Pontificia Universidad Católica del Ecuador

CREACIÓN DE UN CORTO ANIMADO EN 3D Y LA INFRAESTRUCTURA TÉCNICA NECESARIA PARA SU REALIZACIÓN TANTO EN HARDWARE Y EN SOFTWARE DE LICENCIA GNU GPL.

DISERTACIÓN PREVIA PREVIA A LA OBTENCIÓN DEL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS Y COMPUTACIÓN EN LA FACULTAD DE INGENIERÍA

Autor: Juan Carlos Reascos Laverdi

Director: Ingeniero Oswaldo Luna

Quito, abril 2012

DEDICATORIA

Este trabajo lo dedico a los que han desafiado a la tecnología inspirados por el arte, a los creativos que junto a personas ingeniosas han hecho del bit y el byte un lienzo, también a esas personas que a pesar de tener tantas limitaciones siguen creando, siguen soñando, como ese grupo de Geeks que hemos trabajado sin parar por tanto tiempo.

Al verdadero sentimiento que nos aproxima a un Dios, que es el amor, gracias a mi padres , a mis hermanos, gracias por creer y estar junto a mi.

INDICE

CAPITULO I..... 1

MARCO TEÓRICO 1
1.1 ANIMACIÓN 1
1.1.1 Definición 1
1.1.2 Breve Reseña Histórica 2
1.1.3 Principios de la Animación 12
1.1.3.1 Estirar y Encoger (Squash and stretch) 12
1.1.3.2 Anticipación 12
1.1.3.3 Puesta en escena (Staging) 13
1.1.3.4 Acción Directa y Pose a Pose 13
1.1.3.5 Acción Continuada y Acción Superpuesta (Follow Through and Overlapping Action) 13
1.1.3.6 Entradas Lentas y Salidas Lentas (Slow In and Slow Out) 13
1.1.3.7 Arcos 14
1.1.3.8 Acción Secundaria 14
1.1.3.9 Timing 14
1.1.3.10 Exageración 14
1.1.3.11 Dibujos sólidos 14
1.1.3.12 Personalidad o Apariencia 14
1.1.4 3D 15
1.1.4.1 Definición 16
. La visión Humana 16
. La perspectiva 17
. Perspectiva tradicional 18
. Perspectiva con un punto de fuga o cónica 19
. Perspectiva con dos puntos de fuga. 19
1.1.4.2 Herramientas Digitales 20
- Autodesk Maya 20
- Autodesk 3ds Max 22
- Autodesk® Softimage 23
- Marionette 24
- Lightwave 24
- Blender 3D 25
1.1.4.3 Actualidad - Estudios de animación. 26
- Pixar 26
. Historia 26

. La creación 29
- DreamWorks SKG Animation 29
. Historia 1994-2003 30
. 2004-actualidad 30
- Blue Sky Studios 31
. Historia 31
. Tecnología 32
. Filmografía 32
- Blender Foundation & Instituto Blender. 33
. Elephants Dream 33
. Producciones 34
- Festivales 36
1.2 SOFTWARE LIBRE 38
1.2.1 LICENCIA GNU-GPL (General Public Licence). *39*
1.2.1.1 DEFINICION 39
1.2.1.2 Principales tipos de licencias libres 40
Copyleft 40
Licencia BSD 41
Creative Commons 42
1.2.1.3 Sistema Operativo Linux 43
HISTORIA 44
Popularidad 45
Distribuciones 46
UBUNTU (Distribución). 47
1.2.2 Animación y Software libre *48*
1.2.2.1 Animación 2D y Software libre. 48
Synfig (Herramienta de Animación 2D) 48
. Historia 49
Pencil (animación digital tradicional 2D) 51
1.2.2.2 Animación 3D y Software libre. 52
Wings 3D (modelado digital 3D) 52
Blender 3D 53
. Origen 53
. Características 54
1.3 EL CORTOMETRAJE 55
1.3.1 Etapas - Pipeline de Animación *55*
1.3.2 Preproducción *56*
1.3.2.1 StoryBoard 57
1.3.2.2 Layouts 58
1.3.2.3 Model Sheets 58
1.3.2.4 Animatic 59
1.3.3 Producción *60*
1.3.3.1 Modelado 60
1.3.3.2 Texturas 61
1.3.3.3 Rigging / Skinning 63
1.3.3.4 Puesta en Escena. 64

Iluminación 64
Componentes y Propiedades de la Luz 65
INTENSIDAD DE LA LUZ 66
Dirección de la Luz 67
Color de la Luz 67
Tamaño de la Fuente de Luz 70
1.3.3.5 Proceso de Animación 70
1.3.4 Postproducción *73*
1.3.5 Cuadro Sinóptico - CAPITULO I *74*

CAPITULO II 75

2. LA HERRAMIENTA 75
2.1 RESEÑA HISTÓRICA 77
2.2 CORTOMETRAJES DESARROLLADOS EN BLENDER 79
2.2.1 Proyecto Orange "Elephants Dream" *80*
Software 83
2.2.2 Big Buck Bunny & Yo Frankie! *84*
2.2.3 Plumíferos *86*
Desarrollo 86
Mejoras 87
2.2.4 Proyecto Durian "Sintel" *88*
Software 89
2.3 MEJORAS ACTUALES PARA BLENDER 90
2.3.1 EVOLUCION *90*
2.3.2 Características Blender3D *95*
2.3.4 Caracteristicas para Blender 2.49b *110*
2.4 FUTURO DE BLENDER 112
2.4.1 Blender 2.56 ..... *112*
2.4.3 Información general *115*
2.4.4 Interfaz renovada *116*
2.4.5 Multiples Ventanas *118*
2.4.6 Nuevo Diseño - GUI Controls *119*
2.4.7 Python Mejorado *120*
2.4.8 Explorador de Archivos *121*
2.4.9 Builds de Blender 3D *122*
2.5 PYTHON EL LENGUALE DE BLENDER 123
2.5.1 Scripts Python *124*
2.5.1.2 Scripting y Blender 124
2.5.2 Plugins *126*
2.5.3 Funcionalidad Python en Blender *126*
2.5.4 La API del motor de juegos de Blender: *126*

CAPITULO III 128

ESTRUCTURA DEL CORTOMETRAJE 128

PIPELINE ANIMACIÓN "LEROY" 128
3.1 PREPRODUCCION: 130
3.1.1 SINOPSIS (historia) *130*
3.1.2 GUIÓN *130*
3.1.3 DISEÑO PERSONAJES *133*
3.1.3.1 Leroy: 133
3.1.3.2 MOMO: 136
3.1.3.3 Nave Momo: 137
3.1.3.4 Nave Vehículo de Carga: 139
3.1.3.5 Nave Wincha: 140
3.1.4 STORYBOARD *141*
3.1.5 ANIMATIC *144*
3.2 PRODUCCION: 147
3.2.1 ADMINISTRACION DE ARCHIVOS *147*
3.2.1.1 SVN Subversion 149
3.2.1.2 Cliente Subversion 150
3.2.2 MODELADO *154*
3.2.2.1 Modelado Organico (plantas) 159
3.2.2.2 Modelado Escenario (landscapes) 161
3.2.3 TEXTURAS *163*
3.2.3.1 Mapeo UV 166
3.2.4 RIGGING *170*
3.2.4.1 IK y FK 176
3.2.5 Sistema de Huesos en extremidades - Constraints. *180*
3.2.6 SKINING *191*
3.2.7 INTEGRACION ARCHIVOS *194*
3.2.8 ANIMACION *203*
3.2.8.1 ACCIÓN CICLO CAMINAR. 205
3.2.8.2 EDITOR NLA. 209
3.3 POST-PRODUCCION 212
3.3.1 ILUMINACION *212*
3.3.2 COMPOSICION POR NODOS. *218*
3.3.3 HARDWARE Y RENDER *223*
3.3.4 RENDIMIENTO *225*
3.3.4.1 RENDER 226
3.3.4.2 Administración y Editor de Video. 228
CONCLUSIONES **233**
RECOMENDACIONES **236**
BIBLIOGRAFIA **238**

CAPITULO I

Marco Teórico

El estatuto teórico del presente estudio está integrado por tres importantes ámbitos. El primero se relaciona con la animación, su significación, historia y evolución; además de una revisión de los conceptos sobre gráficos en tres dimensiones, las herramientas informáticas más usadas para el manejo de objetos tridimensionales y la actualidad en el desarrollo de productos en este formato.

El segundo ámbito se enfoca en el software libre, en sus principios y filosofía; así como también la exposición de los programas, más sobresalientes y conocidos, creados para animación dentro de esta corriente informática.

Para finalizar, el estudio se enfoca en el proceso de creación de un cortometraje animado, desde su análisis en pre-producción, su desarrollo y el trabajo final de afinación en post-producción.

Se considera que el proceso de construcción de un cortometraje engloba:

1.1 Animación

Se da inicio a la investigación con el estudio del campo de la animación; los principales hitos que marcaron su evolución, las invenciones que se destacaron a lo largo de su historia, los principios que la rigen y finalmente la aparición de los gráficos por computadora.

1.1.1 Definición

Existen varias perspectivas desde las que se puede definir a la animación, en la presente investigación ha podido evidenciar dos de ellas, estas dependen del valor artístico o técnico al que se lo aplique, por ejemplo el origen etimológico dice que "animación" es proviene del latín lexema anima, del vocablo en latín para referirse a alma

o espíritu; el verbo "animar" significa literalmente "dar vida a", refiriéndose a todo aquello que no la tuviere.

[1]Una definición más técnica podría ser la que dice el animador norteamericano Gene Deitch, "animación es el registro de fases de una acción imaginaria creadas individualmente, de tal forma que se produzca ilusión de movimiento cuando son proyectadas a una tasa constante y predeterminada, superior a la de la persistencia de la visión en la persona."

Según ese autor las animaciones son una fantasía, una ilusión de nuestros sentidos. El cerebro es engañando. Todo es una ilusión óptica. Se cree estar viendo una corriente constante de movimiento en la pantalla Pero en realidad, esto no es así. Lo que el ojo está viendo son una serie de fragmentos discontinuos. A su vez, el ojo humano lo ve como una continuidad y así es como lo interpreta el cerebro

El fenómeno de la persistencia de la visión genera una acción continua e interrumpida casi tan real como la realidad. La sucesión estroboscópica de los fotogramas crea la apariencia de movimiento.

1.1.2 Breve Reseña Histórica

La animación ha tenido un largo camino desde que se realizaban las mismas en papel y lápiz hasta poder llegar al píxel, pero la idea de plasmar un registro de eventos, sea esta por la importancia histórica de este o por el simple hecho de mantenerlos vivos y recordarlos, estos han llevado a inventar y crear dispositivos cada vez más sofisticados con la simple idea de recrear movimiento.

Sensación de Movimiento, esta ilusión creada por la secuencia de imágenes ha sido parte de nuestra evolución, historiadores que se remontan a la prehistoria , han realizado descubrimientos en los que en varias pinturas rupestres nuestros antepasados trataban de expresar movimiento con sus insipientes dibujos, ya que desde sus orígenes, el hombre ha tratado de comunicarse mediante grafismos o dibujos que con sus primeras presentaciones que se tiene conocimiento no solo intentó representar la realidad que les

[1] http://es.wikipedia.org/wiki/Gene_Deitch

rodeaba, animales, astros, al propio ser humano, etc. Sino también sensaciones, como la alegría de las danzas, o la tensión de las cacerías.

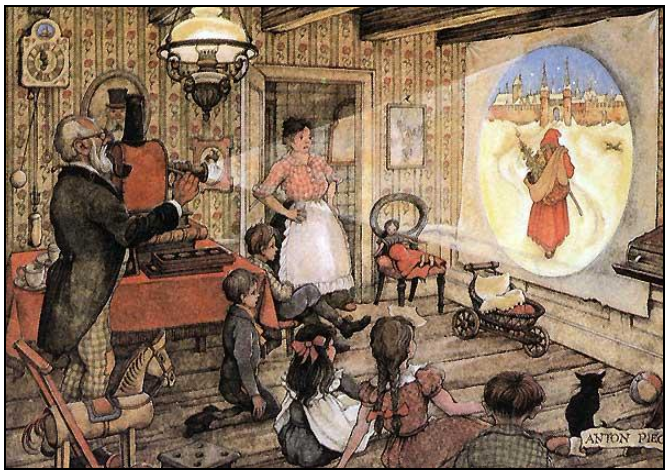
Después culturas más civilizadas como en Egipto y Grecia aportaron con importantes avances tecnológicos propios de su época, de igual forma trataban de representar el movimiento de varias de sus creaciones.

Mas adelante Leonardo Da Vinci también experimentó con la figura en movimiento, como se puede comprobar en su ilustración de las proporciones humanas, en las que dibuja las que parecen ser dos fases de una misma acción.

Las invenciones.

[2]El primer intento que se conoce de una animación mediante la proyección de imágenes data de 1640, cuando el alemán Anthonasius Kircher inventó el primer proyector de imágenes: "la linterna mágica", en la que, mediante grabados en cristales, era capaz de proyectar diferentes fases consecutivas del movimiento, cambiando los cristales de forma mecánica. En una de sus proyecciones representaba a un hombre mientras dormía, abriendo y cerrando la boca.





Título: Linterna Mágica

Fuente: http://www.esacademic.com/dic.nsf/eswiki/729179

[2] http://www.esacademic.com/dic.nsf/eswiki/729179

[3]A finales del siglo XVIII una gran parte de los espectáculos usaban la linterna mágica para producir shows, también conocidos como phantasmagoria shows. Una variedad de imágenes horríficas era proyectada en nubes de humo, daban una apariencia aterradora junto con imágenes que se movían por las paredes. El proyector se ubicaba frecuentemente detrás de una pantalla translúcida fuera de la vista del público, eso añadía un toque de misterio al espectáculo, Uno de los showmen mas famosos phantasmagoria fué Étienne Gaspard Robertson (1763-1863), quién modifico la linterna, la llamó "phantascope" y le dio mas movilidad para así obtener sensación mas espectacular a sus proyecciones.



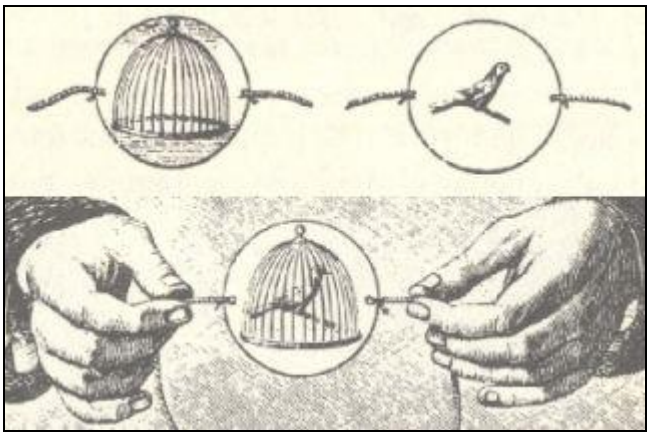
Título: Linterna Mágica
Fuente: http://wladislawdraculesti.blogspot.com/
Elaboración: Juan Carlos Reascos Laverdi.

Uno de los principales Hitos marcados en la historia de la animación fue en 1824, cuando Peter Mark Roget descubrió el "Principio de Persistencia de la Visión", fundamento

3

http://books.google.com.ec/books?id=1EvqT7YflYIC&printsec=frontcover&dq=animacion&hl=es&ei=lMXETMbyAoS0lQfo3IDCCg&sa=X&oi=book_result&ct=result&resnum=5&ved=0CD4Q6AEwBA#v=onepage&q&f=false

en el que se basan todas las imágenes proyectadas que conocemos hoy en día. Demostraba que el ojo humano retiene la imagen que ve durante el tiempo suficiente para ser sustituida por otra, y así sucesivamente, hasta realizar un movimiento completo, como se ve en su "taumatropo".



Título: Taumatropo
Fuente: http://www.antiquus.es/catalogo.php?sm=4&g=15
Elaboración: Juan Carlos Reascos Laverdi.

Más invenciones a partir el descubrimiento de la persistencia de la visión----------

Ninguno pasó de la categoría de juguete hasta la llegada del "Phenakistoscopio" de Joseph Antoine Plateau, en 1831, en el que conseguía plasmar un movimiento completo mediante el uso de dibujos.

Entre las bases del origen de la animación está el mismo juego de sombras y la proyección de siluetas de papeles recortados creados por la cultura china.

[4]*La animación mas vieja que el Cine*

La animación apareció antes que el propio cinematógrafo. En 1888 el francés Émile Reynaud, padre del cine de animación, inventó el praxinoscopio, uno de los muchos juguetes ópticos de la época, en el cual se utilizaba una técnica pre-cinematográfica de

[4] http://es.wikipedia.org/wiki/Cine_de_animaci%C3%B3n

animación. Posteriormente lo perfeccionó con su "teatro óptico", que permitía proyectar películas animadas dotadas de argumento en una pantalla para un público, y, acompañadas de música y efectos sonoros, mantuvo un espectáculo de dibujos animados desde 1892 hasta finales del siglo XIX.



Título: Praxinoscopio
Fuente: http://www.maquinascientificas.es/15el%20praxinoscopio.htm
Elaboración: Juan Carlos Reascos Laverdi.

El siguiente pionero del cine de animación fue el francés Émile Cohl, que desde 1908 realizó los primeros cortometrajes de dibujos animados

Muchos fueron los inventos y sus creadores pero todos partían del mismo principio que fue la persistencia de la visión.

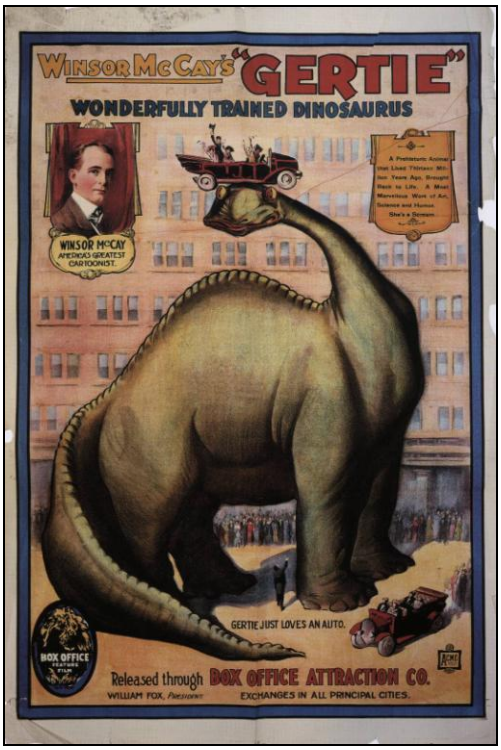
Todas estas invenciones fueron creadas con el fin de distraer al público, estas fueron presentadas en ferias y locales en los que se podían presentar creaciones innovadoras.

Animación tradicional

[5]Con "Gerty El dinosaurio" (1914) es un cortometraje animado de 1914 dirigido por Winsor McCay que inspiró a varias generaciones de animadores. Aunque no fue el

[5] http://68revoluciones.com/?p=72

primer dibujo animado, como algunos piensan, fue el primero en presentar un personaje con personalidad propia. El reto fue volver a la vida a un animal de la prehistoria valiéndose solo de 10.000 dibujos hechos a mano, supuso un antes y después en la animación, lo convierte en el predecesor de los creados posteriormente por Walt Disney.



Título: Gertie "el dinosaurio"
Fuente: http://www.esacademic.com/dic.nsf/eswiki/527409
Elaboración: Juan Carlos Reascos Laverdi.

Hubieron las primeras producciones realizadas con animación tradicional, por ejemplo en 1917 Realiza el primer largometraje de dibujos animados de la historia: "El Apóstol" por Quirino Cristiani (una sátira sobre Hipólito Irigoyen un político de la época, con libreto de Alfonso de Laferrére, producida por Federico Valle, dibujos de Diógenes Taborda y maqueta del Arq. Andrés Ducaud). Link: http://www.quirinocristiani.com.ar/

[6] En 1928 Disney formó la Walt Disney Productions, y se aventuró en el cine sonoro con Steamboat Willie (1928). Esta película y La danza macabra (Skeleton Dance, 1929),

[6] http://es.wikipedia.org/wiki/Dibujo_animado

que inauguró sus Sinfonías Tontas, le proporcionaron fama. En 1932 utilizó por primera vez el color para Árboles y flores (Flowers and Trees); y, en 1937, con El viejo Molino (The Old Milis), reveló las posibilidades de efectos tridimensionales que podía lograr mediante su procedimiento «Multiplane», que implicaba enfocar con gran exactitud imágenes animadas dispuestas en una determinada secuencia de planos.



Título: Estudios Animación Clásica
Fuente: http://animacionclasica.blogspot.com/
Elaboración: Juan Carlos Reascos Laverdi.

Durante toda la década de los 30 perfeccionó a sus personajes Mickey y Minnie, el Pato Donald, Pluto, Goofy y otros muchos. En 1937 estrenó su primer largometraje de dibujos animados, Blancanieves y los 7 enanitos, seguido por Pinocho (Pinocchio) y Fantasía (ambas de 1940), Dumbo (1941) y Bambi (1942).

Título: Logo Estudios Disney
Fuente: www.waltdisney.com
Elaboración: Juan Carlos Reascos Laverdi.

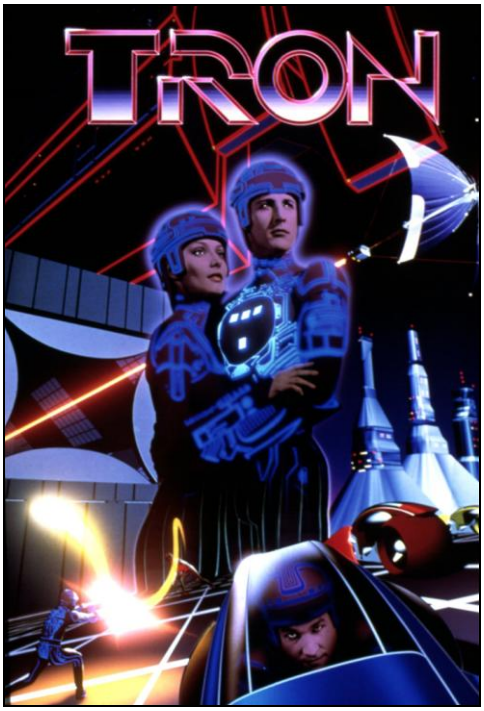
[7]La industria CGI aparece en el cine y la televisión y estos recurren a las imágenes generadas por computadora ya que en ciertas situaciones es más económico que el recurrir a métodos físicos como la construcción de maquetas complicadas para creación de efectos, o alquiler de mucho vestuario para escenas de multitudes de personas, y también porque permiten la obtención de imágenes que no serían factibles de ningún otro modo.

Las dos primeras películas en las que intervineron imágenes generadas por computadora de forma importante, Tron (1982) y Last Starfighter (1984),

[7] http://www.tech-faq.com/es/lo-que-es-cgi-animacion.html

http://knol.google.com/k/historia-de-la-animaci%C3%B3n-por-computadora-cgi#

http://es.wikipedia.org/wiki/Imagen_generada_por_computadora



Título: TRon – la película
Fuente: http://www.disney.es/tron/
Elaboración: Juan Carlos Reascos Laverdi.

Varios studios han sido los que han marcado un sitial en el desarrollo CGI tales como Industrial Light and Magic (starwars), Blue Sky (Ice Age), Dreamworks (Shrek), Pixar (Cars, Up, etc).

Existen otros estudios que han realizado producciones de manera independiente por ejemplo en Mexico en el año 2006 un estudio mexicano lanzó su primer largometraje animado realizado en flash (sw animación digital 2D), esta película fue: Una Película de Huevos.

La «Factoría Disney» se ha aliado en los últimos años en un acuerdo de larga duración con «Pixar», una empresa de animación que realiza las películas que Disney coproduce y distribuye por todo el mundo. De esta colaboración han surgido películas muy interesantes como Toy Story (1995), Toy Story 2 (1999) de John Lasseter, Monster Inc (2001), dirigida por Pete Docter, y Buscando a Nemo (2003), dirigida por Lee

Unkrich, Los Increíbles (The Incredibles), 2006, Cars, 2007, Ratatouille, 2008, WALL·E, 2009, Up.

Todas en tres dimensiones, utilizando las última técnicas digitales.



Título: Estudios Pixar
Fuente: http:// http://www.uhu.es/cine.educacion/cineyeducacion/figurasdisney.htm
Elaboración: Juan Carlos Reascos Laverdi.

El 30 de septiembre de 2010, se estrenó on-line y se liberó el último cortometraje realizado por la Fundación Blender llamado Sintel, esta ya ha realizado varias obras previas como: Elephants Dreams (marzo 2006), Bick Buck Bunny (abril 2008), junto al juego que fue desarrollado en paralelo llamado Yo Frankie!.



Título: Sintel
Fuente: http://www.sintel.org
Elaboración: Juan Carlos Reascos Laverdi.

1.1.3 Principios de la Animación

[8]Los 12 Principios de la Animación

Los doce principios de la animación tradicional fueron creados en los años treinta por los primeros animadores en la compañía Disney, denominados the Nine Old Men (los Nueve Viejos). Aún hoy siguen vigentes y han cambiado poco, desde ese tiempo la mayoría de animadores se han regido a estos principios y estos son:

1.1.3.1 Estirar y Encoger (Squash and stretch)

Éste es el primero de los 12 principios, y consiste en exagerar las deformaciones de los cuerpos flexibles, para lograr un efecto más cómico, o más dramático.

1.1.3.2 Anticipación

El principio de la anticipación ayuda a guiar la mirada del público al lugar donde está a punto de ocurrir la acción. Es ideal para 'anunciar la sorpresa'. Así, a mayor anticipación menor es la sorpresa, pero mayor el suspense.

[8] http://es.wikipedia.org/wiki/Doce_principios_%28animaci%C3%B3n%29

1.1.3.3 Puesta en escena (Staging)

Con este principio se traduce las intenciones, el ambiente de la escena a posiciones y estos a acciones específicas de los personajes. Poniendo en escena las posiciones claves de los personajes definiremos la naturaleza de la acción.

1.1.3.4 Acción Directa y Pose a Pose

Éstas en realidad son dos técnicas de animación diferentes. En la acción directa se crea una acción continua, paso a paso, hasta concluir una acción impredecible, un ejemplo de la aplicación de esta técnica es la usada stopmotion.

En la acción pose a pose se desglosa los movimientos en series estructuradas de poses clave, de esta forma se crea Keyframes los que desglosan un movimiento específico y así poder dar fluidez a la animación.

1.1.3.5 Acción Continuada y Acción Superpuesta (Follow Through and Overlapping Action)

Estas dos técnicas ayudan a enriquecer y dar detalle a la acción. En ellas el movimiento continúa hasta finalizar su curso. En la acción continuada, la reacción del personaje después de una acción dice cómo se siente el personaje. En la acción superpuesta, movimientos múltiples se mezclan, se superponen, e influyen en la posición del personaje.

Diferentes cosas se mueven a diferente tiempo.

Con esta premisa se debe tener en cuenta que existen elementos que aunque conservan la misma animación que otros, ésta ocurre con cierto desfase. Un ejemplo muy común sería la cola de un perro; cada hueso se mueve ligeramente después que el anterior, permitiendo conseguir el efecto de látigo en mayor o menor medida.

1.1.3.6 Entradas Lentas y Salidas Lentas (Slow In and Slow Out)

Con este principio se consigue un efecto gracioso al acelerar el centro de la acción, mientras que se hacen más lentos el principio y el final. La mayoría de los cuerpos deben acelerar y detener de manera gradual.

1.1.3.7 Arcos

Al utilizar los arcos para animar los movimientos de un personaje se le esta dando una apariencia natural, ya que la mayoría de las criaturas vivientes se mueven en trayectorias curvas, nunca en líneas perfectamente rectas. Si no se utiliza estos arcos, podemos dar un toque siniestro, robótico, a la animación.

1.1.3.8 Acción Secundaria

Este principio consiste en los pequeños movimientos que complementan a la acción dominante. En el caso de animar a un jinete montado a caballo, el vaivén del primero estaría provocado por el galopar del segundo, por lo que eso sería una animación secundaria.

1.1.3.9 Timing

Es el momento preciso y el tiempo que tarda un personaje en realizar la acción, y que proporciona emoción e intención a la actuación, además este puede definir ciertas características de un personaje. Y con este principio se tiene en cuenta la noción del tiempo; de saber cuánto tiempo han de tardar las cosas para que ocurran.

1.1.3.10 Exageración

Normalmente, la exageración ayuda a los personajes a reflejar la esencia de la acción. Una gran parte de esta exageración puede ser obtenida mediante el Squash and Strech. Exagerar poses, gestos o movimientos dotará al personaje de vida.

1.1.3.11 Dibujos sólidos

Un modelado y un sistema de esqueleto sólido, o un dibujo sólido, ayudarán al personaje a cobrar vida. El peso, la profundidad y el balance simplificarán posibles complicaciones en la producción debidas a personajes pobremente modelados. Además, hay que poner atención a las siluetas al alinear los personajes con la cámara.

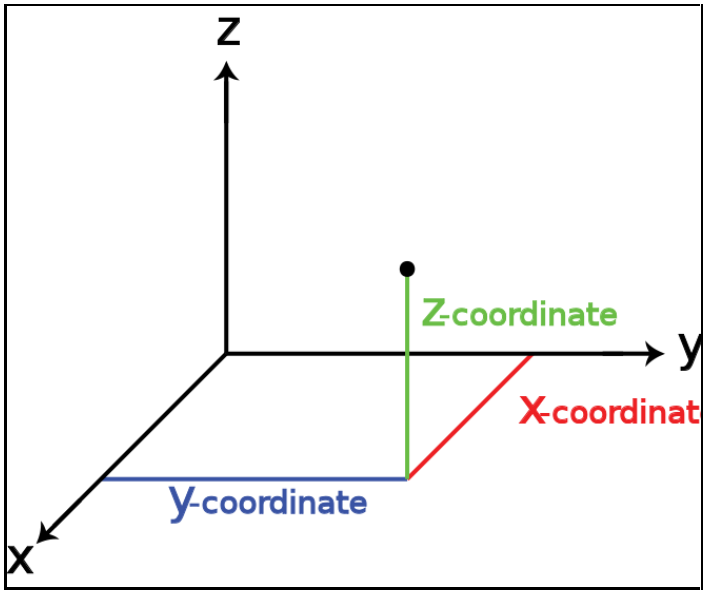
1.1.3.12 Personalidad o Apariencia

La personalidad, o la apariencia, como se le llamó en un principio, facilita una conexión emocional entre el personaje y el público. Se debe desarrollar los personajes hasta darles una personalidad interesante, con un conjunto de deseos y necesidades claras que marquen su comportamiento y sus acciones.

El animador que sabe aplicar un buen acting se diferencia del resto por que no se limita a mover controladores, sino que dota al personaje de personalidad; lo hace actuar.

1.1.4 3D

En geometría y análisis matemático, un objeto o ente es tridimensional si tiene tres dimensiones. Es decir cada uno de sus puntos puede ser localizado especificando tres números dentro de un cierto rango. El estudio de las 3 Dimensiones basado en la geometría de los 3 ejes, han producido una revolución en todas las aplicaciones Graficas asistidas por computador, han surgido la posibilidad de dibujar figuras con una precisión mejor que con una regla y un compás



Título: Coordenadas ordenadas
Fuente: http://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm
Elaboración: Juan Carlos Reascos Laverdi.

1.1.4.1 Definición

En la creación de gráficos en 3D, técnicamente se dice que este mundo tridimensional es el real, la computadora sólo simula gráficos de este tipo, representando en la pantalla las tres dimensiones largo, ancho y la profundidad de las imagenes. En la computación se utilizan los gráficos en 3D para crear animaciones, gráficos, películas, juegos, realidad virtual, diseño, etc.

[9]El proceso de la creación de gráficos tridimensionales comienza con un grupo de fórmulas matemáticas y se convierte en un gráfico en 3D. Las fórmulas matemáticas describen objetos poligonales, tonalidades, texturas, sombras, reflejos, transparencias, translucidez, refraxiones, iluminación (directa, indirecta y global), profundidad de campo, desenfoques por movimiento, ambiente, punto de vista, etc. Toda esa información constituye un modelo en 3D. Hay casos en que se usa las tres dimensiones en computación para simular una realidad virtual, la ciencia y la técnica han aprovechado este recurso para los más variados y complicados proyectos, muestra de ello son tanto los esfuerzos que dedican al tema los principales centros de investigación del mundo entero como el interés que demanda la industria en estas aplicaciones.

. La visión Humana

[10]La visión humana asume el concepto de perspectiva de distintas formas. Probablemente es familiar la idea de las vías del tren totalmente rectas sobre un suelo llano alejándose hacia el horizonte. Los rieles parecen converger hacia un solo punto en la distancia. La convergencia de estas líneas es un rasgo básico de la perspectiva. El caso de las vías es extremo, en pocas observaciones diarias son tan evidentes los efectos de la visión en perspectiva. Es posible que la mente humana no se de cuenta de estos aspectos.

[9] http://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm

[10] http://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm



Título: Vista Rieles Tren Perspectiva
Fuente: http://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm
Elaboración: Juan Carlos Reascos Laverdi.

Los ojos absorben a gran velocidad cantidad de imágenes, que la mente mezcla y compone para obtener una imagen global de la escena. El cerebro organiza los contornos y formas de acuerdo a su relación espacial. En el caso de la imagen de los rieles, nuestra vista dice que éstos convergen y al final se unirán, pero la mente asume que esas líneas son paralelas y así seguirán infinitamente. Esta es una interpretación de la realidad, los rieles realmente son paralelos. Resulta mucho más fácil moverse por un mundo que la mente comprende aunque nuestra vista no.

. La perspectiva

Uno de los objetivos de esta tesis es introducirnos a las técnicas matemáticas y gráficas que subyacen en las aplicaciones de diseño 3D por ordenador y la animación. Estas herramientas permiten el diseño y la visualización de escenas. De esta forma se sintetizan imágenes tridimensionales que incluyen no sólo la geometría de los objetos, sino también sus propiedades ópticas.

Para saber más sobre la Geometría en 3 Dimensiones debemos tener un conocimiento más amplio sobre la visión Humana, de la perspectiva, de cómo funciona nuestro cerebro para realizar las respectivas interpretaciones de las imágenes vistas. De tal forma que todo los avances tecnológicos que se han dado últimamente en el desarrollo de software para manejo tridimensional, han sido posibles gracias a la interpretación de nuestra visión a trabes de éstos.

Actualmente se utilizan muchos y diversos métodos empíricos, mecánicos como también informáticos, para crear imágenes en perspectiva. Existen métodos muy específicos para construir una perspectiva entre ellas están:

. Perspectiva tradicional

La teoría tradicional de la perspectiva, sitúa al observador en un punto estacionario y que mira a otro punto en la distancia denominado centro de visión. Esto es equivalente al emplazamiento de una cámara y del target (punto hacia el que se dirige la cámara) es una escena 3D. La línea que une nuestro ojo y el centro de visión normalmente se denomina línea de visión.

Se pueden trazar líneas desde el punto del observador a cualquiera de los objetos de la escena; estas líneas se proyectan sobre un plano teórico suspendido entre el observador y la escena y que es determinado por el plano de ésta.

El plano sobre el que está el observador se denomina plano de tierra, el suelo o la superficie sobre la que la mayoría de los objetos de la escena descansan. El plano de tierra se extiende en profundidad hasta la altura de la visión, esto es, hasta la altura del horizonte.

La altura del punto de visión, o posición de la cámara, es también la altura del horizonte de la escena. Todas las líneas contenidas en planos paralelos al de tierra convergen en puntos situados en el horizonte. Se puede pensar en el horizonte como un plano infinito que ese extiende en la distancia manteniendo siempre la altura respecto del plano de tierra. A medida que los objetos se alejan en la distancia, parecen estar cada vez más cercanos a reposar sobre el horizonte.

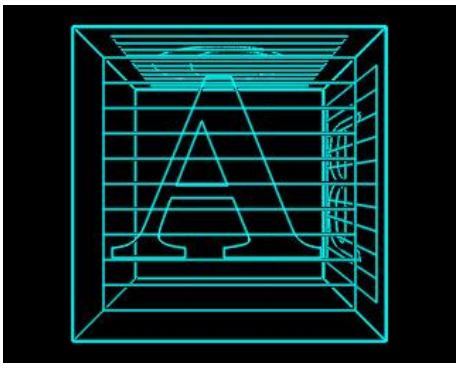
El horizonte es importante porque todas las líneas horizontales (líneas que están sobre planos paralelos al de tierra) visualmente convergen en puntos de fuga situados en él.

Líneas situadas en planos por debajo del punto de vista convergen hacia arriba hasta la línea del horizonte, mientras que líneas sobre el punto de vista convergen hacia abajo. Líneas que están directamente a nivel del punto vista coinciden con el horizonte y se visualizan como una sola línea.

. Perspectiva con un punto de fuga o cónica

[11]El mundo está basado primordialmente en ángulos rectos. La perspectiva tiene su mayor efecto en líneas paralelas y en este tipo de ángulos; por ello, es bastante común hablar de perspectiva en relación con un simple cubo.

Cuando se observa el cubo de forma paralela a una de sus caras, sólo las líneas perpendiculares a nosotros convergen en el horizonte. El punto de fuga reposa en esta línea y coincide con el centro de visión. Las otras aristas del cubo tienen puntos de fuga a una distancia infinita a cada lado (es decir, no existe punto de fuga). Estas líneas no convergen y son paralelas al observador y a su horizonte. Por la existencia de un solo punto de fuga esta vista se denomina, en lengua anglosajona, one point perspective.



Título: Perspectiva con un punto de fuga o cónica.
Fuente: http://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm
Elaboración: Juan Carlos Reascos Laverdi.

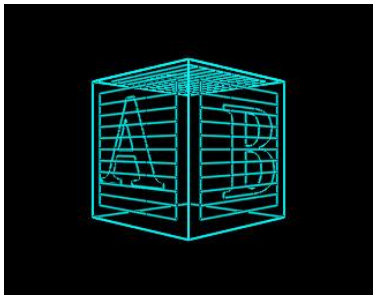
. Perspectiva con dos puntos de fuga.

Si no estamos en un plano paralelo al cubo, existirá un punto de fuga para cada una de las dos caras visibles. Éstos se sitúan fuera del ángulo de visión, en la línea del horizonte, a izquierda y a derecha. Por la existencia de dos puntos de fuga esta visión se denomina two point perspective. Mientras que la perspectiva cónica debe ser paralela a una de las caras del cubo, la perspectiva con 2 puntos de fuga puede situarse desde cualquier ángulo del mismo plano que la anterior. Se ha de tener en cuenta que hay que

[11] http://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm

mantener la línea de visión para cerciorarse que las líneas verticales siguen siéndolo, para ello la cámara y el target deben estar nivelados con el plano de tierra.

Como estos planos verticales permanecen constantes, es muy fácil para los delineantes determinar distancias utilizando la perspectiva con dos puntos de fuga.



Título: Perspectiva con dos puntos de fuga
Fuente: ttp://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm
Elaboración: Juan Carlos Reascos Laverdi.

1.1.4.2 Herramientas Digitales

Desde cuando se emprendió el camino hacia el desarrollo de herramientas para el manejo de gráficos por computador, se ha tenido varias generaciones las cuales han tenido programas donde principalmente se destacaban los siguientes grupos principalmente: juegos de video, animación y efectos especiales siendo estos los que motiven al desarrollo de una industria que crece día a día.

A continuación se mencionará brevemente las principales herramientas desarrolladas para la creación de gráficos tridimensionales.

- ***Autodesk Maya***

También conocido como Maya, es uno de los principales referentes de software 3D, este es un programa informático dedicado al desarrollo de gráficos tridimensionales, efectos especiales y animación. Surgió a partir de la evolución de Power Animator y de la fusión de Alias y Wavefront, dos empresas canadienses dedicadas a los gráficos generados por ordenador. Más tarde Silicon Graphics (ahora SGI), el gigante informático, absorbió a Alias-Wavefront, que finalmente ha sido absorbida por Autodesk.

El programa posee diversas herramientas para modelado, animación, render, simulación de ropa y cabello, dinámicas (simulación de fluidos), etc.

Además Maya es el único software de 3D acreditado con un Oscar gracias al enorme impacto que ha tenido en la industria cinematográfica como herramienta de efectos visuales, con un uso muy extendido debido a su gran capacidad de ampliación y personalización.

Autodesk Maya se comercializa en dos variantes: "Maya Complete" (una versión básica que incluye los módulos de modelado, animación, render, dinamicas/particulas) y "Maya Unlimited" (la versión más avanzada, que dispone de los módulos de la versión "Maya Complete", más los de Fluids, Hair, Cloth, el nuevo NCloth etc.). Ahora la versión Unlimited tiene un precio parecido al de la mayoría de programas de este tipo, pero solía ser considerablemente más caro.

El Maya Personal Learning Edition (PLE) está disponible para uso no comercial, y es gratuita. Las imágenes renderizadas con esta versión tienen el logotipo de la compañía impreso en forma de marca de agua digital.

Jurassic Park, The Abyss y Terminator 2: Día del Juicio Final

[12]Películas en las que se empleó Maya.

102 DALMATIANS

A.I. ARTIFICIAL INTELLIGENCE

ATLANTIS

BEHIND ENEMY LINES

BLACK HAWK DOWN

FINAL FANTASY: THE SPIRITS WITHIN

HARRY POTTER & THE CHAMBER OF SECRETS

[12] http://es.wikipedia.org/wiki/Autodesk_Maya

HARRY POTTER & THE SORCERER'S STONE

ICE AGE

INSPECTOR GADGET

THE IRON GIANT

JURASSIC PARK III

THE LORD OF THE RINGS: THE FELLOWSHIP OF THE RING

THE LORD OF THE RINGS: THE TWO TOWERS

THE MATRIX

MEN IN BLACK II

MINORITY REPORT

- ***Autodesk 3ds Max***

[13](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

[13] http://es.wikipedia.org/wiki/Autodesk_3ds_Max

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.

Actualmente se encuentra en la versión Autodesk 3ds Max 2011.

[14] - *Autodesk® Softimage*

Softimage, co. era una subsidiaria de Avid Technology, Inc., situada en Montreal (Canadá). El 23 de octubre de 2008 fue adquirida por Autodesk por aproximadamente 35 millones de dólares, finalizando así su historia como división e integrándose dentro de Autodesk Media & Entertainment. Produce programas para animación 3D, composición y efectos especiales. Su principal producto es Softimage XSI, empleado para la creación de animaciones por ordenador en nuevas películas, en comerciales y videojuegos.

La compañía fue comprada en 1994 por Microsoft y posteriormente, en 1998, paso a manos de Avid Technology.

Para crear una arquitectura más avanzada, abierta, para mejorar la integración del Mental Ray, y para competir con Maya, Softimage hacia el 2000 desarrolló un paquete de nueva generación llamado Softimage XSI, que remplazo a Softimage 3D. Tiempo más tarde fue absorbido por Autodesk y ahora sobrevive como hermano de 3d studio max y maya en una gran familia.

