
| 3ds Max | Windows 32 / 64 bits | Direct X , OpenGL , la representación de software , Heidi | MAXScript , C ++ (SDK) | \$ 3.495 (febrero 2011) |
| Blender | Windows , OSX , Linux , FreeBSD , Irix , Solaris | Representación de Software , OpenGL | C , Python | Software libre. (GPLv2) |

Tabla IX. Comparativa detalles técnicos

6.3 Herramientas de Modelado

| Paquete | 3D Max | Blender |
|--------------|-----------|-----------|
| Nurbs | Si | Si |
| Patch | Si | Si |

| | | |
|-----------------------|-----------|-----------|
| Sub D | Si | Si |
| Polígono | Si | SI |
| Suaviza 3 gons | Si | Si |

Tabla X. Tabla herramientas de modelado

| | |
|--------|---------|
| 3d max | blender |
| 100 | 100 |

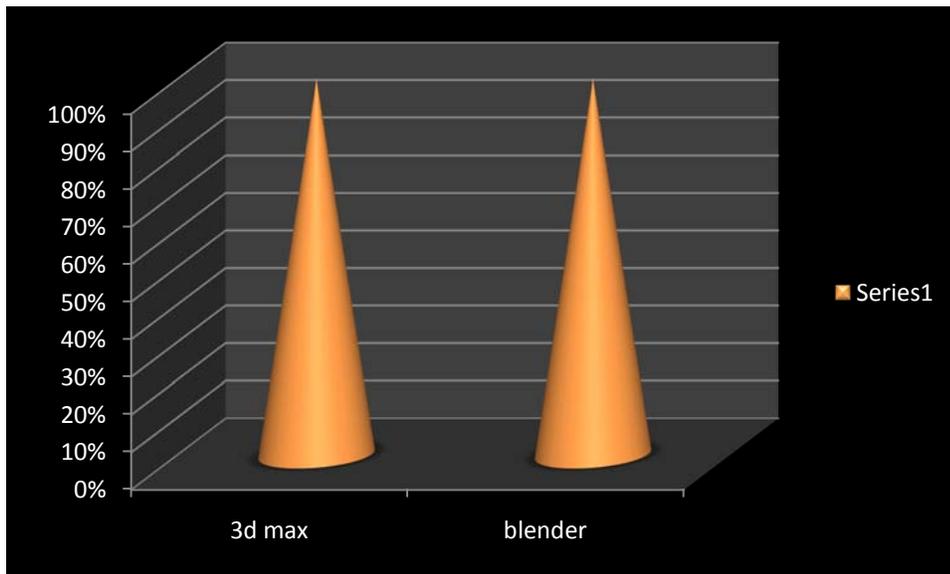


Tabla XI. Herramientas de modelado

6. 4 Viable Polígono Límite (3GHZ p4 (o equivalente), 1 GB de RAM media, polígonos de cuatro)

| Paquete | 3d Max | Blender |
|-----------|--------|---------|
| 10,000 | Si | Si |
| 50,000 | Si | Si |
| 100,000 | Si | Si |
| 500,000 | Si | Si |
| 1,000,000 | Si | Si |

Tabla XII. Tabla limite viable del polígono

| | |
|--------|---------|
| 3d max | blender |
| 100 | 100 |

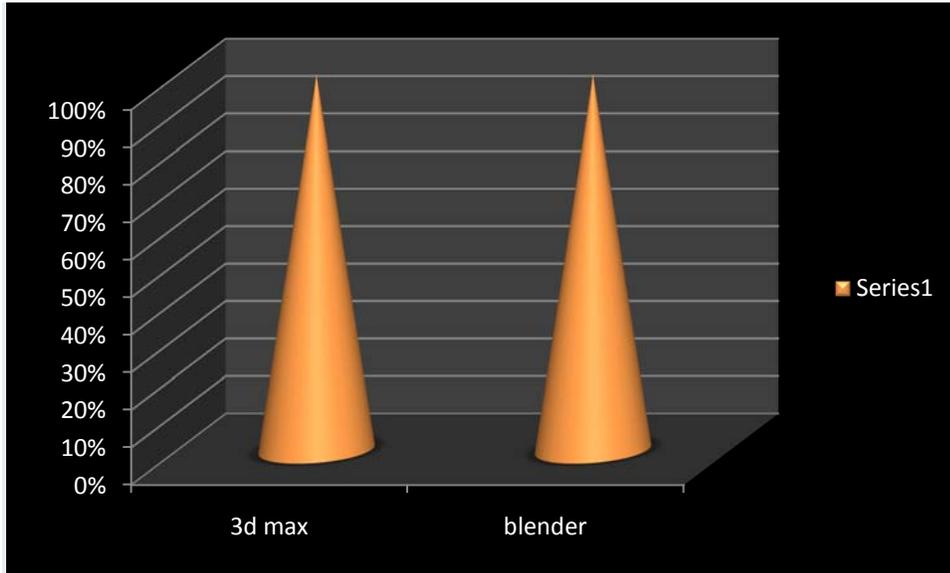


Tabla XIII. Límite viable del polígono

6.5 UV Unwrap

| Paquete | 3d Max | Blender |
|---------|--------|---------|
| Básico | Si | Si |
| Piel | Si | Si |
| LSCM | Si | Si |

| | | |
|---------------------------|----|----|
| ABF ++ | Si | Si |
| UV de Múltiples conjuntos | Si | Si |
| UV Subdividida | Si | Si |

Tabla XIV. Tabla UV Unwrap

| | |
|--------|---------|
| 3d max | blender |
| 100 | 100 |

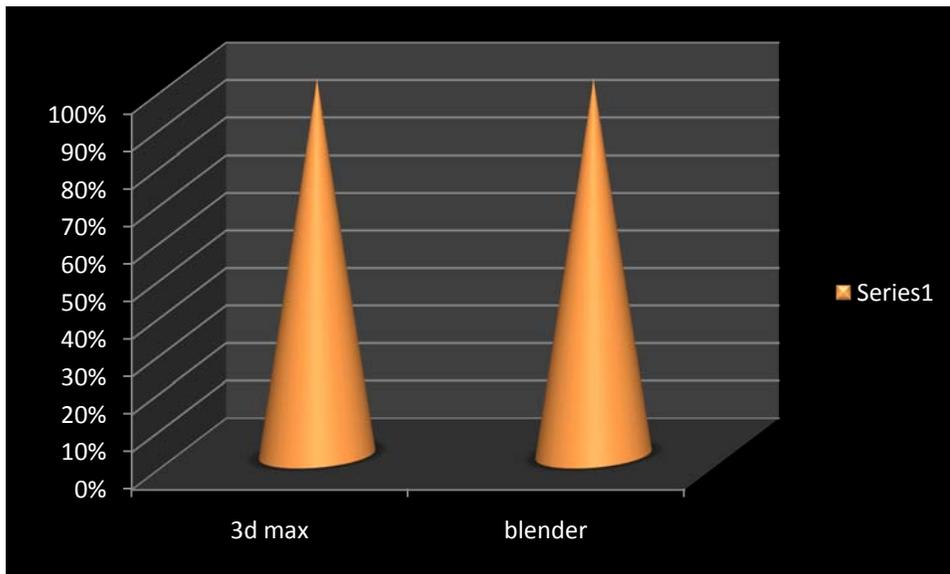


Tabla XV. Unrwrap

6.6 Texturizado y Pintura

| Paquete | 3d Max | Blender |
|--|---------------------|---------|
| Nodo base Texturizado | A través de plug-in | Si |
| 2D pintura | A través de plug-in | Si |
| 3D pintura | A través de plug-in | Si |
| Pintura simultáneamente en varios objetos | Si | Si |
| Capa base de pintura | No | No |
| Pintura al mismo tiempo materiales múltiples | No | No |
| Vertex pintura | Si | Si |

Tabla XVI. Tabla Texturizado y pintura

| | |
|--------|---------|
| 3d max | blender |
| 50% | 71,42% |

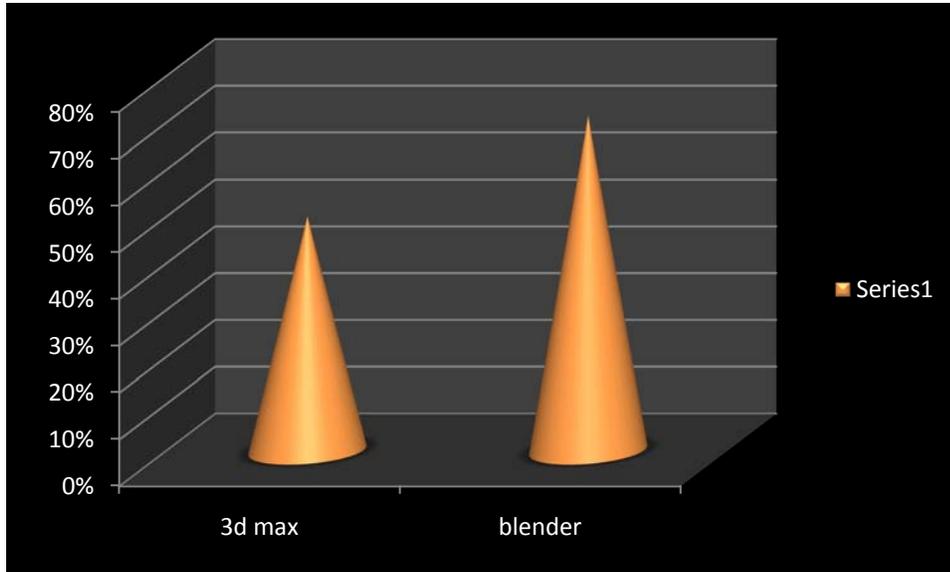


Tabla XVII. Texturizado y pintura

6.7 Animación y Simulación

| Paquete | 3d Max | Blender |
|---------------|---------------------|---------|
| Simple | Si | Si |
| Personaje | Si | Si |
| De partículas | Si | Si |
| Física | Si | Si |
| Softbody | Si | Si |
| Paño | Si | Si |
| Cabello | Si | Si |
| Líquido | A través de plug-in | Si |
| Humo | Si | Si |

| | | |
|----------|----------------------------|----------------------------|
| Llama | A través de plug-in | Si |
| Multitud | Si | A través de plug-in |

Tabla XVIII. Tabla animación y simulación

| | |
|--------|---------|
| 3d max | blender |
| 90,9 % | 95,45 % |

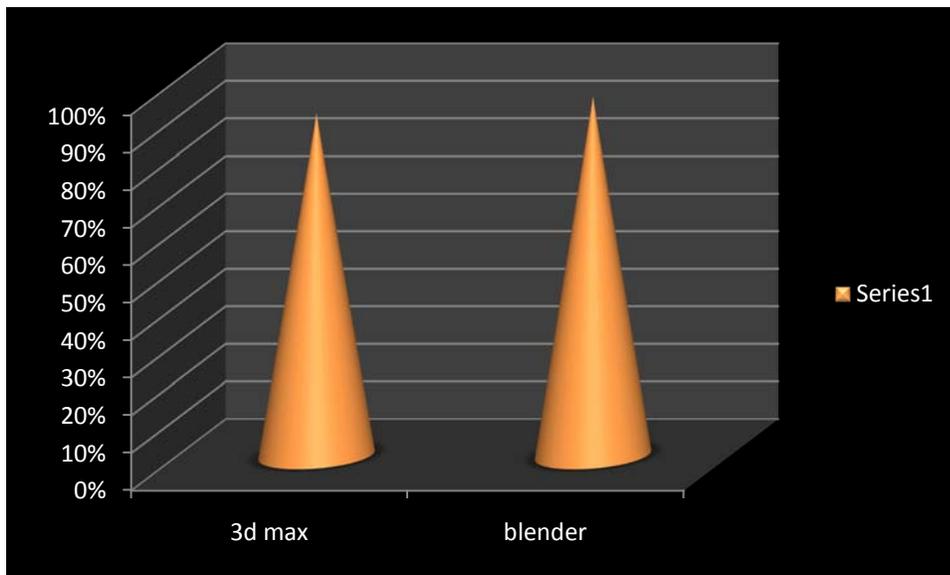


Tabla XIX. Animación y simulación

6.8 Prestación

| | | |
|-----------------------------|---|--|
| Paquete | 3d Max | Blender |
| Procesador Primario | Scanline , Mental Ray , trazadores de luz, Quicksilver , motor de radiosidad (Lightscape) | Blender |
| RIB Apoyo | A través de plug-in | A través de plug-in |
| Otros renderizado res | Arion Render , Brasil r / s, finalRender ,finalToon , Fryrender , Gelato , Ilustre! , Krakatoa ,MaxwellRenderer , VRay | Yafray , Mental Ray , Sunflow , Kerkythea ,Indigo , luxrender , Vray , POV-Ray |
| NET Render | Si | A través de plug-in |
| # Nodos de red | Unlimited | Unlimited |

Tabla XX. Prestación

6.9 Interfaz y personalización

| Paquete | 3d Max | Blender |
|------------------------|-----------|-----------|
| Diseño de pantalla | Si | Si |
| De telas | Si | Si |
| Mouse enlaces | Si | Si |
| Menús | Si | Si |
| Botones | Si | Si |
| Estantes | Si | No |
| Barras de herramientas | Si | Si |
| Vistas Personalizadas | No | Si |

Tabla XXI. Tabla interfaz y Personalización

| | |
|--------|---------|
| 3d max | blender |
| 87,5 | 87,5 |

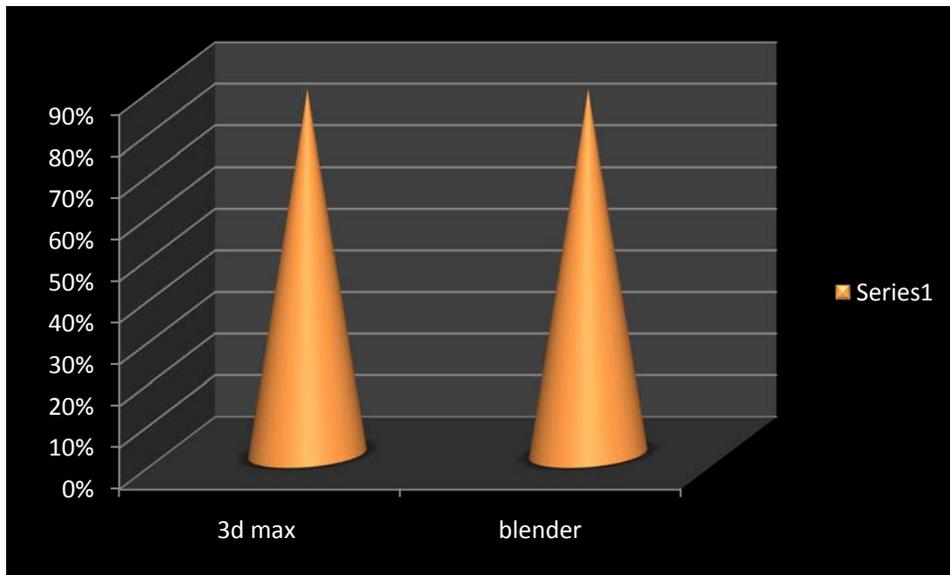


Tabla XXII. Interfaz y Personalización

6. 10 Formatos de archivo admitidos importar

| Paquetes | 3d Max | Blender |
|----------|--------|---------|
| OBJ | Si | Si |
| 3DS | Si | Si |

| | | |
|------|----|----|
| DXF | Si | Si |
| FBX | Si | No |
| LWO | No | Si |
| JPEG | Si | Si |
| TIFF | Si | Si |
| PSD | Si | Si |
| BMP | Si | Si |
| GIF | Si | Si |
| PNG | Si | Si |
| AI | Si | No |

Tabla XXIII. Tabla Archivos permitidos para importar

| | |
|--------|---------|
| 3d max | blender |
| 91,66 | 83,33 |

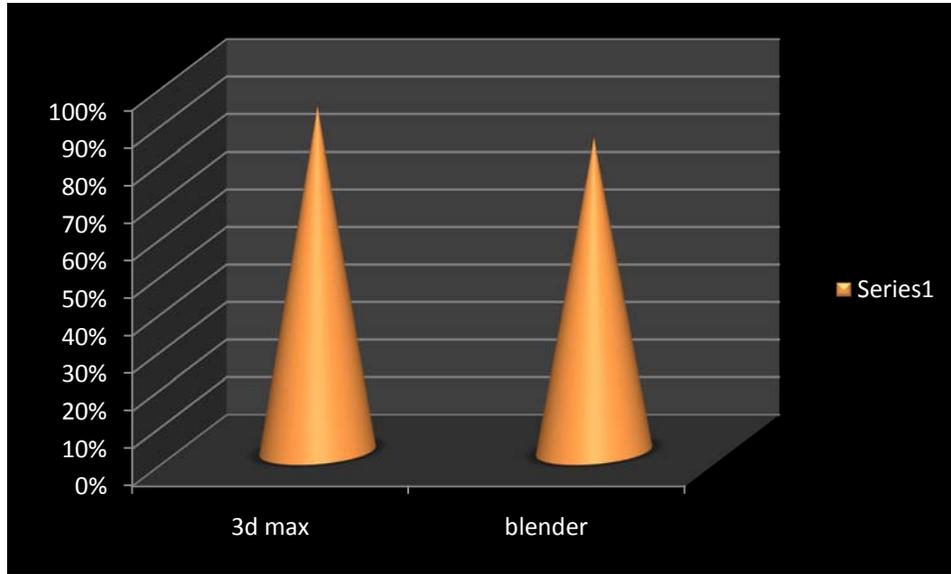


Tabla XXIV. Archivos permitidos para importar

6. 11 Formatos de archivo admitidos exportar

| Paquetes | 3d Max | Blender |
|----------|--------|---------|
| OBJ | Si | Si |
| 3DS | Si | Si |
| DXF | Si | Si |

| | | |
|------|------------------------|----|
| FBX | Si | No |
| LWO | A través de un plug-in | Si |
| JPEG | Si | Si |
| TIFF | Si | Si |
| PSD | Si | Si |
| BMP | Si | Si |
| GIF | Si | Si |
| PNG | Si | Si |
| AI | Si | No |