Actualmente se encuentra en la versión: Autodesk® Softimage® 2011

[14] http://usa.autodesk.com/adsk/servlet/pc/index?siteID=123112&id=13571168

- ***Marionette***

[15]Marionette es el software propietario desarrollado y usado por los Estudios de animación Pixar para la animación de sus películas y cortometrajes. Marionette no está disponible para la venta y es utilizado solamente por Pixar. Consecuentemente poco se sabe, fuera de Pixar, sobre los funcionamientos detallados de este software.

Pixar elige utilizar un sistema propietario en lugar de los productos comerciales disponibles y usados por otras compañías porque puede corregir la codificación del software para resolver sus necesidades. Un ejemplo de esto que corrige se demuestra en el DVD de Los Incredibles; donde en su material extra, se explica que las versiones anteriores del Marionette los modelos generados no podían realizar las transformaciones necesarias por los animadores de modo que el equipo interno del desarrollo del Marionette creó una nueva versión que incluyó esta característica.

- ***Lightwave***

[16]LightWave (o, más correctamente, LightWave 3D) es un programa de gráficos por computadora para Modelado 3D, renderizado y animación.

Lightwave se creó mediante la fusión de un programa de animación llamado Videoscape y una aplicación complementaria de modelado 3D llamada Aegis modeler. La unión de éstos daría lugar al actual LightWave 3D desde el año 1990.

LightWave 3D siempre ha sido multiplataforma y hoy día funciona en Mac OS X y Windows en 32 y 64 bits.

Lightwave 3d se divide en dos subprogramas: Modeler y Layout; En modeler se realiza el modelado del objeto con una filosofía orientada a capas. En Layout se realiza el rigging o configuración del esqueleto de una malla, Dinámicas, FX, Render, configuración

[15] http://www.worldlingo.com/ma/enwiki/es/Marionette_%28software%29
[16] http://es.wikipedia.org/wiki/LightWave

de cámaras y luces. El editado de materiales y texturas se realiza tanto en Modeler como en Layout mediante el menu "Surface Editor".

Modeler y Layout están ligados en funcionamiento por medio de un programa llamado HUB, por ejemplo si tengo cargado en Layout el objeto hecho en modeler, puedo modificarlo desde modeler y los cambios aparecen en Layout automáticamente.

- Blender 3D

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

Actualmente se encuentra en la versión 2.55 beta, en miras de la versión 2.6. Esta trae consigo una mayor velocidad y estabilidad en las herramientas de modelado y la corrección de más de 200 bugs encontrados en la versión 2.53. Pequeñas mejoras que se suman a la pequeña revolución que supuso la versión 2.5, con una profunda remodelación de su código fuente, y con el estreno de una larga lista de nuevas funcionalidades, entre las que se encuentran:

* Interfaz renovada, que facilita la utilización de sus herramientas.

* Personalización de los atajos de teclado.

* Mayor facilidad para la creación de plugins mediante Python.

* Nuevo sistema de acceso a los datos y operaciones de una escena.

* Soporte 64 bits para Windows, Linux y Mac

* Modelado en Multiresolución

* Editores de animación con F-Curves, edición no lineal, splines IK para trayectorias, un nuevo sistema de simulación basada en fluidos para humo y partículas, optimización de raytracing, gestión del color.

[17] http://www.blender.org

1.1.4.3 Actualidad - Estudios de animación.

La animación ha tomado un giro evolutivo hacia la tecnología, pues las producciones que se han venido realizando desde hace 24 años desde cuando en el festival **Siggraph** (conferencia mayor sobre gráficos por computadora en general), se presento un cortometraje animado realizado en 3D por Pixar llamado Luxor Junior - 1986, en este pudieron demostrar que se podía crear de manera digital y basados en los principios clásicos de la animación, una obra totalmente realizada en el computador.

Luxo Jr. convenció tanto al público como a los animadores de las posibilidades de la tecnología de animación por computadora, estableciendo el escenario de una revolución en la narrativa cinematográfica, Esto marco una tendencia y los estudios de animación fijaron su atención en el uso de estas nuevas herramientas tecnológicas.

Entre los principales estudios que se crearon están:

- Pixar

Pixar Animation Studios es una compañía de animación por ordenador especializada en la producción de gráficos en 3D y con sede en Emeryville, California (Estados Unidos). A lo largo de su exitosa historia ha cosechado numerosas condecoraciones por sus cortometrajes, largometrajes y logros técnicos, como 22 premios Óscar de la Academia, 4 Globos de Oro o 3 Grammys.

Pixar fue la creadora del primer largometraje comercial totalmente animado por ordenador, Toy Story, cuyo estreno tuvo lugar el 22 de noviembre de 1995 en Estados Unidos. Después de Toy Story ha producido otras 10 películas: Bichos (1997), Toy Story 2 (1999), Monsters, Inc. (2001), Buscando a Nemo (2003), Los increíbles (2004), Cars (2006), Ratatouille (2007), WALL·E (2008), Up (2009) y Toy Story 3 (2010).

. Historia



Título: Logotipo Pixar Studios
Fuente: http://www.pixar.com
Elaboración: Juan Carlos Reascos Laverdi.

[18]Pixar se fundó como The Graphics Group (una división de Lucasfilm), que se lanzó en 1979 con la contratación del Dr. Ed Catmull del Instituto de Tecnología de Nueva York (NYIT). En NYIT, los investigadores trabajaron en una película experimental llamada The Works, aunque nunca fue estrenada. Cuando el grupo se independizó de LucasFilm, el equipo trabajó en crear al precursor de RenderMan, llamado Motion Doctor que les permitió a los animadores del cel tradicionales trabajar con la animación de la computadora con un mínimo de experiencia.

Seguidamente, el equipo empezó trabajando en las sucesiones de cualquiera de las películas producidas para Lucasfilms o en las películas con Industrial Light & Magic. Después de años de éxito en notable investigación, e hitos importantes en las películas como en Star Trek II: La ira de Khan y Young Sherlock Holmes] el grupo fue comprado en 1986 por Steve Jobs poco después de que dejara Apple Inc. quien pagó US$5 millones a George Lucas e invirtió $5 millones de dólares de capital en la compañía. La compañía recientemente independiente se encabezó por Dr. Catmull, Presidente y Gerente, y Dr. Alvy Ray Smith, Vicepresidente Ejecutivo y Director. Los trabajos sirvieron como base del presidente.

[18] http://es.wikipedia.org/wiki/Pixar

Inicialmente, Pixar era una compañía de hardware gráfico cuyo producto base fue la Pixar Image Computer, un sistema vendido sobre todo a agencias estatales y la comunidad médica. Uno de los principales compradores de estas computadoras eran los estudios Disney, que utilizaban el dispositivo como parte de sus proyectos secretos CAPS, usando las máquinas y el software para emigrar el proceso de animación de tinta a un método automatizado y más eficiente.

La etapa de cortometrajes se vio interrumpida cuando en mayo de 1991 Pixar y Disney unieron fuerzas mediante un contrato en el que se estipulaba la realización de dos largometrajes. Después del estreno de estas, se realizó un nuevo contrato para cinco películas más.

Según esto ambas compañías compartirían gastos y beneficios a partes iguales donde Pixar obtenía además de parte de las ganancias, que su nombre figurara como "realizadora" pero no como propietaria.

La primera en ver la luz fue Toy Story, dirigida por el mismo John Lasseter antiguo trabajador de Walt Disney, recibiendo un Oscar especial por la primera película animada de la historia: Toy Story. Fue la película más taquillera de 1995 recaudando más de 360 millones de dólares, además de ser el primer largometraje de animación integral por ordenador y la primera película digital en recibir un Oscar. Con Monsters, Inc.transformó en imágenes la perfección. Los 2.320.413 pelos de Sully hicieron temblar de nuevo las taquillas y alzar a Pixar a lo más alto del panorama cinematográfico. El proceso digital. Todo comienza con el planteamiento de las diferentes posibilidades y su enfoque narrativo.

En 2004 Pixar y los estudios Disney rompen relaciones.

El 24 de enero de 2006 Disney adquiere Pixar Animation Studios por 7.400 millones de dólares y cede el control de su estudio de animación a los directores creativos de Pixar. Steve Jobs se convierte en uno de los mayores accionistas de Disney.

Sin embargo como el intercambio de activos es una operación de Disney y Pixar, decidieron extender su contrato de distribución hasta 2007 con una sola producción, "Ratatouille" .

. La creación

Los encargados de producción proceden a establecer cuál será el aspecto de los personajes, los decorados, los colores y de todo aquello que deba animarse, desde el protagonista principal hasta una simple silla.

Es ahora cuando comienza realmente la fase de producción, en la que intervienen más de mil ordenadores con un total de 2 Terabits de memoria RAM y 60 de espacio en disco.

La etapa final es la renderización, una operación de 6 horas que incluso puede llegar a durar hasta 90 por cada fotograma. Para finalizar y como en cualquier otro filme se añaden los efectos sonoros y la banda sonora, compuestas en su mayoría por Randy Newman, ganador del Oscar a la mejor canción original por Monsters Inc. y nominado en varias ocasiones a la mejor partitura. Y es que si por algo destaca Pixar es por su empeño a la hora de realizar cualquier tipo de proyecto, en los que siempre resalta una sola pero difícil palabra: calidad.

- DreamWorks SKG Animation



Título: Logotipo Dreamworks
Fuente: http://www.dreamworksanimation.com/
Elaboración: Juan Carlos Reascos Laverdi.

[19]DreamWorks SKG Animation, Inc. Es un estudio de animación independiente estadounidense, que producen principalmente una serie de éxitos comerciales por computadora de películas animadas, incluyendo Shrek, El Espanta Tiburones, Madagascar, Bee Movie, Kung Fu Panda, Monsters vs. Aliens y Cómo Entrenar a tu Dragón. Se formó por la fusión de la función de división de animación de DreamWorks SKG y Pacific Dates and Images (PDI). Originalmente bajo la bandera de DreamWorks SKG, se inició como una empresa pública en 2004.

. Historia 1994-2003

El 12 de octubre de 1994, DreamWorks SKG fue creada y fundada por un trío de jugadores de entretenimiento, director y productor Steven Spielberg, la música David Geffen ejecutivo y ex ejecutivo de Disney Jeffrey Katzenberg. PDI creado muchos logotipos animados para la televisión y anuncios para empresas como NBC y Sky Movies. Que pasó en los efectos visuales de películas a partir de 1991 con una contribución a Terminator 2: Judgment Day. La nueva unidad de producción generados por computadora a partir de las películas Antz en 1998. En el mismo año, DreamWorks SKG produjo El Príncipe de Egipto utilizando técnicas de animación tradicional.

. 2004-actualidad

De 2004 a 2008, el estudio se dedicó exclusivamente a la producción de películas de animación CG en casa y se comprometió a realizar 2 largometrajes de animación por ordenador al año. No se esperan más largometrajes en las tradicionales 2D.

DreamWorks Animation también tiene una asociación con Aardman animation, una compañía de animación de stop-motion en Bristol, Inglaterra. En esta asociación DreamWorks que participa en la producción de películas de stop-motion en Bristol, y también Aardman que participa en algunas de las películas CG realizadas en los EE.UU.

[19] http://www.dreamworksanimation.com/

Esta alianza terminó después del estreno de Lo que el agua se Llevó en noviembre de 2006.

- Blue Sky Studios



Título: Logotipo Bluesky
Fuente: http:// www.blueskystudios.com
Elaboración: Juan Carlos Reascos Laverdi.

[20]Blue Sky Studios es un estudio de animación CGI especializado en la animación de personajes generada por ordenador y renderizado foto-realistas de alta resolución. Además de sus largometrajes de animación, que incluyen Ice Age (2002), Robots (2005) y Horton (2008), Blue Sky ha trabajado en muchas películas de alto perfil, sobre todo en la integración de la acción en vivo con animación generada por computadora.

. Historia

Blue Sky fue fundada en febrero de 1987 por una serie de artistas y técnicos que habían trabajado en la película de Disney Tron, mientras trabajaban en MAGI/Synthavision. A lo largo de finales de los años 1980 y la década de 1990, el estudio se concentró en la producción de anuncios de televisión y efectos visuales para películas. Algunos de los anuncios más memorables en los que Blue Sky trabajó durante este período de tiempo fueron un spot de Chock full o'Nuts con un grano de café parlante, y una serie de anuncios Usando sus herramientas propietarias de animación, el estudio

[20] http:// www.blueskystudios.com

elaboró más de 200 spots para clientes como Chrysler, M&M/Mars, General Foods, Texaco y el Cuerpo de Marines de los Estados Unidos.

Después de su adquisición por 20th Century Fox en 1997, Blue Sky se fundió con la casa de VFX de Los Ángeles, VIFX. Finalmente, VIFX fue vendida a Rhythm & Hues Studios, y Blue Sky se volvió a centrar únicamente en tareas de animación.

. Tecnología

El estudio es notable por su propietario Renderer CGI Studio. Inicialmente desarrollado por Eugene Troubetzkoy, Carl Ludwig, Tom Bisogno y Michael Ferraro, CGI Studio se caracteriza por su uso de trazado de rayos en lugar del renderizado REYES que prevalecía en toda la industria del CG.

. Filmografía

* Rio (2011)
* Ice Age: La edad de hielo (2002, candidato a un premio Óscar)
* Robots (2005)
* Ice Age 2: El deshielo (2006)
* Dr. Seuss' Horton hears a Who! (2008)
* Ice Age 3: El origen de los dinosaurios (2009)
* Rio (2011)
* Ice Age 4 (2012)
* Leaf Men (2012)
* Spore (TBA)

- Blender Foundation & Instituto Blender.

La Fundación Blender (en inglés: "Blender Foundation"), es una organización sin ánimo de lucro que se encarga del desarrollo de Blender, un programa de código abierto para modelado 3D. Dirigida por Ton Roosendaal, el autor original del programa, fue fundada gracias a las donaciones, que permiten a Ton Roosendaal dedicarse profesionalmente al desarrollo de Blender y mantienen la infraestructura, posibilitando el desarrollo del programa, su distribución y otras actividades. La meta principal de la fundación es brindar acceso a la tecnología 3D gracias a Blender.

Como dueños del sitio web de Blender, la fundación entrega varios recursos para apoyar a la comunidad que utiliza Blender. En particular, organiza un "Conferencia Blender" anual en Amsterdam para discutir el futuro del programa, y atiende un stand en el SIGGRAPH, una conferencia mayor sobre gráficos por computadora en general.

Se han creado 4 proyectos que son: 3 cortometrajes y 1 juego los que serán descritos a continuación.

. Elephants Dream



Título: Elephants dream
Fuente: http:// www.elephantsdream.org
Elaboración: Juan Carlos Reascos Laverdi.

Elephants Dream (Sueño de los Elefantes) es el primer cortometraje de la historia de animación realizada prácticamente usando sólo software libre y que además se ofrece de forma pública con licencia o condiciones basada en la filosofía del conocimiento libre. Duración del corto : 11 minutos.

. Producciones

La producción comenzó en septiembre de 2005 y fue financiada y desarrollada por el estudio Orange con un equipo internacional de siete artistas y animadores. Originalmente conocida por el nombre de Machina, fue rebautizada como Elephants Dream para cuadrar más exactamente con el elemento central del argumento.

La película fue anunciada en mayo de 2005 por Ton Roosendaal, presidente de la Fundación Blender y líder en el desarrollo principal de Blender, un conjunto de herramientas para el desarrollo de modelos, animaciones o escenas en 3D. Poco después del citado anuncio, la Fundación Blender ofreció la posibilidad de reservar la película en soporte físico (DVD) como parte de su propia auto financiación, además de ofrecer como extra la posibilidad de aparecer en los créditos finales de la película.

Los objetivos del proyecto son varios:

- exhibir o promocionar el talento profesional de las personas que integran el equipo,
- demostrar la validez de los programas libres
- que el resultado final revierta y sirva para mejorar aún más en el propio software especialmente Blender, analizando y añadiendo nuevas mejoras, resolviendo errores, etc
- ofrecer por primera vez un film de animación de gran calidad, de forma que el material empleado se pueda obtener, modificar o distribuir libremente

Durante el desarrollo de Elephants Dream y, a fin de acrecentar la sofisticación del resultado, fueron añadidas o mejoradas numerosas características, como un avanzado sistema de animación, renderizado por pasadas, mejoras en el motor gráfico de

simulación de partículas, un nuevo esquema de materiales y post-procesamiento nodal que permite complejos efectos como la simulación física de pelo, piel, fusión de fotogramas, aplicaciones de filtros no lineales, etc.

El contenido de la película es libre, es decir está publicado con una licencia Creative Commons que ofrece el código fuente del trabajo 3D utilizado (modelos, texturas, animaciones..) de modo que, además de servir como promoción de Blender y el software libre, pueda devolver a la comunidad parte de su propio trabajo. Se puede conseguir en un soporte físico o bien descargarse desde la página web del proyecto sin coste alguno.

La presentación del film fue el 24 de marzo de 2006 en la ciudad europea de Ámsterdam. Posteriormente será proyectada en varios festivales internacionales de cinematografía, incluido Cannes.

"Además la Blender Foundation ha creado 2 cortometrajes y un juego totalmente liberados bajo la licencia CC (creative commons)".

Los mismos que serán presentados más adelante.



Título: Big Buck Bunny
Fuente: http:// www.bigbuckbunny.org/
Elaboración: Juan Carlos Reascos Laverdi.



Título: Yo frankie (game)
Fuente: http:// www.bigbuckbunny.org/
Elaboración: Juan Carlos Reascos Laverdi.



Título: Sintel
Fuente: http:// www.sintel.org/
Elaboración: Juan Carlos Reascos Laverdi.

- Festivales

Actualmente existen varios festivales de animación entre los que tenemos:

Los festivales de animación acogen tanto largometrajes comerciales como cortometrajes de todos los tipos, y son el principal medio de difusión para estos últimos. Los principales festivales de animación del mundo son los siguientes:

* Siggraph, 2010 EEUU Loas Angeles.

* Annecy, Francia.

* Blender Conference, Holanda.

* Che Blender, Argentina.

* Zagreb, Croacia.

* Anima Mundi, en São Paulo Brasil.

* Animacam, Festival Internacional de Animación Online.

En el mundo hispanoparlante los más relevantes son:

* Anima, en Córdoba, Argentina.

* LOOP - Festival Latinoamericano de Animación & Videojuegos Colombia.

* Animadrid, en Pozuelo de Alarcón y Animacor, en Córdoba, España.

* Creanimax, En Guadalajara, Jalisco, México.

* Tecnotoon Animation Fest, En Ciudad de México, Distrito Federal.

* Animación a la Carta, en Caracas , Venezuela.

* ANIMEC,Festival de Animación, en Quito , Ecuador.



Título: Animec
Fuente: http:// www.animecfest.com/
Elaboración: Juan Carlos Reascos Laverdi.

1.2 SOFTWARE LIBRE



Título: Software Libre

Elaboración: Juan Carlos Reascos Laverdi.

El software libre (en inglés free software, esta denominación también se confunde a veces con gratis por la ambigüedad del término en el idioma inglés) es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, cambiado y redistribuido libremente. Según la Free Software Foundation, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar el software y distribuirlo modificado.

El software libre suele estar disponible gratuitamente, o al precio de costo de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a "software gratuito" (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("software comercial"). Análogamente, el "software gratis" o "gratuito" incluye en ocasiones el código fuente; no obstante, este tipo de software no es libre en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

Existen actualmente programas libres para computador como: sistemas operativos de estaciones y también de servidores, no está por demás decir que los servidores mas robustos usan software libre como sus sistemas operativos por ejemplo Google, existen varias distribuciones para este tipo de trabajo como son Centos, Red Hat y para

estaciones de trabajo de igual forma podemos encontrar distribuciones como Ubuntu, Open Suse, etc, además de programas de las mas variadas necesidades como Open Office que es un suite ofimática símil de Microsoft Office, otro ejemplo sería Gimp que es un editor de imágenes símil de Photoshop, claro que Gimp es libre, así por dar un ejemplo de todos las herramientas que se dispone en el mundo del software libre.

1.2.1 LICENCIA GNU-GPL (General Public Licence).

Las licencias que cubren la mayor parte del software están diseñadas para quitar la libertad de compartirlo y modificarlo. Por el contrario, una licencia GNU-GPL o licencia libre, pretende garantizarle la libertad de compartir y modificar software libre, para asegurar que el software es libre para todos sus usuarios. Esta Licencia se aplica a la mayor parte del software de la Free Software Foundation y a cualquier otro programa si sus autores se comprometen a utilizarla.

Cuando hablamos de licencias libres, se está refiriéndo a libertad, no a precio. Estas Licencias están diseñadas para asegurarnos de que tenga la libertad de distribuir copias de software libre (y cobrar por ese servicio si quiere), de que reciba el código fuente o que pueda conseguirlo si lo quiere, de que pueda modificar el software o usar fragmentos de él en nuevos programas libres, y de que sepa que puede hacer todas estas cosas.

1.2.1.1 DEFINICION



Título: logotipo GNU
Fuente: http:// www.gnu.org/
Elaboración: Juan Carlos Reascos Laverdi.

[21]La Licencia Pública General de GNU o más conocida por su nombre en inglés GNU General Public License o simplemente sus siglas del inglés GNU GPL, es una licencia creada por la Free Software Foundation en 1989 (la primera versión, actualmente se encuentra en la versión 3), y está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

Existen varias licencias "hermanas" de la GPL, como la licencia de documentación libre de GNU (GFDL), la Open Audio License, para trabajos musicales, etcétera, y otras menos restrictivas, como la MGPL, o la LGPL (Lesser General Publical License, antes Library General Publical License), que permiten el enlace dinámico de aplicaciones libres a aplicaciones no libres.

1.2.1.2 Principales tipos de licencias libres

Existen varios tipos de licencias libres entre las cuales se tiene: GPL, AGPL, BSD, MPL y derivadas, Copyleft y Cretive Commons. A continuación se detallara las 2 más importantes y que son de mucha utilidad para trabajos como el que se desarrolla en esta tesis.

Copyleft



Título: logotipo Copyleft
Fuente: http://www.gnu.org/copyleft/copyleft.es.html
Elaboración: Juan Carlos Reascos Laverdi.

[21] http:// www.gnu.org/

[22]Es una característica de algunas licencias utilizadas para regular las restricciones impuestas por el derecho de autor de obras o trabajos, tales como programas informáticos, arte, cultura y ciencia, es decir prácticamente casi cualquier tipo de producción creativa.

El nombre surge como oposición al copyright tradicional. Se considera que una licencia es copyleft cuando además de otorgar permisos de copia, modificación y redistribución de la obra protegida, contiene una cláusula que impone la misma licencia a las copias y a las obras derivadas.

Sus partidarios la proponen como alternativa a las restricciones que imponen las normas planteadas en los derechos de autor, a la hora de hacer, modificar y distribuir copias de una obra determinada. Se pretende garantizar así una mayor libertad para que cada receptor de una copia, o una versión derivada de un trabajo, pueda, a su vez, usar, modificar y redistribuir tanto el propio trabajo como las versiones derivadas del mismo.

Muchas licencias de software libre, como las que utilizan los sistemas operativos BSD, el Sistema de Ventanas X y el servidor web Apache, no son licencias copyleft puesto que no exigen al titular de la licencia la distribución de los trabajos derivados bajo la misma licencia. En la actualidad, se debate sobre qué licencia proporciona mayor grado de libertad. En este debate se consideran cuestiones complejas como la propia definición de libertad y qué libertades son más importantes. A veces se dice que las licencias copyleft tratan de maximizar la libertad de todos aquellos destinatarios potenciales en el futuro (inmunidad contra la creación de software privativo), mientras que las licencias de software libre sin copyleft maximizan la libertad del destinatario inicial (libertad para crear software privativo). Con un enfoque similar, la libertad del destinatario (que es limitada por el copyleft) puede distinguirse de la libertad del propio software (la cual es garantizada por el copyleft).

Licencia BSD

[22] http://www.gnu.org/copyleft/copyleft.es.html

Esta licencia muy conocida por su posición "aun más libre" que la GPL es siempre un punto de discusión cuando se habla de licencias de Software Libre.

Técnicamente una licencia del tipo BSD no tiene Copyleft, por lo que es posible hacer versiones modificadas no libres. Esto se ha dado mucho en la práctica, por ejemplo, Microsoft uso código de BSD en el sistema de red de Windows y muchos componentes de FreeBSD han sido usados en MacOS X.

La oposición entre la licencia BSD y la GPL es clara en ese punto: la primera considera que si el software es libre no debe imponer ninguna restricción en su distribución aunque esto signifique que un "alguien use el software para su propio beneficio y no comparta el código", la GPL ve en esto un problema: el software libre termina favoreciendo a quienes no les importa la libertad de los usuarios.

Existe otro punto por el cual las licencias BSD no eran consideradas libres por GNU: su cláusula de publicidad (también conocida como "4-clause") que obligaban a hacer mención a la Universidad de Barkeley, lo cual era una restricción adicional.

En el año 1999, esta cláusula fue eliminada y a partir de allí la licencia BSD se fragmento en dos tipos: las que incluían la "4-clause" y las que no. Es por eso que muchas veces encontramos software bajo "BSD-old"/"4-Clause BSD" y "BSD-new"/"BSD Revised".

Creative Commons



Título: logotipo Creative Commons
Fuente: http://creativecommons.org/
Elaboración: Juan Carlos Reascos Laverdi.

[23]Las licencias Creative Commons o CC están inspiradas en la licencia GPL (General Public License) de la Free Software Foundation. No son, sin embargo, un tipo de licenciamiento de software. La idea principal es posibilitar un modelo legal ayudado por herramientas informáticas, para así facilitar la distribución y el uso de contenidos.

Existe una serie de licencias Creative Commons, cada una con diferentes configuraciones o principios, como el derecho del autor original a dar libertad para citar su obra, reproducirla, crear obras derivadas, ofrecerla públicamente y con diferentes restricciones, como no permitir el uso comercial o respetar la autoría original.

Una de las licencias que ofrecía Creative Commons es la que llevaba por nombre "Developing Nations" (Naciones en Desarrollo), la cual permitía que los derechos de autor y regalías por el uso de las obras se cobraran sólo en los países desarrollados del primer mundo, mientras que se ofrecían de forma abierta en los países en vías de desarrollo. Esta licencia ha sido retirada por problemas comerciales.

Aunque originalmente fueron redactadas en inglés, las licencias han sido adaptadas a varias legislaciones en otros países del mundo. Entre otros idiomas, han sido traducidas al español, al portugués, al gallego, al euskera y al catalán a través del proyecto Creative Commons International. Existen varios países de habla hispana que están involucrados en este proceso: España, Chile, Guatemala, Argentina, México, Perú, Colombia, Puerto Rico y Ecuador ya tienen las licencias traducidas y en funcionamiento, en tanto que Venezuela se encuentra en proceso de traducción e implementación de las mismas. Brasil también tiene las licencias traducidas y adaptadas a su legislación.

1.2.1.3 Sistema Operativo Linux

[24]GNU/Linux es uno de los términos empleados para referirse a la combinación del núcleo o kernel libre similar a Unix denominado Linux, que es usado con herramientas de sistema GNU. Su desarrollo es uno de los ejemplos más prominentes de software libre; todo su código fuente puede ser utilizado, modificado y redistribuido libremente por

[23] http://creativecommons.org/

[24] http://www.gnu.org/gnu/linux-and-gnu.es.html

cualquiera bajo los términos de la GPL (Licencia Pública General de GNU) y otra serie de licencias libres.

A pesar de que Linux (núcleo) es, en sentido estricto, el sistema operativo, parte fundamental de la interacción entre el núcleo y el usuario (o los programas de aplicación) se maneja usualmente con las herramientas del proyecto GNU. Sin embargo, una parte significativa de la comunidad, así como muchos medios generales y especializados, prefieren utilizar el término Linux para referirse a la unión de ambos proyectos.

A las variantes de esta unión de programas y tecnologías, a las que se les adicionan diversos programas de aplicación de propósitos específicos o generales se las denomina distribuciones. Su objetivo consiste en ofrecer ediciones que cumplan con las necesidades de un determinado grupo de usuarios. Algunas de ellas son especialmente conocidas por su uso en servidores y supercomputadoras, donde tiene la cuota mas importante del mercado. Según un informe de IDC, GNU/Linux es utilizado por el 78% de los principales 500 servidores del mundo, otro informe le da una cuota de mercado de % 89 en los 500 mayores supercomputadores. Con menor cuota de mercado el sistema GNU/Linux también es usado en el segmento de las computadoras de escritorio, portátiles, computadoras de bolsillo, teléfonos móviles, sistemas embebidos, videoconsolas y otros dispositivos.

HISTORIA

El proyecto GNU, que fue iniciado en 1983 por Richard Stallman, tiene como objetivo el desarrollo de un sistema operativo Unix completo y compuesto enteramente de software libre. La historia del núcleo Linux está fuertemente vinculada a la del proyecto GNU. En 1991 Linus Torvalds empezó a trabajar en un reemplazo no comercial para MINIX que más adelante acabaría siendo Linux.

Cuando Torvalds liberó la primera versión de Linux, el proyecto GNU ya había producido varias de las herramientas fundamentales para el manejo del sistema operativo, incluyendo un intérprete de comandos, una biblioteca C y un compilador, pero como el proyecto contaba con una infraestructura para crear su propio sistema operativo,

el llamado Hurd, y este aún no era lo suficiente maduro para usarse, comenzaron a usar a Linux a modo de continuar desarrollando el proyecto GNU, siguiendo la tradicional filosofía de mantener cooperatividad entre desarrolladores. El día en que se estime que Hurd es suficiente maduro y estable, será llamado a reemplazar a Linux.

Entonces, el núcleo creado por Linus Torvalds, quien se encontraba por entonces estudiando en la Universidad de Helsinki, llenó el "espacio" final que había en el sistema operativo de GNU.

La designación "Linux" al principio fue usada por Torvalds sólo para el núcleo. El núcleo fue, sin embargo, con frecuencia usado junto con otro software, especialmente con el del proyecto de GNU. Esta variante de GNU rápidamente se hizo la más popular, ya que no había ningún otro núcleo libre que funcionara en ese tiempo. Cuando la gente comenzó a referirse hacia esta recopilación como "Linux", Richard Stallman, el fundador del proyecto de GNU, solicitó que se usara el nombre GNU/Linux, para reconocer el rol del software de GNU. En junio de 1994, en el boletín de GNU, Linux fue mencionado como un "clon libre de UNIX", y el Proyecto Debian comenzó a llamar a su producto GNU/Linux. En mayo de 1996, Richard Stallman publicó al editor Emacs 19.31, en el cual el tipo de sistema fue renombrado de Linux a Lignux. Esta "escritura" fue pretendida para referirse expresamente a la combinación de GNU y Linux, pero esto pronto fue abandonado en favor de "GNU/Linux".

Popularidad

La creciente popularidad de GNU/Linux se debe, entre otras razones, a su estabilidad, al acceso al código fuente (lo que permite personalizar el funcionamiento y auditar la seguridad y privacidad de los datos tratados), a la independencia de proveedor, a la seguridad, a la rapidez con que incorpora los nuevos adelantos tecnológicos (IPv6, microprocesadores de 64 bits), a la escalabilidad (se pueden crear clusters de cientos de computadoras), a la activa comunidad de desarrollo que hay a su alrededor, a su interoperatibilidad y a la abundancia de documentación relativa a los procedimientos.

Dentro del segmento de supercomputadoras, a noviembre de 2009, el uso de este sistema ascendió al 89,2% de las computadoras más potentes del mundo por su confiabilidad, seguridad y libertad para modificar el código, De acuerdo con TOP500.org.

GNU/Linux, además de liderar el mercado de servidores de Internet debido, entre otras cosas, a la gran cantidad de soluciones que tiene para este segmento, tiene un crecimiento progresivo en computadoras de escritorio y portátiles. Además, es el sistema base que se ha elegido para el proyecto OLPC: One Laptop Per Child.

Distribuciones

Una distribución Linux (coloquialmente llamada distro) es una distribución de software basada en el núcleo Linux que incluye determinados paquetes de software para satisfacer las necesidades de un grupo específico de usuarios, dando así origen a ediciones domésticas, empresariales y para servidores. Por lo general están compuestas, total o mayoritariamente, de software libre, aunque a menudo incorporan aplicaciones o controladores propietarios.

Además del núcleo Linux, las distribuciones incluyen habitualmente las bibliotecas y herramientas del proyecto GNU y el sistema de ventanas X Window System. Dependiendo del tipo de usuarios a los que la distribución esté dirigida se incluye también otro tipo de software como procesadores de texto, hoja de cálculo, reproductores multimedia, herramientas administrativas, etcétera. En el caso de incluir herramientas del proyecto GNU, también se utiliza el término distribución GNU/Linux.

Distribuciones más usadas para PC de escritorio Actualmente.

* Ubuntu, Fedora, OpenSuse

Distribuciones más usadas para equipos portátiles

* Ubuntu, openSUSE, Linux Mint

Distribuciones más usadas para Workstations.

* Novell SLED, RHEL Desktop

Distribuciones más usadas para Servidores.

* RHEL Server, Novell SLES, CentOS, Red Hat

Distribuciones LiveCD de linux

* KNOPPIX

* Ubuntu, Mandriva, DamSmall Linux

Distribuciones dirigidas hacia la seguridad

* BackTrack, Wifiway, Wifislax

Distribuciones más usadas para multimedia

* Ubuntu Studio.

UBUNTU (Distribución).

[25]Ubuntu es una distribución Linux basada en Debian GNU/Linux que proporciona un sistema operativo actualizado y estable para el usuario medio, con un fuerte enfoque en la facilidad de uso e instalación del sistema. Al igual que otras distribuciones se compone de múltiples paquetes de software normalmente distribuidos bajo una licencia libre o de código abierto. Estadísticas web sugieren que el porcentaje de mercado de Ubuntu dentro de las distribuciones Linux es de aproximadamente 50%, y con una tendencia a subir como servidor web.

Está patrocinado por Canonical Ltd., una compañía británica propiedad del empresario sudafricano Mark Shuttleworth que en vez de vender la distribución con fines lucrativos, se financia por medio de servicios vinculados al sistema operativo y vendiendo soporte técnico. Además, al mantenerlo libre y gratuito, la empresa es capaz de aprovechar los desarrolladores de la comunidad en mejorar los componentes de su sistema operativo. Canonical también apoya y proporciona soporte para cuatro derivaciones de Ubuntu: Kubuntu, Xubuntu, Edubuntu y la versión de Ubuntu orientada a servidores (Ubuntu Server Edition).

Su eslogan es Linux for Human Beings (Linux para seres humanos) y su nombre proviene de la ideología sudafricana Ubuntu («humanidad hacia otros»).

[25] http://www.ubuntu.com

Cada seis meses se publica una nueva versión de Ubuntu la cual recibe soporte por parte de Canonical, durante dieciocho meses, por medio de actualizaciones de seguridad, parches para bugs críticos y actualizaciones menores de programas. Las versiones LTS (Long Term Support), que se liberan cada dos años, reciben soporte durante tres años en los sistemas de escritorio y cinco para la edición orientada a servidores.

Existen distribuciones que están soportadas comercialmente, como Fedora (Red Hat), openSUSE (Novell), Ubuntu (Canonical Ltd.), Mandriva, y distribuciones mantenidas por la comunidad como Debian y Gentoo.

1.2.2 Animación y Software libre

Bajo el slogan "El conocimiento es para todos", Organismos como la Fundación Blender ha venido liberando todo el material realizado en cada uno de sus cortometrajes animados: modelos, riggs, props, escenarios y demás elementos son entregados de forma libre para que toda la comunidad y las personas que deseen aprender, estudiar y tal vez crear sus propias historias con los mismos personajes lo puedan realizar sin preocuparse de caer en demandas por uso indebido de material ya que estos están bajo la licencia CC.

1.2.2.1 Animación 2D y Software libre.

Al igual que en el software comercial existe Flash que se trata de una herramienta en la que uno de sus usos es animación 2D, en software libre también hay varias herramientas creadas para ese fin, a continuación están las mas herramientas más significativas.

Synfig (Herramienta de Animación 2D)



Título: logotipo Synfig
Fuente: http:// www.synfig.org
Elaboración: Juan Carlos Reascos Laverdi.

Última versión estable	0.62.02
Sistema operativo	Multiplataforma
Licencia	GPL

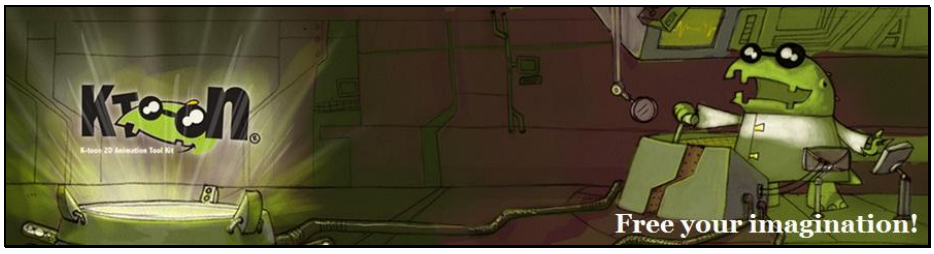
[26]Synfig es una poderosa herramienta de animación y diseño 2D vectorial, programa opensource creado por Robert Quattlebaum, con la ayuda adicional de Adrian Bentley, diseñado para producir películas animadas con pocas personas y recursos.

. Historia

Synfig fue diseñado originalmente para el ahora difunto Voria Studios, pero fue lanzado el 2005 bajo la licencia open source GNU General Public License.

La meta de los programadores era crear una herramienta capaz de producir una película de animación de calidad con menos gente y recursos. Al estar basado en vectores elimina la tarea del tweening manual, produciendo una suave y fluida animación sin que el animador tenga que dibujar cada cuadro individualmente.

- **KToon (Herramienta de Animación 2D)**



Título: logotipo Ktoon
Fuente: http://www.ktoon.net/

[26] http:// www.synfig.org

Sistema operativo: Unix-like
Licencia: GNU General Public License

[27]KTooN es una aplicación informática para el diseño y creación de animaciones 2D inspirada por animadores para animadores. Esta herramienta, desarrollada bajo las dinámicas de una comunidad abierta, se encuentra cubierta por los términos de la licencia GPLv3, es decir, que se trata de Software Libre.

. Historia

Este proyecto nace en Cali (Colombia), en el año 2002, como una iniciativa de dos jóvenes emprendedores miembros de las empresas Toonka Films y Soluciones Kazak, en busca de crear una herramienta de software que incentivara la evolución de la incipiente industria de la animación 2D a nivel Latinoamericano, en ese momento. En función de este objetivo, se determina desde el inicio que la aplicación será de carácter libre y compatible con la mayor cantidad de sistemas operativos, dando prioridad a plataformas libres como GNU Linux.

Durante esta primera fase de planificación, se escoge el nombre de "KTooN" para la herramienta, se realiza el diseño de las primeras interfaces del programa y se escoge Qt como el lenguaje de programación para la creación del software.

Dentro de las funcionalidades que hacen parte de la ruta de desarrollo del programa se encuentran:

* Tweening

* Partículas

* Soporte de Sonido

* Librería de Objetos prediseñados

[27] http://www.ktoon.net

* Soporte de Huesos (Bones), Pivotes y Keyframes

* Y Morphing entre otras.

Ktoon no incluye lenguajes de programación como (No ActionScript for you!), interactividad y demás, ya que actualmente está planteado como un software de animación; pero en planes si que lo incluyen y, mientras continué recibiendo el apoyo de la comunidad muy probablemente se tendrían en el futuro.

Se encuentra solo disponible para Linux al momento.

Pencil (animación digital tradicional 2D)



Título: logotipo Pencil
Fuente: http://www.pencil-animation.org/
Elaboración: Juan Carlos Reascos Laverdi.

Licencia: GNU General Public License
Sistema operativo: Multiplataforma

[28]Pencil es una aplicación de dibujo y animación 2D, disponible para GNU/Linux, Mac OS X y Windows, que permite realizar animaciones de forma tradicional, fotograma a fotograma para unirlos y formar la animación deseada, utilizando bitmaps y gráficos vectoriales.

Pencil no está pensado para realizar animaciones para web, si no que está pensado para realizar animaciones tradicionales de la misma forma que se las realiza con lápiz y papel.

[28] http://www.pencil-animation.org

1.2.2.2 Animación 3D y Software libre.

Wings 3D (modelado digital 3D)



Título: logotipo Wings 3D
Fuente: http://www.wings3d.com/
Elaboración: Juan Carlos Reascos Laverdi.

Licencia: GNU General Public License
Sistema operativo: Multiplataforma

Wings 3D es un programa de modelado 3D libre inspirado en otros programas similares, como Nendo y Mirai, ambos de Izware.

Está disponible para la mayoría de plataformas, incluyendo Windows, Linux y Mac, utilizando el entorno de Erlang.

Está diseñado para modelar y texturizar elementos formado con un número de polígonos menor. Cuando se compara con otros programas de 3D (como puede ser Blender) se observan las diferencias existentes sobre todo en la interfaz gráfica de usuario, la cual aporta una mayor flexibilidad a cambio de ciertas limitaciones en algunas áreas. Por ejemplo, Wings 3D no puede manejar animaciones, sólo trae un renderizador de OpenGL , se puede exportar los modelos tridimensionales para poder ser usados en animaciones en programas externos como Blender 3D.

Sin embargo, Wings 3D es de muy fácil manejo y un sistema de iconos muy intuitivo. Aun careciendo de un potente renderizador, Wings 3D puede combinarse con otros programas como POV-Ray, YafRay o Kerkythea para realizar imágenes de alta calidad.

Blender 3D



Título: logotipo Blender 3D
Fuente: http://www.blender.org/
Elaboración: Juan Carlos Reascos Laverdi.

Licencia: GNU General Public License
Sistema operativo: Multiplataforma

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

El programa fue inicialmente distribuido de forma comercial, 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.

. Origen

Originalmente, el programa fue desarrollado como una aplicación propia por el estudio de animación holandés NeoGeo; el principal autor, Ton Roosendaal, fundó la empresa "Not a Number Technologies" (NaN) en junio de 1998 para desarrollar y distribuir el programa.

[29] http://www.blender.org

La compañía llegó a la bancarrota en 2002 y los acreedores acordaron ofrecer Blender como un producto de código abierto y gratuito bajo los términos de la GNU GPL a cambio de 100.000 €. El 18 de julio de 2003, Roosendaal creó sin ánimo de lucro la Fundación Blender para recoger donaciones; el 7 de septiembre se anuncia la recaudación como exitosa (participaron también ex empleados de NaN) y el código fuente se hizo público el 13 de octubre.

El domingo 13 de octubre de 2002, Blender fue liberado al mundo bajo los términos de la Licencia Pública General de GNU (GPL). El desarrollo de Blender continúa hasta nuestros días conducido por un equipo de voluntarios procedentes de diversas partes del mundo y liderados por el creador de Blender, Ton Roosendaal.