Tabla XXV. Tabla archivos permitidos para exportar

| | |
|--------|---------|
| 3d max | blender |
| 95,83 | 83,33 |

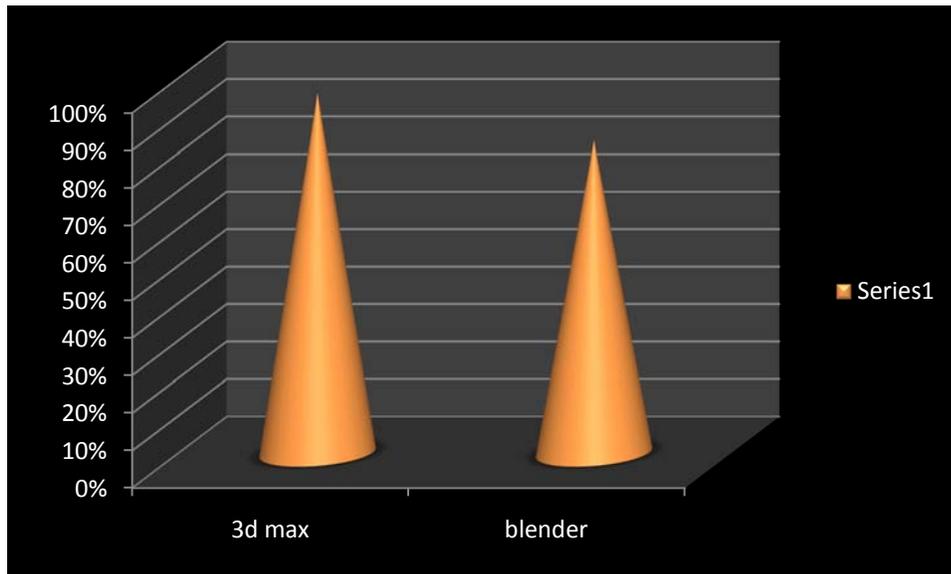


Tabla XXVI. Archivos permitidos para exportar

Las extensiones de archivo que tienen por defecto para grabar en estos programas son:

3D Max: .max

Blender: .blend

Siendo el .3ds una base fundamental para importar o exportar documentos entre sí.

Basado en parámetros de comparación 3D Max y Blender cumplen las mismas funciones, encontrando así este cuadro:

6. 12Cuadro comparativo Herramientas para modelado 3d

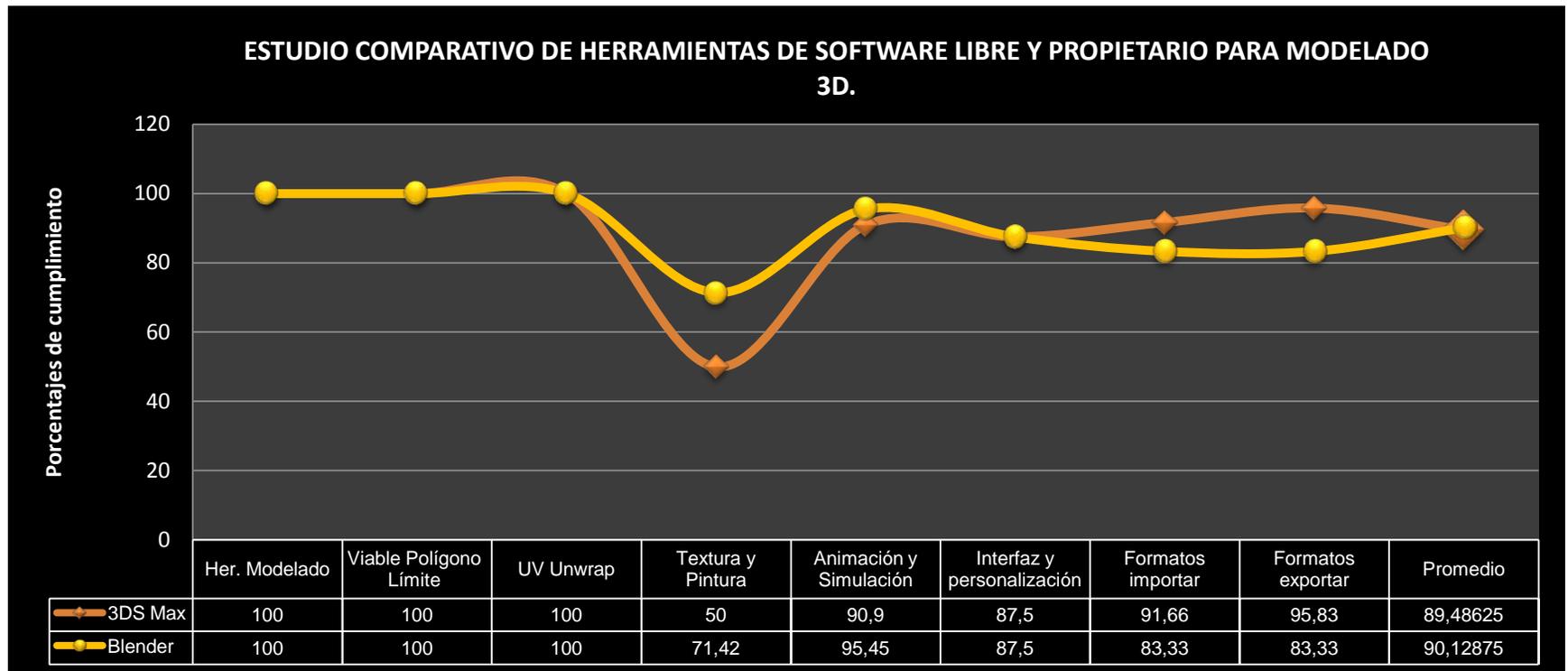


Tabla XXVII. Cuadro comparativo herramientas para modelado 3D

Manejo malla editable en blender y 3d max

Dentro de ambos programas se encuentra fácilmente los procesos necesarios para la edición de una malla editable.

3D Max

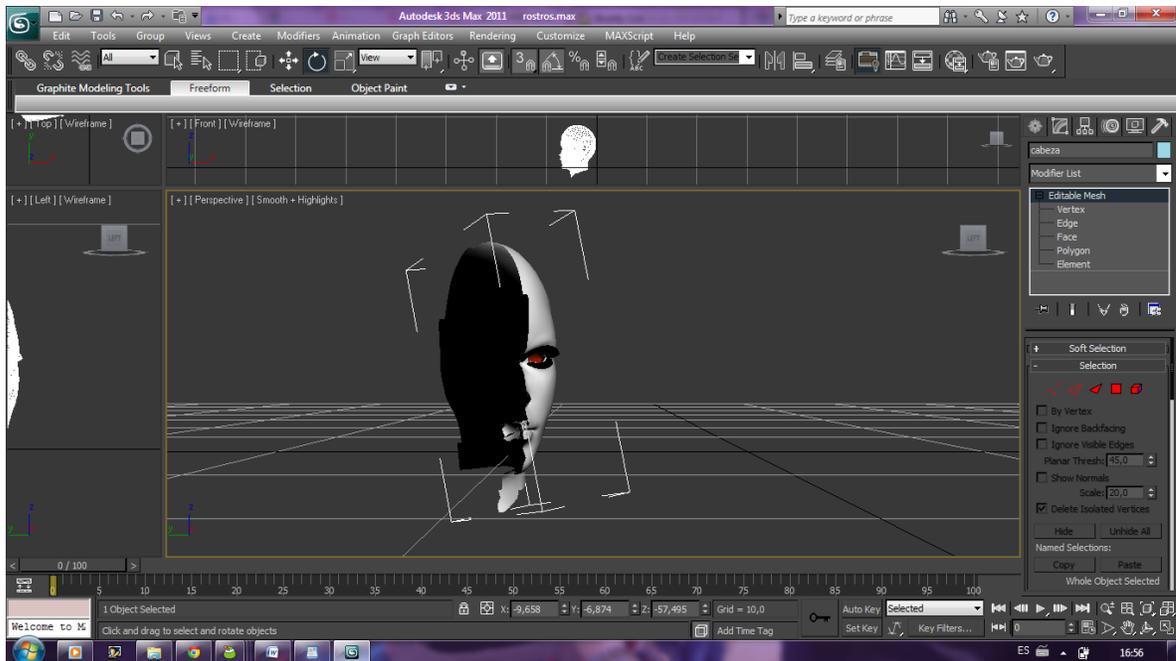


Fig. VI.45 Modelado rostro 3ds Max I

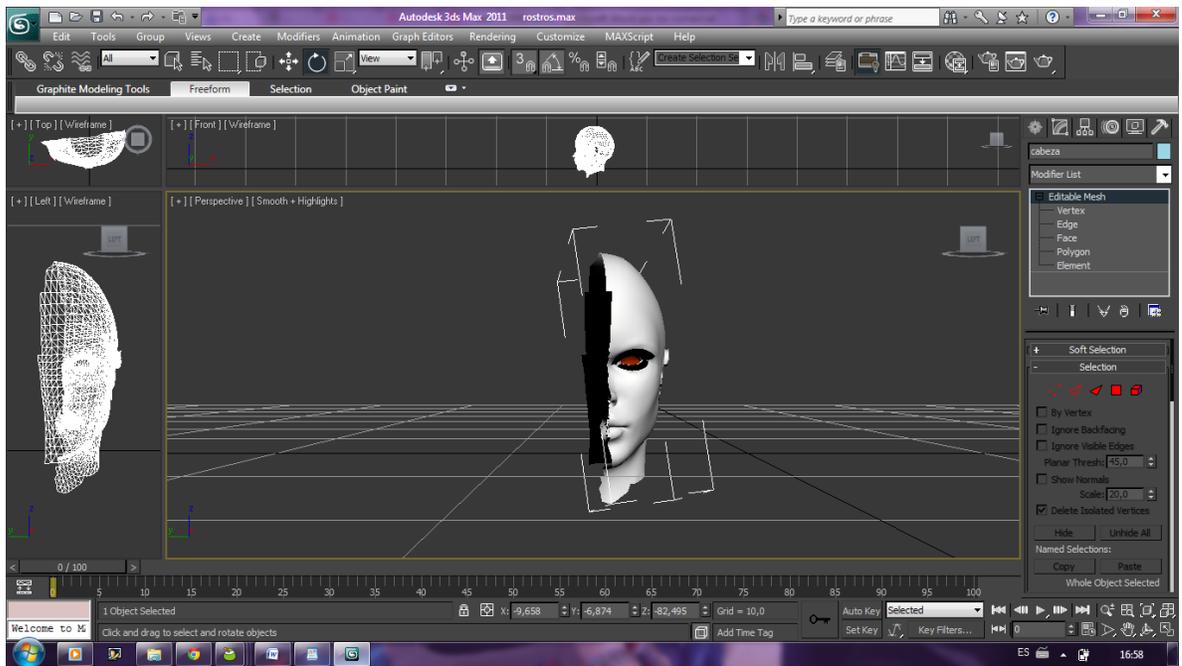


Fig. VI.46 Modelado rostro 3ds Max II

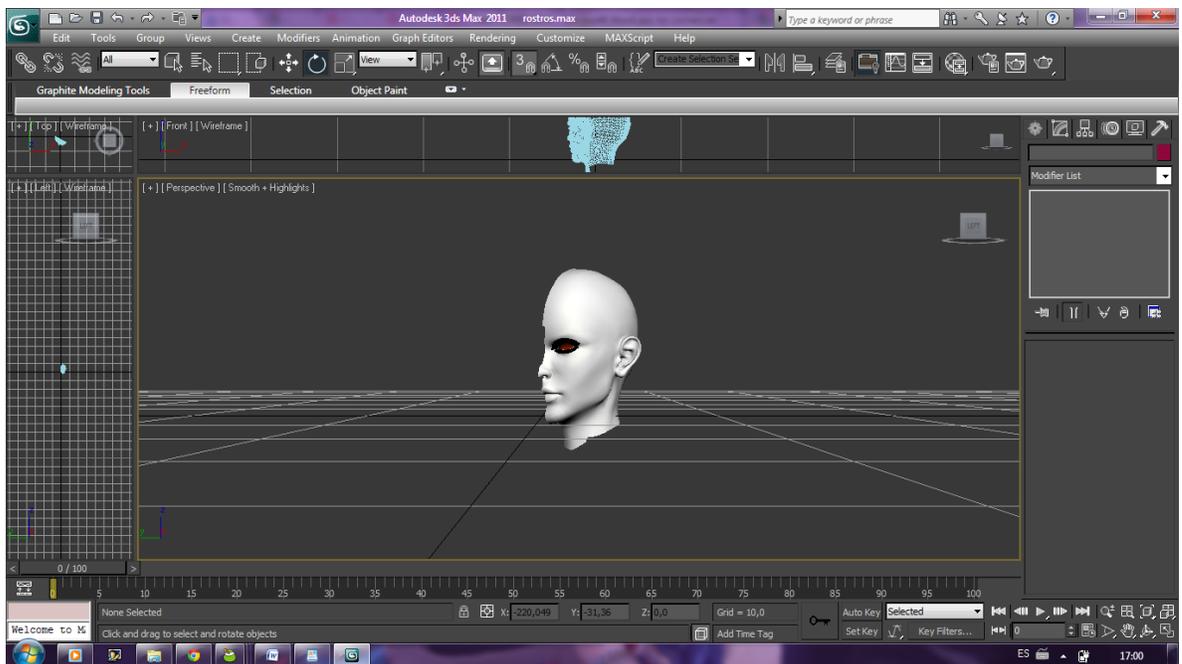


Fig. VI.47 Modelado rostro 3ds Max III

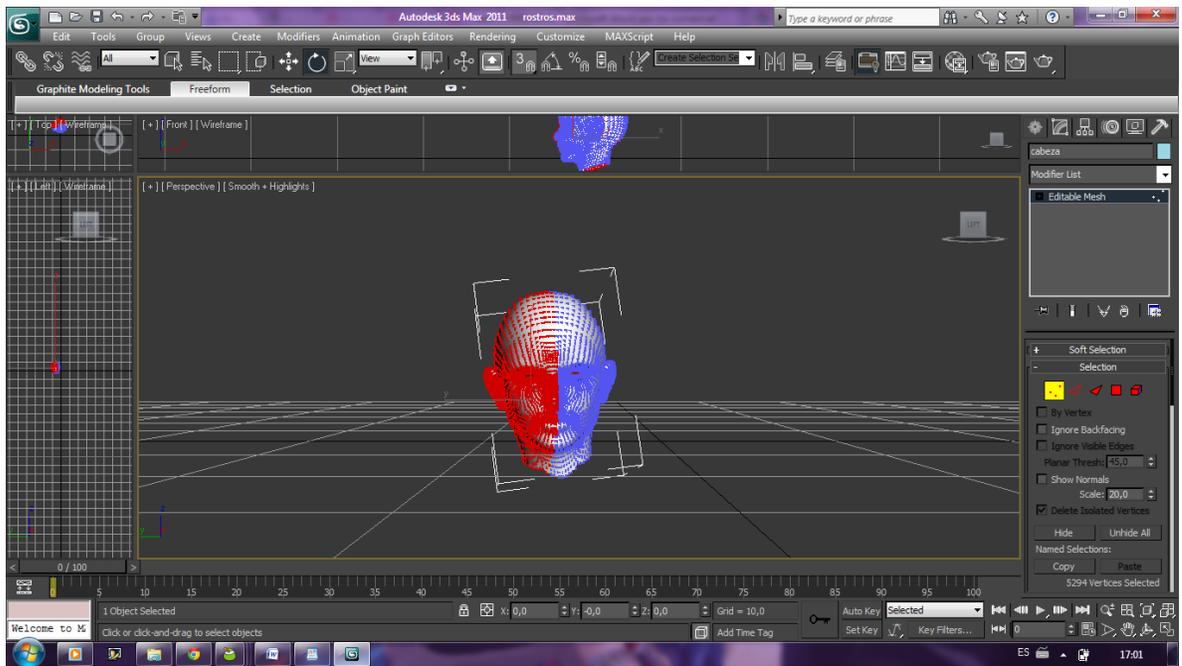


Fig. VI.48 Modelado rostro 3ds Max IV

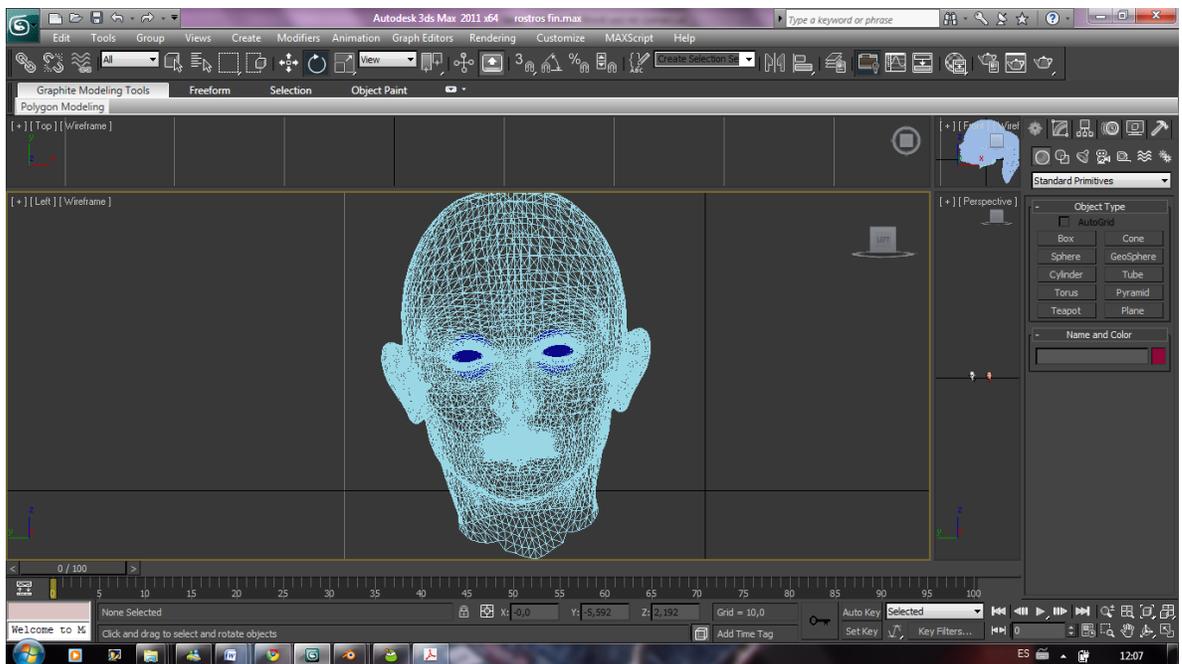


Fig. VI.49 Modelado rostro 3ds Max V

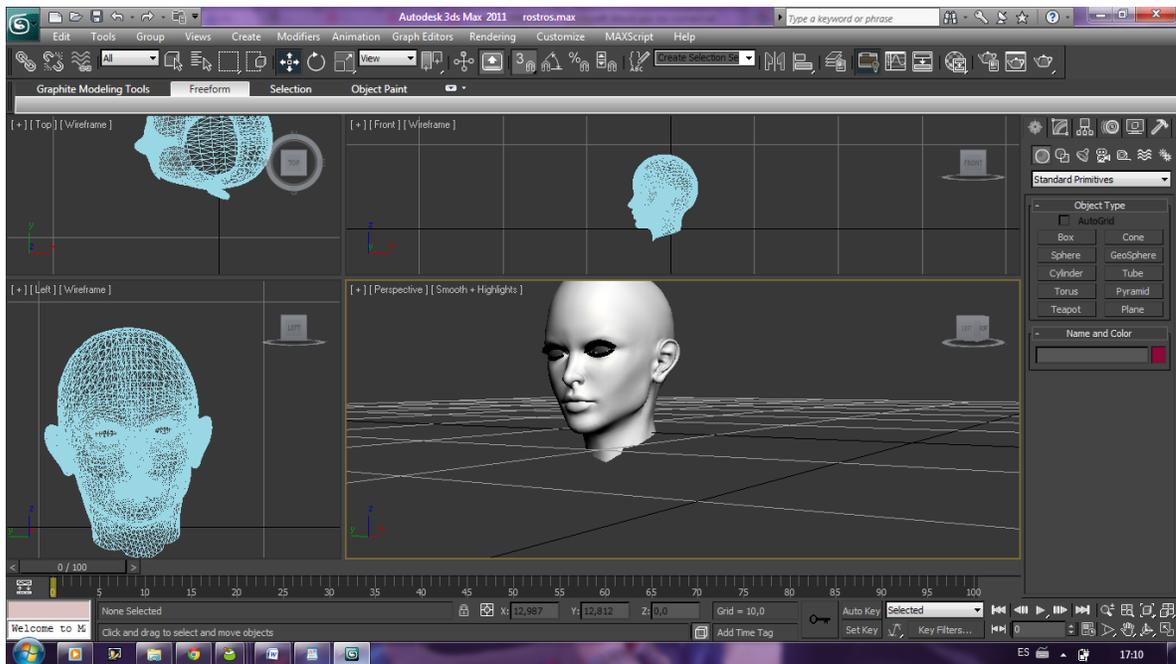


Fig. VI.50 Modelado rostro 3ds Max VI

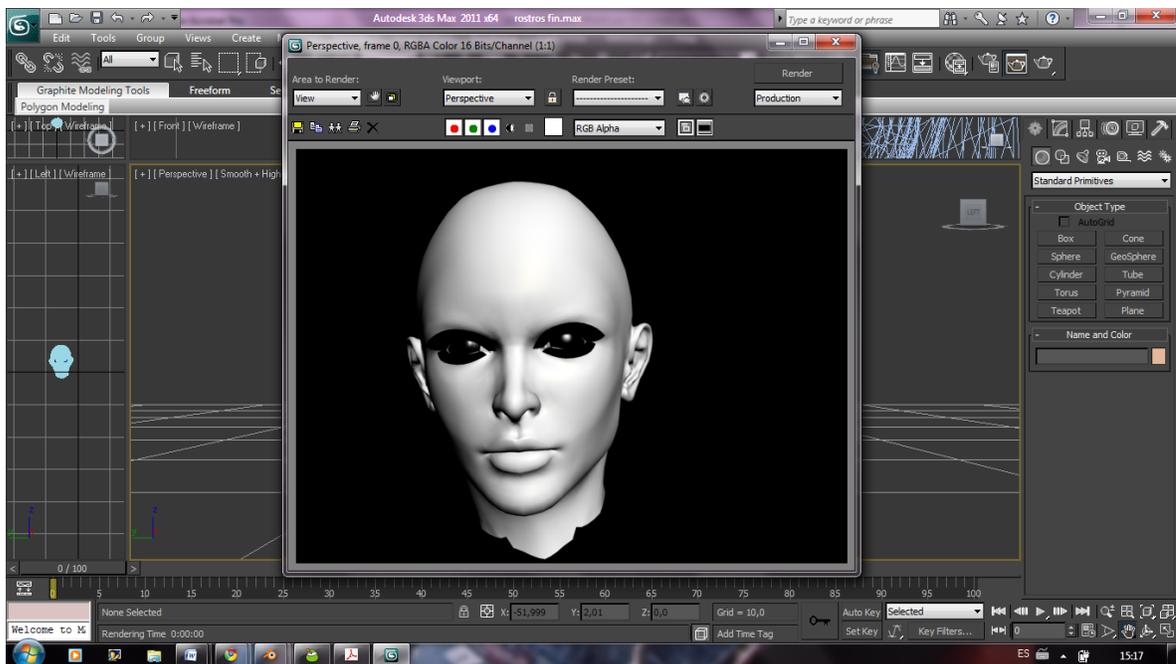


Fig. VI.51 Render Modelado rostro 3ds Max VII

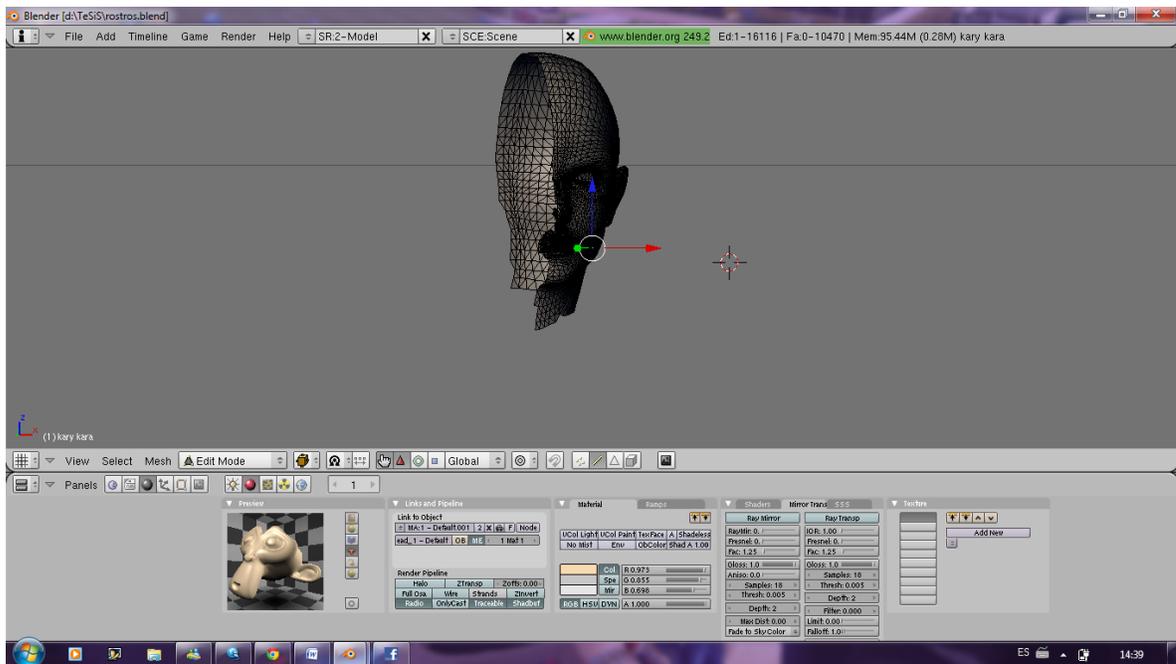


Fig. VI.51 Modelado rostro Blender I

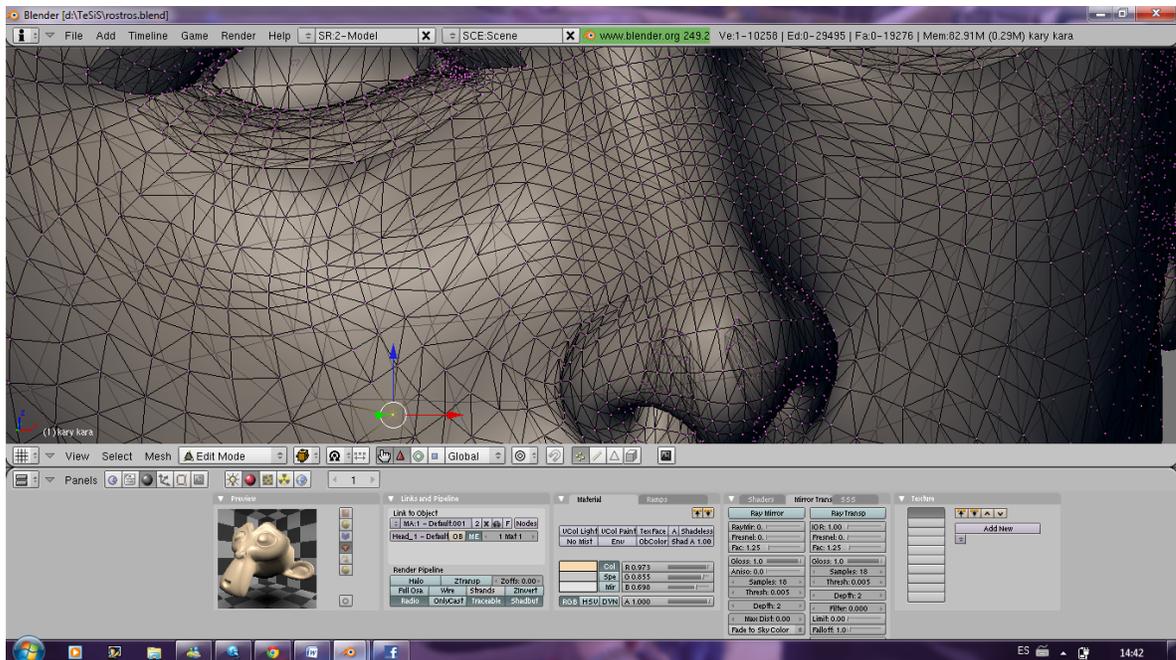


Fig. VI.52 Modelado rostro Blender II

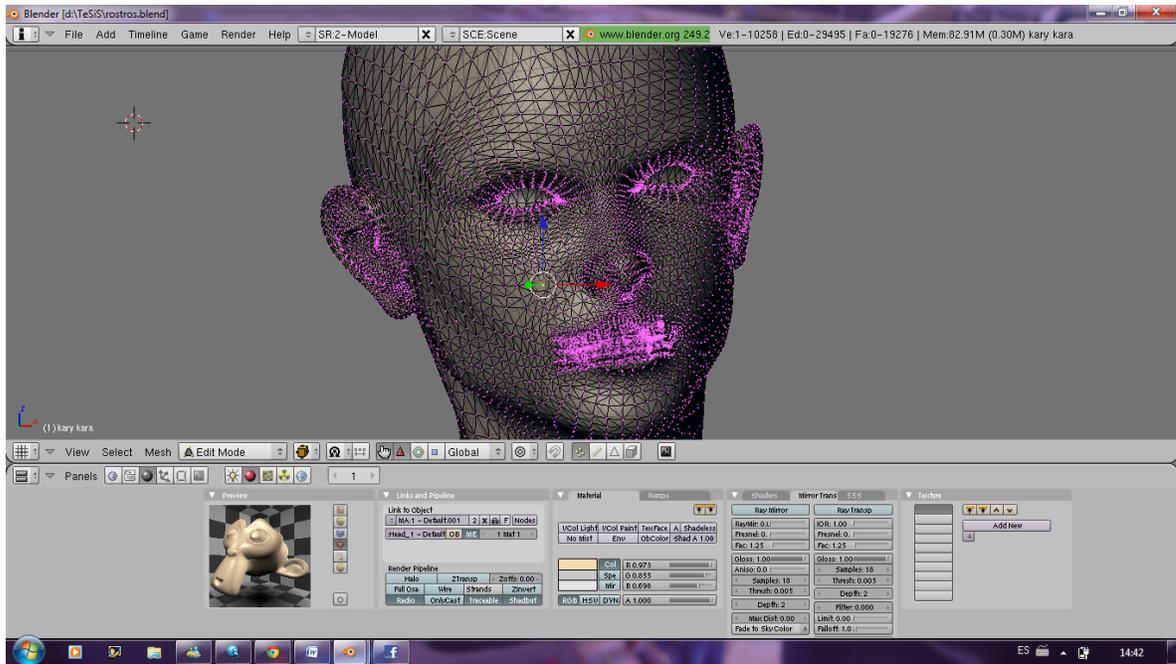


Fig. VI.53 Modelado rostro Blender III

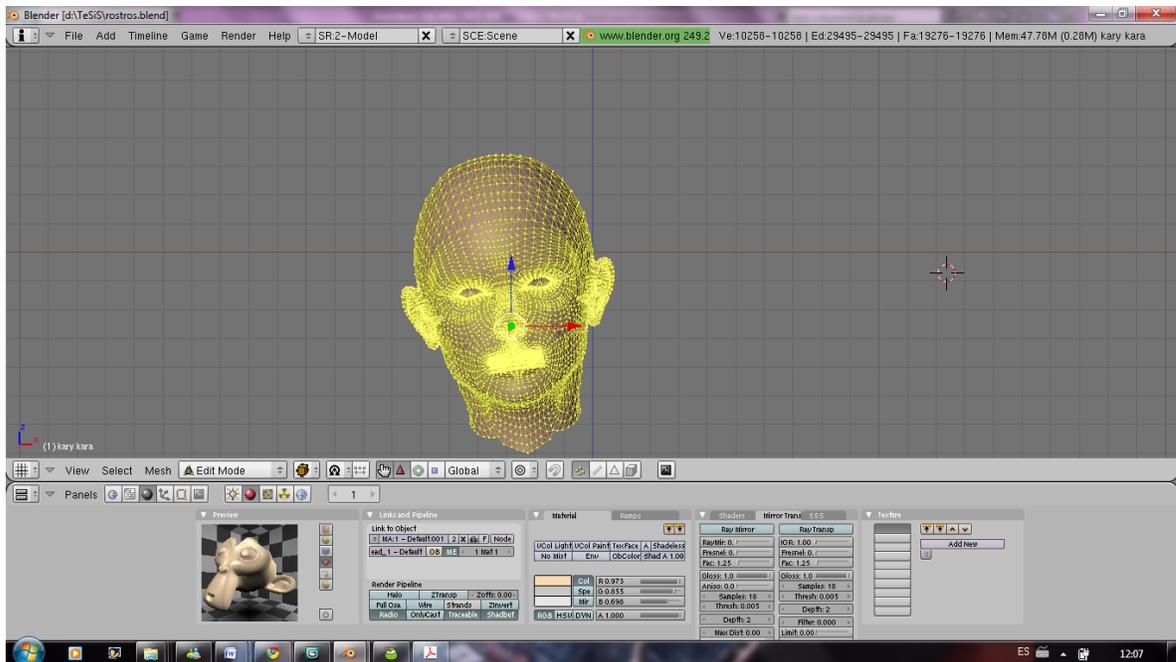


Fig. VI.54 Modelado rostro Malla editable

Render

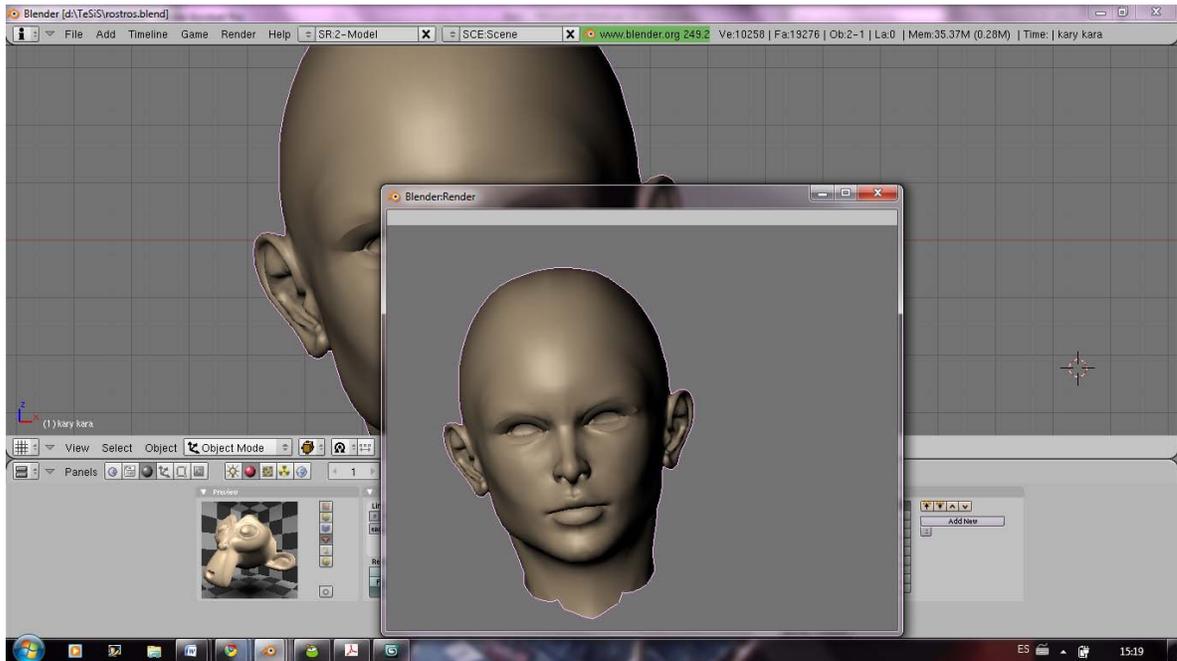


Fig. VI.58 Renderizado Blender

En el tema de renderizado ambos programas resultan veloces sin generar ningún problema y dando un finalizado de alta calidad.

CONCLUSIONES

1. Es posible obtener óptimos y similares resultados en el trabajo terminado utilizando software libre o software propietario para modelado en 3D.
2. Para el diseñador o una pequeña empresa que inicia, blender aparece como alternativa de software libre, que cumple los requerimientos de desempeño encontrados en software propietario como en este caso 3d Max, sin tener que incurrir en altos costos.
3. Autodesk 3d Max limita su uso a plataforma Windows de 32 o 64 bits; Blender está disponible para Windows , OSX , Linux , FreeBSD ,Irix , Solaris
4. El uso de 3d Max está ampliamente difundido a pesar de su costo, especialmente por su compatibilidad con otros productos usados a nivel industrial y de grandes empresas como son el resto de productos Autodesk y Adobe.