. Características

* Multiplataforma, libre, gratuito y con un tamaño de origen realmente pequeño comparado con otros paquetes de 3D, dependiendo del sistema operativo en el que se ejecuta.

* Capacidad para una gran variedad de primitivas geométricas, incluyendo curvas, mallas poligonales, vacíos, NURBS, metaballs.

* Módulo de animación.

* Compositor de Audio y Video.

* Motor de Juegos

* Raytracer interno y externos.

* Scripting.

*Multiformato.

*Simulación softbodies, partículas y físicas.

1.3 EL CORTOMETRAJE

Un cortometraje es una producción audiovisual, multimedia o cinematográfica que dura sustancialmente menos que el tiempo medio de una película de producción normal. Una de las finalidades del cortometraje es conseguir la atención del espectador desde el primer plano, a través de la historia que se le presenta, y "soltarlo" de manera sorpresiva, absurda, violenta, humorística, inexplicable, nostálgica, es decir, que "movilice" al espectador.

Todo comienza con una idea, dependiendo de la que se desee trabajar habrá diferentes procesos de producción para desarrollar ese proyecto.

A continuación se describen las etapas de desarrollo de un cortometraje animado en 3D.

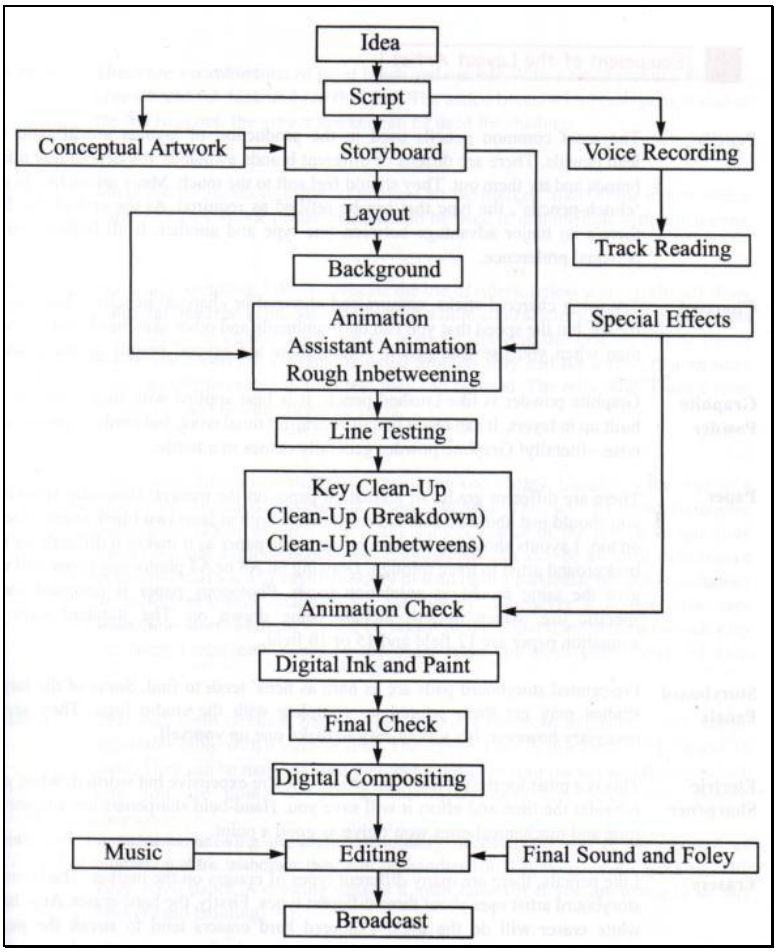
1.3.1 Etapas - Pipeline de Animación

Un pipeline es un sistema de trabajo que hace que un conjunto de personas trabajen compenetradamente en un flujo de trabajo, en una gran producción es bastante complejo, en un pipeline de una producción grande como pixar hay que fusionar todo el trabajo de manera más optima posible para que el trabajo de las personas se encajen de tal manera que no solo haya un flujo de trabajo continuo, sino que haya una optimización del mismo en el tiempo y una preparación para que una vuelta a atras en el flujo para la resolución de problemas no afecte al flujo.

Esto quiere decir por ejemplo que una vez termine alguien de diseñar un personaje, inmediatamente lo toma alguien para modelarlo y luego alguien para texturarlo (o a la misma vez depende del pipeline), cuando este termina, alguien del equipo lo riggea, y asi sucesivamente hasta que el personaje este terminado.

El pipeline puede integrar a diferentes estudios, depende del proyecto y la necesidad del mismo.

A continuación El pipeline diseñado para un proyecto de animación tradicional, que puede llegar a ser parecido a un pipeline para un proyecto de animación en 3D.



Título: Pipeline animación
Fuente: http://www.blender.org/
Elaboración: Juan Carlos Reascos Laverdi.

Las etapas de un cortometraje animado son las diferentes fases del proyecto y son el concepto y la preproducción, la producción y la post-producción.

1.3.2 Preproducción

Qué es?

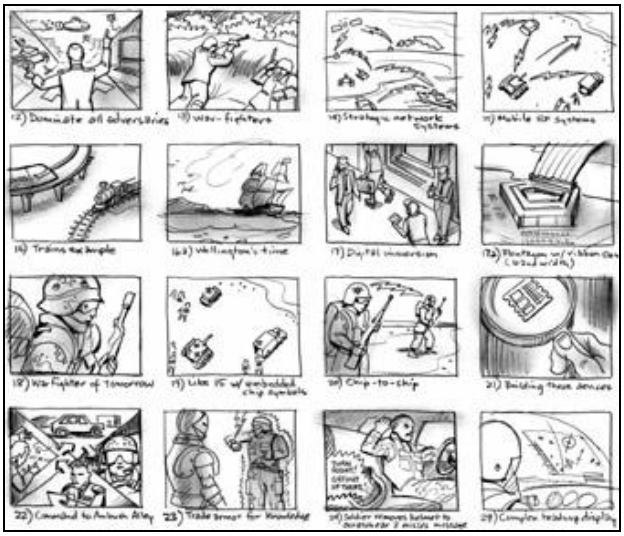
En la etapa de preproducción se realizan las primeras etapas del pipeline de animación en donde se crea el concepto del cortometraje, la historia que se va a contar,

para después desarrollar guión los diálogos entre nuestros personajes y la música que se va a tener en el desarrollo del mismo.

Los componentes más importantes de una preproducción son el Script o Guión, el Story Board, los Layouts o diseño de personajes o de escenarios , Hojas de Modelos (poses principales de personajes) y el animatic.

1.3.2.1 StoryBoard

Una vez realizado el guión se realizarán la creación de los personajes con sus respectivas cualidades, personalidad y rasgos físicos o psicológicos característicos de cada uno de ellos, después de este proceso pasamos al desarrollo del story board que es un preliminar de secuencias de los respectivos planos del cortometraje, un storyboard o guión gráfico es un conjunto de ilustraciones mostradas en secuencia con el objetivo de servir de guía para entender una historia, previsualiza una animación o puede seguir la estructura de una película antes de animarse, por ejemplo en un cortometraje se puede realizar un desglose de planos y así tener una referencia más cercana a la realidad del proyecto, acompañada por notas que describen lo que sucede en cada escena como por ejemplo movimientos de cámaras, etc. Ayuda a la visualización y comunicación de la idea de la historia, este detalla la escena y los cambios que se pueden realizan en la animación como también provee un recordatorio del plan original, puede se útil como referencia y ayuda a finalizar la idea de la historia.



Título: logotipo Storyboard
Fuente: http://www.blender.org/
Elaboración: Juan Carlos Reascos Laverdi.

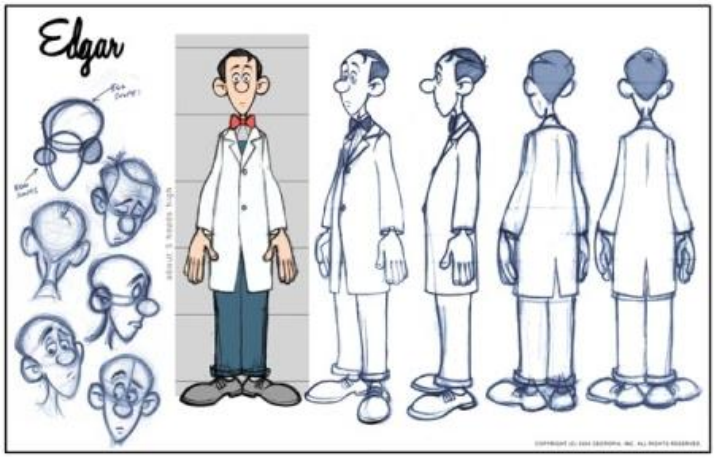
1.3.2.2 Layouts

Una vez que el story board ha sido aprobado por el director, este es enviado la departamento de de Layout (departamento de diseño) en donde se trabaja junto al director para diseñar los personajes, las locaciones y trajes que van a ser usados en el cortometraje, gracias a este proceso se pueden preparar el set-up de las escenas, mostrando como quedan definidas las posiciones de los personajes durante el cortometraje.

1.3.2.3 Model Sheets

Estos son precisamente un grupo de ilustraciones que muestran todas las posibles expresiones que puede tener un personaje como también las poses que pueden adoptar. Estas son creadas con el fin de mantener de forma precisa los detalles de cada uno de los personajes y para mantener los diseños de los personajes uniforme, mientras que diferentes animadores pueden trabajar en ellos sin perder detalle alguno y poder mantener una uniformidad de estos entre escena y escena.

Durante esta etapa, se termina el diseño de los personajes de manera que cuando se inicia la producción, sus modelos pueden ser enviadas al departamento de modelado que son responsables de crear los modelos finales tridimensionales de los personajes.



Título: logotipo Blender 3D
Fuente: http://cg.tutsplus.com/articles/step-by-step-how-to-make-an-animated-movie/
Elaboración: Juan Carlos Reascos Laverdi.

1.3.2.4 Animatic

[30]Una vez desarrollado el story board, se puede decir que esta casi estructurado el cortometraje, pero para poder tener un acercamiento al tiempo de cada toma, del cuadre y movimientos de cámara nos valemos de un "animatic" o tira leica, la idea del animatic es describir de forma general las acciones de los personajes toma por toma, esta motiva mucho al tener una visión general, aunque sea en este caso, una visualización en bajo polígono "simple", sin animación, sin texturas y en esta se puede incluir audio. La reproducción de esta especie de esbozo de la pieza final, sirve al equipo de animación para encontrar fallos en la duración, composición de las escenas y en el guión. Con esto se sabe o se tiene una idea de hacia dónde se desea ir con la historia del cortometraje.

[30] http://cg.tutsplus.com/articles/step-by-step-how-to-make-an-animated-movie/

1.3.3 Producción

Una vez que el guión ha sido aprobado, el proyecto entra en la fase de producción. Es aquí donde el trabajo real puede dar inicio, con base en los lineamientos establecidos durante la preproducción. Algunas partes principales son el diseño, modelado, texturizado, iluminación, rigging y animación.

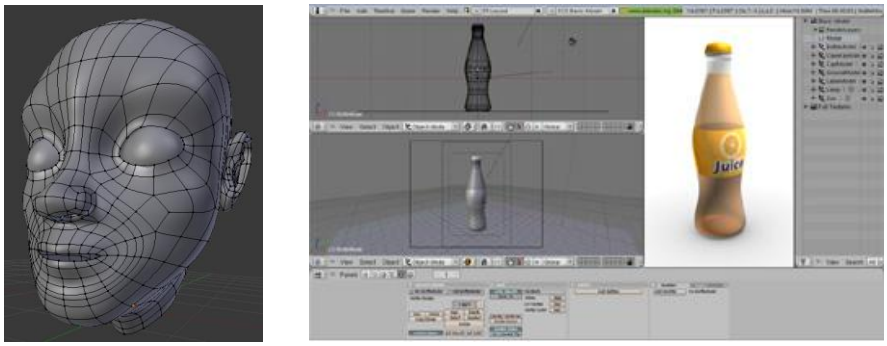
1.3.3.1 Modelado

El equipo de modelado se suelen dividir en dos o más departamentos. Mientras que los modeladores orgánicos tienden a tener conocimientos en escultura y especializado en la construcción de los personajes y otras superficies de forma libre u orgánicas. Se encargan de crear modelos humanoides bípedos o cuadrúpedos sean estos apegados a la realidad o a modelos humanos o no que pueden ser fantáticos.

Los modeladores inorgánicos "más técnicos" a menudo tienen un diseño más industrial o el fondo de arquitectura, y como tal modelo de los vehículos, armas, objetos y edificios.

Trabajando en estrecha colaboración con los directores de arte, efectos visuales supervisores y supervisores de animación, el equipo de modelado 3D encargado de convertir el concepto de arte 2D además de las maquetas tradicionalmente esculpidas en alto detalle en modelos 3D tridimensionales.

Una vez los modelos tridimensionales son aprobados, estos están disponibles para ingresar al departamento de rigging/skinning, para después ir poder añadir texturas o mapeados según el modelo lo necesite.



Título: modelado 3D

Fuente: http://www.blender.org/
Elaboración: Juan Carlos Reascos Laverdi.

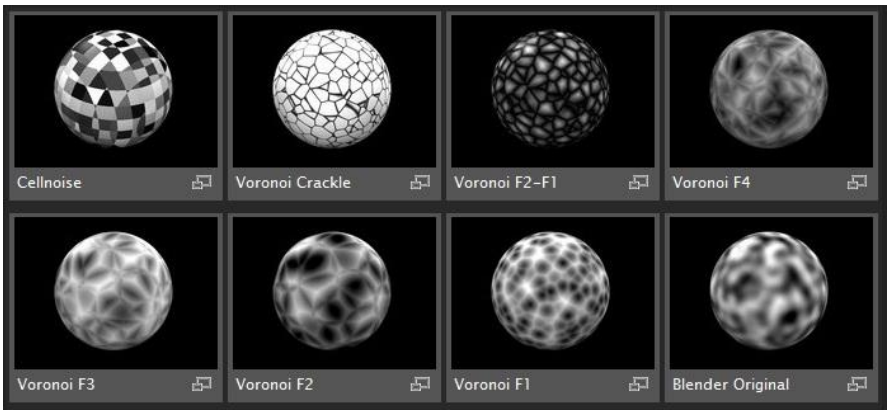
1.3.3.2 Texturas

Una vez que los modelos realizados en 3D y están aprobados, estos pasan al departamento de texture paint, que es el lugar donde son elaboradas todas las texturas de estos y son preparados los modelos tridimensionales de los personajes para ser riggeados para después ser animados.

Existen varias técnicas de texturización, estas dependen de la calidad o estética que el director necesite en su obra, estas pueden ser las texturas procedurales o Mapeado UV.

Las texturas procedurales son texturas definidas con algoritmos matemáticos. En general son relativamente fáciles de utilizar, puesto que no necesitan ser mapeadas de ninguna forma específica lo que no significa que las texturas procedurales no puedan llegar a ser muy complejas.

Este tipo de texturas son 3D 'real'. Con ello quiere decir que encajan perfectamente en las aristas y continuan apareciendo del mismo modo para el que han sido pensadas, incluso cuando se las corta; como si un bloque de madera hubiera sido realmente cortado en dos. Las texturas procedurales no son filtradas ni se les aplica antialias. Eso difícilmente representa un problema: el usuario puede mantener fácilmente las frecuencias especificadas dentro de límites aceptables.



Título: Texturas Procedurales

Fuente: http:// http://wiki.blender.org/index.php/Doc:ES/Manual/Textures/Types/Procedural
Elaboración: Juan Carlos Reascos Laverdi.

Otra técnica para dar texturas es el mapeado UV. El mapeado UV consiste en la asignación de una textura 2D a las caras de un modelo 3D siguiendo las coordenadas de la superficie (las coordenadas UV precisamente). Es uno de los métodos más populares y efectivos para texturizar personajes animados, pero a la vez es uno de los más difíciles y tediosos de utilizar cuando se quieren obtener resultados óptimos. Blender soporta texturizado UV.

Por ejemplo, la piel de una cabeza humana nunca se verá lo suficientemente bien cuando es generada proceduralmente. Las arrugas en un rostro humano, o las saltaduras de pintura en un automóvil no aparecen en lugares al azar sino que dependen de la forma del modelo y de su uso. Las imágenes pintadas manualmente le dan al artista control total sobre el resultado final. En lugar de andar jugando con deslizadores numéricos, los artistas pueden controlar cada pixel sobre la superficie. Esto comúnmente significa más trabajo, pero los resultados valdrán la pena. Un mapa UV es una forma de asignar una parte de una imagen a un polígono en el modelo. Cada vértice del polígono es asignado a un par de coordenadas 2D que definen que parte de la imagen es mapeada. Estas coordenadas 2D se llaman UVs (comparar con las coordenadas XYZ en 3D). La operación de crear estos mapas UV se conoce también como "despliegue" ("unwrap" en inglés), debido a que todo ocurre como si la malla fuera desenvuelta o desplegada sobre un plano 2D.



Título: texturas UV
Fuente: httphttp://wiki.blender.org/index.php/Doc:ES/Manual/Textures/UV/Unwrapping_a_MeshElaboración: Juan Carlos Reascos Laverdi.

1.3.3.3 Rigging / Skinning

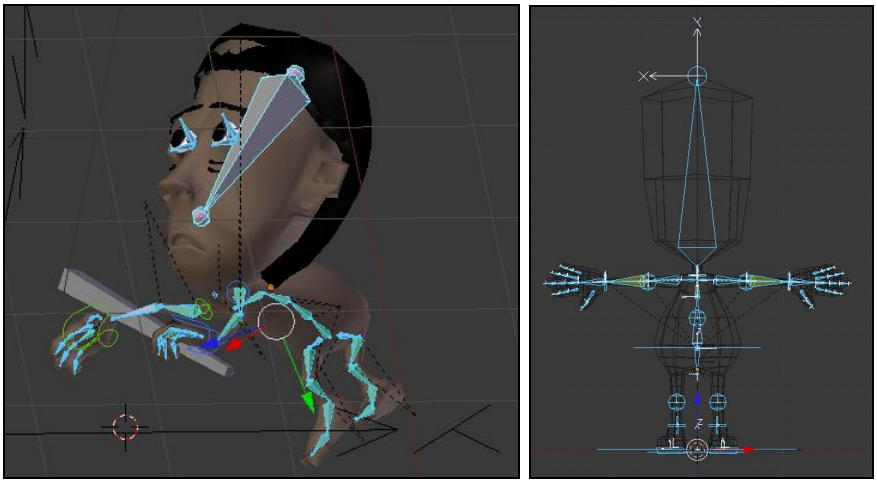
A continuación, los respectivos modelos son llevados al departamento de rigging/skinning junto a un animador y a un Técnico para la creación del sistema de huesos (rig), con este, se crea un esqueleto tanto para animación corporal y también se crea un sistema mas elaborado de huesos y shapes mas detallados para realizar la animación facial.

El sistema de huesos que se implemente depende de las necesidades del equipo de animadores, ya que ellos van a ser los encargados de manipular dicho sistema, por lo que el equipo de rigging puede realizar este proceso en una malla a través de scripts realizados en Python, ya que los huesos utilizados son también objetos y Python se vale de estos para poder generar scripts, de esta forma por ejemplo se puede dar más opciones a un Hueso para que este tenga un control automático de IK o FK. De esa forma ganar flexibilidad al momento de animar un personaje sin tener que ir al panel de opciones del hueso y hacer el mismo proceso.

Este proceso se termina con el Skinning del modelo, es decir que a la malla de un personaje es emparentado un sistema de huesos, con este proceso la malla adopta los

movimientos del sistema de huesos que fue creado previamente, para poder así poder realizar las animaciones que se tengan que realizar en la fase de producción.

Antes de enviar los personajes dotados de esqueletos y con sus respectivas texturas, son realizadas varias pruebas en base a los requerimientos de la construcción de cada personaje como también con el animatic, en muchas ocasiones es necesario realizar ajustes en el Rig de algún personaje ya que no se suele llegar a realizar las diferentes poses requeridas en la animación.



Título: Rigging
Fuente: http://www.blender.org/
Elaboración: Juan Carlos Reascos Laverdi.

1.3.3.4 Puesta en Escena.

Iluminación

Basados en los layouts elaborados en post-producción se realiza la propuesta de iluminación o fotografía con esto se logra la estética visual que tendrá el cortometraje, pero en iluminación se tiene muchos conceptos a los que se debe atener. Los objetivos de la iluminación en gráficos tridimensionales de computadora son más o menos los mismos que en la iluminación en un entorno real. La iluminación cumple la función básica de resaltar o disimular las formas de los objetos visibles desde el punto de vista de la cámara. Le otorga a la imagen bidimensional del monitor una ilusión de profundidad tridimensional. Pero no solamente eso. También le aporta a la imagen su personalidad, su

carácter. Una misma escena iluminada de distintas formas puede provocar un sentimiento de alegría, de pesar, de temor, etc. y puede lograr esto en formas que van de lo dramático a lo sutil. Conjuntamente con la personalidad y el carácter, la iluminación inunda la escena con una emoción que es transmitida directamente al espectador.

La iluminación intentar simular un entorno real dentro de uno artificial, esta puede llegar a ser una tarea agotadora. Pero aún logrando una imagen 3D que luzca absolutamente foto-realista, eso no garantizará que ésta transmita una emoción suficientemente intensa como para satisfacer las expectativas de una audiencia. Lograr imágenes 3D foto-realistas puede ser difícil. Impregnarlas de emociones profundas puede ser más difícil aún. Sin embargo, con una estrategia de iluminación planificada en consonancia con el ánimo y emoción que se quiere expresar en la imagen, se logrará que el proceso sea más fácil.

Componentes y Propiedades de la Luz

Cada fuente de luz puede ser descompuesta en 4 componentes distintos para ser analizada.

- Intensidad
- Dirección
- Color
- Tamaño

Varios especialistas prefieren combinar todos estos términos bajo otro término que denominan "Calidad de Luz". Esta calidad de luz es determinada por la contribución de cada uno de estos 4 componentes a la iluminación global de la escena.

Aquí algunos ejemplos importados del mundo de la fotografía. Los fotógrafos son los maestros de la iluminación y que hay mucho para aprender de su trabajo.

Por ejemplo, la calidad de luz varía con el tiempo en un entorno natural y los fotógrafos escogen sus sujetos de acuerdo a la calidad de la luz disponible a determinada hora del día. Algunos fotógrafos solamente hacen tomas luego del alba y justo antes de la

puesta del sol porque la luz en esos momentos del día tiene colores más fuertes y arroja sombras largas y dramáticas. Es posible simular ese tipo de entorno dentro de un programa 3D y obtener imágenes realmente dinámicas. Una luz proveniente de un ángulo bajo es apropiada para la iluminación lateral, la cual da una buena calidad tridimensional general a los sujetos. Iluminarlos desde atrás con luces similares puede producir unas siluetas increíbles.

Por ejemplo se iluminar los personajes utilizando luces de bajo ángulo que crean sombras largas y resaltan la forma del personaje. Este tipo de iluminación es también muy apropiado para paisajes.

En condiciones de cielo encapotado, tal como un día nublado, los fotógrafos realizan tomas de acercamiento de flores, por ejemplo. La iluminación encapotada se traduce en una fuente de luz de un área considerable (una fuente de luz de Área en Blender). Otros ejemplos de fotografías que pueden ser tomadas en una condición encapotada (o con una fuente de luz de gran área) son retratos o fotografías de alimentos. Esto no necesariamente tiene por qué hacerse en exteriores. La frase operativa aquí es "Fuente de Luz de Gran Área".

Debería ahora pasar a describir individualmente cada una de los 4 componentes de la luz.

INTENSIDAD DE LA LUZ

La intensidad de la luz puede ser definida simplemente como la cantidad de luz emitida desde una fuente. A medida que se incrementa la intensidad de una luz, desde cero hasta valores muy elevados, comienzan a suceder cosas interesantes a los objetos iluminados por dicha luz. Abajo hay una serie de imágenes que muestran algunos de estos efectos. Mi propósito era resaltar una esfera ubicada entre otros objetos y para esto hice un anáisis de los efectos que cambiar la intensidad de la luz en la escena tenía sobre toda la composición.

Dirección de la Luz

Imaginemos una escena donde existe luz de intensidad y color equivalente, incidiendo sobre un rostro humano en todas direcciones por igual y sobre un fondo negro. Veríamos solamente una silueta bidimensional del rostro recortada sobre el fondo.

¿Por qué? Porque los rayos de igual color e intensidad 'pintarán' todos los lados del rostro con el mismo color y la misma intensidad. Si una sombra tendiera a formarse, sería instantaneamente borrada por los rayos que inciden sobre esa región.

La razón por la cual somos capaces de reconocer la forma de un objeto, es que los rayos de luz de distintas intensidades que golpean al objeto desde distintas direcciones, 'pintan' al objeto con brillos y sombras.

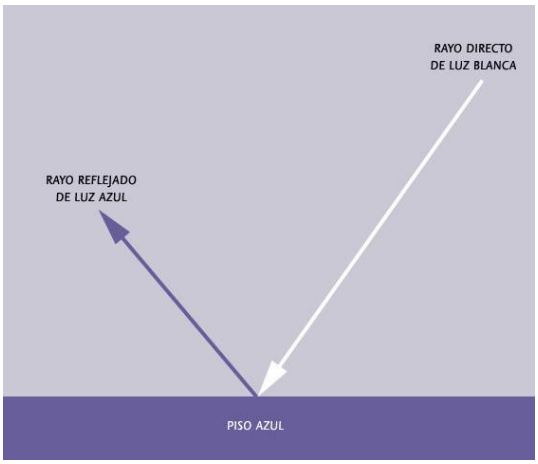
La dirección de la luz proveniente de una fuente puede mejorar la forma del sujeto, así como la emotividad general de la escena. También puede arruinar lo que se está intentando capturar en la imagen. Para dar profundidad al objeto que está siendo iluminado, se debe ubicar la fuente principal de luz, o luz principal, a un cierto ángulo con respecto a la cámara para producir brillos y sombras. Hacer esto creará o resaltará la ilusión de profundidad en el objeto, debido al desvanecimiento gradual de brillo a sombra sobre la superficie del mismo. Lo que se aprecia en el monitor es en realidad una imagen bidimensional, y la ilusión de tridimensionalidad es creada por los brillos y sombras en el objeto.

Color de la Luz

Como se indicó anteriormente, la razón por la que somos capaces de reconocer la forma de un objeto es que rayos de luz de distintas intensidades, provenientes de distintas direcciones, impactan en él, 'pintándolo' con brillos y sombras. Para hacer más completa esta afirmación, se tendría que agregar aquí que nuestra habilidad para reconocer la forma de un objeto depende de la habilidad de los rayos de luz, de distintas intensidades y colores, de impactarlo desde distintas direcciones, 'pintandolo' con brillos y sombras.

El color de una luz directa depende de su fuente de irradiación. La luz blanca está compuesta por todos los posibles colores existentes. Un rayo de luz blanca cambiará de color si encuentra un obstáculo que no sea ni blanco ni negro. Si impacta a un objeto blanco, el mismo rayo es reflejado. Si el objeto es de color negro, el objeto absorbe toda la luz, sin importar de qué color era originalmente y nada es reflejado. De manera que básicamente al observar un objeto totalmente negro, se ve de ese color porque no hay luz que ingrese al ojo proveniente de esa dirección. Para probar este punto, se puede cerrar los ojos por un segundo.

En la fig. abajo, es posible ver un rayo blanco de luz directa, que es reflejado por un piso azul. El piso absorbe todos los colores del rayo que lo impacta, a excepción del azul que es reflejado. Nótese que la luz es reflejada con un ángulo idéntico al ángulo de impacto con relación al piso.



Título: Propiedades de la Luz
Fuente: http://www.warpedspace.org/lightingT/part1_es.htm
Elaboración: Juan Carlos Reascos Laverdi.

Generalmente hablando, los colores saturados representan una proximidad cercana, mientras que los no saturados representan distancia. Un buen ejemplo para citar aquí son las mañanas brumosas. A medida que los objetos se alejan, tienden a perder la saturación de su color. Resumiendo, los colores brillantes y saturados tienden a permanecer en frente, mientras los menos saturados encuentran su lugar en el fondo.

Las mañanas normalmente presentan un tinte azulado. Alrededor del mediodía, se obtiene una luz más o menos balanceada. Hay, en alguna medida, algo de luz azul presente (reflejada desde el cielo), pero su efecto no es muy pronunciado. La luz del atardecer está típicamente caracterizada por los tonos cálidos y anaranjados.



Título: Propiedades de la Luz
Fuente: http://www.warpedspace.org/lightingT/part1_es.htm
Elaboración: Juan Carlos Reascos Laverdi.

Si observamos atentamente el siguiente conjunto de tres imágenes. Las sombras cambian su posición en las primeras dos. La fig. 1 representa otra vez un mediodía veraniego y la 2 representa el atardecer. La fig 3 muestra una escena iluminada por una Luna elevada en el cielo. El tinte azul está presente para dar la ilusión de noche.



Título: Propiedades de la Luz
Fuente: http://www.warpedspace.org/lightingT/part1_es.htm
Elaboración: Juan Carlos Reascos Laverdi.

Tamaño de la Fuente de Luz

El tamaño de la fuente tiene un efecto preponderante en la sensibilidad general de la escena. Una fuente de luz de reducido tamaño arroja sombras muy definidas y apreciables, incorporando un elemento de tensión a la imagen. Un ejemplo de una fuente de luz de tamaño reducido podría ser la lamparita de una linterna, la cual en efecto arroja sombras muy definidas.

Una fuente de luz de un área mayor arroja sombras mucho más suaves (menos apreciables), aportando un sentimiento de relajación a la escena.

1.3.3.5 Proceso de Animación

Después de desglosar el cortometraje en escenas y gracias a la ayuda del animatic se procede a animar éstas, una por una.

La animación es un proceso utilizado para dar la sensación de movimiento a imágenes o dibujos. Para realizar animación existen numerosas técnicas que van más allá de los dibujos animados. Los cuadros se pueden generar dibujando, pintando, o fotografiando los minúsculos cambios hechos repetidamente a un modelo de la realidad o a un modelo tridimensional virtual; también es posible animar objetos de la realidad y actores.

Concebir una animación tiende a ser un trabajo muy intensivo y tedioso. Por esto la mayor parte de la producción proviene de compañías de animación que se han encargado de organizar esta labor. Aun así existe la animación de autor (que tiene relación con la animación independiente), en general más cercana a las artes plásticas. Ésta surge del trabajo personal de uno o de unos pocos artistas. Algunos se valen de las nuevas tecnologías para simplificar la tarea.

En el cine existe un estándar de 24 fotogramas por segundo. Ésa es la tasa a la que graban las cámaras y proyectan los proyectores. Se toma una fotografía de la imagen cada veinticuatroavo de segundo para crear la sensación de movimiento, existen varias técnicas de animación y la técnica utilizada depende mucho del director y la historia , como se la desee contar.

[31]Entre las técnicas más conocidas tenemos:

- **Dibujos animados:** Los dibujos animados se crean dibujando cada fotograma a un constante de 24 cuadros por segundo.
- **Stop motion:** Animación de objetos, muñecos, marionetas, figuras de plastilina u otros materiales así como maquetas de modelos a escala. Se utiliza la grabación "fotograma a fotograma" o "cuadro a cuadro" (frame a frame).
- **Pixelación:** Es una variante del stop-motion, en la que los objetos animados son auténticos objetos comunes (no modelos ni maquetas), e incluso personas. Al igual que en cualquier otra forma de animación, estos objetos son fotografiados repetidas veces, y desplazados ligeramente entre cada fotografía.
- **Animación de recortes:** Más conocido en inglés como cutout animation, es la técnica en que se usan figuras recortadas, ya sea de papel o incluso fotografías. Los cuerpos de los personajes se construyen con los recortes de sus partes. Moviendo y reemplazando las partes se obtienen diversas poses, y así se da vida al personaje.
- **Otras técnicas:** Cualquier materia que pueda ser fotografiada, puede utilizarse para animar. Existen muchas técnicas de animación que sólo han sido utilizadas por unos y que son desconocidas. Entre éstas se incluyen: pintura sobre cristal, animación de arena, pantalla de agujas, pintura sobre celuloide, tweening. Pero también es posible reproducirlo por ordenador.

La fase de animación en un proyecto de este tipo requiere ciertas etapas en las que se podrá ir perfeccionando los detalles que van desde la propuesta del animador para los personajes en la escena, como la actuación de éstos o acting, timing, spacing y el uso apropiado de los principios de animación descritos anteriormente.

Los animadores varían de acuerdo al proyecto.

[31] http://es.wikipedia.org/wiki/Animaci%C3%B3n

Estas etapas serán descritas en Pases, entendiéndose como "pases" la definición de las diferentes etapas en donde el animador desarrolla la animación de la escena asignada con los personajes incluidos en ella, así el primer pase de la animación de una escena entregada a un animador, es la propuesta misma de éste manejada básicamente con poses principales o claves (keyframes), en las que se muestra las principales acciones del personaje "líneas de acción" estas son las poses mas extremas que puede tener el personaje en una escena para acentuar una acción o intención, además de la interacción del mismo con el entorno o los posibles vínculos con los demás personajes involucrados en la escena designada. Junto al director se resuelven los diferentes problemas que se puedan encontrar en la resolución de la animación de esta.

Pudiendo el director realizar cambios drásticos en caso de que la animación no sea la que el director esperaba, ya sea por actuación (acting), técnicas de animación o por dramaturgia.

El siguiente Pase es en donde el animador como el director, han resuelto la acciones principales y la interacción de este con el entorno y los demás personajes. Es ahora cuando la animación de esta escena pasa al proceso de creación de in-betweens, entre cada una de las poses principales que fueron creadas, en este paso se crea una fluidez en la animación, como en los grandes estudios de animación en donde se producía animación clásica, los animadores con más experiencia creaban las poses principales de los personajes en la escena y los animadores principiantes realizaban las poses intermedias o in-between.

Para finalizar se tiene el tercer pase de animación en donde se realiza la animación secundaria, dado que un personaje influye sobre su entorno y este puede producir su propia animación, por ejemplo si tenemos un personaje que está colgado sobre las ramas de un árbol, este además de tener su propia animación, las ramas el árbol y demás tiene su propia animación que es secundaria para la escena pero es muy importante al momento de realizar una animación completa.

1.3.4 Postproducción

En esta etapa de la realización de un cortometraje animado, en el que el trabajo de animación está completo el cual se ha pulido junto con el manejo del guión y la propuesta ya aceptada de arte se procede a juntar todas estas en una composición final de las escenas, el mismo que incluye temas ya expuestos como manejo de texturas tanto del entorno como de los personajes, además del manejo de la iluminación que va a ser necesaria en cada una de las escenas dependiendo del storyboard junto a la propuesta de arte que previamente fue aceptada por el director.

Además se puede adicionar a esta etapa un trabajo de FX efectos especiales, que sean necesarios y que fueron detallados en la fase de pre-producción.

En caso de tener diálogos los personajes, éstos diálogos son grabados con anterioridad a la fase de animación ya que son muy necesarios para el proceso llamado "lip-sync", en que se sincroniza los movimientos de la boca de los personajes con los fonemas producidos por las palabras.

Finalmente se realiza un trabajo de Audio en la que se agregan según sea el caso sonido ambiente, musicalizaciones, o sonidos creados para denotar alguna acción o también llamados Foley.

Una vez que se tiene la mezcla final se procede al render del proyecto en el cual están ubicados el proceso de animación, el proceso de iluminación, texturas, landscape, musicalización, Foley y composición obteniendo así el proyecto deseado.

1.3.5 Cuadro Sinóptico - CAPITULO I

TG: Creación de un Corto animado en 3D y la infraestructura Técnica necesaria para su realización tanto en Hardware y en Software de licencia GNU GPL.



CAPITULO II

La herramienta elegida para realizar esta tesis es Blender 3D, esta tiene licenciamiento GNU GPL la utilización de esta herramienta tiene como varios objetivos, conocer y explicar las utilidades que tiene esta herramienta para el desarrollo de proyectos multimedia, además de explicar los conceptos fundamentales de la creación de gráficos 3D y la evolución del software libre en el campo de programas tridimensionales.

2. La herramienta

[32]Blender 3D es un programa informático que ha venido integrando una serie de herramientas para la creación de una amplia gama de contenidos 2D y 3D. Blender ofrece en un paquete un amplio espectro de funcionalidad para el modelado, texturizado, iluminación, animación y post-procesado de vídeo. Por medio de su arquitectura abierta, ofrece interoperabilidad entre plataformas, extensibilidad, una superficie de apoyo increiblemente pequeña y un flujo de trabajo altamente integrado. Blender es una de las aplicaciones Open Source de gráficos 3D más populares del mundo.

Destinado al amplio mundo de los profesionales de la comunicación, desarrolladores y artistas, Blender puede usarse para crear visualizaciones 3D, tanto imágenes estáticas como videos de alta calidad, mientras que la incorporación de un motor 3D en tiempo real permite la creación de contenido interactivo que puede ser reproducido independientemente.

Entre sus principales características tenemos:

- Paquete de creación totalmente integrado, ofreciendo un amplio rango de herramientas esenciales para la creación de contenido 3D, incluyendo modelado, mapeado uv, texturizado, rigging, skinning, animación, simulación de partículas y otros, scripting "Python", renderizado, composición, post-producción y creación de juegos

[32] http://es.wikipedia.org/wiki/Blender

- Multiplataforma, con una interfaz basada en OpenGL, lista para ser usada en todas las versiones de Windows (98, NT, 2000, XP, 7), Linux, OS X, FreeBSD, Irix, Sun y otros sistemas operativos.
- Arquitectura 3D de alta calidad permitiendo un rápido y eficiente desarrollo.
- Más de 200.000 descargas de cada lanzamiento (usuarios) en todo el mundo.
- Foro de soporte comunitario para preguntas, respuestas y críticas en http://BlenderArtists.org (inglés), http://www.3dpoder.com (español) y nuevos servicios en http://BlenderNation.com
- Ejecutable de pequeño tamaño, para una fácil distribución con un tamaño de origen realmente pequeño comparado con otros paquetes de 3D, dependiendo del sistema operativo en el que se ejecuta.
- Capacidad para una gran variedad de primitivas geométricas, incluyendo curvas, mallas poligonales, vacíos, NURBS, metaballs.
- Junto a las herramientas de animación se incluyen cinemática inversa, deformaciones por armadura o cuadrícula, vértices de carga y partículas estáticas y dinámicas.
- Edición de audio y sincronización de video.
- Características interactivas para juegos como detección de colisiones, recreaciones dinámicas y lógica.
- Posibilidades de renderizado interno versátil e integración externa con potentes trazadores de rayos o "raytracer" libres como kerkythea, YafRay o Yafrid.
- Blender acepta formatos gráficos como TGA, JPG, Iris, SGI, o TIFF. También puede leer ficheros Inventor.
- Motor de juegos 3D integrado, con un sistema de ladrillos lógicos. Para más control se usa programación en lenguaje Python.

- Simulaciones dinámicas para softbodies, partículas y fluidos.
- Modificadores apilables, para la aplicación de transformación no destructiva sobre mallas.
- Sistema de partículas estáticas para simular cabellos y pelajes, al que se han agregado nuevas propiedades entre las opciones de shaders para lograr texturas realistas.

2.1 Reseña Histórica

[33]En 1988, Ton Roosendaal co-fundó el estudio de animación holandés NeoGeo. NeoGeo rápidamente se convirtió en el estudio más grande de animación 3D en Holanda y en una de las más destacadas casas de animación en Europa. NeoGeo creó producciones que fueron premiadas (European Corporate Video Awards de 1993 y 1995) para grandes clientes corporativos tales como la compañía multinacional de electrónica Philips. En NeoGeo, Ton fue el responsable tanto de la dirección artística como del desarrollo interno del software. Después de una cuidadosa deliberación, Ton decidió que la actual herramienta 3D utilizada en el estudio de NeoGeo era demasiado vieja y voluminosa de mantener y actualizar y necesitaba ser reescrita desde el principio. En 1995, esta reescritura comenzó y estaba destinado a convertirse en el software de creación 3D que ahora conocemos como Blender. Mientras NeoGeo continuaba refinando y mejorando Blender, Ton se dio cuenta que Blender podría ser utilizado como una herramienta para otros artistas fuera del estudio NeoGeo.

En 1998, Ton decidió crear una nueva compañía llamada Not a Number (NaN) derivada de NeoGeo para fomentar el mercado y desarrollar Blender. En la base de NaN, estaba el deseo de crear y distribuir gratuitamente una suite de creación 3D compacta y multiplataforma. En ese momento, esto fue un concepto revolucionario ya que la mayoría de los programas comerciales de modelado costaban miles de dólares. NaN esperaba

[33] http://es.wikipedia.org/wiki/Blender#Historia

conseguir una herramienta de modelado y animación de un nivel profesional al alcance del público en general. El modelo de negocio de NaN consistía en proporcionar productos comerciales y servicios alrededor de Blender. En 1999, NaN asistió a su primera conferencia en el Siggraph en un esfuerzo aún mayor para promocionar Blender. La primera convención del Siggraph para Blender en 1999 fue un auténtico éxito y provocó un enorme interés tanto de la prensa como de los asistentes a la convención. ¡Blender fue un gran éxito y se confirmó su tremendo potencial!

En aras del gran éxito del Siggraph, a principios del año 2000, NaN consiguió una financiación de 4,5 millones de euros procedente de unos inversores. Este gran aporte de dinero permitió a NaN expander rápidamente sus operaciones. Pronto NaN alardeó de tener más de 50 empleados trabajando alrededor del mundo intentando mejorar y promocionar Blender. En el verano del 2000, Blender 2.0 fue publicado. Esta versión de Blender integraba un motor de juegos a la suite 3D. Al final del 2000, el número de usuarios registrados en el sitio web de NaN sobrepasó los 250.000.

Desafortunadamente, las ambiciones y oportunidades de NaN no coincidieron con las capacidades de la compañía ni con la realidad del mercado de la época. Este sobredimensionamiento de la empresa condujo a una reestructuración creando una compañia (NaN) mas pequeña y con nuevos fondos procedentes de los inversores. Seis meses mas tarde, el primer producto comercial de NaN, Blender Publisher fue lanzado. Este producto fue dirigido al emergente mercado de medios interactivos en 3D basados en entornos web. Debido a las decepcionantes ventas y al continuo clima de dificultades económicas, los nuevos inversores decidieron dar por terminadas las actividades de NaN. Esto también incluía parar el desarrollo de Blender. Si bien existían claramente defectos en la actual versión de Blender, con una arquitectura interna del software compleja, características inacabadas y una IGU no muy común, la magnífica ayuda de la comunidad y los clientes que habían comprado Blender Publisher en el pasado provocó que Ton no pudiera permitir que Blender desapareciera en el olvido. Como relanzar una nueva compañía con un equipo suficientemente grande de desarrolladores no era factible, en marzo de 2002, Ton Roosendaal fundó la organización no lucrativa Blender Foundation (Fundación Blender).