5. Tanto Blender como 3d max son incompatibles entre sí (ninguno de los dos programas la extensión nativa del otro), sin embargo se puede exportar entre sí usando la extensión .3ds.
6. Para un diseñador principiante o con poco conocimiento de los conceptos del modelado, la interfaz gráfica de usuario de ambos programas no es amigable, para que su uso se facilite se requiere el manejo de conceptos de modelado.
7. Debido a que es un tema nuevo hay gran dificultad en encontrar bibliografía, porque se ha encontrado información básicamente en internet.

RECOMENDACIONES

1. Incluir en los programas de enseñanza el uso de alternativas de software libre en la carrera de diseño gráfico para ampliar el horizonte profesional del futuro diseñador
2. La institución (ESPOCH) al ser una Universidad pública está en la obligación ética de hacer conocer a sus estudiantes que existen alternativas en cuanto a herramientas (programas) de enseñanza y trabajo que posibilitan el cumplimiento del decreto 1014.
3. Para evitar el problema de incompatibilidad de formatos de extensiones entre distintos programas procurar el uso de archivos exportados en extensión .3ds que prácticamente es reconocida por todos los programas de modelado.
4. Dotarse de una variedad de libros que permitan hacer investigaciones sobre temas referentes y así incrementar la posibilidad de incrementar el conocimiento de los estudiantes de la escuela.

RESUMEN

Se realizó un estudio comparativo de herramientas de software libre y propietario que ayudó a identificar herramientas de modelado 3D para realizar trabajos de modelado de rostros humanos con terminados gráficos de buena calidad. Estudiándose los softwares 3DS Max (propietario) y Blender (sw libre) ya que son los mayores exponentes en su rama con la finalidad de demostrar que éstos tienen la misma funcionalidad.

Fue utilizando método analítico-sintético para el estudio de parámetros comparativos tales como: herramientas de modelado, textura y pintura, animación, interfaz, etc; y para el proceso de modelado en se utilizó el método experimental.

Se tomó como parámetros a evaluar un porcentaje de 100/100 para lo siguiente: herramientas de modelado, viable polígono límite, uvwrap, interfaz y personalización: Blender y 3DS Max alcanzaron un 100%; en textura y pintura: Blender alcanzó un 71,42%, 3DS Max alcanzó un 50%; en animación y simulación: Blender alcanzó un 95,45%, 3DS Max alcanzó un 90,9%; en formatos para importar: Blender 83,33%, 3DS Max 91,66% y formatos de exportación: Blender 83,33%, 3DS Max 95,83%.

Las dos alternativas estudiadas, 3DS Max y Blender, permiten obtener productos terminados de calidad profesional en modelado 3D. Tienen usabilidad y manejo de herramientas similares facilitando al diseñador el uso de ambos programas. El uso de Blender beneficia la disminución de costos por adquisición de software de diseño.

SUMMARY

A study was done to compare the tools of free software and a privately owned software that help to identify 3D modeling tools required to perform work on modeling of the human faces with specific forms of high quality graphics. Software 3DS Max (privately owned software) and Blender (free software) are the most famous in this kind of work and the purpose of this is to show that they are similar according to their functions.

The method used was analytic-synthetic, using these comparative parameters: modeling tools, texture and paint, animation, interface, etc. and for the modeling process an experimental method was used.

It was taken as evaluation parameters a 100/100 percentage for the following: modeling tools, viable polygons' boundary, unwrap, interface and personalization: Blender and 3ds Max reached a level of 100%; in texture and paint, Blender reached the value of 71.42%; 3DS Max reached 50%; in animation and simulation modeling, Blender reached 95.45%; 3DS Max reached 90.9%; in formats to import: Blender 83.33%, 3DS Max 91.66% and in formats to export, Blender 83.33%, 3DS Max 95.83%.