El primer objetivo de la Fundación Blender fue encontrar una manera de continuar el desarrollo y la promoción de Blender como un proyecto de código abierto basado en la

comunidad de usuarios. En julio de 2002, Ton logró obtener de los inversores de NaN un "sí" para que la Fundación Blender llevara a cabo su plan de que Blender fuera código abierto. La campaña de "Liberad a Blender" tenía que obtener 100.000 EUR para que la Fundación pudiese comprar los derechos del código fuente y los de propiedad intelectual de Blender a los inversores de NaN y, posteriormente, liberar Blender a la comunidad de código abierto. Con un entusiasta grupo de voluntarios, entre los que se encontraban varios ex-empleados de NaN, fue lanzada la campaña de "Liberad a Blender". Para el deleite y sorpresa de todo el mundo, la campaña alcanzó el objetivo de 100.000 EUR en tan sólo 7 semanas. El domingo 13 de octubre de 2002, Blender fue liberado al mundo bajo los términos de la Licencia Pública General de GNU (GPL). El desarrollo de Blender continúa hasta nuestros días conducido por un equipo de valientes y dedicados voluntarios procedentes de diversas partes del mundo y liderados por el creador de Blender, Ton Roosendaal.

2.2 Cortometrajes desarrollados en Blender

Blender ha sido el principal programa para la creación y desarrollo de varios cortometrajes , como herramienta de modelado, en el manejo de texturas, iluminación, animación, además se utilizó otras herramientas para el trabajo de pre y post producción como renderizadores externos como Yafray, etc para la gestión del conjunto del proyecto.

Ubuntu ha sido el sistema operativo utilizado y GNOME o KDE los escritorios gráficos de la mayoría de éstos.

Además del software libre utilizado, como excepción se han usado Reaktor (un programa modular de sonido) y sistemas Mac organizados en racimos (comúnmente llamados "granjas de render" cuando se habla de 3D o efectos visuales por derivar del inglés "render farm") para renderizar la parte final del proyecto.

Entre los cortometrajes realizados tenemos:

- Elephants Dream
- Plumíferos
- Big Buck Bunny

- Sintel
- Durian

Además del video juego llamado Yo Frankie! Y el primer largometraje realizado con Blender3D en Latinoamérica llamado Plumiferos.

A continuación tenemos las principales características de cada cortometraje.

2.2.1 Proyecto Orange "Elephants Dream"



Título: Elephants Dream
Fuente: http://orange.blender.org/
Elaboración: Juan Carlos Reascos Laverdi.

La producción comenzó en septiembre de 2005 y fue financiada y desarrollada por el estudio Orange con un equipo internacional de siete artistas y animadores. Originalmente conocida por el nombre de Machina, fue rebautizada como Elephants Dream para cuadrar más exactamente con el elemento central del argumento.

Es una pequeña historia de dos personajes: un chico joven llamado Emo y Proog, dos personas que comparten un mundo surrealista o fantástico en el que están inmersos

y que varía según van moldeando sus propios pensamientos. Proog, que comprende lo que está sucediendo, está fascinado por éste y sus misterios, sin embargo Emo pasa del desconocimiento a cansarse de lo que le rodea.

Esto terminará enfrentándoles entre la realidad que desea Proog: una especie de mundo extraño industrial de criaturas mecánicas y Emo con una visión totalmente distinta, más viva. Finalmente, Emo "crea" a un titán para matar a Proog hasta que éste, horrorizado, le da un golpe seco y lo mata.

Los objetivos del proyecto son varios:

- exhibir o promocionar el talento profesional de las personas que integran el equipo
- demostrar la validez de los programas libres
- que el resultado final revierta y sirva para mejorar aún más en el propio software especialmente Blender, analizando y añadiendo nuevas mejoras, resolviendo errores, etc
- ofrecer por primera vez un film de animación de gran calidad, de forma que el material empleado se pueda obtener, modificar o distribuir libremente
- Durante el desarrollo de Elephants Dream y, a fin de acrecentar la sofisticación del resultado, fueron añadidas o mejoradas numerosas características, como un avanzado sistema de animación, renderizado por pasadas, mejoras en el motor gráfico de simulación de partículas, un nuevo esquema de materiales y post-procesamiento nodal que permite complejos efectos como la simulación física de pelo, piel, fusión de fotogramas, aplicaciones de filtros no lineales, etc.
- El contenido de la película es libre, es decir está publicado con una licencia Creative Commons que ofrece el código fuente del trabajo 3D utilizado (modelos, texturas, animaciones..) de modo que, además de servir como promoción de Blender y el software libre, pueda devolver a la comunidad parte de su propio trabajo. Se puede conseguir en un soporte físico o bien descargarse desde la página web del proyecto sin coste alguno.

- La presentación del film fue el 24 de marzo de 2006 en la ciudad europea de Ámsterdam. Posteriormente será proyectada en varios festivales internacionales de cinematografía, incluido Cannes.

Software

Para el desarrollo del corto de animación, se han utilizado casi completamente las siguientes herramientas de software libre:

* Blender
* CinePaint
* GIMP
* Seashore
* SubVersion
* DrQueue
* GNOME y KDE
* Inkscape
* OpenEXR
* Python
* Seashore
* Twisted (software)
* Ubuntu Linux
* Verse (protocolo)

Link

http://orange.blender.org/

2.2.2 Big Buck Bunny & Yo Frankie!





Título: Peach Project
Fuente http://www.bigbuckbunny.org/
http://www.yofrankie.org/
Elaboración: Juan Carlos Reascos Laverdi.

Big Buck Bunny es un corto animado del Instituto Blender, El Instituto Blender es parte de la Fundación Blender. Como la película previa de la fundación, Elephants Dream, esta película se ha realizado usando software libre. El trabajo comenzó en octubre de 2007 y la película se estrenó del 10 de abril de 2008 en Ámsterdam.

La película fue financiada por la fundación Blender, donaciones de la comunidad Blender, las pre-ventas del DVD de la película y patrocinio comercial. Tanto el producto final como los datos de producción, incluyendo los datos de animación, los personajes y las texturas son lanzados bajo la licencia Creative Commons Attribution License.

Yo Frankie! (nombre código Apricot) es un videojuego desarrollado en 2008 por Instituto Blender, parte de la Fundación Blender. Es distribuido bajo licencia CC-BY 3.0. El

juego está basado en 'Frankie', un personaje de la película de Big Buck Bunny. Es hecha usando Blender.



Título: Peach Project
Fuente http://www.yofrankie.org/
Elaboración: Juan Carlos Reascos Laverdi.

Links:

http://www.bigbuckbunny.org/

http://www.yofrankie.org/

2.2.3 Plumíferos



Título: Plumíferos (la película)
Fuente http://www.plumiferos.com
Elaboración: Juan Carlos Reascos Laverdi.

Plumíferos, aventuras voladoras es un largometraje de animación por computadora argentino producido por Manos Digitales Animation Studio. Una de las particularidades de este proyecto es el uso de herramientas de software libre como Blender para las tareas de modelado, animación y renderizado 3D; sobre plataforma Ubuntu Linux.

Algunas escenas de la película se mostraron por primera vez en las conferencias de Blender en 2005 y 2006, y una broma fue lanzada más adelante mostrando a uno de los pájaros auditando para el papel. El primer trailer oficial fue mostrado el 10 de marzo de 2007 en el Festival Internacional de Cine de Mar del Plata, en donde fue organizada una presentación sobre la película.

Estrenada el 18 de febrero de 2010, Plumíferos se ha convertido en el primer largometraje de animación 3D realizado íntegramente en la Argentina, y el primero en el mundo en utilizar Blender sobre Linux para la totalidad de la generación de la imagen 3D.

Desarrollo

Luego de más de 3 años de desarrollo y temores acerca de la cancelación del proyecto, Plumiferos estreno en los cines de Argentina el 18 de Febrero del 2010.

Mejoras

Al igual que Elephants Dreams, este proyecto trajo grandes mejoras para Blender, realizandose más de 80 nuevas funcionalidades y cambios en Blender

Para el desarrollo de esta película han trabajado aproximadamente 30 personas en el estudio y varias más repartidas alrededor del mundo, como programadores, animadores etc. Cabe destacar el gran apoyo recibido por la comunidad de Blender, que ha colaborado desinteresadamente, tanto en su ejecución, como resolviendo errores de Blender o realizando nuevas funcionalidades que ManosDigitales iba posteando en su □ wishlist□ . Blender Fundation colaboró enviando a varios artistas del proyecto Elephants Dreams para intercambiar impresiones y ayudar en su desarrollo.

Plumíferos desarrolló una versión de Blender modificada por Manos Digitales; se llama, PlumiBlender y puede bajarse sin cargo para Windows, Linux y Mac desde www.graphicall.org/builds/plumiblender.php ; allí también están disponibles las fuentes. Esta posibilidad de modificar una aplicación para adaptarla a las necesidades de un proyecto sin pagar un centavo es única del software libre y constituye uno de los motores más potentes de su popularidad.

LINK: http://www.plumiferos.com/

LINK: http://www.lanacion.com.ar/nota.asp?nota_id=1081146

Plumiblender Link: http://www.graphicall.org/builds/plumiblender.php

Conferencia: http://video.google.com/videoplay?docid=4482334295160768412#

2.2.4 Proyecto Durian "Sintel"



Título: Durian Project
Fuente http://www.sintel.com
Elaboración: Juan Carlos Reascos Laverdi.

Sintel es un cortometraje de ambiente fantástico. Sintel es el nombre de la protagonista. En neerlandés sintel quiere decir ascua o brasa. Sintel es el tercer cortometraje realizado por la Fundación Blender, prácticamente enteramente usando sólo software libre y que, además, se ofrece de forma pública y gratuita con licencia y condiciones de difusion y reutilizacion basadas en la filosofía del conocimiento libre. Los otros dos cortometrajes anteriores fueron Elephants Dream y Big Bick Bunny.

La duración de Sintel es de 14 minutos.

La producción comenzó en junio de 2009.

El filme se financió mediante patrocinadores públicos y privados y también mediante pre-venta del DVD del film. Los patrocinadores, incluso los privados pequeños, aparecen en los créditos del film.

El contenido de la película es libre, es decir está publicado con una licencia Creative Commons que ofrece el código fuente del trabajo 3D utilizado (modelos, texturas, animaciones..) de modo que, además de servir como promoción de Blender y el software libre, pueda devolver a la comunidad parte de su propio trabajo. En otoño de 2010 se podrá adquirir estos materiales en un soporte físico de 4 DVDs.

Software

La presentación del film fue el 27 de septiembre de 2010. Posteriormente se difundió el 30 de septiembre en internet, en Youtube.

Para el desarrollo de este corto de animación, se han utilizado las siguientes herramientas de software libre:

* Blender (Modelado, Animación, Renderizado, Composición, Edición y Exportación)
* GIMP (Concept Art y pintado de texturas)
* MyPaint (Concept Art)
* Al.chemy (Concept Art)
* Inkscape (Dibujos Vectoriales)
* OpenEXR (Formato de Salida de Archivos)
* SubVersion (Administración de Archivos)
* Python (Programación y Scripts)
* GNOME (Entorno Gráfico y herramientas)
* Ubuntu Linux (Sistema Operativo)

Blender ha sido el principal programa para la creación y desarrollo de la película, el resto de los programas o herramientas se han utilizado en gran parte para tareas de pre y post producción y gestión del conjunto del proyecto. Ubuntu ha sido el sistema operativo utilizado, con GNOME como entorno gráfico de escritorio.

Además del software libre anterior, para la realización del sonido se ha usado programas propietarios.

2.3 Mejoras Actuales para Blender

Como en el caso de los otros films anteriores de la Fundacion Blender, los desarrolladores de Blender trabajaron intensamente siguiendo las necesidades del equipo de producción del film para mejorar Blender. Se han realizado mejoras en Blender, entre otros, en la interfaz de usuario, en el sistema de partículas (lluvia, polvo), en el esculpido, el sombreado, el sistema de rendición y en la simulación de humo. Estas modificaciones se han ido incorporando a las versiones de Blender desde la 2.50 alfa hasta la 2.54 beta.

Link: http://www.sintel.org/

2.3.1 EVOLUCION

Blender ha tenido un desarrollo vertiginoso y en pocos años ha logrado destacacarse, gracias a la comunidad y a la fundación Blender se tiene actualmente la versión estable 2.49b y las versiones beta 2.5x y en un futuro cercano tener la versión 2.6.

Uno de los mayores problemas que se tenían con Blender era su interfaz, hasta la versión 2,49b fue utilizada y la mayoría de inconvenientes por la que los usuarios nuevos de Blender no proseguían con el uso de la herramienta fue que esta no era amigable, ya que blender no se basaba en el típico manejo de ventanas de Windows, además de no tener un orden definido para los diferentes manipuladores y modificadores.

A continuación la evolución de Blender y su hoja de ruta.

Release logs: http://www.blender.org/development/release-logs/

- enero de 1995 - Desarrollo de Blender en el estudio de animación NeoGeo.
- 1.23 - enero de 1998 - Versión para SGI (IrisGL) publicada en la web.
- 1.30 - abril de 1998 - Versión para Linux y FreeBSD, se porta a OpenGL y a X.
- 1.3x - junio de 1998 - Creación de NaN.
- 1.4x - septiembre de 1998 - Versión para Sun y Linux Alpha publicada.
- 1.50 - noviembre de 1998 - Primer manual publicado.
- 1.60 - abril de 1999 - C-key (nuevas caracteristicas necesitan ser desbloqueadas, $95), la version de Windows se libera.
- 1.6x - junio de 1999 - Versión para BeOS y PPC publicada.
- 1.80 - junio de 2000 - Fin de la C-key, Blender es totalmente gratuito de nuevo.
- 2.00 - agosto de 2000 - Motor en tiempo-real y reproductor interactivo.
- 2.10 - diciembre de 2000 - Nuevo motor, física y Python.
- 2.20 - agosto de 2001 - Sistema de animación de personajes.
- 2.21 - octubre de 2001 - Blender Publisher lanzado.
- 2.2x - diciembre de 2001 - Versión para Mac OSX publicada.
- 13 de octubre de 2002 - Blender se convierte en código abierto, primera Conferencia de Blender.
- 2.25 - octubre de 2002 - Blender Publisher está de nuevo disponible gratuitamente.

- Tuhopuu1 - octubre de 2002 - Se crea la rama experimental de Blender, un lugar de pruebas para los programadores.
- 2.26 - febrero de 2003 - La primera versión de Blender siendo código abierto.
- 2.27 - mayo de 2003 - La segunda versión de Blender siendo código abierto.
- 2.28x - julio de 2003 - La primera de las series 2.28x.
- 2.30 - octubre de 2003 - En la segunda conferencia de Blender, la interfaz 2.3x es presentada.
- 2.31 - diciembre de 2003 - Actualización a la interfaz de la versión 2.3x, ya que es estable.
- 2.32 - enero de 2004 - Gran revisión de la capacidad del render interno.
- 2.33 - abril de 2004 - Oclusión Ambiental, nuevos procedimientos de Texturas, ¡el motor de juego ha vuelto!.
- 2.34 - agosto de 2004 - Grandes mejoras: Interacciones de Partículas, mapeado LSCM UV, integración funcional de YafRay, Pliegues compensados en Subdivisión de Superficies, Sombreado de Inclinación (Ramp), OSA completo y muchas muchas más.
- 2.35 - noviembre de 2004 - Otra versión llena de mejoras: Object hooks, curve deforms and curve tapers, particle duplicators y mucho más.
- 2.36 - febrero de 2005 - Versión más estable, más trabajo en la escena, pocas nuevas mejoras, mapeado Normal y de Desplazamiento, principalmente.
- 2.37 - junio de 2005 - Un gran avance: herramientas de Transformación y controles, Softbodies, Force fields, deflections, incremental Subdivision Surfaces, sombras transparentes, y renderizado multihilo.
- 2.40 - diciembre de 2005 - Un gran avance: reescritura del sistema de esqueletos, teclas de formas, piel con partículas, fluidos y cuerpos rígidos.
- 2.41 - enero de 2006 - Principalmente añadidos en el Motor de juego y montones de correcciones.

- 2.42 Jul 2006 The Node release. Over 50 developers contributed Nodes, Array modifier, vector blur, new physics engine, rendering, lipsync and many other features.

- 2.43 Feb 2007 The Multi release: Multi functions: multi-resolution Meshes, multi-level UV Textures, multi-layer images and multi-pass rendering and baking, Sculpting, Retopology, multiple additional Matte, Distort and Filter Nodes, modeling and animation improvements, better painting with multiple brushes, fluid particles, proxy objects, Sequencer rewrite, and post-production UV texturing. whew! Oh, and a website rewrite And yes, it still has multi-threaded rendering for multi-core CPUs. With Verse it is still multi-user, allowing multiple artists to work on the same scene collaboratively. Lastly, render farms still provide multi-workstation distributed rendering.

- 2.44 May 2007 The SSS release: The big news, in addition to two new modifiers and re-awakening the 64-bit OS support, was the addition of subsurface scattering, which simulates light scattering beneath the surface of organic and soft objects.

- 2.45 Sep 2007 No new feature have been added, but serious effort has been put in tracking bugs and fixing them. Some performance issues have also been addressed.

- 2.46 May 2008 The Peach release was the result of a huge effort of over 70 developers providing enhancements to the core and patches to provide hair and fur, a new particle system, enhanced image browsing, cloth, a seamless and non-intrusive physics cache, rendering improvements in reflections, AO, and render baking; a mesh deform modifier for muscles and such, better animation support via armature tools and drawing, skinning, constraints and a colorful Action Editor, and much more. It was the release following Project Peach

- 2.47 Aug 2008 Bugfix release

- 2.48 Oct 2008 The Apricot release: cool GLSL shaders, lights and GE improvements, snap, sky simulator, shrinkwrap modifier, python editing improvements

- 2.49 Jun 2009 The Pre-Re-Factor release added significant enhancements to the core and GE. Core enhancements include node-based textures, armature sketching (called Etch-a-Ton), boolean mesh operation improvements, JPEG2000 support, projection painting for direct transfer of images to models, and a significant Python script catalog. GE enhancements included video textures, where you can play movies in-game

(!), upgrades to the Bullet physics engine, dome (fish-eye) rendering, and more API GE calls made available.

- 2.49b- septiembre de 2009 - actualizada API de Python, documentos del motor de juegos y corrección de errores.

- 2.50 Apha 0 - Diciempre de 2009 - versión alfa que se desarrolla en conjunto con el proyecto DURIAN.

- 2.50 Apha 1 - Enero de 2010 - primera versión alfa que se desarrolla en conjunto con el proyecto DURIAN.

- 2.50 Apha 2 - Marzo de 2010 - segunda versión alfa - corrección de bugs y mejor estabilidad.

- 2.53 Beta - Julio de 2010 - Esta versión se denomina "beta" porque es ahora para completar la función de su mayor parte. El tema principal que aún está trabajando en la API de Python,Mejoras en La Interfaz de Usuario,Modelado,Animacion del sistema,Simulacion de Fisica y Render.

- 2.54 Beta - Entre 2,53 y 2,54 hubo una extensa operación de renombrado ha cambiado la secuencia de comandos API mucho, con repercusiones también para la carga de 2,53. Mezcla con la animación en 2,54. Make sure you try the FCurve/Driver 2.54 fix entry from the Help menu. No se pierda la FCurve / Driver 2,54 fijar la entrada en el menú Ayuda.

- 2.55 Beta - Corrección de errors.

- 2.56 Beta - Diciembre 2010 - Prueba y búsqueda de bugs antes de lanzarlo.

- 2.57 Beta – Abril 2011 – última versión estable.

2.3.2 Características Blender3D



Título: Blender 3D
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

Entre las principales características de Blender tenemos lo siguiente:

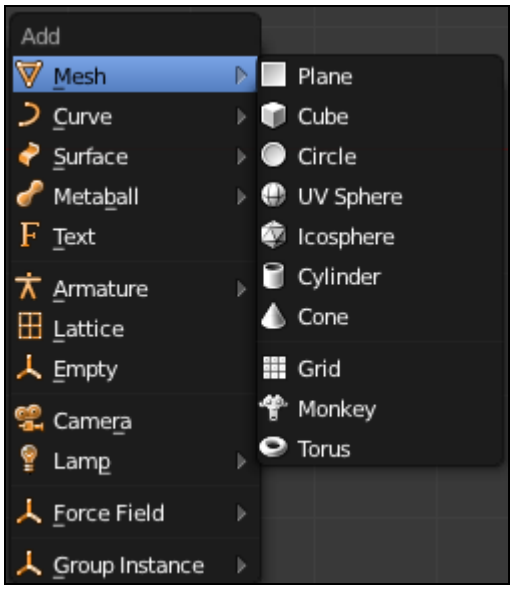
LINK: http://wiki.blender.org/index.php/Doc:ES/Manual

- Multiplataforma, libre, gratuito y con un tamaño de origen realmente pequeño comparado con otros paquetes de 3D, dependiendo del sistema operativo en el que se ejecuta.



Título: Blender 3D - Multiplataforma
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

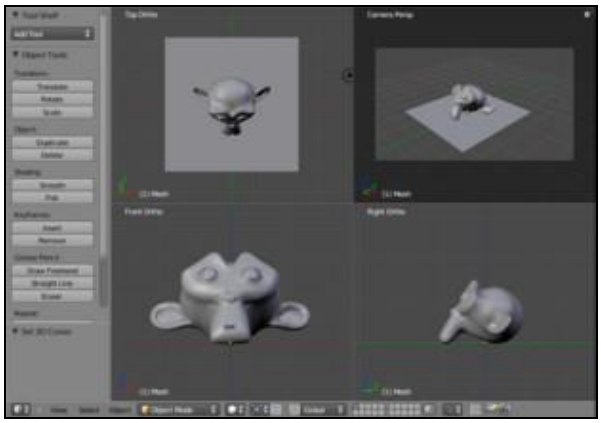
- Capacidad para una gran variedad de primitivas geométricas, incluyendo curvas, mallas poligonales, vacíos, NURBS, metaballs.



Título: Blender 3D – objetos geométricos mas usados en blender 3D.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Herramientas de modelado, operaciones básicas de transformación para estos objetos. Rotación R, traslación G y escalado S.

- Interfaz con sistema de ventanas personalizable (sub regións).



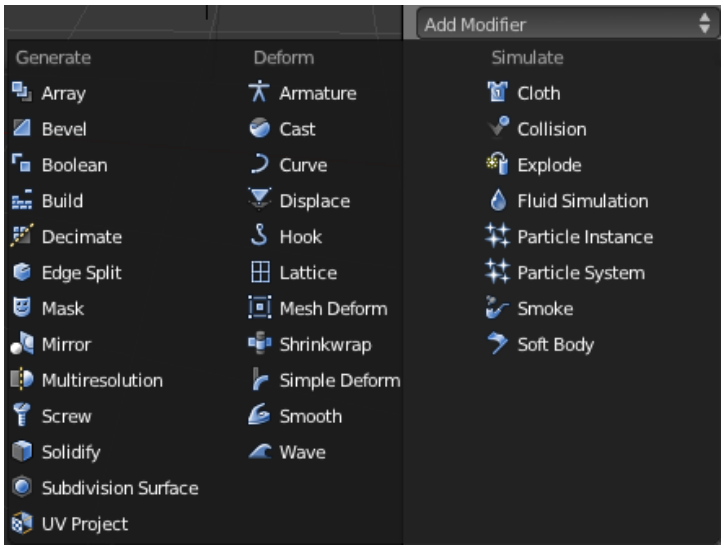
Título: Blender 3D – sistema subregions
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Shortcuts para trabajar de forma eficiente tanto en modo Objeto como en modo Edición.



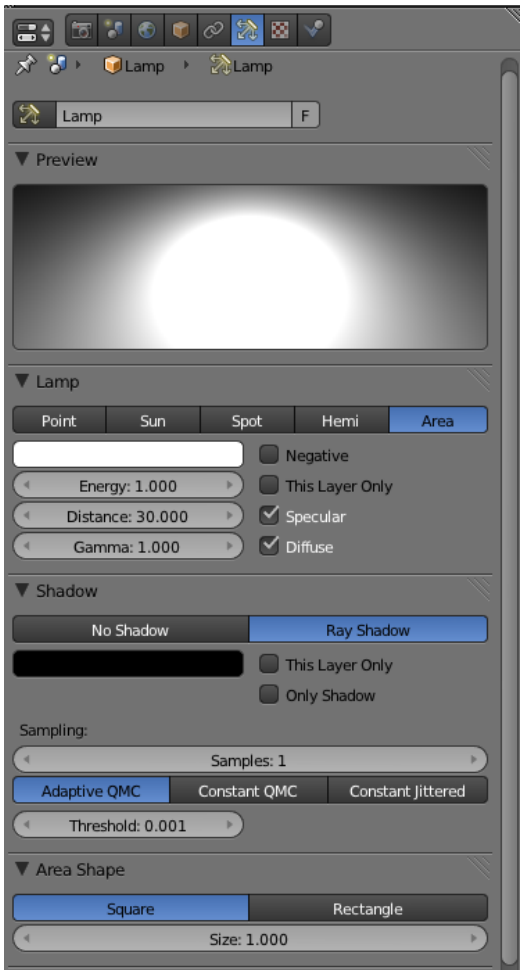
Título: Blender 3D – teclado abreviado
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Modificadores de Generación, Deformación y Simulación, apilables, para la aplicación de transformación no destructiva sobre mallas.



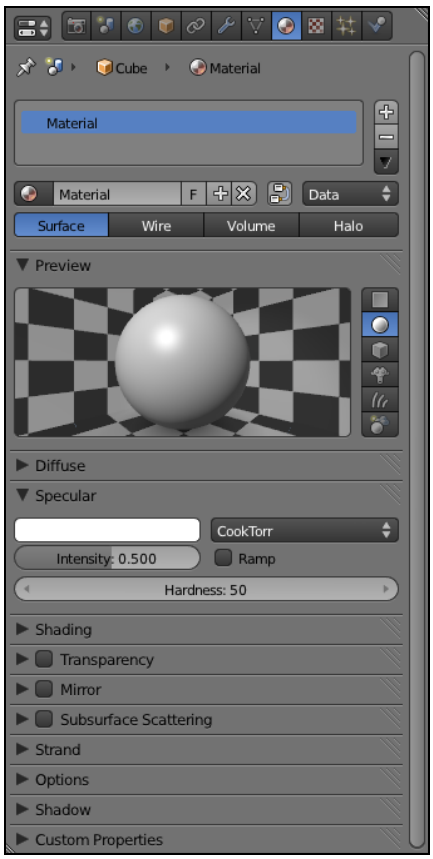
Título: Blender 3D - modificadores
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Iluminación, con sus tipos de luces como: spot, de área, Hemi, Sol. Sombras tipo: rayrace, sombras de buffer y halos volumétricos, Radiosidad y Bake con radiosidad, además del tipo de iluminación: Oclusión Ambientas, Luz Indirecta.



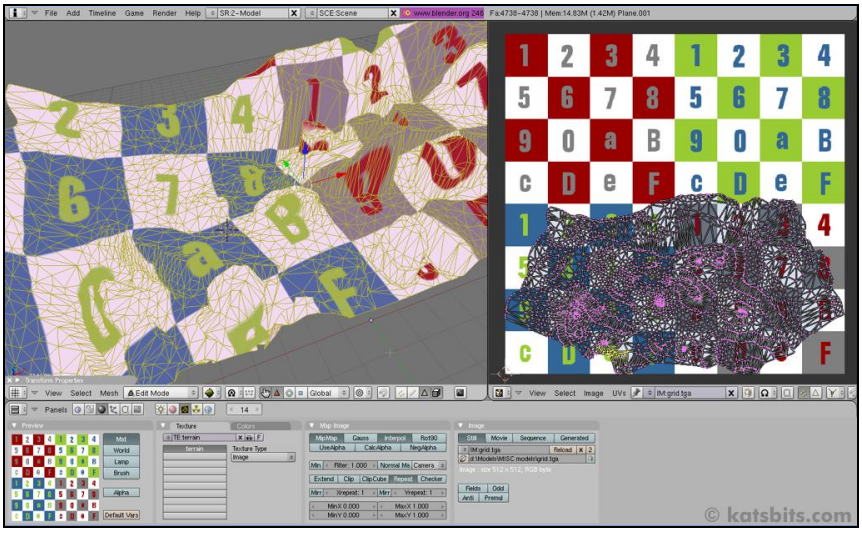
Título: Blender 3D – sistema de iluminación.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Manejo de materiales procedurales, con sus propiedades Diffuse shaders, specular shaders, ambient light effect, color ramps, raytraced reflexions raytraced transparency, surface scattering (SSS), strands. Además del manejo de nodos, vertex paint y Halos.



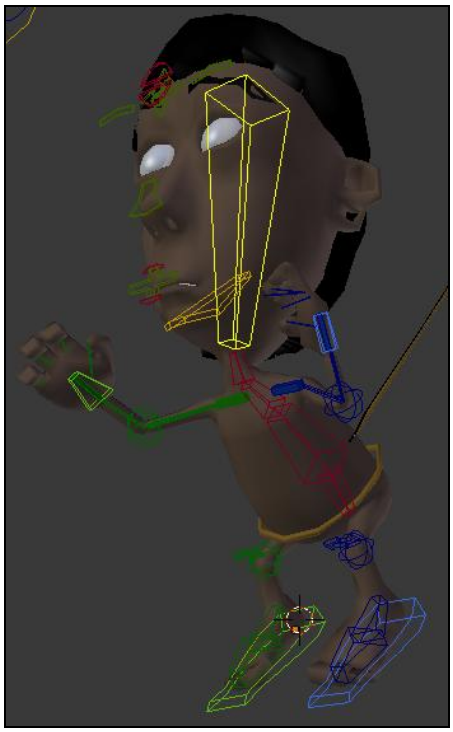
Título: Blender 3D – controles para textura.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Texturas UV, pinturas de vértices, desenvoltura de malla, edición de imagen para uso en UV.



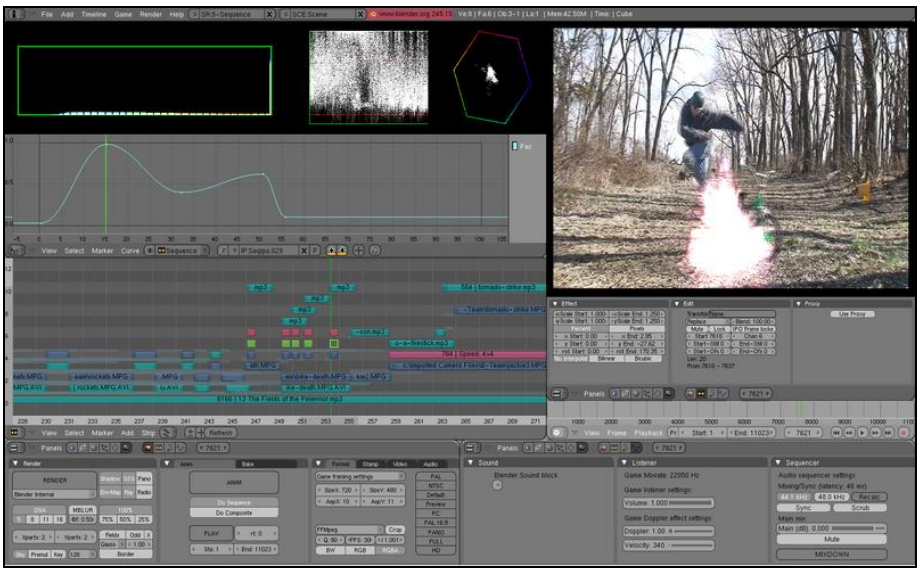
Título: Blender 3D – mapeado UV.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Junto a las herramientas de animación se incluyen cinemática inversa, deformaciones por armadura o cuadrícula, vértices de carga y partículas estáticas y dinámicas.



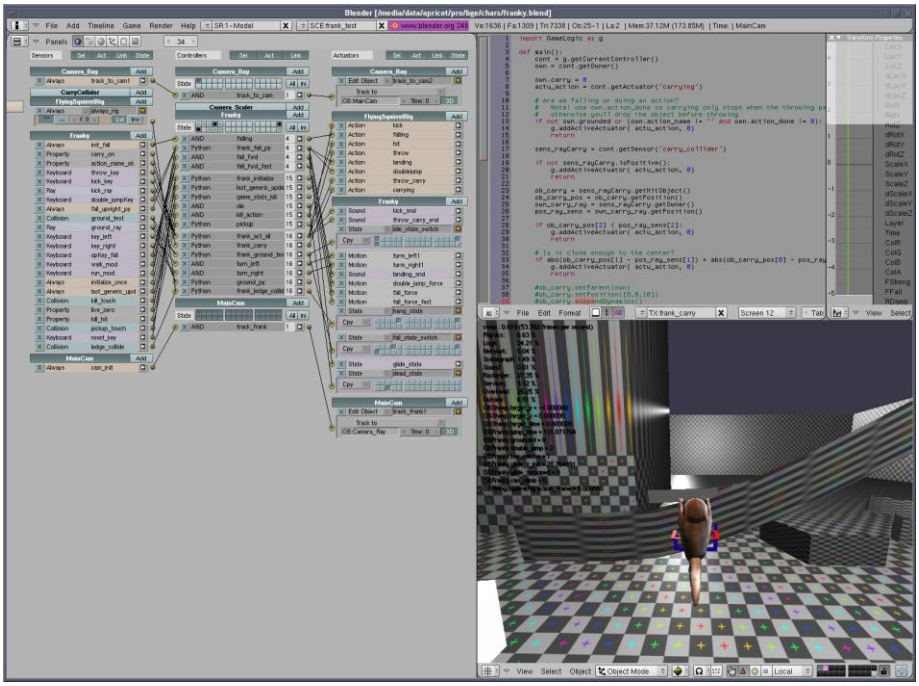
Título: Blender 3D – sistema de huesos Rigging.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Edición de audio y sincronización de video.



Título: Blender 3D – Editor de video.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Características interactivas para juegos como detección de colisiones, recreaciones dinámicas y lógica.
- Motor de juegos 3D integrado, con un sistema de ladrillos lógicos. Para más control se usa programación y desarrollo en lenguaje Python.



Título: Blender 3D – Game engine – sistema de bloques.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Posibilidades de renderizado interno versátil e integración externa con potentes trazadores de rayos o "raytracer" libres como kerkythea, Lux Render, YafRay o Yafrid.

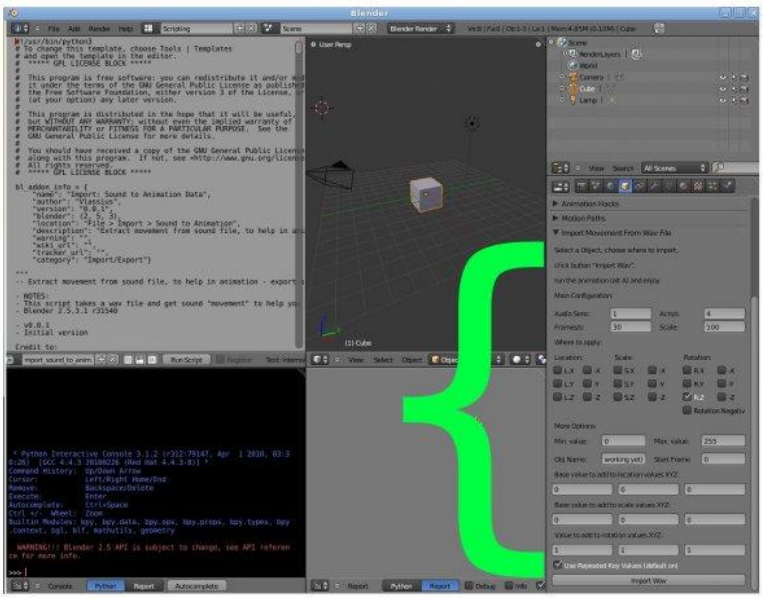


Título: Raytracer externos – LUX render
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Lenguaje Python para automatizar o controlar varias tareas. El mezclador / API Python puede hacer lo siguiente:

    - Editar los datos de la interfaz de usuario puede (Escenas, Mallas, partículas, etc)
    - Modificar las preferencias del usuario, mapas de teclado y la interfaz.
    - Ejecutar las herramientas con configuración personalizada.
    - Crear elementos de interfaz de usuario, como menús, cabeceras y paneles
    - Crear nuevas herramientas.
    - Crear herramientas interactivas.
    - Crear nuevos motores de renderizado que se integran con Blender.

- Definir nuevos ajustes en los datos existentes de Blender.
- Dibujar en la vista 3D utilizando los comandos de OpenGL desde Python.



Título: Blender 3D – lenguaje Python
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- Blender acepta formatos gráficos como TGA, JPG, Iris, SGI, o TIFF. También puede leer ficheros Inventor.
- Simulaciones dinámicas para softbodies, partículas y fluidos.



Título: Blender 3D – simulación softbodies.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

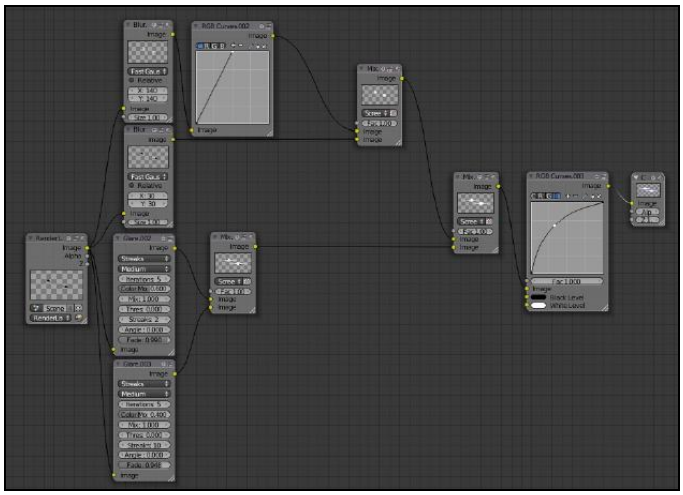
- Sistema de partículas estáticas para simular cabellos y pelajes, al que se han agregado nuevas propiedades entre las opciones de shaders para lograr texturas realistas.



Título: Blender 3D – sistema partículas (cabello)
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

- El sistema de nodos de composición. Mediante los nodos se puede aplicar diferentes ajustes a los elementos de la escena (que se generan en capas de render), permitiendo un preciso control sobre el resultado final y además, permitiendo realizar optimizaciones sobre diferentes capas.

Sistema de nodos



Título: Blender 3D – sistema de nodos para composición
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

Resultado Final



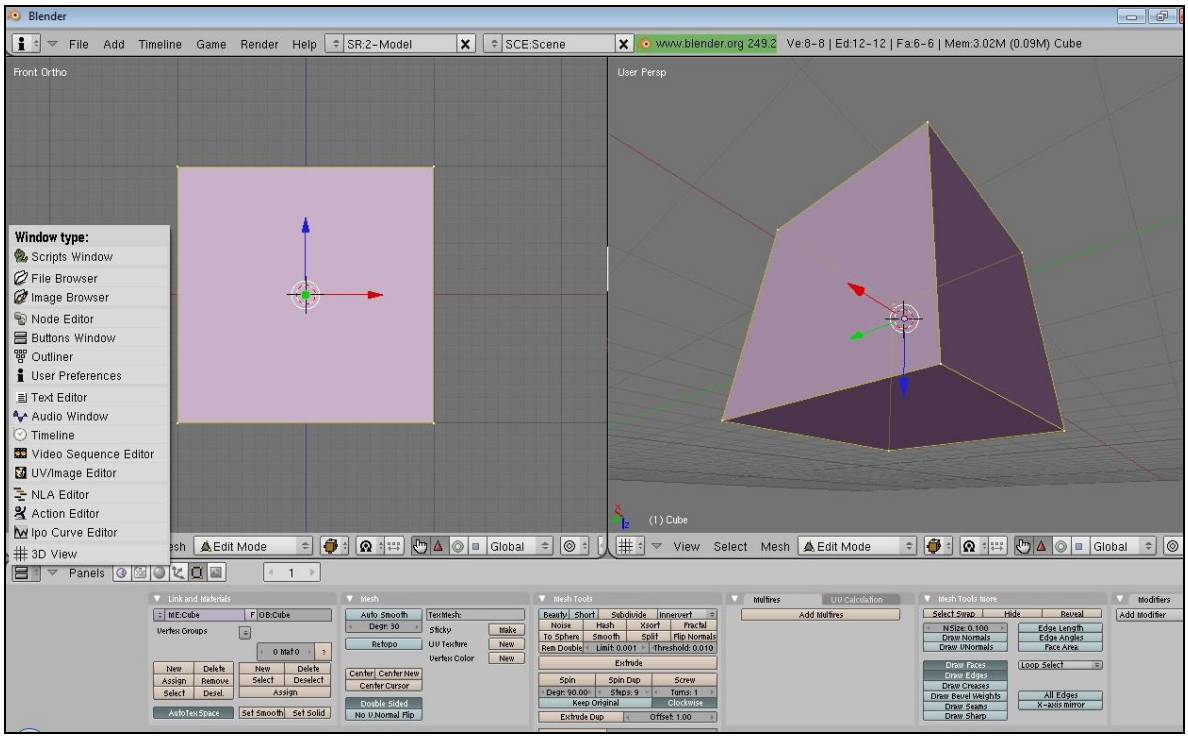
Título: Blender 3D – resultado sistema de nodos
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

2.3.3 Blender 2.49b (última versión estable).

Carateristicas Blender: http://www.blender.org/development/release-logs/blender-249/



Título: Blender 3D versión 2.49b
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.



Título: Blender 3D – interfaz usuario UI – versión 2.49
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

Esta última versión estable de Blender fue la que traía una interfaz poco amigable la cual fue el motivo por el que muchos usuarios de este tipo de software abandonaron o nunca quisieron tomarlo como su herramienta de trabajo.

Todas las características que se crearon hasta esta versión como sistema de nodos, scripting con python, manejo de UV mapping, los sistemas de Animación lineal y no lineal y demás, fueron evolucionando hasta ser totalmente estables.

2.3.4 Caracteristicas para Blender 2.49b

-Video Textura

-Textura en Nodos

-Projection Painting

-Real-time Dome Rendering

-Game Engine mejorado

-Libreria Bullet actualizado para manejo fisicas.

-JPEG2000 support para soportar imágenes JEPG en Alta Calidad y mejor compresión.

-Extensiones para Scritps en Python.

-Correcciones de fallos (bug´s) presentados en la primera versión 2.49.

Esta fue la última versión que la Fundación Blender presentara con su interfaz antigua, la evolución hacia una nueva cara del programa estaba realizándose desde el año 2.007.

Un nuevo blender estaba por aparecer con un rediseño total, mucho mas ordenado con nuevas características y una nueva versión.