The two studied alternatives, 3DS Max and Blender, both permit the finished products to obtain a professional quality in 3D modeling. They have usability and management of similar tools, which facilitates designers to use both programs. The use of is beneficial for the reduction of costs in software design acquisition.

BIBLIOGRAFÍA

EDITORIAL ALFAOMEGA, El Gran Libro de 3D Studio MAX 7/7.5 Media active.
México, Alfaomega, 2005, 797 p.

CHISMAR, John P., 3D Studio Max Animación Edición Especial. Madrid, Prentice
Hall, 2000, 470 p.

_____, 3ds Max 4 Animación Edición Especial. Madrid, Prentice Hall,
2002, 661 p.

GARCÍA CORZO, Juan Carlos, Autodesk 3ds-max Design Avanzado. Perú,
Megabyte, 2008, 703 p.

BIBLIOGRAFIA EN INTERNET

SOFTWARE Y SU CLASIFICACIÓN

Concepto Software

<http://epymes.galeon.com/enlaces763223.html>

2010/09/09

<http://es.wikipedia.org/wiki/Software>

2010/09/09

<http://software.grilk.com/historia.htm>

2010/09/09

Tipos de Software

<http://informaticxp.net/clasificacion-y-tipos-de-software>

2010/09/09

http://tecnomestros.awardspace.com/tipos_software.php

2010/09/09

<http://www.mitecnologico.com/Main/ClasificacionDelSoftware>

2010/09/09

Clasificación del software

<http://www.alegsa.com.ar/Dic/software.php>

2010/09/09

<http://www.cosaslibres.com/software.html>

2010/09/09

http://www.ithinkweb.com.mx/capacita/soft_ware.html

2010/09/09

<http://www.sindominio.net/biblioweb-old/telematica/softlibre/node1.html>

2010/09/09

SOFTWARE LIBRE

Software Libre

<http://apereza.wordpress.com/2007/09/17/las-4-libertades-del-software-libre/>

2010/09/09

<http://docencia.etsit.urjc.es/moodle/>

2010/09/09

http://es.wikipedia.org/wiki/Historia_del_software_libre_y_de_c%C3%B3digo_abierto