Información técnica:

http://www.blender.org/development/release-logs/blender-249/

2.4 Futuro de Blender

2.4.1 Blender 2.56

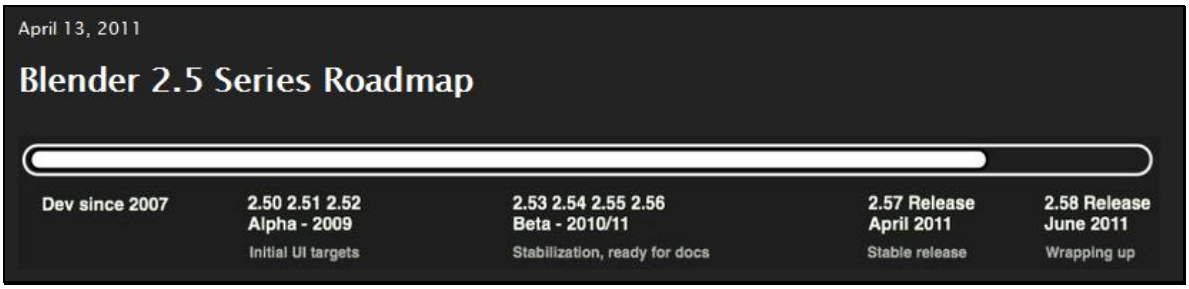
Esta es la primera versión estable de la serie 2.5x, esta representa varios años de rediseño y trabajo de desarrollo.

Esta versión"stable" no solamente es porque tiene las características completas sino que en esta versión se han tenido la corrección de 1000 bugs y actualizaciones que se realizaron desde que se lanzó la primera versión 2.5 beta.

En los próximos meses se trabajará e terminar varios pendientes y gracias a la retroalimentación que brinde la comunidad blender en el mundo estos reportar´an nuevos bugs para poder corregirlos, para que la versión 2.58 que será la versión final de la serie 2.5x sea la que venga con mayor número de mejoras. Para así continuar con la nueva serie versiones que serían la 2.6x.



Título: Blender 3D – última versión estable 2.57
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.



Título: Blender 3D – roadmap – hoja ruta hasta la versión 2.58
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

2.4.2 Blender 2.5x (versión Beta).

Mejoras en las versiones 2.5x



Título: Blender 3D – versión Beta 2.5x
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

La fundación Blender en enero 2010, presento la versión Beta 2.5 alpha1 la que se vino trabajando desde el año 2007, actualmente se encuentra en la versión beta 2.57 es el sexto lanzamiento de una versión beta en miras de la nueva versión que sería la nueva cara de Blender y representara muchos años de rediseño y trabajo de desarrollo.

Esta versión beta llamada asi porque la mayoría de las características están listas, por ejemplo, El API de Python ha tenido muchos cambios, en estas versiones beta se han corregido cerca de 440 bugs.

Esta versión ha sido usada mayormente en la producción del Open project "Durian" del cortometraje "Sintel".

Es decir que mientras se desarrollaba estas versiones beta 2.5x, paralelamente se producía el cortometraje sintel, fueron desarrolladas muchas nuevas características como en Sculpting el mismo es ahora mas estable y con nuevas brochas para su manejo.

Otras características que estaban incorporadas en la versión estable 2.49b, no han sido aun incorporadas para la versión 2.5 o ahora trabajan de diferente manera, como en el caso de los "add –ons". En la versión 2.49b se podía realizar importaciones o exportaciones de archivos según sea el archivo que se deseaba importar o exportar esta opción estaba en el menú herramientas del programa. Ahora si es necesario hacer este tipo de funciones nosotros las activamos desde el menú Add-ons. Existen add-ons para muchas funciones, en esta nueva versión estos add-on se los maneja a manera de plug-ins.

2.4.3 Información general

La interfaz por defecto (la que seguramente se quiere cambiar). Ésta se compone de Editores, Cabeceras, Botones de contexto, Paneles y Controles.

- En Blender, llamamos Ventana (llamado en este apartado Editor pero mas adelante sera solo "Ventana") a la parte del software que responde a una función específica (ventana de Vista 3D, ventana de Propiedades, ventana de Edición de secuencia de vídeo, ventana de Editor de nodos ...). Cada ventana tiene su propia Cabecera (Header) en la parte superior o en el fondo.

- Los Botones contextuales permiten el acceso a las opciones. Son como las fichas, y que a menudo se colocan en una ventana dentro de su cabecera (como el Editor de propiedades).

- En cada uno de los editores, las opciones se agrupan en Paneles para organizar lógicamente la interfaz (panel Shadow, panel Color, panel Dimensions ...).

- Las Barras laterales están incluidas en algún editor. En ese caso, los paneles y los controles se agrupan allí. Para la optimización de espacio de trabajo, es posible ocultar temporalmente barras laterales.

- Los paneles contienen Controles. Estos pueden permitirle modificar una función, una opción o un valor. En Blender, hay varios tipos de controles:

- Buttons : Permiten el acceso a una herramienta (Translate, Rotate, Insert Keyframe). Estas herramientas normalmente tienen accesos de teclas para hacer más rápido el trabajo. To display the shortcut, just hover your mouse over a button to see the tooltip.o
- Checkbox.jpg
- Checkboxes : Permiten activar o desactivar una opción. Este control solo puede contener un valor booleano (True/False, 1/0).
- Slider.jpg
- Sliders : Puedes introducir un valor flotante. Estos pueden ser limitados (Desde 0.0 → 100.0) o no (Desde -∞ → +∞). Existen dos tipos de reguladores en Blender.
- List.jpg
- Select menus : Permiten seleccionar un valor de una lista. La diferencia entre esto y un Checkbox es que los valores estan nombrados y no puede haber mas de dos valores en estos menus.

Entre las características del nuevo Blender tenemos:

2.4.4 Interfaz renovada

La interfaz de usuario del nuevo Blender se basa en tres principios fundamentales:

- Nada se superpone : La interfaz de usuario le permite ver todas las opciones pertinentes y las herramientas de una vez sin empujar o arrastrar ventanas por todos lados.
- Nada se bloquea : Las herramientas y opciones de la interfaz, no bloquean al usuario en la mayoria de las parte del Blender. Blender no lanza ventanas que requieran que el usuario rellene datos antes de ejecutar las cosas.

- No Modal : La entrada de los usuarios deberían seguir lo más coherente y predecible como sea posible sin cambiar los métodos de uso general (ratón, teclado) sobre la marcha.

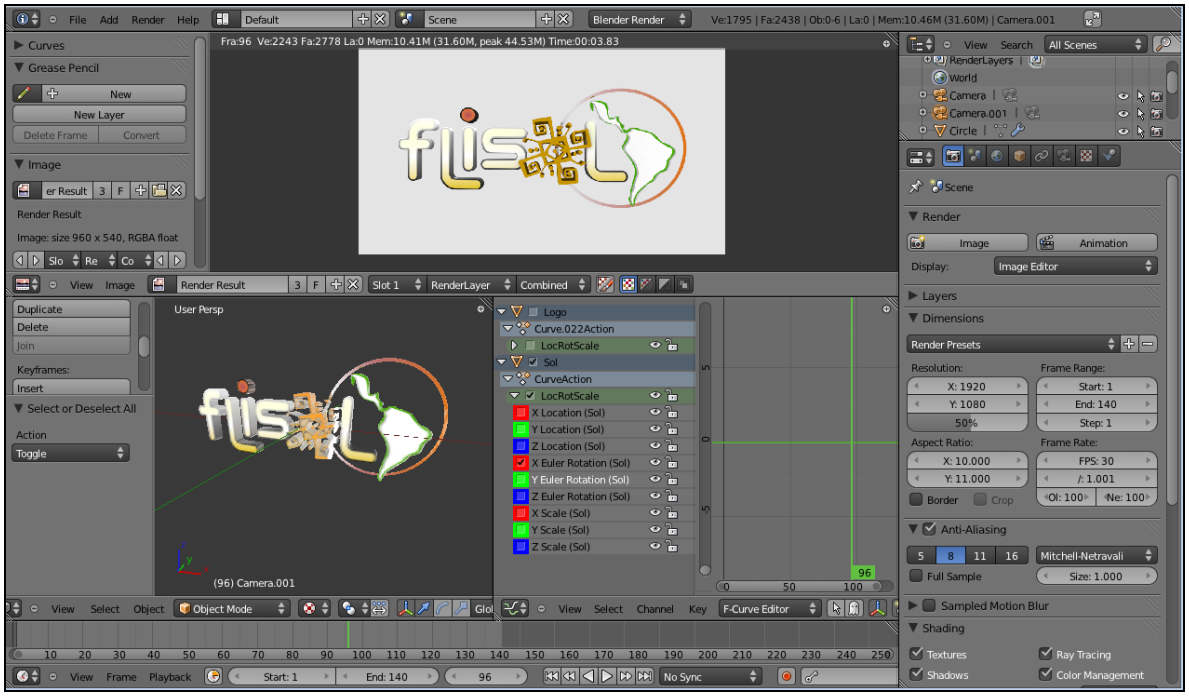
Además, Blender 2.5 permite múltiples ventanas para múltiples pantalla de configuración. Se trata de una excepción a la regla de "Nada se superpone".

Una Interface Poderosa, como la interfaz se dibuja bajo OpenGL, puedes organizar la pantalla de Blender exactamente a tu gusto para cada tarea específica. Esta disposición de la pantalla puede ser nombrada y guardada.



Título: Blender 3D - openGL
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

El sistema de administración de ventanas ha sido una de las características que ha tenido más mejoras, esta permite mayor flexibilidad e intearctividad con el usuario.

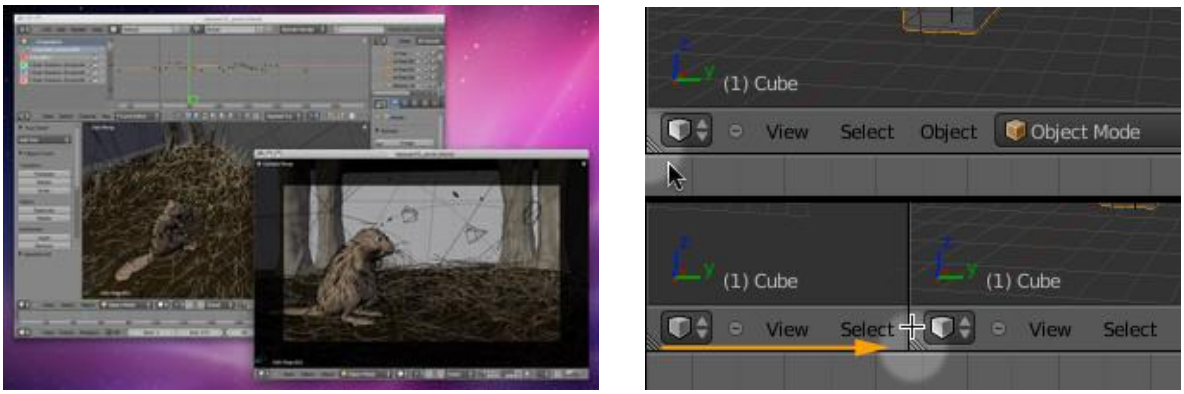


Título: Blender 3D interfaz – versión 2.5x Beta.
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

2.4.5 Multiples Ventanas

Como en la versión anterior Blender. Esta nueva versión soporta múltiples ventanas con cualquier combinación, las mismas que gracias a nuevas características que vienen en esta pueden ser modificadas o suprimidas con facilidad.

Además las ventanas pueden ser unidas o divididas con la nueva característica de "Split zone".

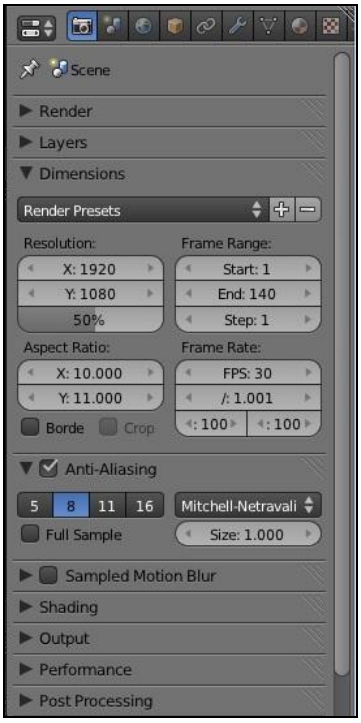


Título: Blender 3D – multiples ventanas
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

2.4.6 Nuevo Diseño - GUI Controls

Lps paneles para manejo de propiedas de objetos en Blender han tenido una completa transformación y reorganización , en las versiones anteriores existía un completo desorden y tuvieron una mayor sistematica revisión.

El nuevo editor de propiedades ha sido rediseñado y reorganizado, ahora es mas flexible y puedes trabajar con mas eficiencia, por ejemplo se puede esconder las propiedades que no están usando en ese momento, por lo que es mas claro y fácil de entender su manejo.



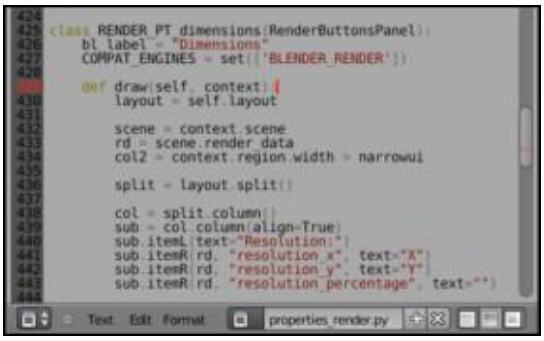
Título: Blender 3D – UI controles mejorados
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

2.4.7 Python Mejorado

Los nuevos diseños de interfaz de usuario son generados por los archivos de script Python, como parte de la instalación estándar de Blender. Estas secuencias de comandos de presentación puede ser totalmente personalizado, con modificaciones, en tiempo real que le permite modificar la interfaz de usuario existente o crear interfaces de usuario completamente nueva a partir de cero. Porque es el mismo sistema, creado por el usuario IU pitón tienen las mismas capacidades y la funcionalidad por defecto.

La nueva interfaz que trae esta nueva versión de blender son generados por python como parte de su nueva instalación estándar. Los scripts generados para la interfaz pueden ser totalmente personalizados, con modificaciones en tiempo real.

Esto permite modificar la interfaz de usuario o crear interfaces de usuario completamente nueva a partir de cero. Porque es el mismo sistema, creado por el usuario IU pitón tienen las mismas capacidades y la funcionalidad.

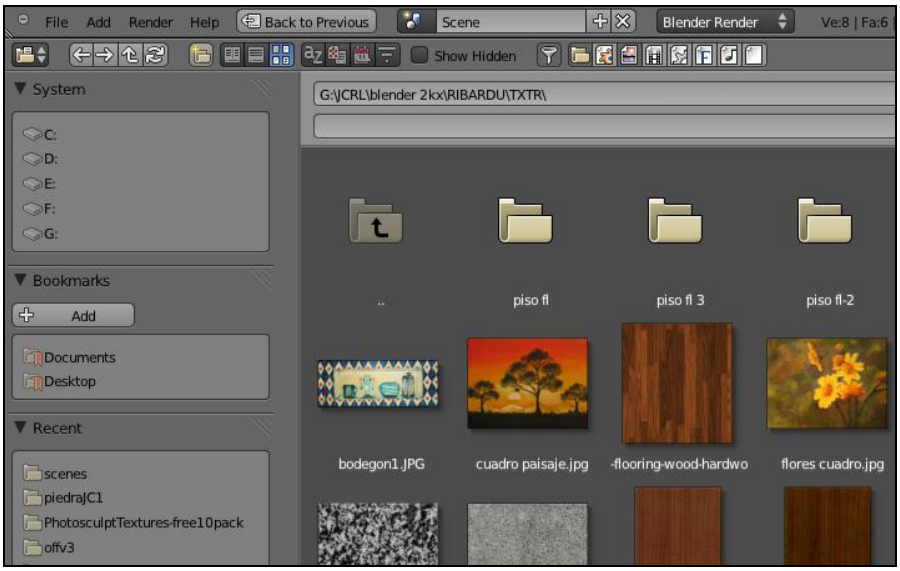


Título: Blender 3D – scripting python
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

2.4.8 Explorador de Archivos

Explorador de archivos interno de Blender ha sido actualizado con un diseño más amigable y más características útiles. Ahora puede cambiar entre diversas (lista / columna / miniatura) puntos de vista, con la carga de fondo de las miniaturas de cualquiera de los formatos de imagen soportados Blender.

Filtros rápidos por tipo de archivo están disponibles, y una nueva lista marcador personalizable. Importadores y Exportadores ofrecen sus ajustes en la región de las propiedades de la izquierda, para ajustar al mismo tiempo, como la selección de archivos.



Título: Blender 3D – administración de ventanas
Fuente http://www.blender.org
Elaboración: Juan Carlos Reascos Laverdi.

Información Técnica:

http://www.blender.org/development/release-logs/blender-256-beta/updated-gui/

http://download.blender.org/documentation/bc2008/evolution_of_blenders_ui.pdf

http://wiki.blender.org/index.php/BlenderDev/Blender2.5/WM_Architecture

http://wiki.blender.org/index.php/BlenderDev/Blender2.5/AreaManager

2.4.9 Builds de Blender 3D

Dado que blender tiene liberado su código fuente, existen diferentes versiones paralelas en las que este viene configurado con muchas opciones que hacen mas potente según el tipo de modificación o adaptación que se haya realizado a blender. Por ejemplo tenemos versiones de blender 2.5x con el raytracer Lux render

Blender Trunk [36114] + LuxRender – Windows (ejemplo de Build, blender + LuxRender)

Link: http://www.graphicall.org/builds/builds/showbuild.php?action=show&id=1855

Link: Graphicall.org

http://www.graphicall.org/builds/

2.5 PYTHON EL LENGUALE DE BLENDER

Python es un lenguaje de scripting de propósito general y tiene una interfaz especial para acceder a todas las funciones internas de Blender en este lenguaje. Los scripts son escritos en este lenguaje para extender las funcionalidades de Blender, sin tener que recompilar y enlazar la distribución binaria. Estos scripts son escritos por usuarios-programadores. La versión recomendada de Python se incluye e instala normalmente con la distribución, sin embargo, también se puede descargar directamente desde desde el sitio oficial de Python, e instalarla por separado. (Cuando se descarge Python por separado, debe prestar atención al número de la versión, ya que la necesitará al compilar Blender. La versión de Python se muestra al iniciar Blender en la ventana principal.

En general, cualquiera sea el momento en el que se instale Python, necesitará establecer una variable de entorno de sistema operativo llamada PYTHONPATH y apuntar al directorio de los Scripts de Blender donde estén instalados los módulos de Python, por ejemplo "C:\Program Files\Blender Foundation\Blender\scripts\bpymodules" para las computadoras con Windows. Las variables de entorno en las computadoras de Windows son configuradas en las opciones avanzadas del Sistema en el Panel de Control.

A diferencia de otros programas con los que se podría estar familiarizado, Blender no es monolítico y estático. Puede extender sus funcionalidades sin tener que modificar la fuente y recompilarlo. Existen dos formas de extender Blender: Scripts Python and Plugins Binarios.

2.5.1 Scripts Python

A parte del ejecutable de Blender, existen, literalmente cientos de scripts escritos por mucha gente de la comunidad:

- Varios scripts están incluídos en Blender.
- No todos los scripts pueden estar incluídos en la versión de lanzamiento de Blender por razones logísticas o de soporte, pero muchos de ellos funcionan de manera confiable y son muy útiles, de modo que han sido catalogados y están disponibles en el web en el Catálogo de scripts y en el nuevo Rastreador de Extensiones.

2.5.1.2 Scripting y Blender

Estas son las formas básicas para ejecutar secuencias de comandos en Blender:

1. Pueden ser cargados o escrito como archivos de texto en la ventana Editor de texto, entonces se ejecuta con ALT + P.
2. A través de la línea de comandos: blender-P <scriptname> comenzará Blender y ejecutar el script. <scriptname> puede ser un nombre de archivo en el sistema de archivos del usuario o el nombre de un texto guardado en una mezcla de Blender de archivo:. 'licuadora myfile.blend textname-P'.
3. A través de la línea de comandos en segundo plano: utilizar la opción '-b' (el orden es importante): la licuadora-b <blendfile>-P <scriptname>. <blendfile> puede ser cualquiera. blend, incluyendo el valor por defecto. B.blend que está en Blender.Get Blender directorio de inicio ("directorio home"). En este modo, ninguna ventana se abrió y el programa va a salir tan pronto como el script finaliza la ejecución.

4. Correctamente las secuencias de comandos registrados pueden ser seleccionados directamente desde los menús del programa.

5. Scriptlinks: estos también se cargan o escrito en la ventana Editor de texto y pueden ser vinculados a los objetos, materiales o escenas utilizando la ficha ScriptLink botones. Los vínculos de script se ejecutan automáticamente cuando sus eventos (onLoad, volver a dibujar, FRAMECHANGED) se ha disparado. Normal puede crear secuencias de comandos (texto) y el enlace otras secuencias de comandos a los objetos y eventos, vea Object.Object.addScriptLink, por ejemplo.

6. Una secuencia de comandos puede llamar a otra secuencia de comandos (que se ejecutará en su propio contexto, con su propio diccionario global) con la función Blender.Run módulo.

La interacción con los usuarios:

Los scripts pueden:

- Simplemente Ejecutar o Salir;
- Mensajes emergentes, menús y el número pequeño y cuadros de texto de entrada;
- Elaborar interfaces gráficas de usuario (GUI) con llamadas OpenGL y los botones de programa nativo que permanecen allí.
- Se unen al evento de un espacio o el código de dibujo (también conocido como controladores de espacio);
- Hacer cambios a la vista 3D (conjunto de capas visibles (s), punto de vista, etc);
- Agarra la cola de entrada principal evento y el proceso (o pasar a Blender) teclado seleccionada, el ratón, vuelva a dibujar eventos.
- Usar Blender para ejecutar secuencias de comandos de otros (ejemplo Blender.Run ());
- Bibliotecas uso externo Python.

2.5.2 Plugins

Hay docenas de plugins que también proporcionan extensibilidad a Blender en dos áreas principales:

- Plugins de texturas - Cambian la manera en los materiales se muestran.
- Secuencias de video - Hacen algo parecido a un video stream.

La instalación por defecto de Blender incluye algunos plugins, pero puede encontrar otros en el Repositorio como el de Kent Mein y en el nuevo Rastreador de Extensiones.

2.5.3 Funcionalidad Python en Blender

Blender tiene una característica muy poderosa que a menudo es pasada por alto. Esta característica es un intérprete de Python totalmente funcional. Esto le permite a cualquier usuario añadir funcionalidades a Blender escribiendo un simple script de Python.

Python es un lenguaje de programación interpretado, interactivo, y orientado a objetos. Incorpora módulos, excepciones, tipeado dinámico, tipo de datos dinámicos de muy alto nivel y clases. Python combina un gran poder con una sintaxis muy sencilla. Está expresamente diseñado para ser usado como una extensión para las aplicaciones que necesiten una interfaz programable, y esto es por lo que Blender lo utiliza.

Estructuras de datos en blender.

Existen programas para manipulación de estructuras de datos, Python de Blender es uno de ellos, tiene una arquitectura orientada a objetos, la interfaz BPython presenta objetos de blender y sus atributos de la misma forma que en la interfa (GUI),

2.5.4 La API del motor de juegos de Blender:

Blender tiene un motor de juego para los usuarios para la creación y programación de juegos en 3D. Este motor permite a los programadores agregar secuencias de comandos para mejorar la interfaz del juego, control, etc. Haciendo la interacción mas compleja y con mayor numero de trucos. El motor de juego de la API es independiente de la API de Blender.

Este motor permite a los programadores agregar secuencias de comandos para mejorar la IA del juego, control, etc, haciendo una interacción más compleja y trucos posibles. El motor del juego de la API es independiente de la API de Blender Python hace referencia este documento

Link: http://www.blender.org/documentation/249PythonDoc/GE/index.html

Link: http://wiki.blender.org/index.php/Release_Notes/Notes246/Python_API

CAPITULO III

ESTRUCTURA DEL CORTOMETRAJE

Primeramente desearía mencionar que lo que me ha motivo a realizar un cortometraje animado en 3D para mi proyecto de tesis, es el poder vincular dos áreas diferentes, pero cuando éstas son trabajadas a la par se pueden lograr resultados impresionantes, estas son el Arte y la Tecnología. "El arte reta a la tecnología, la tecnología inspira al arte" – John Lasseter - CEO Pixar.

Una de las finalidades del cortometraje es tener los conocimientos para el proceso de construcción del mismo, como también es el poder hacer uso de herramientas libres tanto en sistemas operativos como en aplicaciones de manejo de gráficos tridimensionales y utilidades que fui descubriendo a medida que el proyecto avanzaba.

En cuanto a la historia y al desarrollo del cortometraje todo comenzó con una idea, se ha seguido los diferentes procesos y etapas para ir trabajando la misma, como se trata de una pequeña producción cinematográfica en su elaboración hay diferentes etapas como son preproducción, producción y post-producción.

Como este cortometraje es una producción audiovisual, el mismo dura sustancialmente menos que el tiempo medio de una película de producción normal.

A continuación se describen las etapas de desarrollo de "Leroy" un cortometraje animado en 3D.

Pipeline Animación "Leroy"

El sistema de trabajo en este cortometraje se ha reducido lo mas mínimo posible, ya que un trabajo de este tipo requiere de un conjunto de profesionales muy capacitados en su área y se logre un flujo de trabajo necesario para lograr este tipo de producciones.

Para este trabajo se necesitó de la ayuda de un ilustrador para la creación de personajes y el diseño del storyboard, así como también de la información proporcionada

en los foros de las diferentes comunidades de Animación 3D tanto para el uso de la herramienta, en sus varios aspectos, ya que como se trata de una herramienta con crecimiento acelerado y ser libre, no hay mucha información disponible, muchas veces ni en idioma inglés.

De esta manera se logró obtener un flujo de trabajo para la creación de Leroy.

"LEROY"

3.1 PREPRODUCCION:

En la preproducción se plasmó la idea que se tenía respecto al concepto de Leroy, la historia que quería contar, además utilizar esta producción como objeto de estudio para la tesis propuesta.

Teniendo la historia que se va ha contar, procedemos a desarrollar los componentes mas importantes de la preproducción como son: el Guión para crear interacciones entre nuestros personajes y la música, el Story Board, los Layouts o diseño de personajes o de escenarios , Hojas de Modelos (poses principales de personajes) y el animatic.

3.1.1 SINOPSIS (historia)

Un día de esos en que la mala suerte se presenta justo en el momento y en lugar menos esperado.

Después de un largo dia, Leroy está de vuelta a su hogar, sin darse cuenta que no ha cargado combustible, su nave llega a un punto en que no se mueve más, sin su maquina para poder trasladarse y con la próxima estación de servicio a 20Km, Leroy trata de pedir ayuda a los pocos transeuntes de la carretera, sin darse cuenta que puede pasarle algo mucho peor.

3.1.2 GUIÓN

- EXTERIORES. CARRETERA DESIERTA - ATARDECER.

(Año 2700 D.C) La luz del sol cae en la carretera desierta. Ningún vehículo se encuentra en la carretera. La música se escucha de repente por el camino.

Leroy un robot con mucho kilometraje regresa en su nave de provisión, a su estación después de un día cansado en las canteras, escuchando sus canciones que le hacen sentir mejor y si son clásicas mucho mejor, parece que nada lo molesta solo quiere llegar y poner su batería a cargar.

Cuando Leroy empieza a sentir que todo está listo para continuar con su rutina, la música comienza a fallar y la nave se comienza a trabar perdiendo su potencia y su única forma de transporte.

De repente Leroy se queda varado solo pudo orillarse a un lado de la vía y ver en el panel como los medidores de poder están en Cero. Rendido ante su mala suerte, solamente puede hacer dos cosas una ir en busca de ayuda a la estación de reabastecimiento más cercana o pedir ayuda a algún otro robot que se encuentre en la vía y que quiera ayudarlo.

A unos cuantos metros de ahí, un letrero llama la atención de Leroy, el se dirige hacia él, cuando lo ve se da cuenta que está a mucha distancia 20 kilómetros del próximo punto de auxilio y su ánimo se pone de mal en peor.

Como se da cuenta de que no hay visos de que alguien venga o vaya por la carretera, el se resigna a empujar su nave hasta que algo mejor le pueda pasar o termine llegando a donde pueda encontrar ayuda.

Mientras Leroy esta se encuentra en a penosa tarea de empujar su vehículo, a lo Lejos viene un transporte de carga y empieza a caminar hacia su encuentro, mientras el transporte se acerca Leroy da muestras de alivio y salta pidiendo ayuda. El transporte que está programado para salir y llegar a su destino pasa a una buena velocidad dejando detrás a Leroy, nuevamente se ha dejado solo en la carretera Leroy, mientras está renegando por su suerte a lo lejos escucha a otro vehículo acercándose, se trata de una nave ligera que se aproxima a gran velocidad, esta vez Leroy sale muy rápidamente a su encuentro, el momento que llega a su encuentro Leroy trata de hacer contacto con sus ocupantes.

Mientras tanto una grúa del puesto de auxilio que esta programado para retirar los obstáculos en la vía, esta por retirar del camino a la nave de Leroy y dado que la nave se encuentra alejado de donde Leroy está ubicado, simplemente lo engancha para llevarlo.

Leroy se acerca al vehículo con ventanas oscuras trata de establecer comunicación y de repente la ventana del pasajero se abre, por dentro se encuentra un robot de mal aspecto que lo mira de arriba abajo y decide proseguir sin decirle nada.

Leroy demuestra su malestar con gestos al ver partir al vehículo que lo dejo abandonado.

Pero su mala suerte no podía ser peor al darse cuenta que a lo lejos su nave está siendo remolcada y alejándose junto a la nave que hace un momento se fue también.

Quedándose solo en la carretera.

CORTE A NEGRO.

FIN.

3.1.3 DISEÑO PERSONAJES

Basado en la historia, la misma nos entrega las principales cualidades de cada uno de ellos, la construcción de cada uno de ellos y el trabajo de ilustración para las futuras referencias de modelado y animación 3D.

3.1.3.1 Leroy:

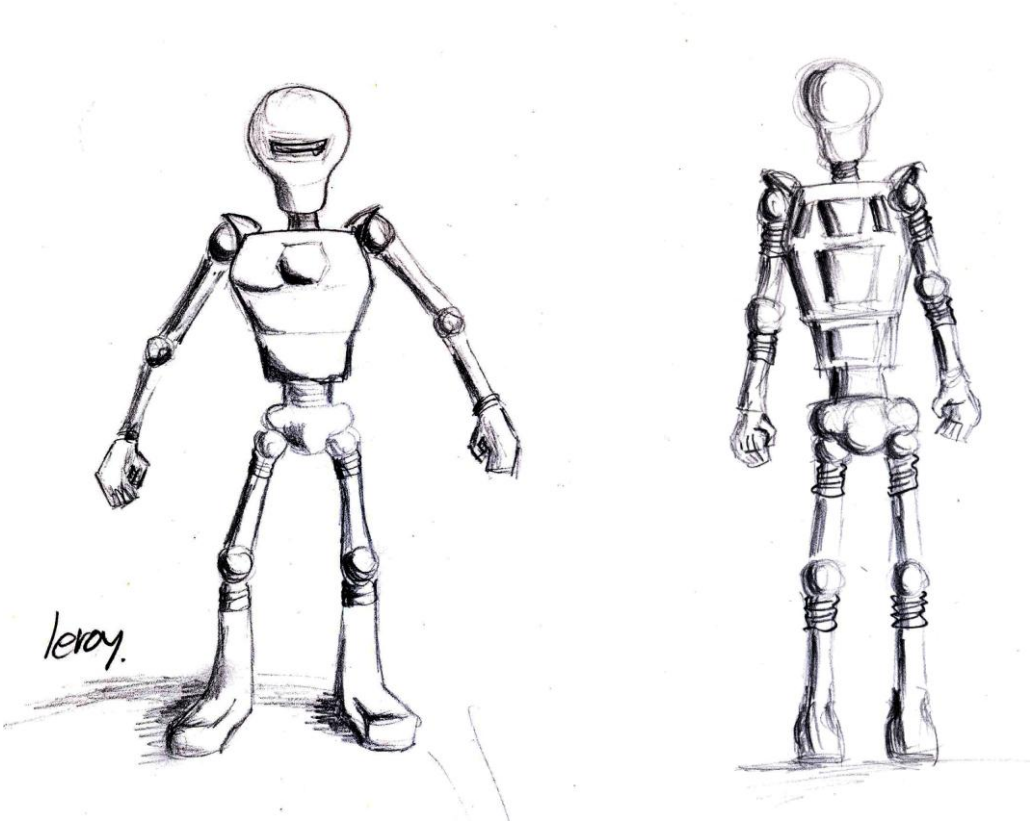
Descripción:

Leroy un robot programado para ser obrero operador de maquinaria en una mina, goza de una supuesta autonomía, tiene su propia estación de recarga a la que va todos los días después de la jornada diaria de trabajo, es ya un robot con mucho kilometraje.

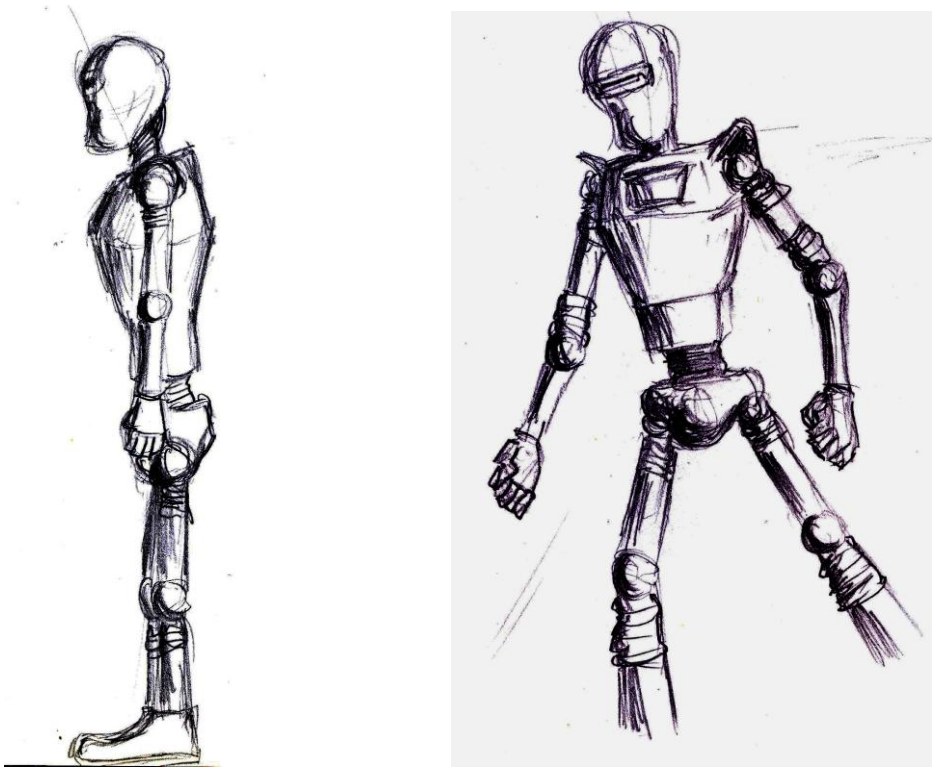
Físicamente.

Bípedo, delgado, Un solo ojo, desgarbado, no tiene boca solo se puede comunicar con su superiores por ondas electromagnéticas.

Bosetos / Model Sheet



Título: Vista Frontal y posterior – personaje Leroy
Elaboración: Juan Carlos Reascos Laverdi.



Vista Lateral y Perspectiva

Título: Vista Lateral y perspectiva – personaje Leroy
Elaboración: Juan Carlos Reascos Laverdi.

3.1.3.2 MOMO:

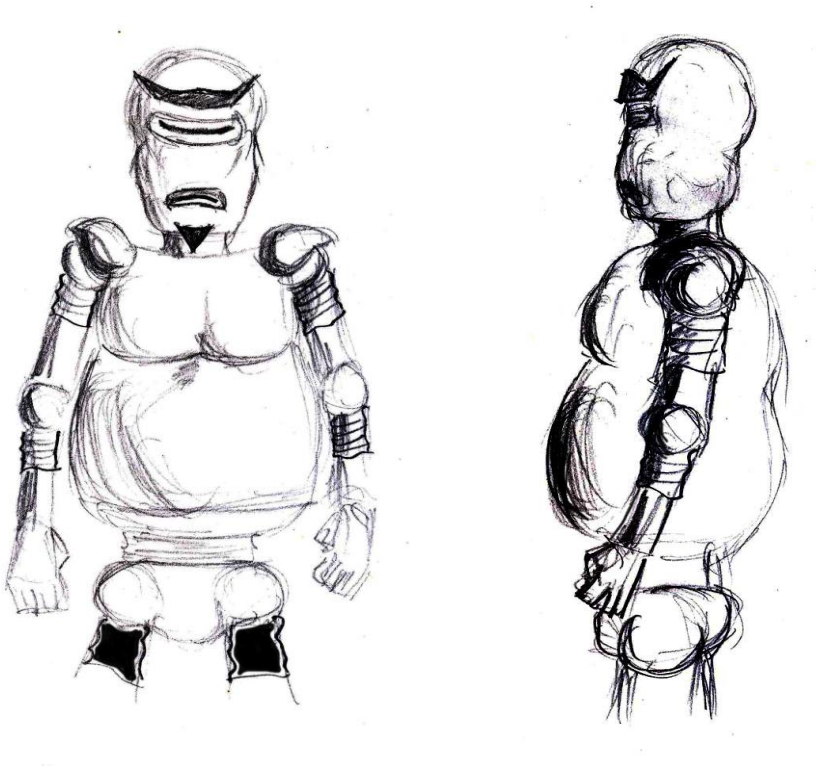
Descripción:

Robot, dedicado a la explotación de robots con defectos, como se trata de un negocio fraudulento evita el contacto con otros bots que le puedan ocasionar problemas como Leroy.

Físicamente.

Bípedo, contextura gruesa, Un solo ojo, obeso,

Bosetos / Model Sheet

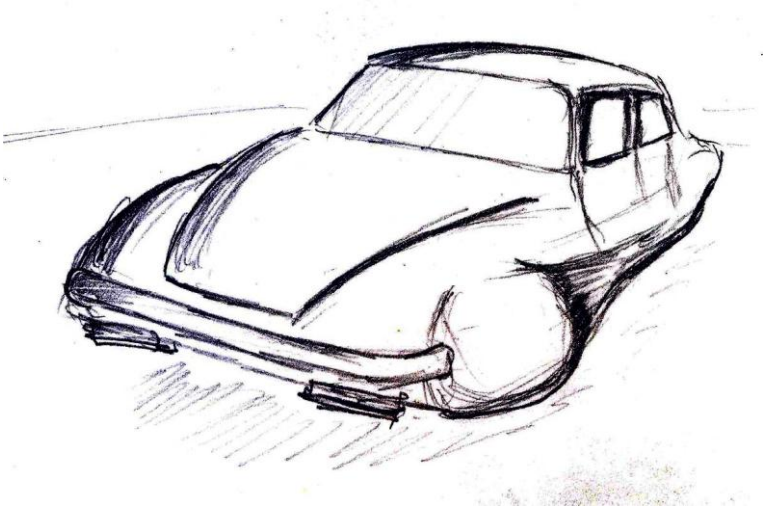


Título: Vista Frontal y Lateral – personaje Momo
Elaboración: Juan Carlos Reascos Laverdi.

Utilería:

Para el desarrollo de este cortometraje se necesitó diseñar varios elementos necesarios como los vehículos, además de otros objetos de utilería como son los 34Props para lograr la composición de las escenas necesarias.

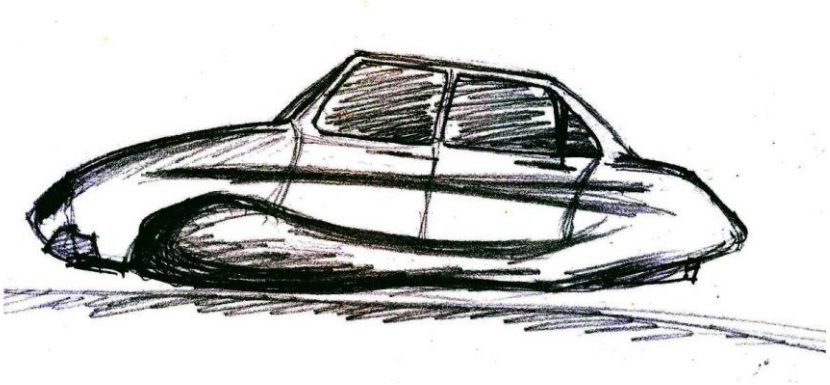
3.1.3.3 Nave Momo:



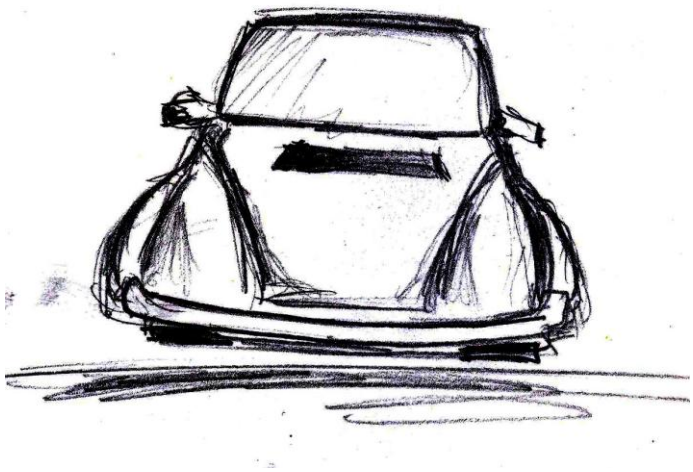
Vehículo Momo – Vista Perspectiva

Título: Vehículo Momo – vista perspectiva
Elaboración: Juan Carlos Reascos Laverdi.

[34] **Props:** es una recreación coleccionable de la utilería (en inglés "prop").

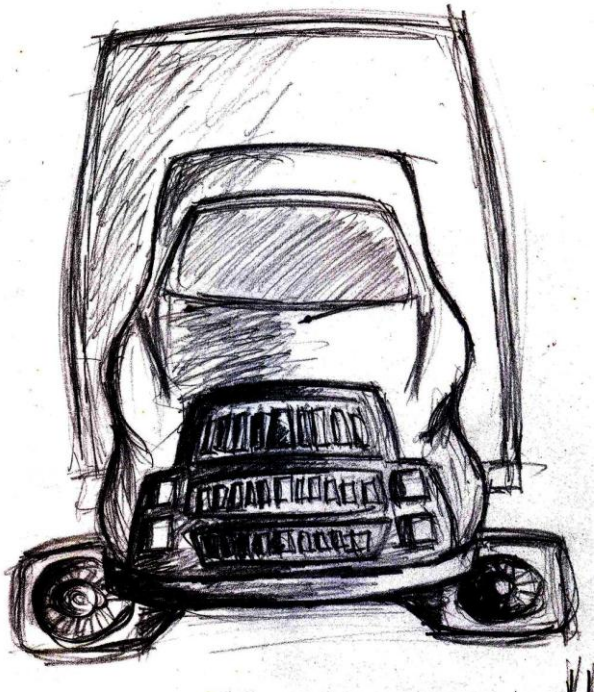


Título: Vehículo Momo – vista Lateral
Elaboración: Juan Carlos Reascos Laverdi.

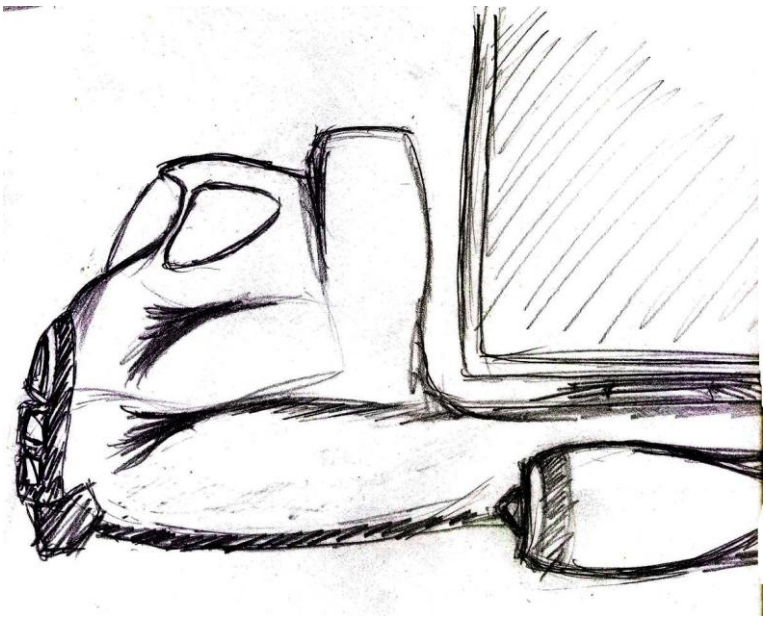


Título: Vehículo Momo – vista Forntal
Elaboración: Juan Carlos Reascos Laverdi.

3.1.3.4 Nave Vehículo de Carga:

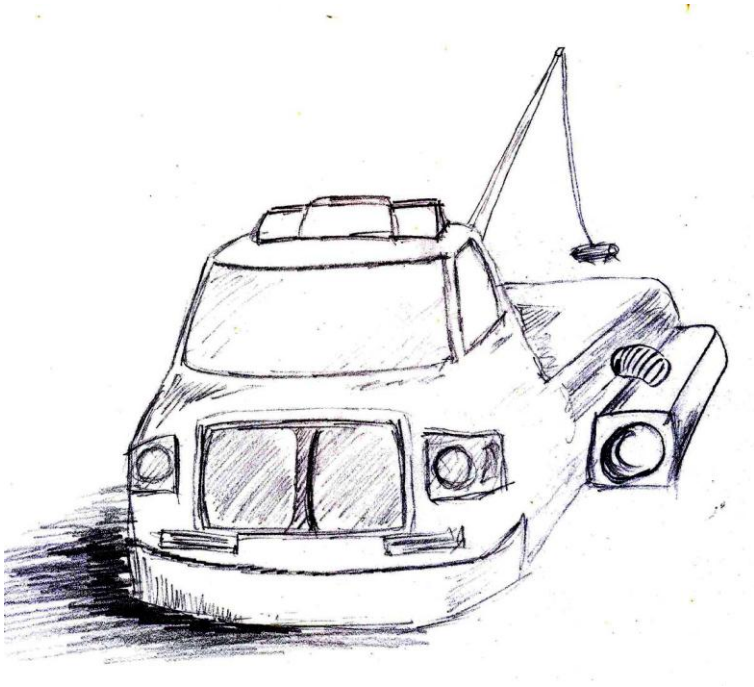


Título: Vehículo Carga – vista Frontal
Elaboración: Juan Carlos Reascos Laverdi.

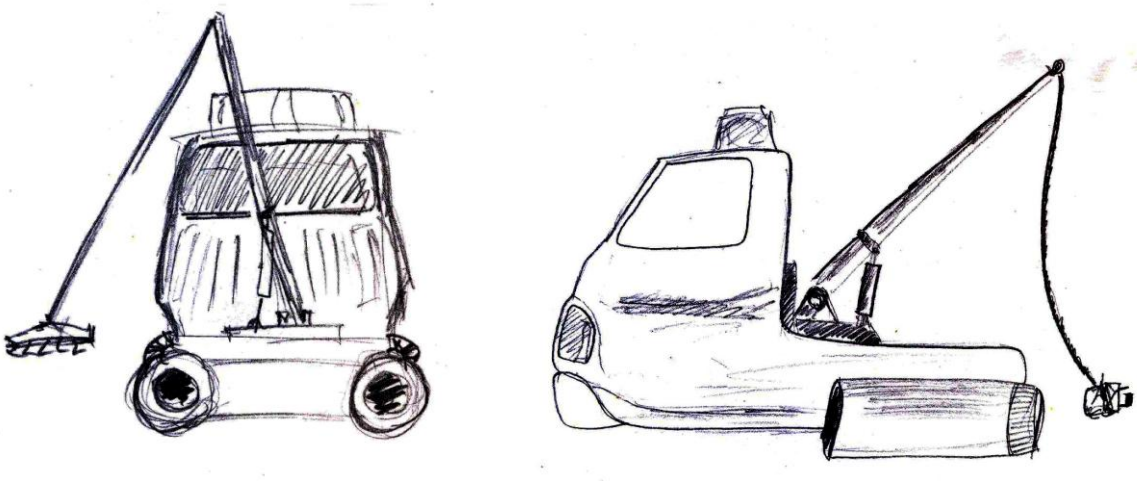


Título: Vehículo Carga – vista Lateral
Elaboración: Juan Carlos Reascos Laverdi.

3.1.3.5 Nave Wincha:

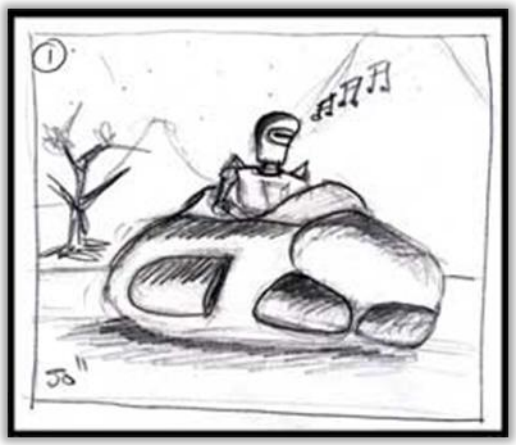


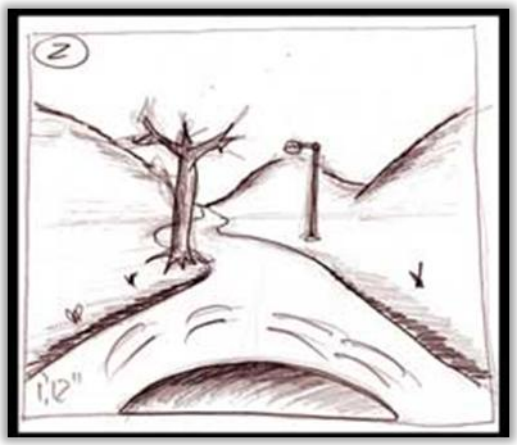
Título: Vehículo Wincha – vista perspectiva
Elaboración: Juan Carlos Reascos Laverdi.



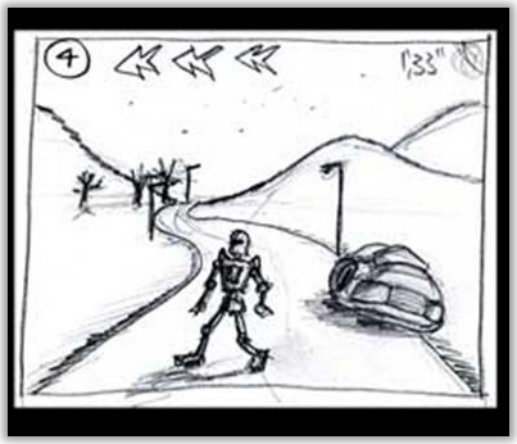
Título: Vehículo wincha – vista Posterior y Lateral
Elaboración: Juan Carlos Reascos Laverdi.

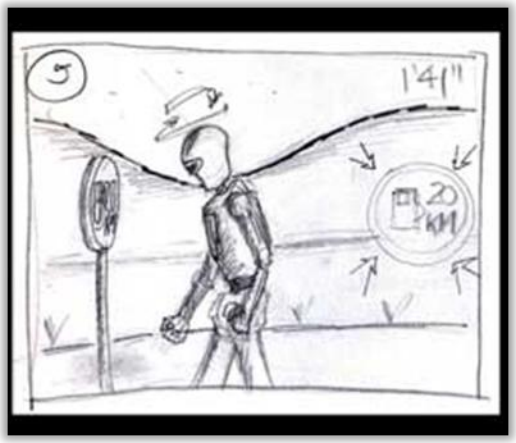
3.1.4 STORYBOARD

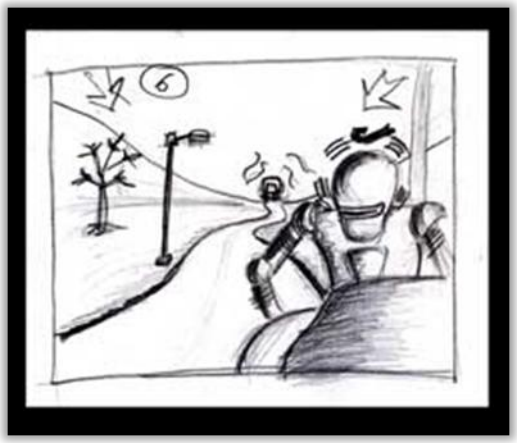


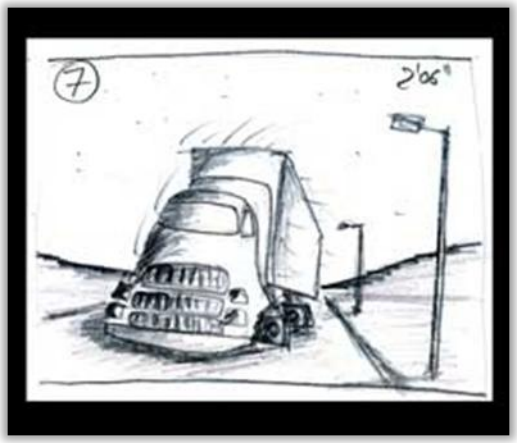


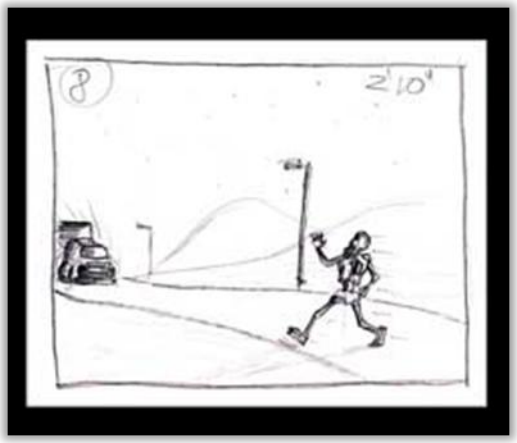




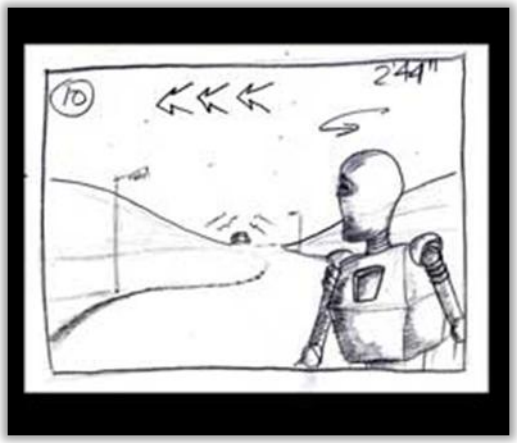

5
1'41"
20 KM

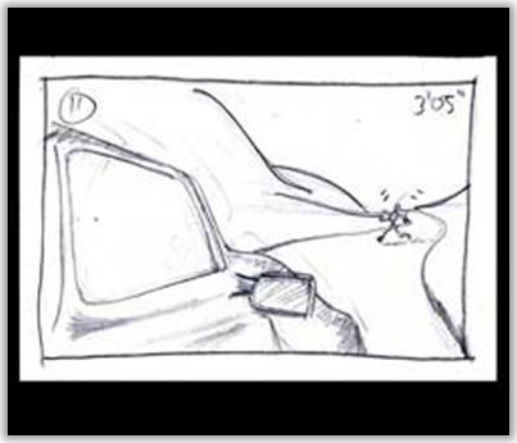

6

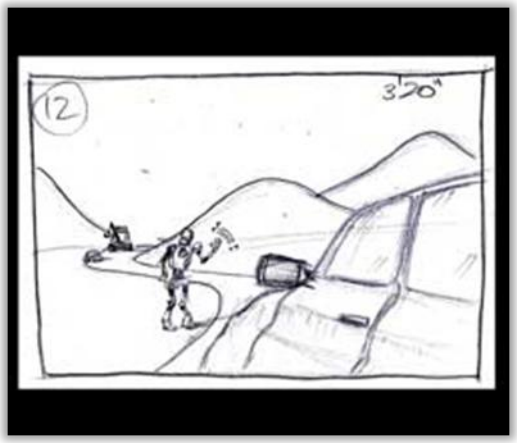

7
2'06"

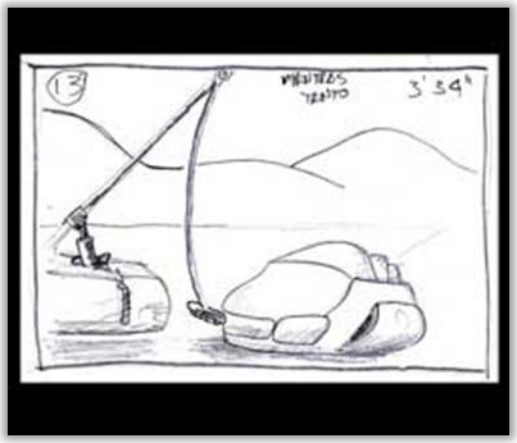

8
2'10"

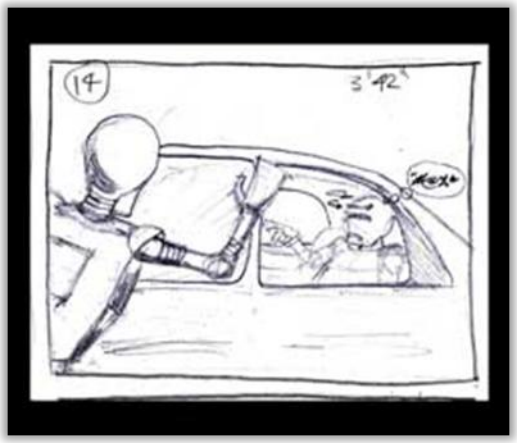

9
2'31"

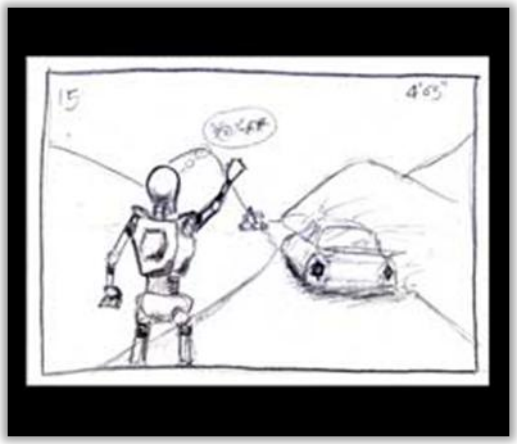

10
2'44"

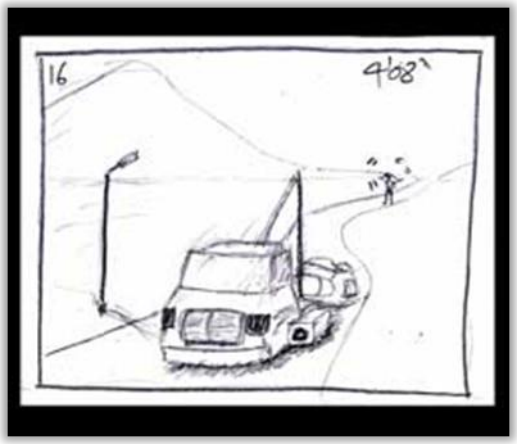

11
3'05"

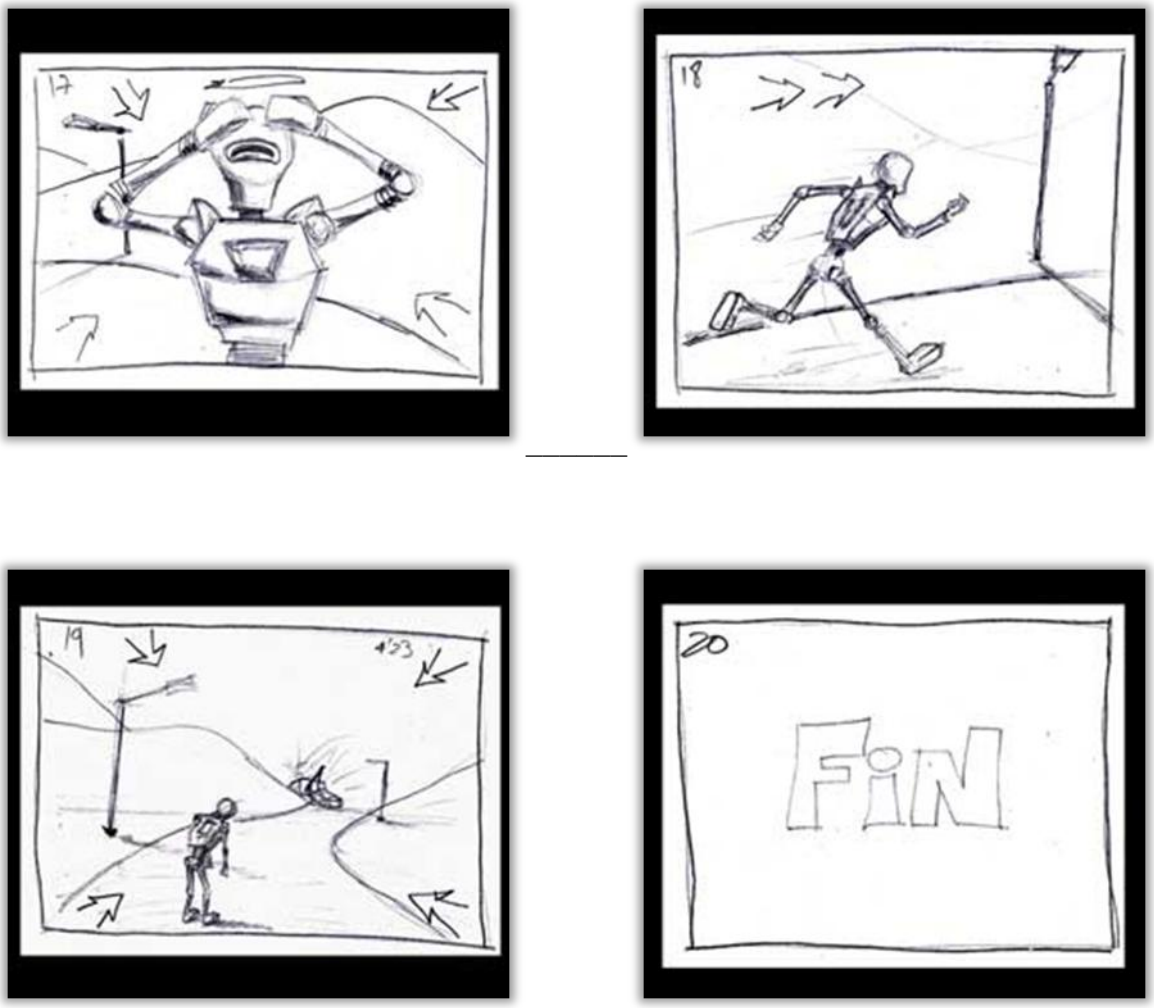

12
3'20"


13
3'34"


14
3'42"


15
4'65"


16
4'68"



Título: Storyboard Cortometraje Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3.1.5 ANIMATIC

Una vez desarrollado el story board, el cortometraje está casi estructurado y fué el momento de tener un acercamiento al timing de cada escena, el encuadre y movimientos de cámara, estos conceptos netamente cinematográficos y de animación. Para tener esta información nos valemos de un ANIMATIC o TIRA LEICA.

De esta forma se obtuvo una visión general de las acciones de los personajes, escena por escena, las visualizaciones de cada una de estas ha sido realizada en bajo

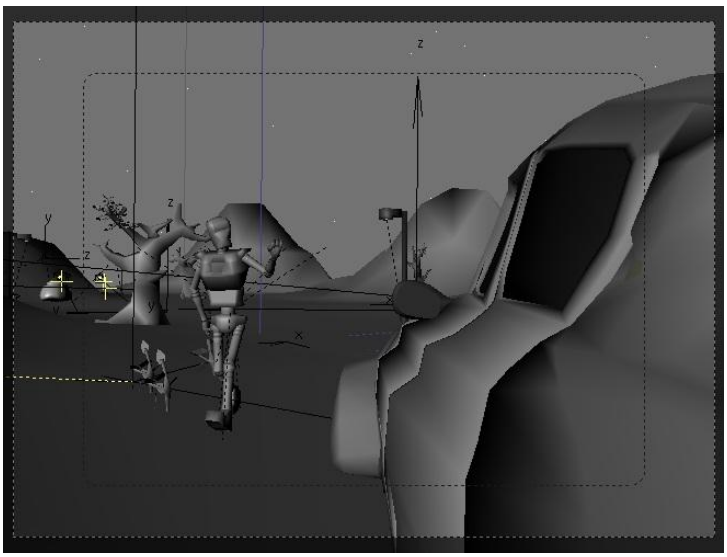
polígono "simple", sin animación, sin texturas como se trata de una producción silente en esta no se ha incluido audio.

La reproducción de este esbozo del cortometraje con los tiempos y encuadres de cada escena, ha sido de mucha ayuda ya que de esta forma se pudo dar forma y ritmo al cortometraje, además de brindar la posibilidad de encontrar fallos en el tiempo empleado en varias escenas.

Leroy tuvo 11 escenas las mismas que fueron realizadas en bajo polígono y basados en el guión y el storyboard.

Tanto los modelos, los objetos, la utilería y los fondos están realizados en bajo polígono, como no se ha modelado al detalle y según la descripción de los models sheets[35], ningún objeto ni personaje esta con un nivel de detalle perfeccionado.

A continuación unos ejemplos de varias de las escenas que se realizaron para el animatic:

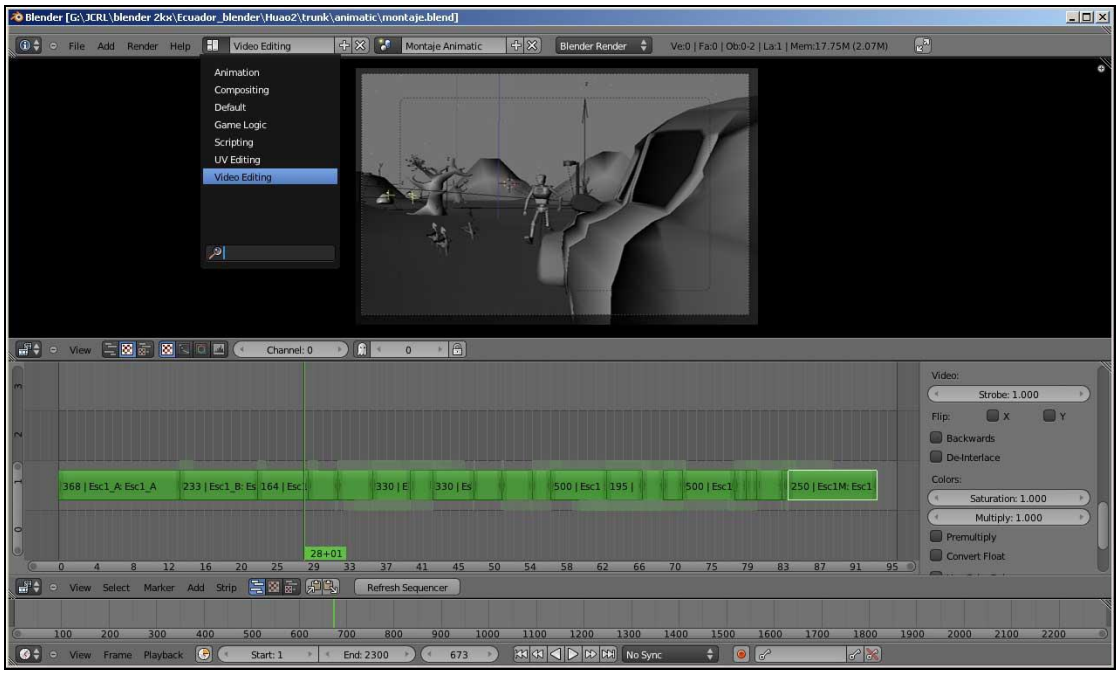


[35] Estos son precisamente un grupo de ilustraciones que muestran todas las posibles expresiones que puede tener un personaje como también las poses que pueden adoptar. Estas son creadas con el fin de mantener de forma precisa los detalles de cada uno de los personajes y para mantener los diseños de los personajes uniforme, mientras que diferentes animadores pueden trabajar en ellos sin perder detalle alguno y poder mantener una uniformidad de estos entre escena y escena.

Título: Animatic Cortometraje Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Este es un ejemplo un frame en la escena 11, en donde se puede apreciar que los elementos que forman parte de la escena están realizados en bajo polígono sin la aplicación de texturas como de un algún estudio de iluminación. Solamente se ha realizado una animación general para visualizar las acciones de los personajes por escena.

Al momento de tener el animatic de todas las escenas, se hizo uso a una de las aplicaciones de Blender como editor de video (video editing), en esta se enlazó cada escena a un solo archivo que reproduce en un solo montaje, de esta forma podemos visualizar el proyecto en su totalidad y así de esta manera poder corregir cualquier falla.



Título: Animatic Cortometraje Leroy – Video Editing.
Elaboración: Juan Carlos Reascos Laverdi.

Una vez finalizado el Animatic donde tenemos un trabajo final de creación de guiones, desarrollo de personajes como de ambientes y también de un animatic con las acciones de los mismos ya aprobados, hemos dado fin a la Pre – Producción.

3.2 PRODUCCION:

En esta fase del desarrollo del cortometraje se da inicio al uso de una metodología para la construcción del mismo y el software que he descrito en los capítulos anteriores como es Blender 3D, ha sido de gran ayuda, ya que en base a esta herramienta, a lo aprendido a través de comunidades y con los proyectos que se han desarrollado en torno a este programa, este se ha podido construir con la metodología empleada y desarrollada para este tipo de proyectos.

3.2.1 ADMINISTRACION DE ARCHIVOS

Para la realización de este proyecto es necesario tener una buena administración de los archivos ya que una buena disposición de éstos hará que su utilización en las distintas fases de la producción y post-producción.

La carpeta contendora del proyecto se llamará LEROY, dentro de ésta tenemos los siguiente archivos.



Título: Administración Carpetas Proyecto Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En el gráfico se muestran las diferentes carpetas que se usan para tener una buena administración del proyecto.

A continuación una descripción de cada una de las carpetas.

Animatic: En esta carpeta se guardara los archivos que pertenecen al animatic de Leroy, es decir todas las escenas que fueron necesarias, para después por medio del compositor de Blender poder jalar cada una de ellas y de esta manera tener un montaje del animatic y tener un aproximado del manejo de los tiempos (timing) del cortometraje.

Bitacora: en este folder se puede almacenar los distintos archivos que formarían parte de la descripción o un detalle de lo que consideramos importante para la construcción del cortometraje, pueden ser textos, imágenes o videos.

Bocetos_personajes: en esta carpeta es muy importante en la fase de preproducción, ya que en esta almacenamos los bocetos como también las referencias para la creación de los diferentes personajes del cortometraje, pueden ser imágenes o videos, en el caso de leroy se realizaron varios intentos a partir del guión para la construcción del personaje, de la misma manera también hay referencias para lo que son los props y utilería.

Chars: La carpeta chars contiene los modelos tridimensionales ya listos de los personajes del cortometraje.

Envs: De la misma manera como en la carpeta chars en ésta encontramos los modelos tridimensionales digitales ya terminados de todos los elementos de utilería, que conforman parte de la locación y sirven de utilería con fines de composición.

Guión: El guión con todos los cambios posibles y hasta tener su versión final están guardados esta carpeta .

Info: Todo lo relacionado con la parte externa al cortometraje y que es necesario para la difusión del mismo, asi como instantáneas del personaje en varias poses para efectos de publicidad, o la creación de banners para el web, etc son alojadas en este folder.

Mattes: En base a las referencias de los ambientes de las locaciones que fueron seleccionadas, en esta carpeta se almacenan los landscapes que son las imágenes que van de fondo o background, ya que no todos los elementos son creados en 3d digital, por ejemplo pueden ser un bosque o el océano.

Pre_personajes: Todos los personajes del cortometraje que son animados deben ir riggeados, es decir con la estructura interna de huesos y todos ellos van a ser agrupados en este folder.

Props: Así como tenemos la carpeta Chars en donde tenemos todos los personajes modelados en 3Ddigital, se tiene esta carpeta en donde tenemos modelados en 3d digital toda los props necesarios para el cortometraje.

Scenes: En base al animatic y al montaje del mismo, en esta carpeta se almacenan Las escenas que son entregadas a cada animador que participa en el cortometraje y con la base del animatic se da paso al proceso de animación de cada un de ellas, para lograr esto el animador puede "jalar" los archivos necesarios de las carpetas que considere importantes para realizar la animación de la misma, mediante varios procesos propios de blender, de esta forma los cambios que se realicen en un personaje serán reflejados a todos los animadores que van a animar el mismo personaje en las escenas de cortometraje.

Sets: Como se lo explico en la sección de Sets, de la misma manera en la carpeta Sets el director ya ha realizado la composición para las escenas del cortometraje y en esta de igual forma se procedió a jalar los archivos de la carpeta Envs y props.

Además hay otras carpeta donde se alojarán los archivos que se generen de la realización de los renders de cada escena y también del montaje final en la carpeta Postproducción.

3.2.1.1 SVN Subversion

Para la administración de las carpeta y los archivos, se utilizó el sistema de control de versiones Subversion que es un sistema para el control de versiones libre y de código fuente abierto.

Es decir, Subversion SVN, este maneja archivos y directorios a través del tiempo. Hay un árbol de archivos en un repositorio central. El repositorio es como un servidor de archivos ordinario, excepto porque recuerda todos los cambios hechos a sus archivos y directorios. Esto permitió recuperar versiones antiguas de los archivos en varias

ocasiones cuando fue necesario volver a una versión anterior de un archivo en especial de la animación de una escena.

http://www.svnrepository.com/

El repositorio se alojó en un hosting dedicado en donde se creo un Trac llamado Leroy, este Trac contendría todas las carpetas que fueron mencionadas anteriormente, una vez que se tenía este Trac configurado, se asignó los usuarios que podían acceder a este Trac y poder guardar, modificar, editar o eliminar cualquiera de los archivos.

Los usuarios que se crearon fueron 3 ya que en este proyecto se necesito la ayuda de un ilustrador y de una persona encargada de realizar el audio para este proyecto.

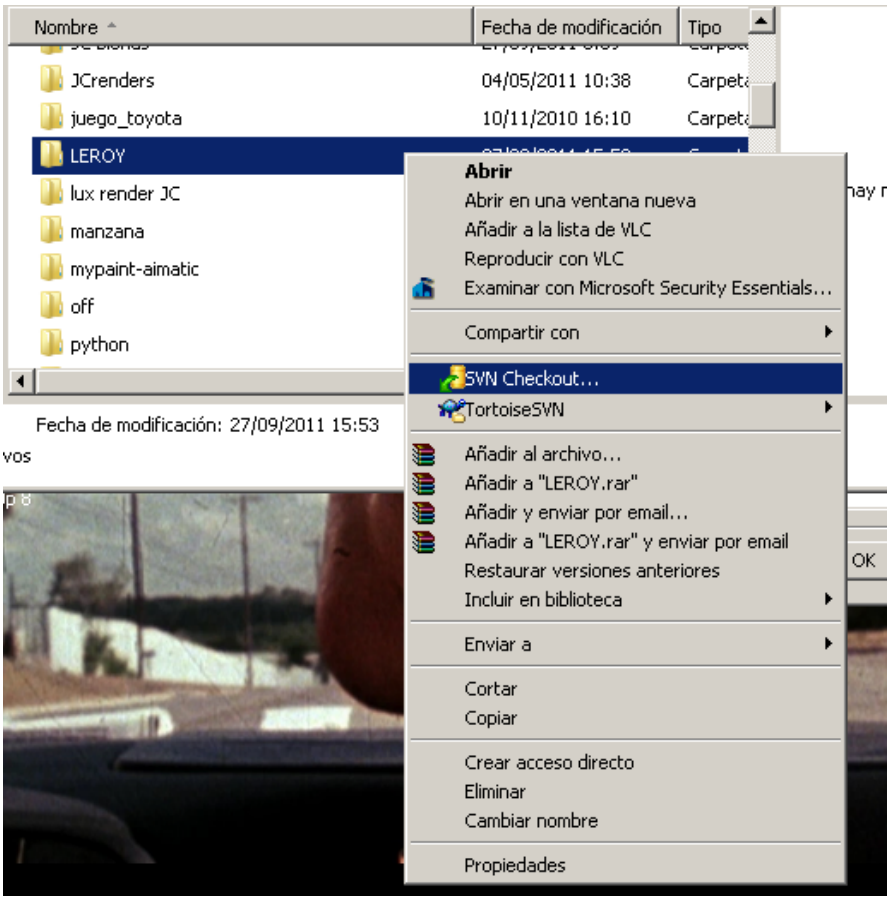
3.2.1.2 Cliente Subversion

El cliente de Subversion utilizado en este proyecto es Tortoise SVN que es un cliente gratuito de código abierto para el sistema de control de versiones Subversion.

Este maneja archivos y directorios a lo largo del tiempo. Los ficheros se almacenan en un repositorio central, que es el Trac creado en el hosting dedicado a SVN subversion.

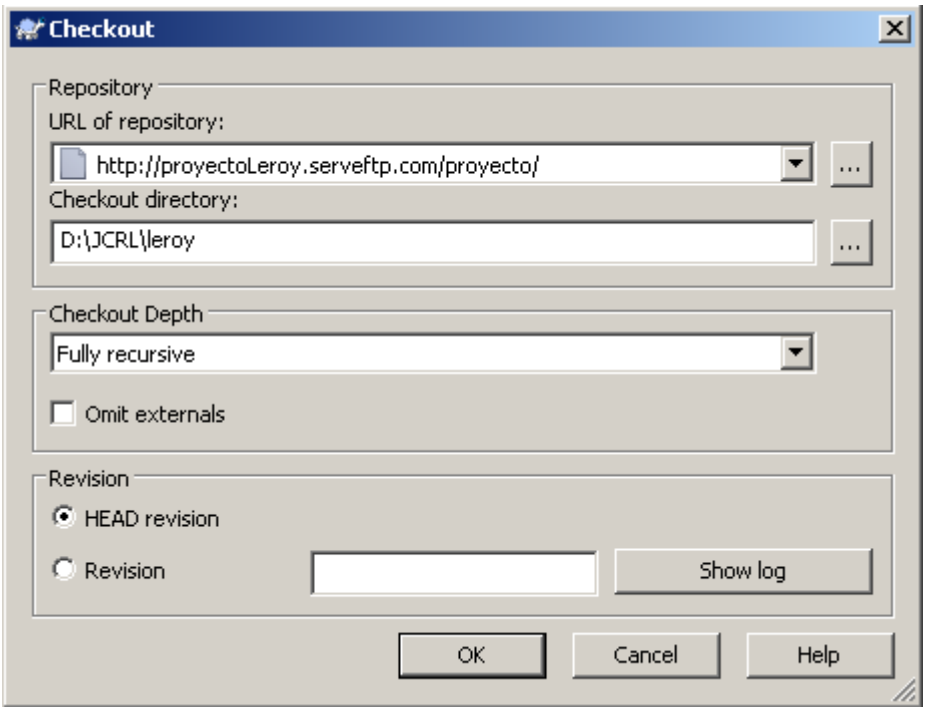
El repositorio es prácticamente lo mismo que un servidor de ficheros ordinario, salvo que recuerda todos los cambios que se hayan hecho a sus ficheros y directorios. Esto permite que un cliente de SVN pueda recuperar versiones antiguas de sus ficheros y examinar la historia de cuándo y cómo cambiaron sus datos, y quién hizo el cambio.

La manera de instalar el cliente de SVN subversión client en una estación de un usuario es a través de de la ejecución de un instalador binario.



Título: Cliente de SVN subversión client - Cortometraje Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

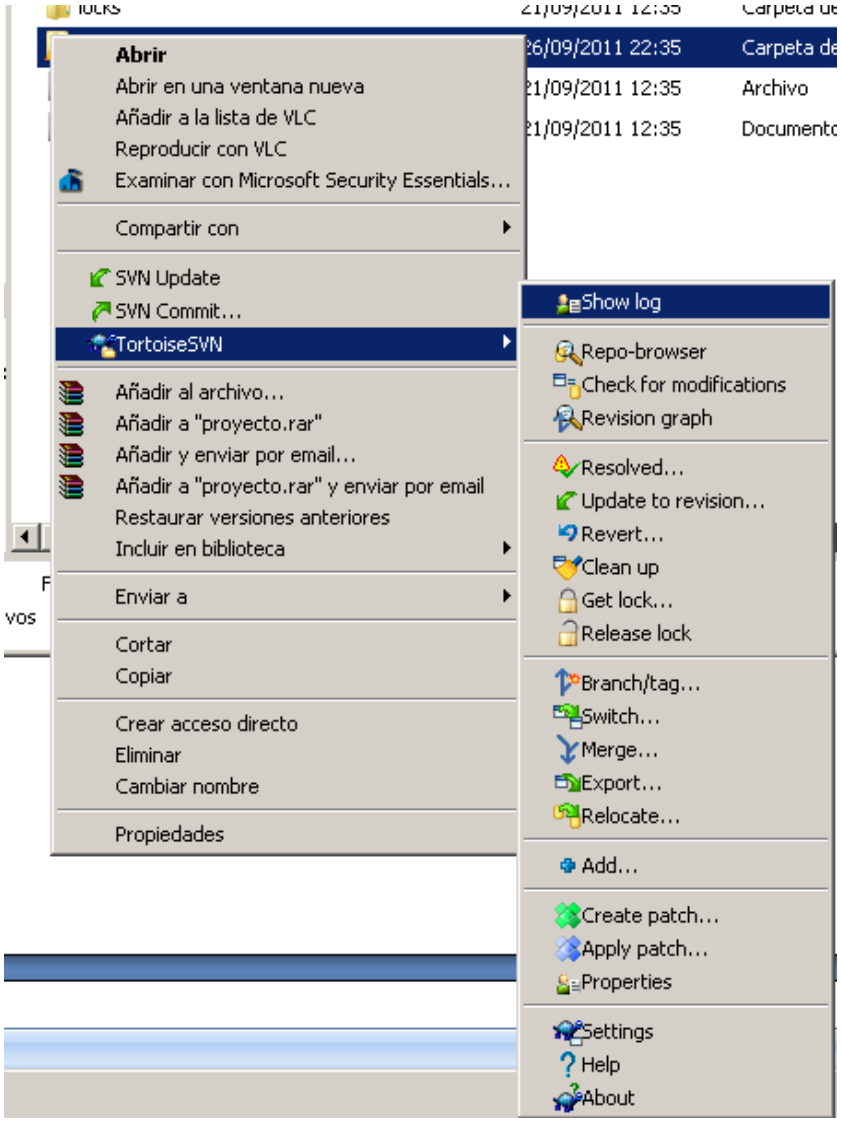
Después en un directorio especifico del proyecto como en la imagen al archivo LEROY accionamos el menú contextual.



Título: cliente de SVN subversión client - Cortometraje Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En el gráfico se muestra la ventana de configuración en donde se ingresa la URL del hosting dedicado a SVN subversion , que es donde esta el repositorio del proyecto, el URL del proyecto es http://proyectoLeroy.serveftp.com/, además de eso tenemos el directorio Local que es la carpeta que contendrá los archivos que se sincronizen con el repositorio en este caso está alojado en D:\JCRL\LEROY

Se tiene diferentes utilidades para el manejo del istema de versiones SVN, el cliente de SVN llamado tortoise, tiene diferentes opciones tanto para mantener actualizado y sincronizado la carpeta local con el repositorio del hosting SVN como son: Update para actualizar la carpeta, Commited para subir un nuevo archivo o actualizar uno ya existente con nuevos cambios.



Título: cliente de SVN subversión client - Cortometraje Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Además de las opciones Update y Commit se tiene mas opciones para poder resolver conflictos en las diferentes versiones que se hayan creado.

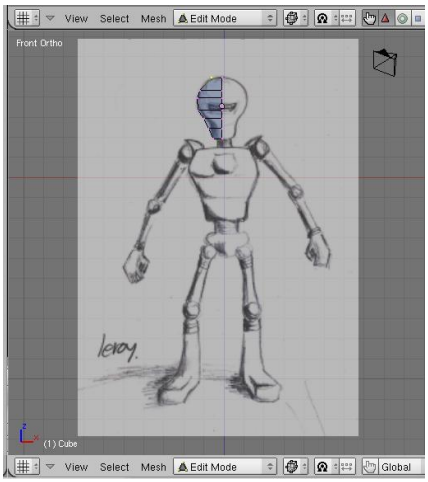
De esta manera se logro realizar un flujo de trabajo constante sin problemas en pérdida de información gracias a este sistema de manejo de versiones.

A continuación seguimos con el proceso de pre_producción del cortometraje

3.2.2 MODELADO

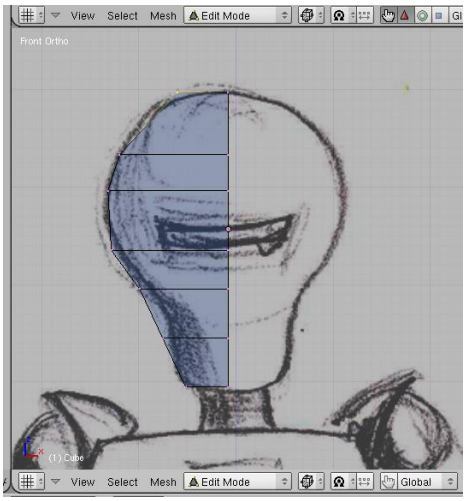
La fase de modelado 3D incluye, el modelado de los personajes, la utilería, los props, los ambientes en caso de ser necesario, el trabajo de modelado en tres dimensiones, en el cortometraje Leroy básicamente es un trabajo de rotoscopia, en donde en base a diseño de personajes estos pasan de estar en papel a digital, cada uno de ellos tiene el mismo tratamiento.