2010/09/10

http://es.wikipedia.org/wiki/Software_libre
2010/09/10

http://gsync.es/~mvidal/docs/FLOSS_history.pdf
2010/09/10

<http://www.cdlibre.org/consultar/catalogo/>
2010/09/10

http://www.codesol.info/portal/index.php?option=com_content&view=article&id=68:-libertades-del-software-libre&catid=71:software-libre
2010/09/10

http://www.foroswebgratis.com/tema-4_libertades_del_software_libre-92220-746591.htm
2010/09/11

Código Abierto

<http://www.gnu.org/philosophy/free-sw.es.html>
2010/09/11

<http://www.gnu.org/philosophy/free-sw.es.html>
2010/09/11

<http://www.hispalinux.es/SoftwareLibre>
2010/09/11

<http://www.ibiblio.org/pub/linux/docs/LuCaS/Manuales-LuCAS/doc-curso-salamanca-10->
2010/09/11

<http://www.softwarelibre.net/>
2010/09/11

<http://www.softwarelibre.org/>
2010/09/11

[softwareLibre/presentaciones/softwarelibre.pdf](http://softwarelibre.org/presentaciones/softwarelibre.pdf)
2010/09/11

SOFTWARE PARA DISEÑO GRÁFICO

<http://diseno-imagen.portalprogramas.com/>
2010/09/13

<http://grupogeek.com/2007/07/30/software-libre-para-diseno-grafico-y-web/>
2010/09/13

<http://www.3dprofesional.com/software/>
2010/09/13

<http://www.desarrolloweb.com/programas/diseno.php>
2010/09/13

<http://www.enlared.biz/diseno-grafico-programas-gratis.html>
2010/09/13

<http://www.forosdelweb.com/f63/cual-mejor-software-para-diseno-grafico-338276/>
2010/09/13

<http://www.hispasof.com/programas-para-diseno-grafico.htm>
2010/09/13

<http://www.mexside.com/software/software-esencial-para-disenadores-graficos>
2010/09/13

<http://www.softbull.com/windows/multimedia-y-diseno/diseno-grafico/3.html>

2010/09/13

MODELADO EN 3D

Introducción

http://es.wikipedia.org/wiki/Autodesk_3ds_Max
2010/09/14

<http://es.wikipedia.org/wiki/Blender>
2010/09/14

<http://miravirtual3d.blogspot.com/2010/03/historia-de-3d-studio-max.html>
2010/09/14

http://wiki.cgsociety.org/index.php/3ds_Max_History
2010/09/14

Software gráfico

<http://www.asterius.com/atari/>
2010/09/14

<http://www.codepixel.com/content/view/806/33/>
2010/09/14

<http://www.foro3d.com/f12/historia-del-3d-studio-44889.html>
2010/09/14

<http://www.horizonteweb.com/biblio/3DStudio.htm>
2010/09/14

<http://www.scribd.com/doc/2228057/3dmax>
2010/09/14

http://www.worldlingo.com/ma/enwiki/es/3ds_Max_release_history
2010/09/14

EXPLICACIONES SOBRE SOFTWARE LIBRE

<http://sinetgy.org/~jgb/articulos/soft-libre-educacion/>
2010/09/17

<http://www.pnte.cfnavarra.es/tecnologia/text/softlibre.htm>
2010/09/17

http://www.smaldone.com.ar/opinion/docs/sl_informatica.html
2010/09/17

INSTALADORES

http://3d.idoneos.com/index.php/Modelado_3D
2010/09/18

<http://modelado3d.tripod.com/>
2010/09/18

<http://modelado3d.tripod.com/>
2010/09/18

<http://usa.autodesk.com/adsk/servlet/index?siteID=123112&id=5659302>
2010/09/18

<http://www.bishop3d.com/>
2010/09/18

<http://www.bishop3d.com/download.htm>
2010/09/18

Descargas

<http://www.blender.org/>
2010/09/18

<http://www.blender.org/download/get-blender/>
2010/09/18

<http://www.dacostabalboa.com/es/sculptris-programa-de-modelado-3d/8547>
2010/09/18

http://www.maxon.net/pages/dyn_files/dyn_htx/htx/welcome_e.html
2010/09/18

<http://www.tutorials3d.com/esp/default.asp>
2010/09/18

<http://www.utilidades-utiles.com/descargar-3d-studio-max.html>
2010/09/18

ENTORNO GRÁFICO 3D MAX

<http://www.programas-gratis.net/b/grafico-3d>
2010/09/18

http://www.taringa.net/posts/downloads/8966907/Crea-Tus-Propios-Juegos-3d_.html
2010/09/18

ANEXOS

7 DECRETO 1014 PRESIDENTE RAFAEL CORREA

N° 1014

RAFAEL CORREA DELGADO

PRESIDENTE CONSTITUCIONAL DE LA REPÚBLICA

CONSIDERANDO:

Que en el apartado g) del numeral 6 de la Carta Iberoamericana de Gobierno Electrónico, aprobada por el IX Conferencia Iberoamericana de Ministros de Administración Pública y Reforma del Estado, realizada en Chile el 1 de Junio de 2007, se recomienda el uso de estándares abiertos y software libre, como herramientas informáticas;

Que es el interés del Gobierno alcanzar soberanía y autonomía tecnológica, así como un significativo

ahorro de recursos públicos y que el Software Libre es en muchas instancias un instrumento para alcanzar estos objetivos;

Que el 18 de Julio de 2007 se creó e incorporó a la estructura orgánica de la Presidencia de la República la Subsecretaria de Informática, dependiente de la Secretaria General de la Administración, mediante Acuerdo N°119 publicado en el Registro Oficial No. 139 de 1 de Agosto del 2007;

Que el numeral 1 del artículo 6 del Acuerdo N° 119, faculta a la Subsecretaría de Informática a elaborar y ejecutar planes, programas, proyectos, estrategias, políticas, proyectos de leyes y reglamentos para el uso de Software Libre en las dependencias del gobierno central; y,

En ejercicio de la atribución que le confiere el numeral 9 del artículo 171 de la Constitución Política de la República;

DECRETA

Artículo 1.- Establecer como política pública para las Entidades de la Administración Pública Central la utilización de Software Libre en sus sistemas y equipamientos informáticos.

Artículo 2.- Se entiende por Software Libre, a los programas de computación que se pueden utilizar y distribuir sin restricción alguna, que permitan su acceso a los códigos fuentes y que sus aplicaciones puedan ser mejoradas.

Estos programas de computación tienen las siguientes libertades:

- a) Utilización del programa con cualquier propósito de uso común
- b) Distribución de copias sin restricción alguna.
- c) Estudio y modificación del programa (Requisito: código fuente disponible)
- d) Publicación del programa mejorado (Requisito: código fuente disponible).

Artículo 3.- Las entidades de la Administración Pública Central previa a la instalación del Software libre en sus equipos, deberán verificar la existencia de capacidad técnica que brinde el Soporte necesario para el uso de este tipo de software.

Artículo 4.- Se faculta la utilización de software propietario (no libre) únicamente cuando no exista una solución de Software Libre que supla las necesidades requeridas, o cuando esté en riesgo la seguridad nacional, o cuando el proyecto informático se encuentre en un punto de no retorno.

Para efectos de este decreto se comprende cómo seguridad nacional, las garantías para la supervivencia de la colectividad y la defensa del patrimonio nacional.

Para efectos de este decreto se entiende por un punto de no retorno, cuando el sistema o proyecto informático se encuentre en cualquiera de estas condiciones:

- a) Sistema en producción funcionando satisfactoriamente y que un análisis de costo beneficio muestre que no es razonable ni conveniente una migración a Software Libre.
- b) Proyecto en estado de desarrollo y que un análisis de costo beneficio muestre que no es conveniente modificar el proyecto y utilizar Software Libre.

Periódicamente se evaluarán los sistemas informáticos que utilizan software propietario con la finalidad de migrarlos a Software Libre.

Artículo 5.- Tanto para software libre como software propietario, siempre y cuando se satisfagan los requerimientos, se debe preferir las soluciones en este orden:

- a) Nacionales que permitan autonomía y soberanía tecnológica.
- b) Regionales con componente nacional.
- c) Regionales con proveedores nacionales.
- d) Internacionales con componente nacional.
- e) Internacionales con proveedores nacionales.
- f) Internacionales.

Artículo 6.- La Subsecretaría de Informática como órgano regulador y ejecutor de las políticas y proyectos informáticos en las entidades del Gobierno Central deberá realizar el control y seguimiento de este Decreto.

Para todas las evaluaciones constantes en este decreto la Subsecretaría de Informática establecerá los parámetros y metodología obligatorias.

Artículo 7.- Encárguese de la ejecución de este decreto los señores Ministros Coordinadores y el señor Secretario General de la Administración Pública y Comunicación.

Dado en el Palacio Nacional en la ciudad de San Francisco de Quito, Distrito Metropolitano, el día de hoy 10 de abril de 2009

Rafael Correa Delgado

PRESIDENTE CONSTITUCIONAL DE LA REPÚBLICA