Haciendo uso de la herramienta Blender 3D, he desarrollado el modelado de los personajes utilizando la técnica de box modelling, en donde a base de figuras geométricas primitivas como son cubos, esferas, etc, se va obteniendo objetos cada vez más complejos a través de cortes, extrusiones y manejo de vértices, lados y caras.



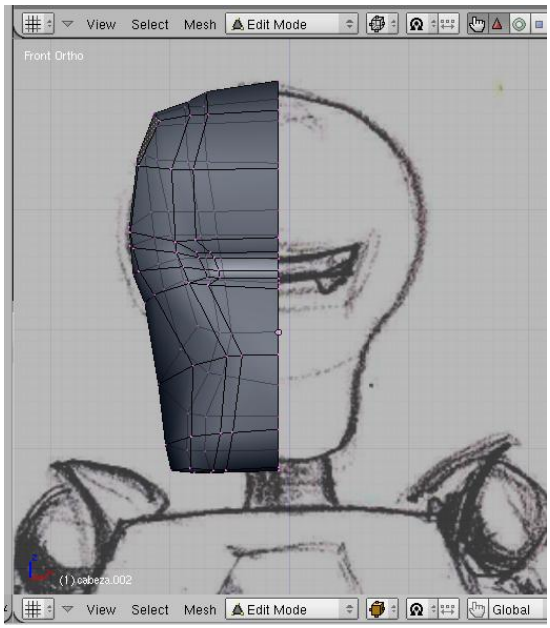
Título: Modelado con Rotoscopia - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Se empieza a modelar cada personaje en base al diseño del mismo , he empezado por la cabeza de Leroy



Título: Modelado con Rotoscopia - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

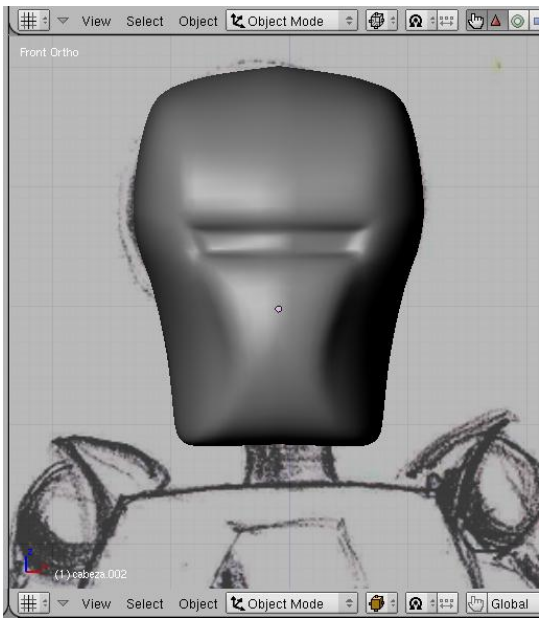
Agregando un simple cubo, he ido aproximando a la figura del personaje, haciendo uso de cortes longuitudinales en la malla ademas de varias extrusiones en el area de los ojos y haciendo unicamente la mitad del rostro para despues obtener lo siguiente:



Título: Modelado con Rotoscopia - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

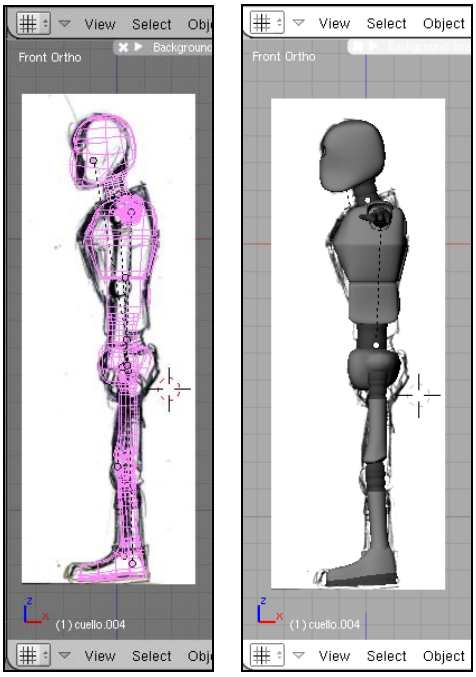
Se puede corregir o realizar variaciones al momento de realizar el modelado de un personaje, claro sin cambiar drasticamente el diseño del mismo, ahora finalmente lo que se debe realizar es un suavizado de la malla para efectos de poder acercarse mas al diseño del robot que es de una forma orgánico, para esto se hace uso de los modificadores de mallas y en este caso el modificador Subsurf. El motivo de haber creado únicamente la mitad del rostro del personaje es que es mas facil hacerlo asi y aplicar un modificador de malla llamado MIrror (espejo), de esta manera obtener el modelado de la cabeza completa, asi podemos modelar cuerpos organicos mas facilmente.

El resultado del modelado de la cabeza de Leroy sería la siguiente:

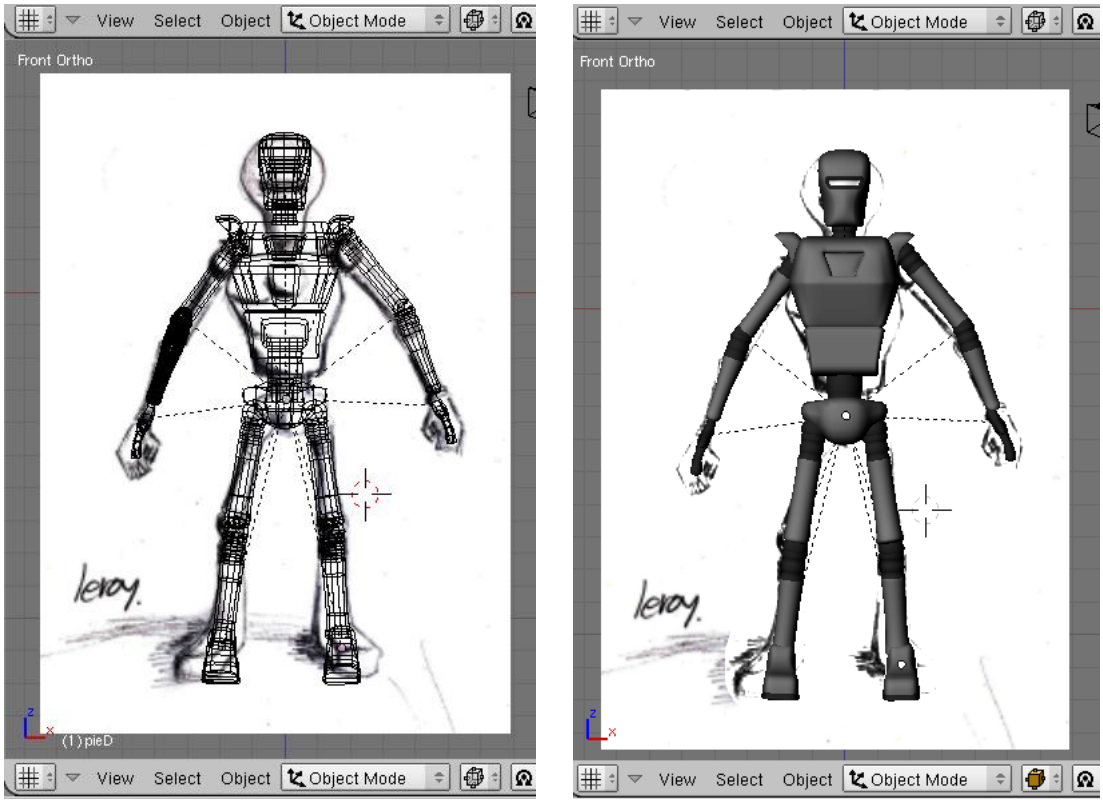


Título: Modelado con Rotoscopia - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Para realizar el modelado completo del personaje, es necesario tener varias vistas del personaje, tanto la vista frontal como la lateral.



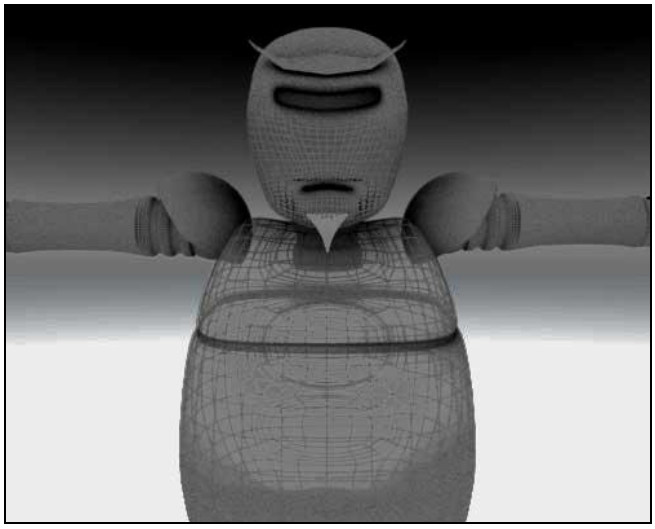
Título: Modelado con Rotoscopia (lateral)- Leroy.
Elaboración: Juan Carlos Reascos Laverdi.



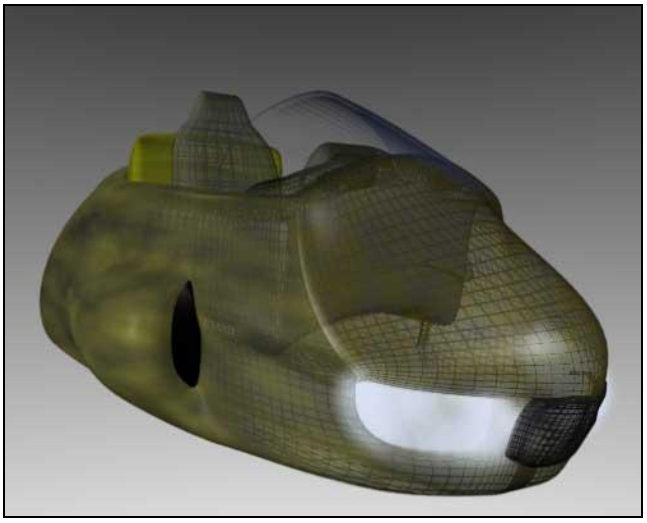
Título: Modelado con Rotoscopia - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En modelado de objetos tridimensionales existen varias reglas importantes que son imprescindibles para cuando estas pasan a otras etapas principalmente la de animación , por ejemplo el uso excesivo de vértices en una malla podrían hacer que el momento de renderizarlo, esta se vuelva una tarea casi imposible por la capacidad de procesamiento que llevaría esta tarea y que a mas de vértices están las texturas e iluminación, etc. Otro ejemplo de construir una malla de manera equivocada es crear en esta los llamados TRIADS cuando lo óptimo en la creación de mallas para animación son CUADS, es decir grupos de cuatro vértices, lo que genera una deformación limpia de la malla al momento de hacer flexiones en esta.

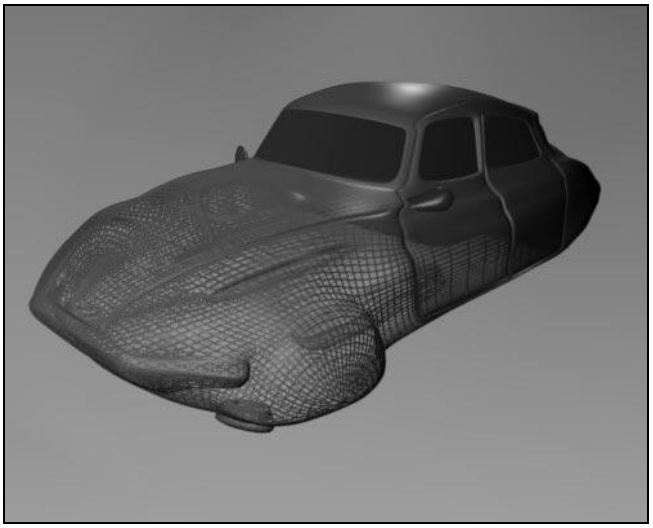
A continuación tenemos las mallas correspondientes a la geometría realizada en personajes , props y objetos de utilería modelados en este cortometraje.



Título: Modelado con Rotoscopia - Momo.
Elaboración: Juan Carlos Reascos Laverdi.



Título: Modelado con Rotoscopia - Props.
Elaboración: Juan Carlos Reascos Laverdi.

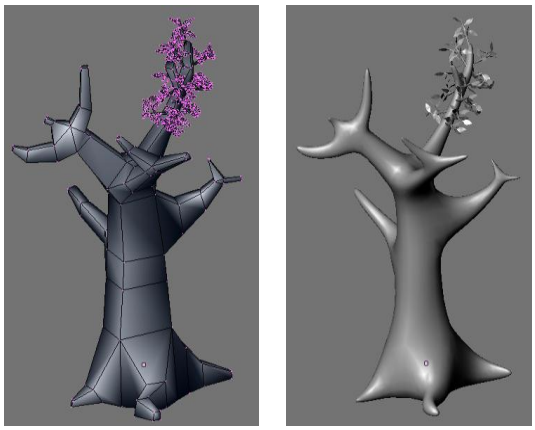


Título: Modelado con Rotoscopia - Props.
Elaboración: Juan Carlos Reascos Laverdi.

3.2.2.1 Modelado Organico (plantas)

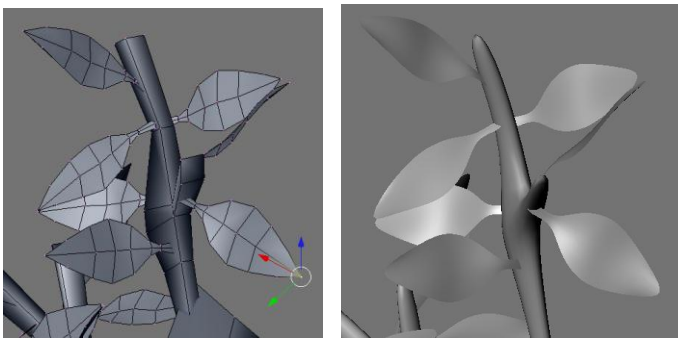
Ya que las plantas son orgánicas , al igual que en personajes humanos se requiere de un tipo de uso del modelado 3D más meticuloso ya que este tipo de modelado es imperfecto como lo es la naturaleza y los arboles como parte de ella, de igual forma la

técnica de modelado es la llamada rotoscopia donde se modela el objeto partiendo de la referencia y haciendo uso de los modificadores para suavizar las superficies. A continuación un ejemplo de cómo se modelo los objetos arboles para el cortometraje.



Título: Modelado con Rotoscopia - Props.
Elaboración: Juan Carlos Reascos Laverdi.

A partir de la figura primitiva CUBO se realizo extrusiones para llegar a la forma básica del objeto en este caso un árbol, haciendo uso de los modificadores del objeto o malla se realizó un suavizado de las superficies obteniendo el resultado esperado.



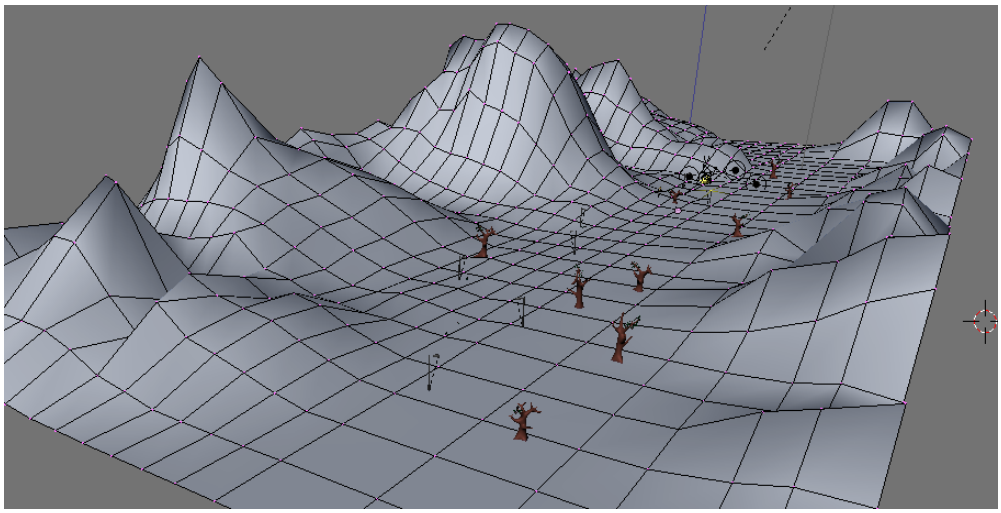
Título: Modelado con Rotoscopia - Props.
Elaboración: Juan Carlos Reascos Laverdi.

Adicionalmente, se modelo por separado objetos de la misma malla, que tienen un mayor nivel de detalle, como son las ramas y las hojas, cabe recalcar que existen algoritmos que generan varios tipos de follaje por ejemplo esta el script llamado Tree from Curves, que fue desarrollado para el cortometraje Big Buck Bunny, en el que se

ingresan los parámetros necesarios para que a través del uso de Curvas Bezier (básicas), se generen los troncos principales y sus subdivisiones o ramificaciones, además de poder setear la cantidad de hojas requeridas y finalmente su posterior uso en texturas y materiales

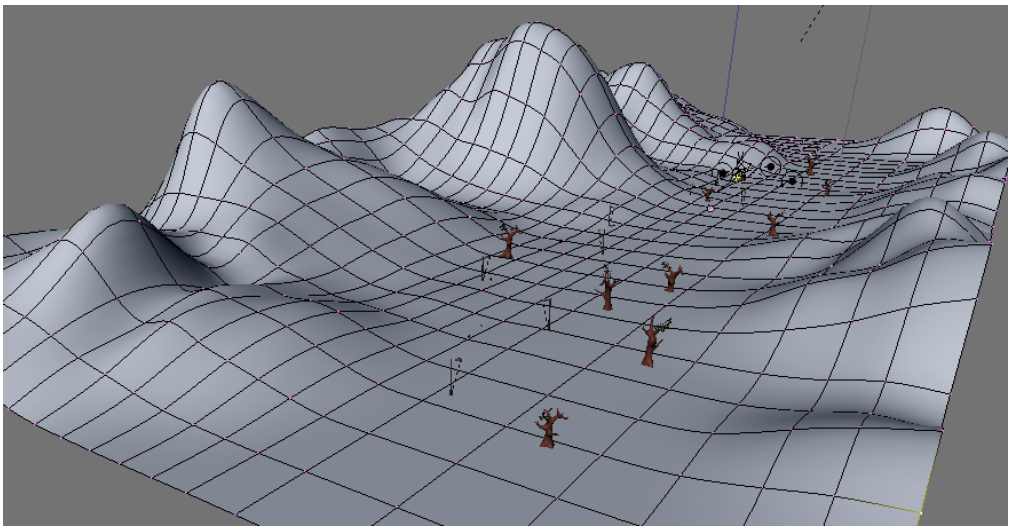
3.2.2.2 Modelado Escenario (landscapes)

En el modelado de nuestro escenario, primeramente se necesitó obtener en base a una referencia real y al storyboard el modelado 3D y de esta manera obtener en bajo polígono la geometría de lo que se necesitaba trabajar



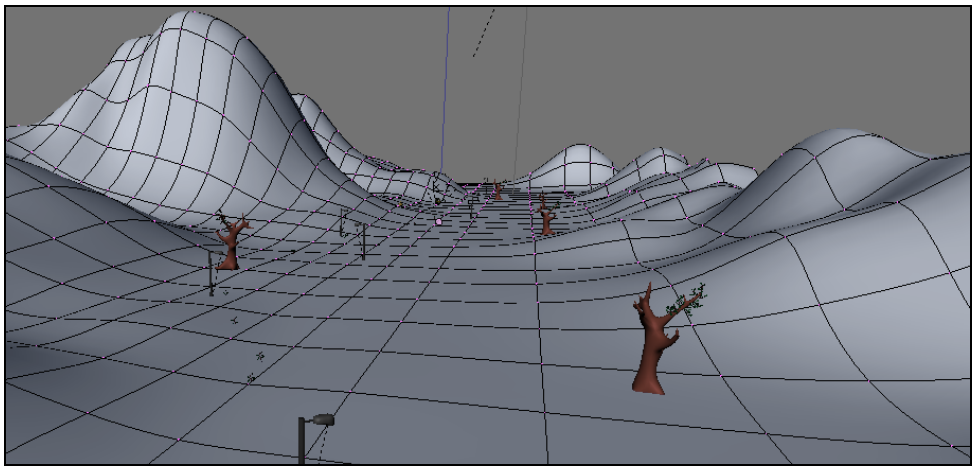
Título: Modelado - Landscape.
Elaboración: Juan Carlos Reascos Laverdi.

A partir de una figura geométrica básica como lo es un plano, se ha subdividido en más partes obteniendo de esta forma muchos vértices con los que se puede trabajar y dar forma en base a nuestra referencia de nuestro escenario o landscape, usando herramientas propias de la herramienta podemos usar un arrastre con cierta influencia de los vértices mas cercanos, para de esta forma crear los picos y formaciones montañosas y, de la misma forma obtener las planicies necesarias para el diseño de la escena.



Título: Modelado - Landscape.
Elaboración: Juan Carlos Reascos Laverdi.

Una vez que se tiene un aproximado del modelado del escenario deseado se ha hecho uso del modificador Subsurf, para dar una mayor suavidad en el modelado.



Título: Modelado - Landscape.
Elaboración: Juan Carlos Reascos Laverdi.

Esta es una vista ya en perspectiva que se usara en el cortometraje.

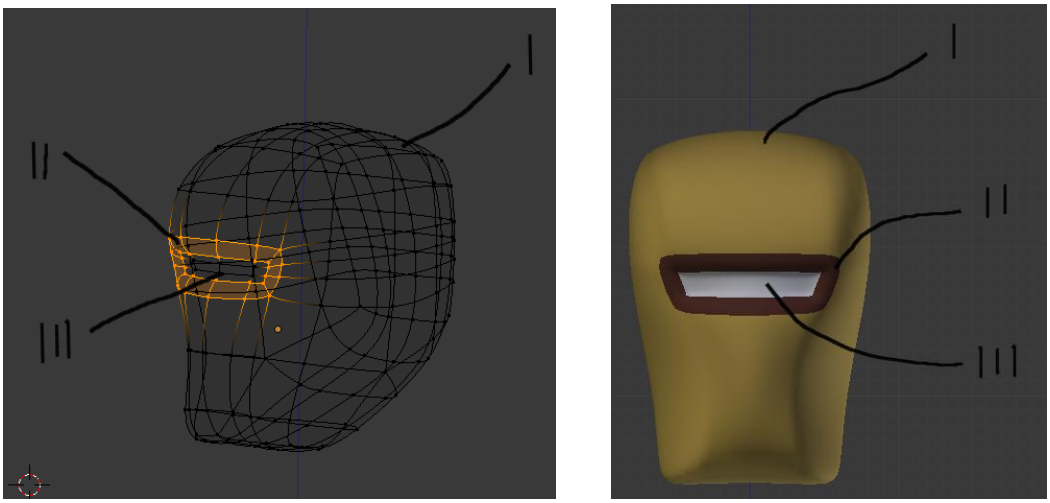
3.2.3 TEXTURAS

Esta fase del proyecto es en el cual se dió tratamiento a las diferentes mallas pertenecientes tanto a los personajes personajes como a los props , utilería y también a los objetos pertenecientes al landscape, previamente modelados y aprobados por el director.

Como parte del manejo de las texturas en Blender 3D se utilizó las llamadas texturas procedurales las que fueron usadas en los personajes del proyecto por ejemplo en el caso de Leroy

1. La geometría perteneciente a las mallas del personaje Leroy que son por ejemplo Cabeza, Tronco, abdomen, extremidades superiores y extremidades inferiores fueron modeladas por separado, ya que se trataba de un robot se necesitaba tener la apariencia y dinámica propia de un bípedo de este tipo.

   Después de tener estos objetos separados, se dio uso a las texturas procedurales propiamente primeramente como ya se tenía un boseto aprobado de la malla, lo que se realizó es un aproximamiento de los materiales de dicho boseto al objeto en digital.

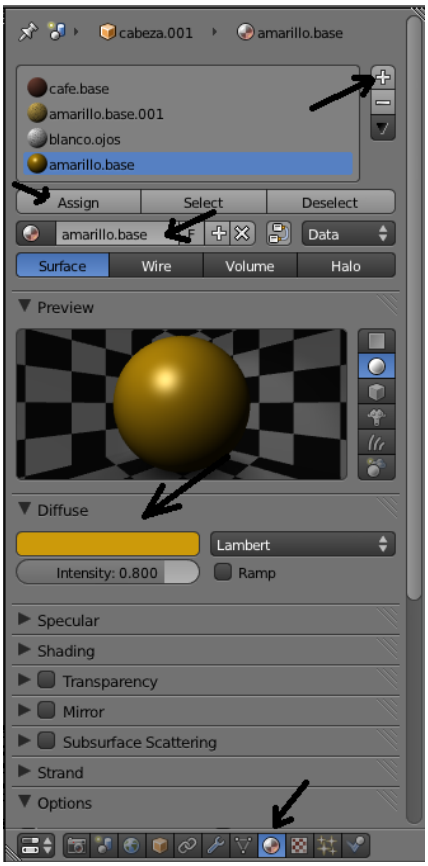


Título: asignación Texturas - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En el gráfico podemos ver que la malla de la cabeza de Leroy es una sola, y que en la misma se ha colocado tres tipos de materiales diferentes, para poder realizar esto se necesitó realizar lo siguiente:

1. En la malla seleccionar los vértices que van a formar parte de la cabeza la base Amarilla. Después en la ventana de propiedades del Objeto se seleccionó Texturas y Materiales en el cual agregamos un Material ,se le asignó un nombre "amarillo.base" , para luego asignar ese material a esos vértices presionamos el botón Assign. Una vez asignado el material a los vértices que seleccionamos podemos configurar el Color , especularidad o Brillo, Shading, Transparencia en caso de que lo tuviere, también se puede configurar la reflectancia en caso de ser un objeto como en el caso de un espejo o configurar el nivel de reflectacia según el material, además de otras opciones como Subsurf scattering que es la transparencia que tienen ciertos objetos al ser traspuestos a la luz.

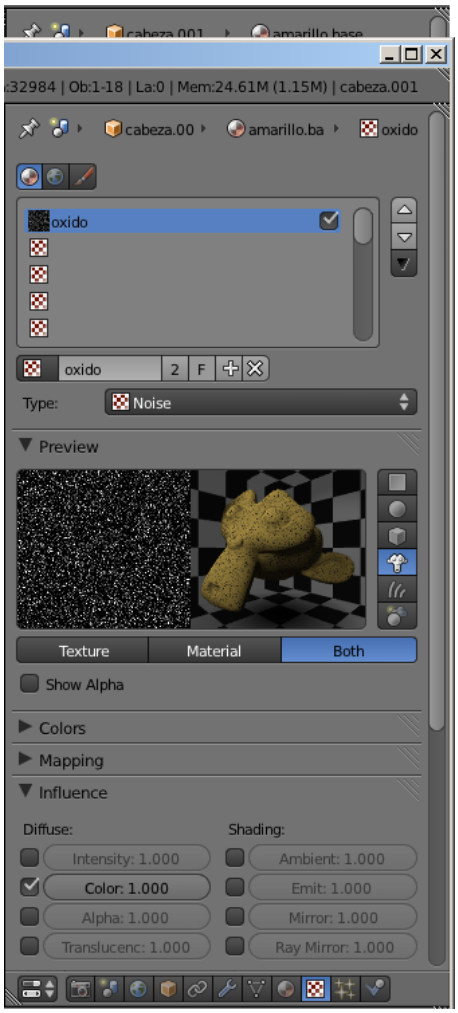
   En el ejemplo de La cabeza y el material amarillo Base el seteo es el siguiente.



Título: Configuración Texturas - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Como se aprecia en la figura en parte superior tenemos los diferentes materiales usados en la cabeza de Leroy como son: café.base, amarillo.base, blanco.ojos y un material de prueba llamado amarillo.base.001.

A cada material se le ha asignado una textura en especial, por ejemplo, al material "amarillo.base" se le agregó un tipo textura llamado Noise que puede simular un poco de aspereza en el material de nuestro personaje, se pueden agregar varias texturas a los materiales que se tengan.

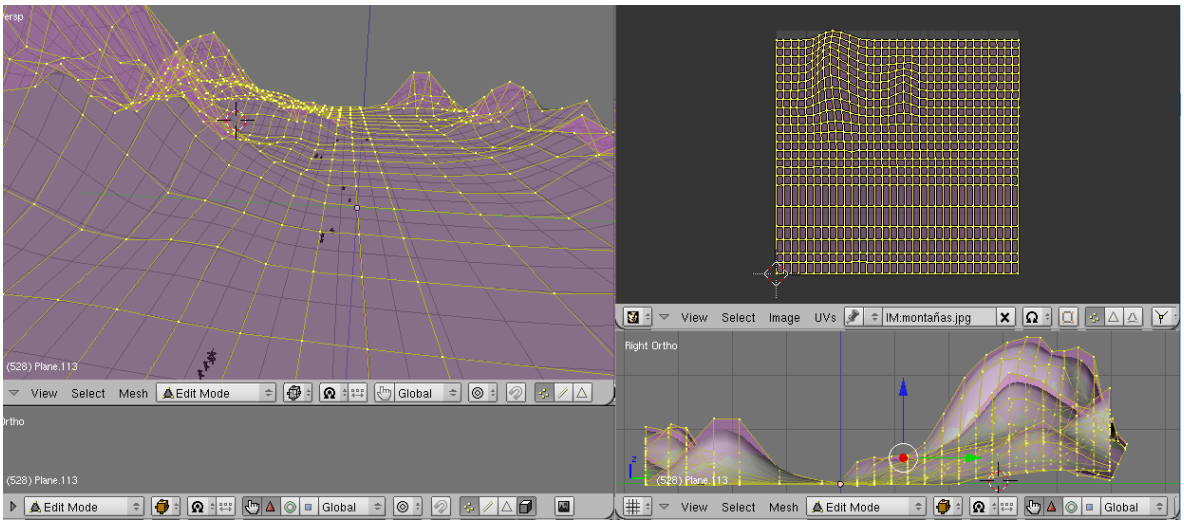


Título: Configuración Texturas - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3.2.3.1 Mapeo UV

El proceso de agregar los diferentes materiales fue el mismo para los personajes y los objetos "props", no así el escenario, ya que dar este tipo de técnica de manejo de materiales a este, puede ser un problema al momento de renderizar el proyecto final, ya que se necesitaría una gran capacidad de computo para lograrlo, por que el manejo de materiales con texturas procedurales implica correr algoritmos para la creación de los mismos.

Para lograr la aplicación de texturas mas realistas para objetos como escenarios sin que nos implique tener una granja de render, hacemos uso de la técnica llamada UV mapping, o en español mapeo UV, en donde a nuestra malla la podemos agregar una textura basada en una imagen, para Leroy se necesitó usar esta técnica en las texturas del escenario, es decir la montañas y el valle.



Título: Configuración Texturas Landscape - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

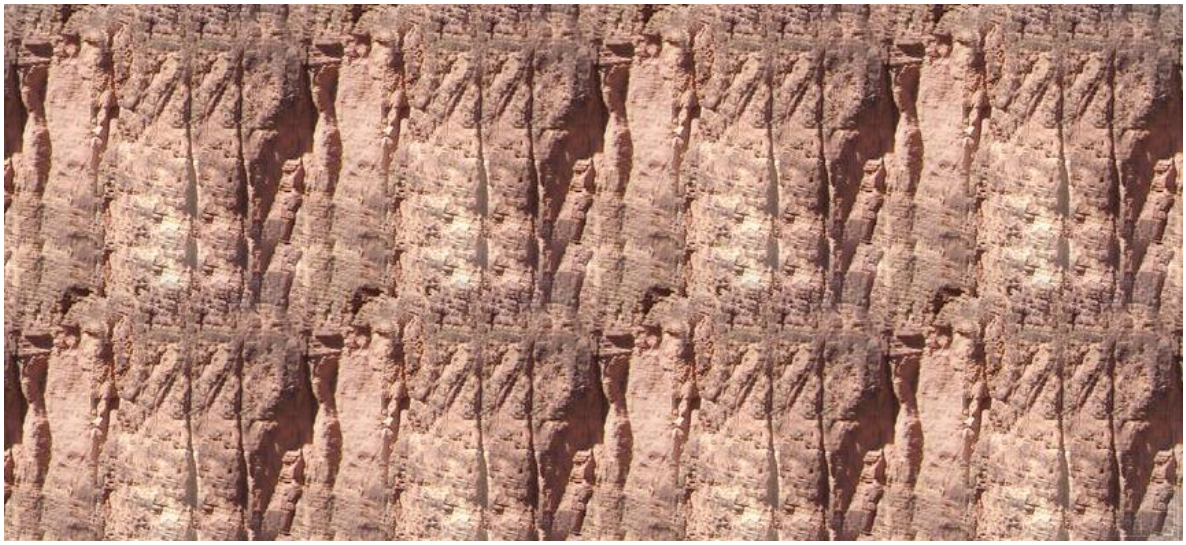
En la figura tenemos la malla que se realizo para las montañas del escenario del cortometraje en sus diferentes vistas, para poder hacer uso del mapeo UV en esta malla primeramente es haberla realizado en Bajo polígono, es decir con solamente los vértices

necesarios, para después en otra ventana en la interfaz en blender, cambiamos la opción de visulizacion de la ventana de trabajo a UV



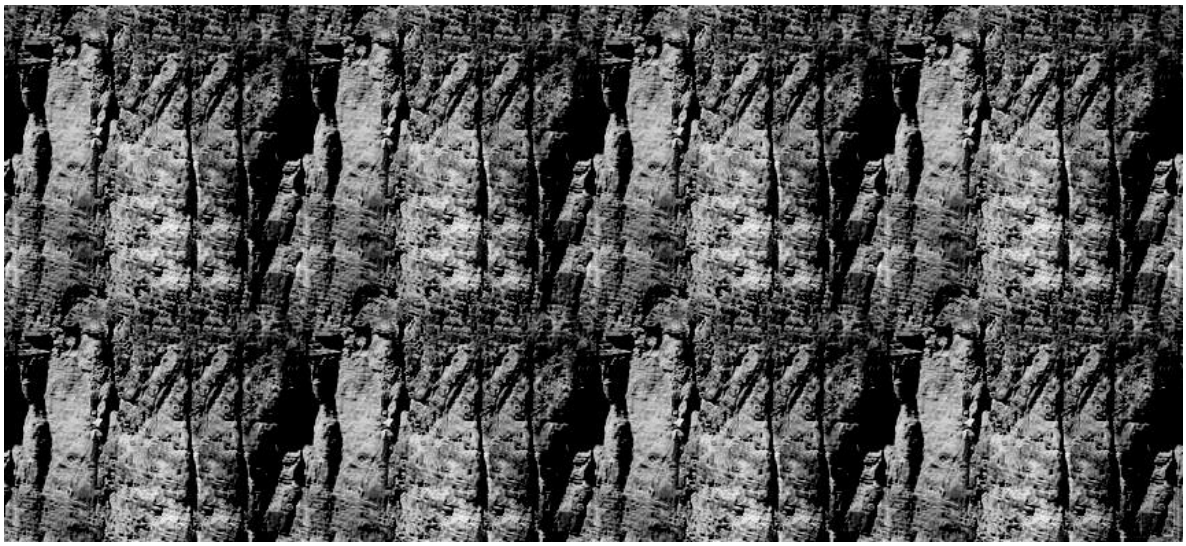
Ahora se hace uso de imágenes mapa de bits, pueden ser varias imágenes para el mismo objeto, estas deben ser realizadas en alta resolución y van a ir añadidas a la malla como un adhesivo, estas imágenes que se agregan como texturas tienen diferentes usos, en el cortometraje Leroy se hizo uso de 2 tipos de imágenes, primeramente la Diffuse y las imágenes Bump.

Las características de las imágenes Diffuse es que este tipo de imagen da las características de color, especularidad, shading entre otras, a continuación la imagen Diffuse usada para crear la textura en el objeto Montañas.



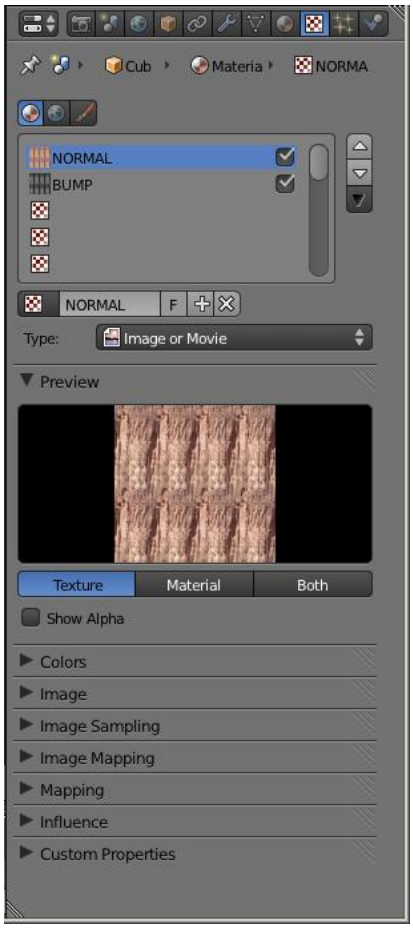
Título: Texturas UV diffuse (landscape) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

La otra clase de imágenes usada en la textura para el objeto Montañas es la llamada Bump, la que sirve para realizar los llamamos Mapas de Relieves estos simulan la impresión de una Superficie 3D, es decir, de relieve. Pero este relieve no va a proyectar ninguna sombra y no obstruirá a otros objetos. Si el ángulo de cámara es rasante en relación a la superficie, nos daremos cuenta de que la superficie no tiene relieve en realidad, a continuación el Mapa de relieves o Bump usada en Leroy.



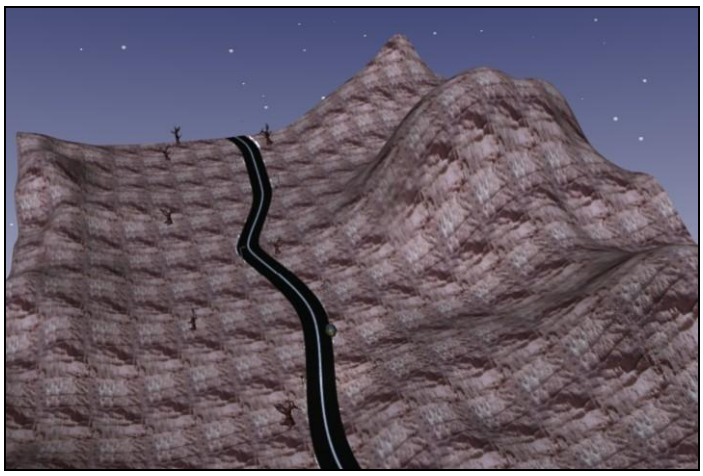
Título: Texturas UV bump (landscape) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Al usar estas 2 imágenes tenemos como resultado un objeto texturizado con detalles de color y relieves que simulan características tridimensionales para objetos tridimensionales, Se pueden agregar imágenes de 32 bits, es decir con canales alfa para trabajar con transparencia y superponer cualquier tipo de imágenes, en el ejemplo del grafico siguiente además de incluir dos imágenes la Normal y la imagen Bump, agregamos una imagen PNG de 32 bits de la carretera necesaria para el cortometraje, ya que de esta manera no fue necesario modelar en 3D dicha carretera y asi liberar de carga computacional extra.



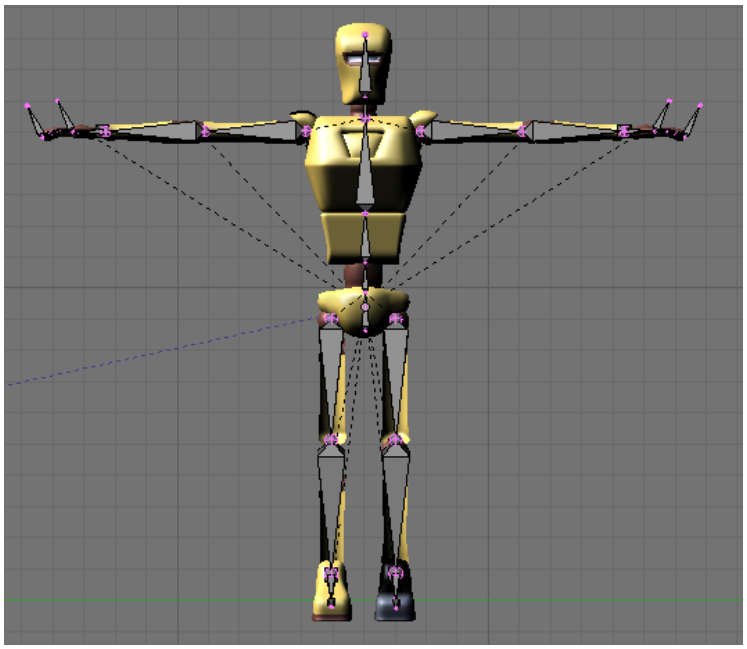
Título: Configuración Texturas UV (landscape) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En la figura esta la caja de propiedades del material del objeto Montañas, como podemos ver al material se ha agragado dos imágenes, las que corresponden al mapa Normal y al Bump.



Título: Textura UV (landscape) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3.2.4 RIGGING

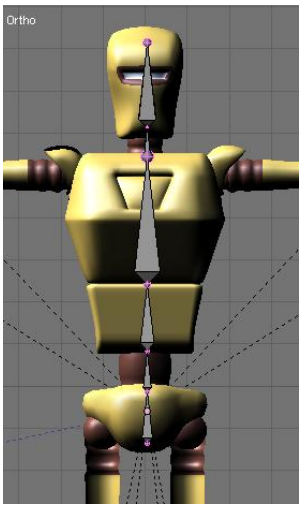


Título: Rigg - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

El riggeado es el proceso por el cual los modelos tridimensionales, digitales que se han realizado, tanto de los personajes, los objetos de utilería, etc, adquieren una estructura ósea por así decirlo, mediante la cual se puede llegar a tener movimientos con

articulaciones similares a las reales, de esta forma las acciones que realice el personaje son mas creíbles y el trabajo de los animadores se facilita.

El proceso de Riggeado de los personajes bípedos del cortometraje fue el mismo, la construcción de la estructura del sistema de huesos puede ser usado en varios modelos de las mismas características.

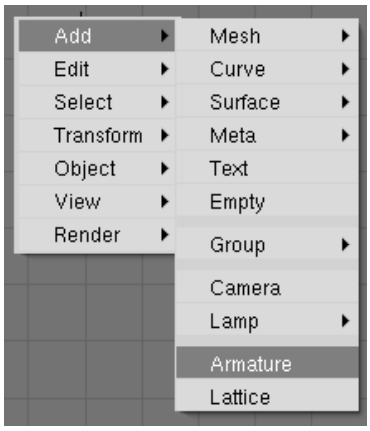


A continuación el proceso de riggeado del personaje Leroy.

Primeramente la malla del personaje tuvo que ser construida para que los diferentes segmentos de huesos queden alineados con los principales loops de vértices, de esta manera no tendremos malformaciones de la misma.

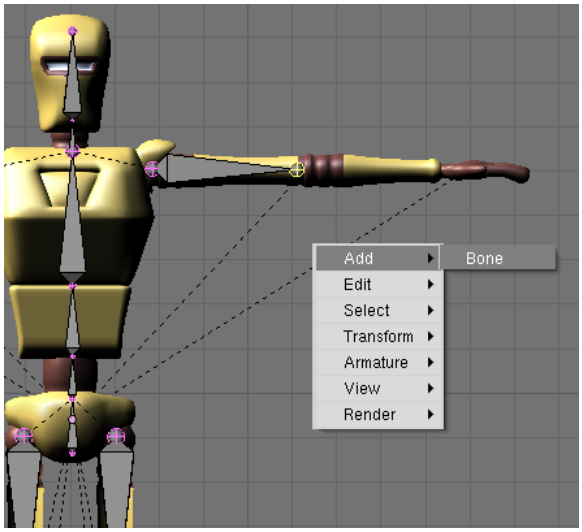
Como se aprecia en la figura se ha creado el principal sistema de huesos del personaje, cada uno de los huesos que pertenecen a la columna principal del mismo, están colocado en las articulaciones para que se genere los movimientos mas fluidos de esta manera no se presentan mal deformaciones en la malla.

Título: Rigg Columna - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.



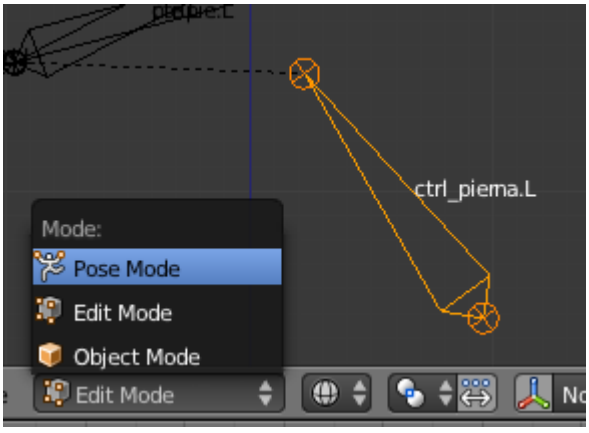
Título: Add Rigg (armadura) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Para crear una estructura con un sistema de huesos "bones", primeramente es necesario tener la malla con en el paso del personaje "Leroy", sobre la malla se agrega un nuevo objeto llamado Armature o Armadura en español, este es el sistema de huesos propiamente dicho, asi fue creado la columna principal del personaje, después se fueron añadiendo huesos.



Título: Add Bone (hueso) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

De la misma manera que existen diferentes modos para trabajar con mallas es decir "modo objeto" y "modo edición". Los huesos también tienen diferentes modos de trabajo como Modo edición, Modo Objeto y Modo Pose.



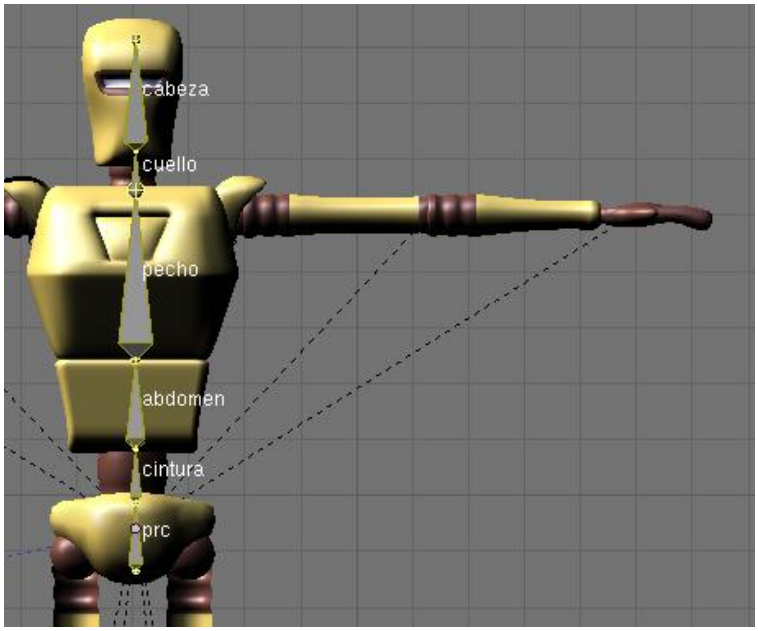
Título: Rigg (modos pose y objeto) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En modo Objeto el sistema de huesos o armadura se trabaja como una sola entidad y se puede trabajar entre varias armaduras no relacionadas entre sí, en modo Edición se trabaja dentro de la armadura, ya sea para aumentar o extruir nuevos huesos o quitarlos y en Modo Pose se realizan las diferentes operaciones con Constraints y se puede animar el personaje.

Después de ser creado la columna principal del personaje, se crean más huesos como en el caso de las extremidades, como estamos dentro de la armadura previamente creada para el personaje, lo que se realiza es únicamente el agregar mas huesos a la misma armadura, ya que pueden haber varias armaduras en escena.

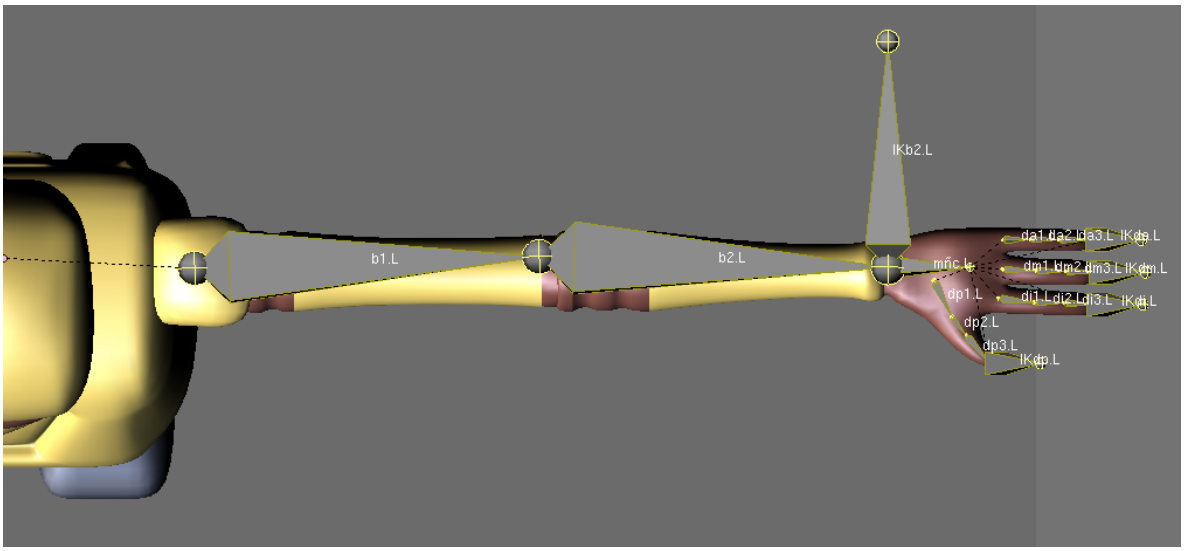
Para agregar huesos a la cadena que ya está creada, es decir para huesos que si están conectados uno tras del otro, solamente se tiene que extruir de esta forma podemos obtener cadenas de huesos como en el caso de las extremidades.

La nomenclatura que se utilice en la construcción de una armadura para un sistema de huesos depende de las necesidades que se tenga para da movimiento en la malla.



Título: Rigg (nomenclatura huesos columna) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

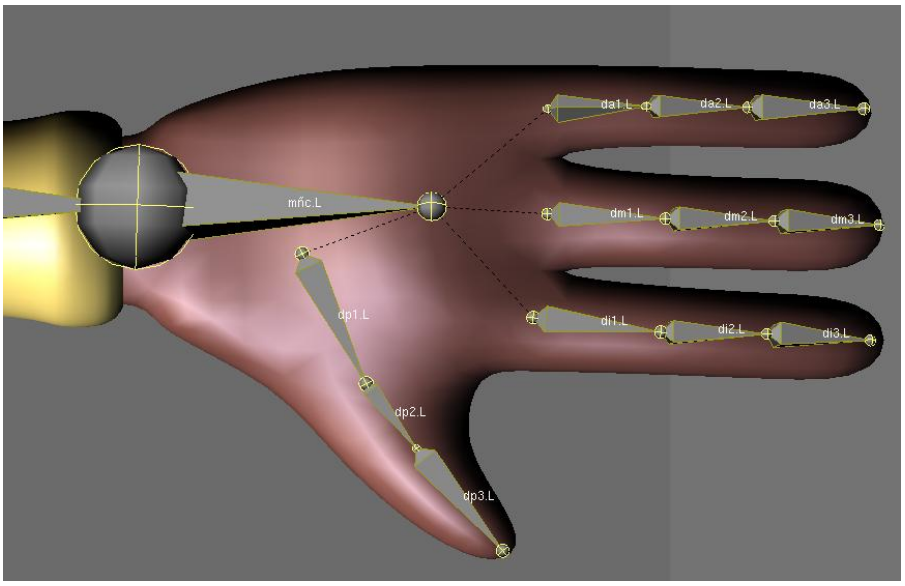
Como se aprecia en la figura en la armadura la nomenclatura que se utilizo en la columna principal del personaje son PRC, CINTURA, ABDOMEN, PECHO, CUELLO, CABEZA.



Título: Rigg (nomenclatura huesos brazo) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

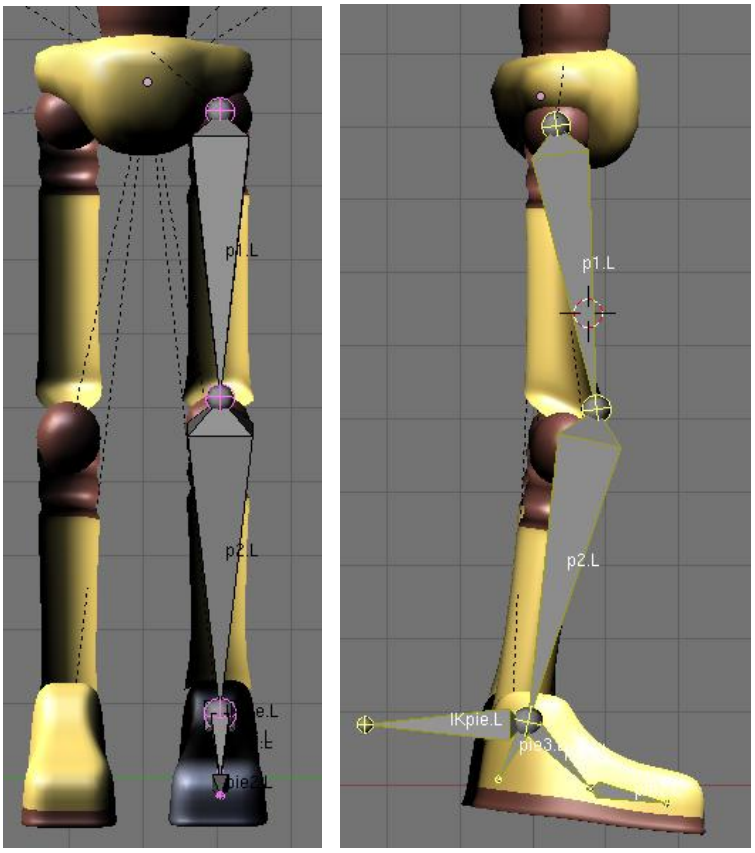
En el caso de las extremidades se utiliza otro tipo de nomenclatura ya que en el caso de Bípedos tenemos que los movimientos de las extremidades superiores y las inferiores tienen las mismas características de movimiento.

Tenemos en la extremidad izquierda superior el hueso llamado b1.L y b2.L, llevan estos la extención .L ya que blender dispone de un algoritmo que duplica cambiendo la extensión de la extremidad por .R, de esta manera solamente creamos los huesos pertenecientes a la extremidad izquierda (.L) y duplicamos otra derecha (.R).



Título: Rigg (nomenclatura huesos mano) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

El mismo tratamiento se lo realiza a las manos como se puede apreciar en el gráfico.

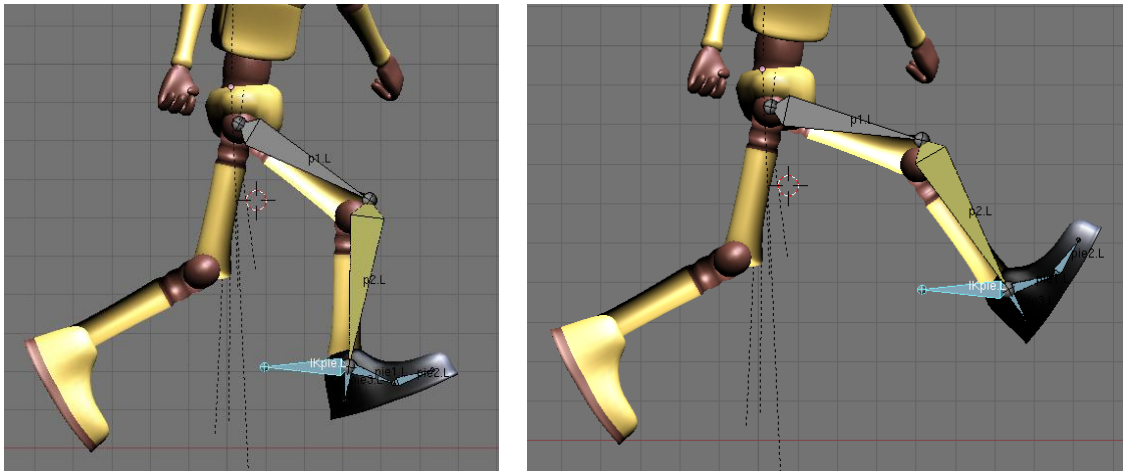


Título: Rigg (nomenclatura huesos piernas) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En el caso de las extremidades inferiores, de igual forma se realizo la construcción del sistema de huesos para la extremidad izquierda con la misma nomenclatura utilizada en el brazo. Existen varias formas de realizar el Riggeado del pie de un bípedo esta depende que tan fácil le sea al animador, es decir mientras el armado del sistema de huesos del pie sea más fácil de usar para los animadores, mas efectivo es el rigg.

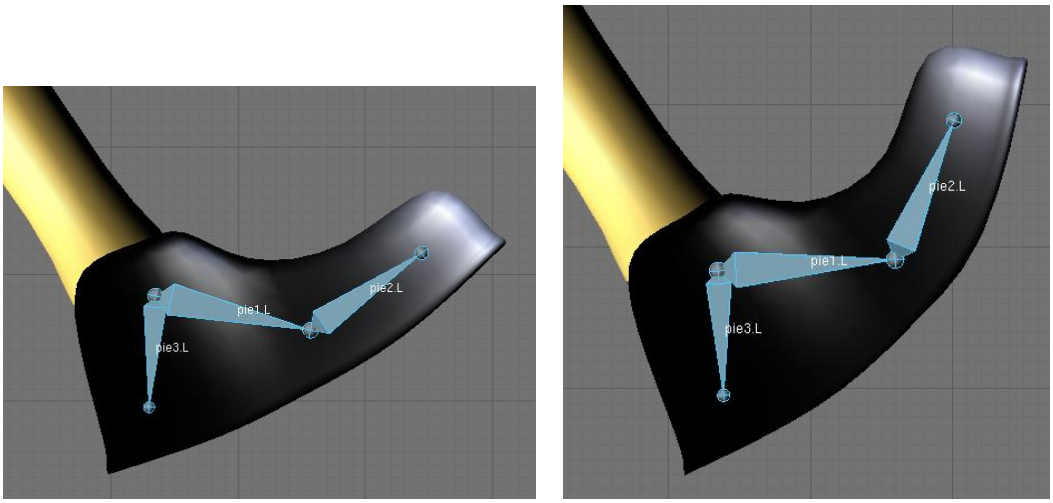
3.2.4.1 IK y FK

O también llamado cinemática inversa y cinemática directa, en una estructura de sistema de huesos da la posibilidad de realizar movimientos, en el caso de IK(cinemática inversa) puesto en una cadena de huesos como en las extremidades, los huesos son arrastrados por el hueso denominado IK, a diferencia de la utilización de FK(cinemática directa), los huesos son posicionados desde la base de la cadena del mismo.



Título: Rigg (sistema IK) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

El hueso IK de la pierna como se lo ve en la grafica, cuando a este se lo cambia de posición los 2 huesos de la pierna, tanto p1.L como p2.L son arrastrados hacia la nueva posición, moviendo únicamente el hueso IK, esto es Cinemática Inversa.



Título: Rigg (sistema FK pie) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En esta cadena de huesos del pie del personaje, no tenemos huesos IK sino que en este caso tenemos los 3 huesos del pie con libre movimiento en donde ninguno de estos huesos es arrastrado por otro hueso IK, sino que el movimiento se genera desde la base,

para el movimiento desde la grafica 1 a la grafica 2 se movió el hueso primeramente el hueso pie1.L y después el hueso pie2.L.

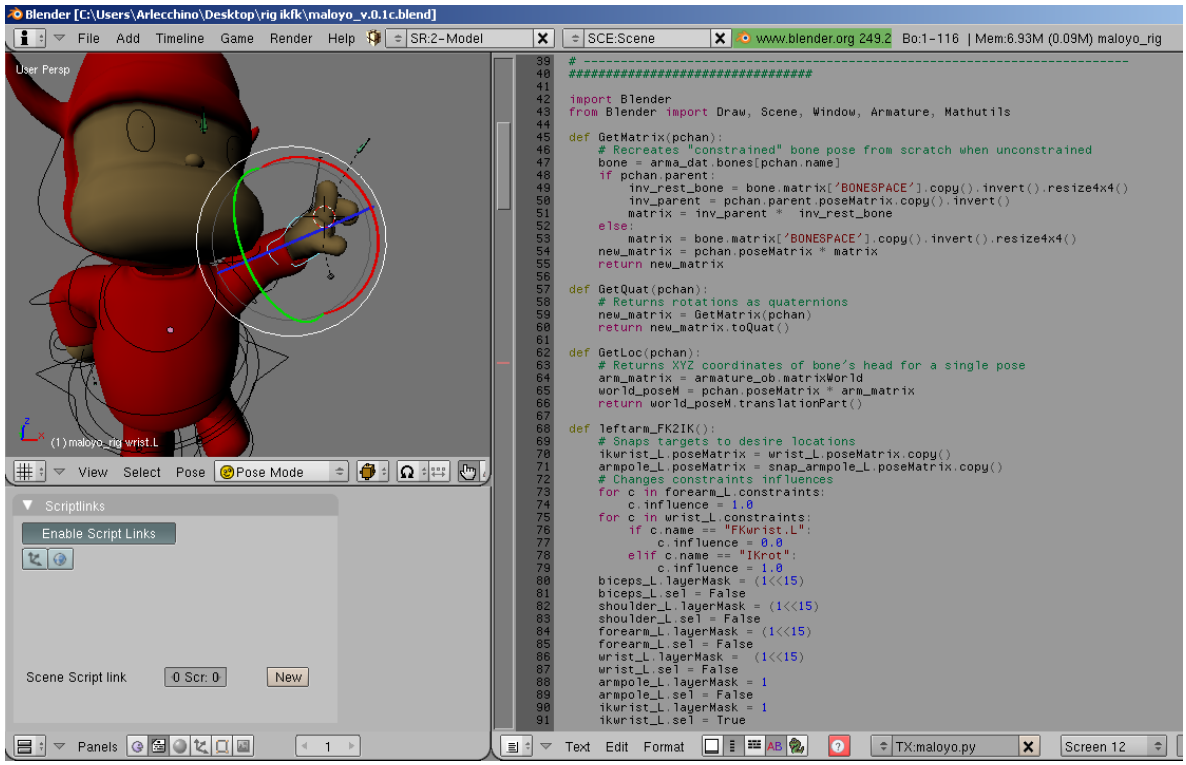
Estos 2 tipos de Riggs son necesarios y cada uno tiene su utilidad.

Los Riggs pueden ser mixtos como por ejemplo en las extremidades superiores, cuando se realiza una animación de personajes usualmente los brazos del personajes tienen en su rigg FK(cinemática directa) ya que la animación de estos es más fluida que si se realizara con IK, pero en el caso de que el personaje tuviera que empujar algún peso se activaría lo que sería el hueso IK(cinemática inversa), se realiza esto para tener las manos totalmente quietas sin movimiento y solamente el resto de la armadura haría el movimiento, para dar la ilusión de que se está haciendo fuerza contra el objeto que se esta empujando.



Título: Rigg (huesos sistema IK, FK) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

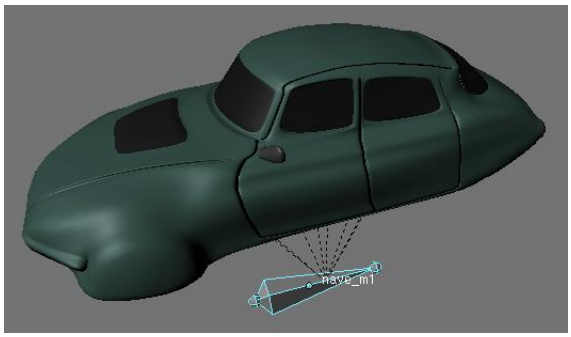
Para el personaje Leroy no usamos Riggs mixtas en las extremidades superiores. Existen scripts que desarrollaron varios miembros de la comunidad para poder realizar los cambios de IK a FK mediante comandos propios del script. Mas información en el siguiente link: http://www.malefico3d.org/blog/?p=28



Título: Rigg (script sistema huesos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

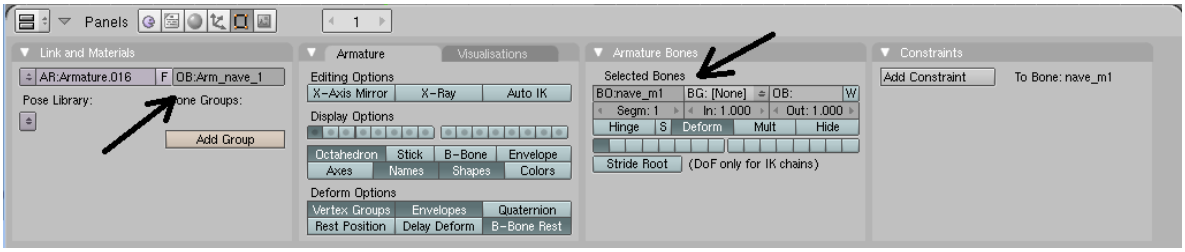
En el Riggeado de los objetos como los vehículos fue necesario una Armadura con un solo hueso ya que los movimientos que realizan estos solo son a lo largo de la carretera.





Título: Rigg (huesos Prop) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

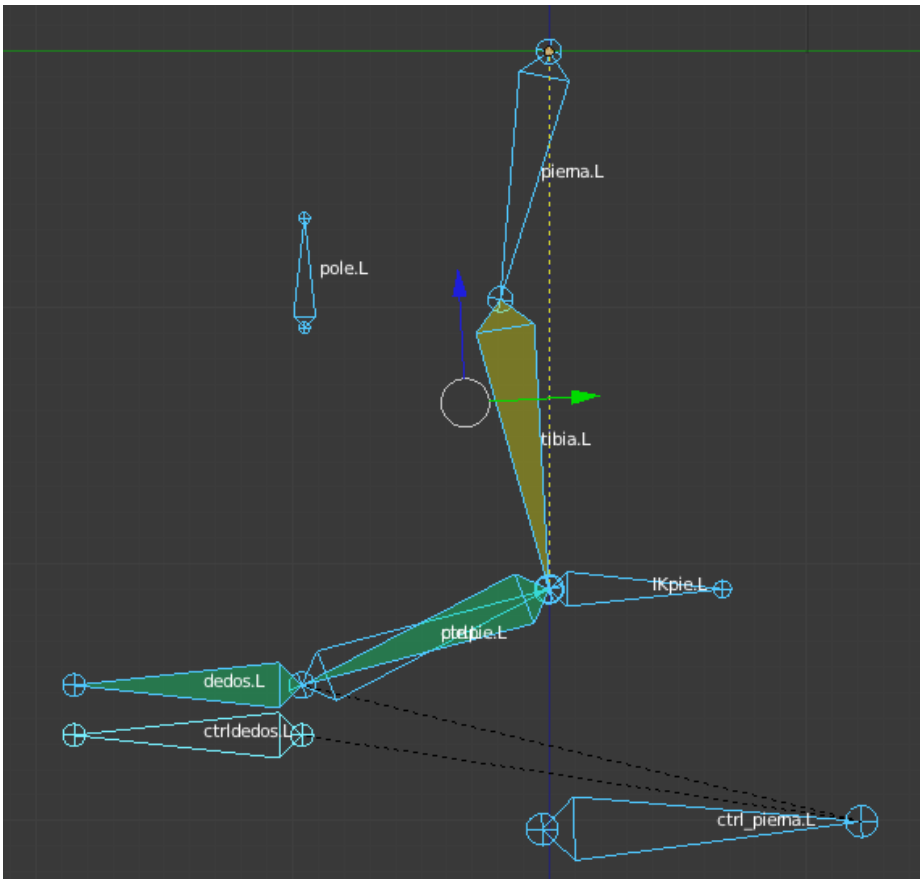
En el caso del vehículo de la gráfica se agrego una armadura "Arm_nave_1" con un solo hueso "nave_m1".



Título: Rigg (ventana propiedades Huesos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3.2.5 Sistema de Huesos en extremidades - Constraints.

Para las extremidades inferiores existen varias configuraciones para el movimiento de ésta llamados "rol de pie", que no es mas que la disposición de los huesos del pie para formar las articulaciones y que de esta manera se pueda generar los movimientos básicos de un pie.

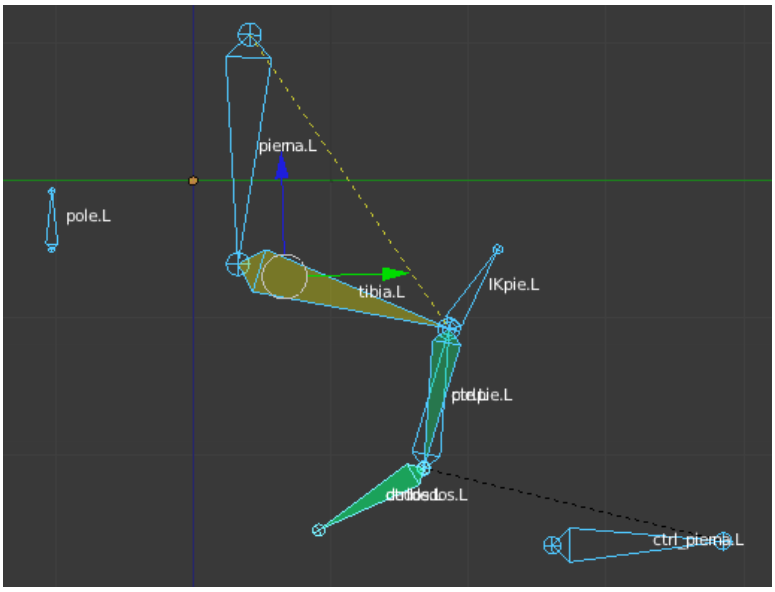


Título: Rigg (sistema huesos pierna) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

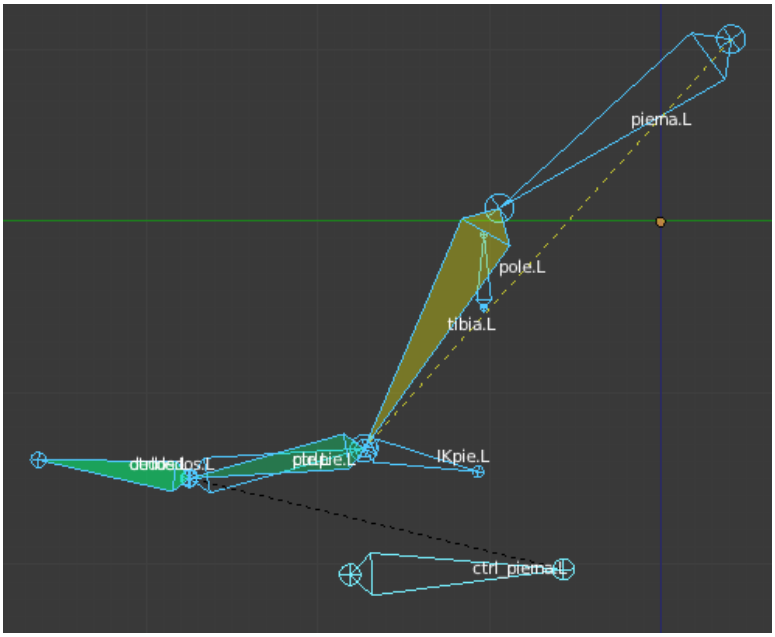
En la gráfica tenemos el Rigg o estructura de huesos del pie del personaje Leroy, cabe destacar que Blender puede exportar los Riggs para otros modelos bípedos que utilicen la misma estructura.

En esta estructura tenemos los huesos principales, además de los huesos para control y los huesos IK, cada uno de ellos es esencial para generar la animación de la pierna del personaje y en especial del pie ya que este tiene diferentes movimientos específicos llamados roles de pie.

Para poder visualizar mejor las funciones de estos huesos a continuación tenemos las graficas de un movimiento realizado por esta estructura o rigg de huesos.



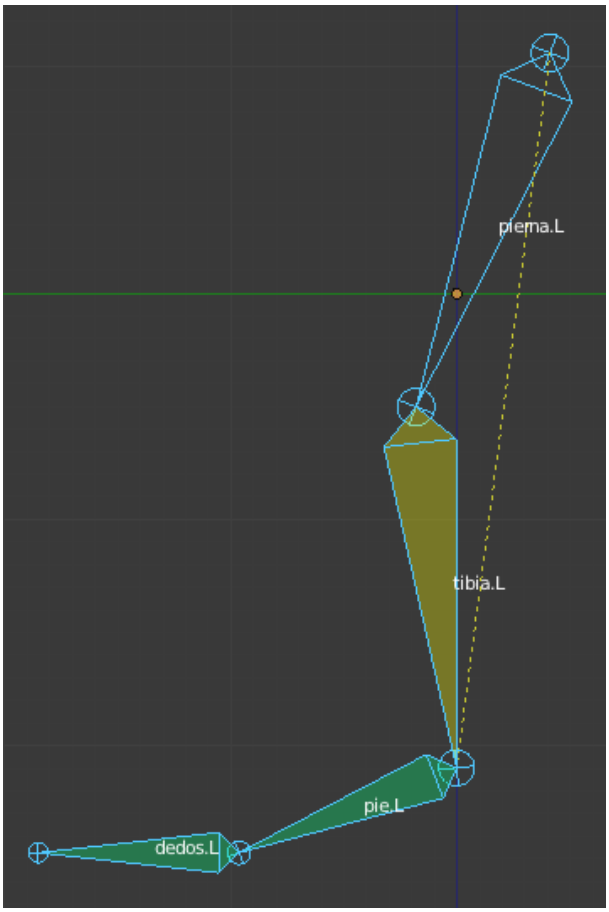
Título: Rigg (sistema huesos pierna) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.



Título: Rigg (sistema huesos pierna) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Como podemos apreciar en las graficas se está realizando el movimiento para crear una patada con la pierna y el movimiento de sus huesos. Ahora usando este ejemplo a continuación se va a describir los diferentes huesos y sus propiedades.

Huesos principales: son los Huesos que van a estar emparentados mediante el proceso de "skining" a la malla del personaje, en este caso de Leroy, en el ejemplo tenemos los huesos: pierna.L, tibia.L, pie.L, dedos.L.

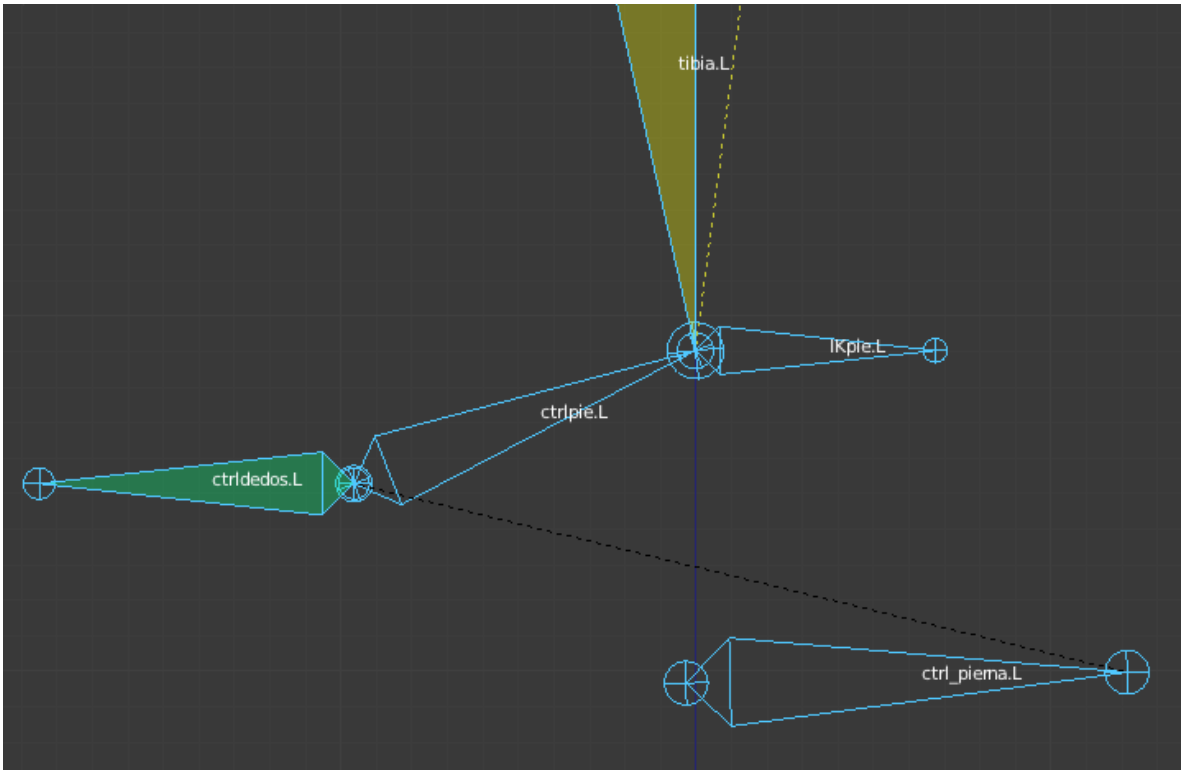


Título: Rigg (sistema huesos pierna) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Huesos Control: son los Huesos que van a generar el movimiento, estos huesos están emparentados a los huesos que van a ser manipulados pero no están conectados a estos, en algunos casos estos huesos tienen que estar sobre puestos como en el caso de los huesos. dedos.L y ctrldedos.L. En otros casos los huesos pueden rotarse en su eje

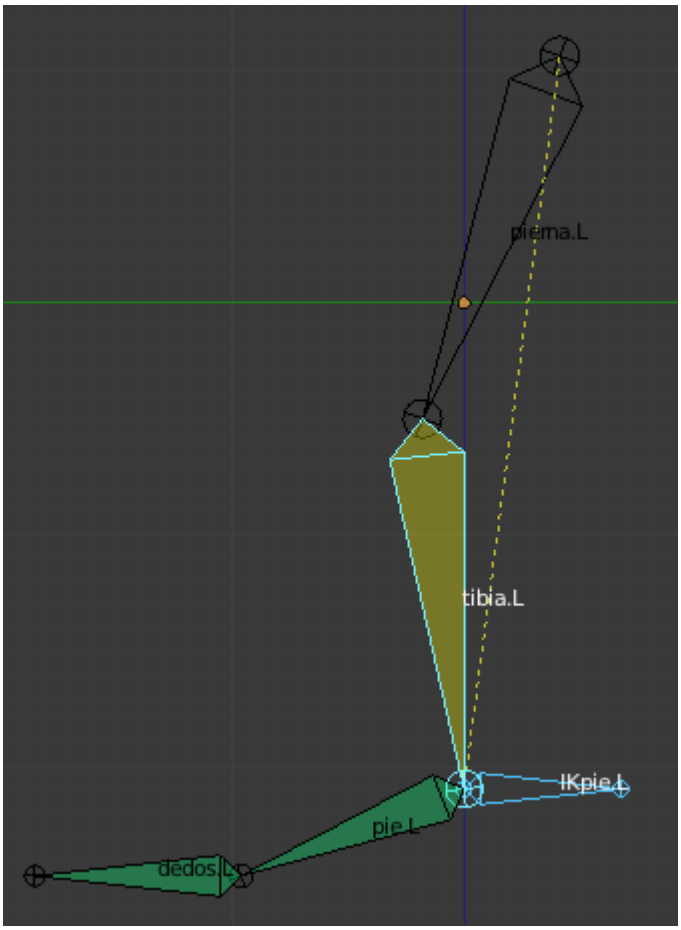
para mas facilidad en el control del hueso a mover como en el caso del hueso pie.L y su hueso controlador ctrlpie.L.

Los huesos para control en este rigg de pierna son: ctrl.dedos.L, ctrlpie.L y ctrl_pierna.L



Título: Rigg (sistema huesos - rol de pie) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

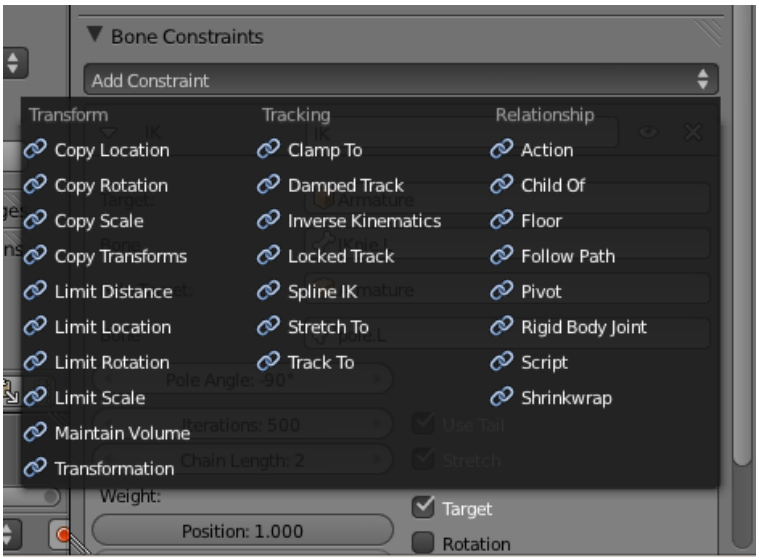
Huesos IK: estos huesos son los encargados de generar movimientos, en la cadena de huesos que están incluidos para que tengan este efecto.



Título: Rigg (sistema huesos pierna) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Constraints o Restricciones: Las constraints son entidades que restringen el movimiento de los huesos y permiten controlar un esqueleto y animarlo de manera más realista y natural. Se aplican siempre en Modo Pose.

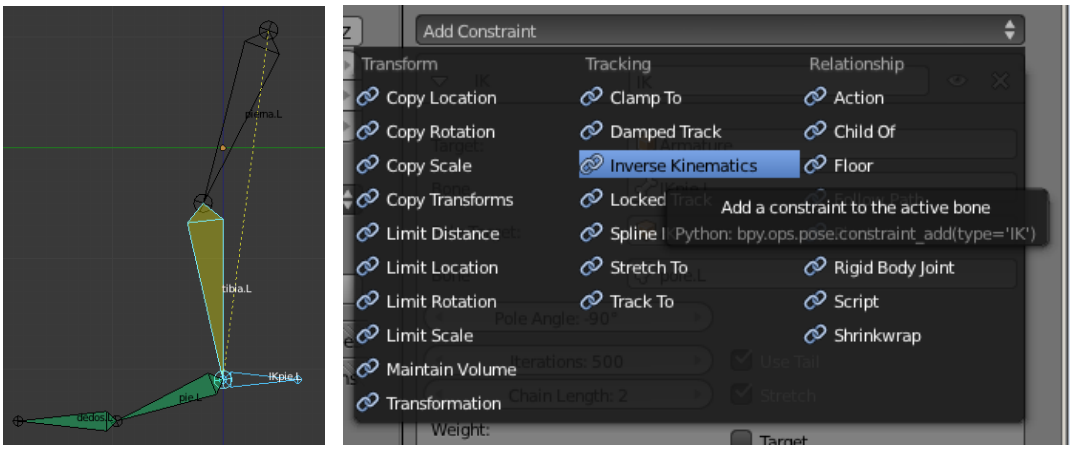
Existen diferentes tipos de Constraints estos según el tipo de control que se necesite dar al mismo.



Título: Rigg (Constraints sistema huesos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

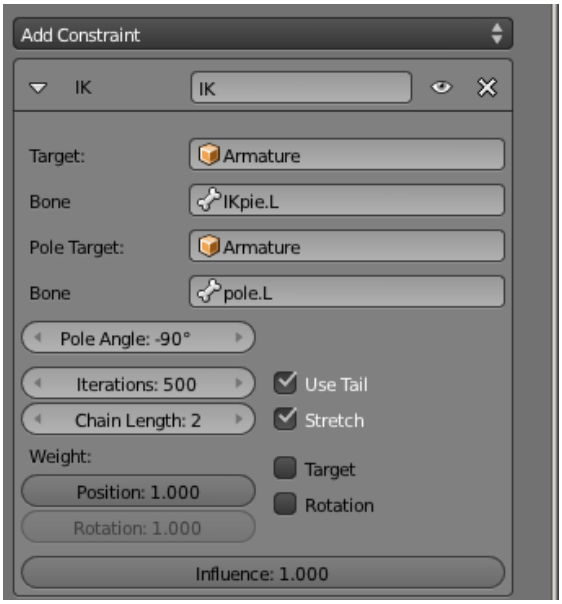
El uso de constraints es aplicable a los huesos de control como a los huesos IK, el uso de estos se debe a que los huesos principales están emparentados mediante "Skininng" a la malla del personaje y estos huesos no pueden tener Constraints, para eso nos valemos de los huesos de control o también de los huesos IK ya que estos son los que en realidad van a ser manipulados por el animador y como se aprecia en la gráfica existen Constraints tanto para Restringir Movimiento "transform" para seguir la trayectoria de un hueso "tracking" o para relacionar un hueso "relationship"

En el ejemplo del Hueso IK se selecciona el primer hueso de la cadena que va a hacer afectado por el hueso IK



Título: Rigg (Constraints sistema huesos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En la primera grafica se ha seleccionado el hueso de color amarillo, después en el menú de propiedades, en la ventana de constraints escojo el constraint de Tracking llamado Inverse Kinematics "IK"

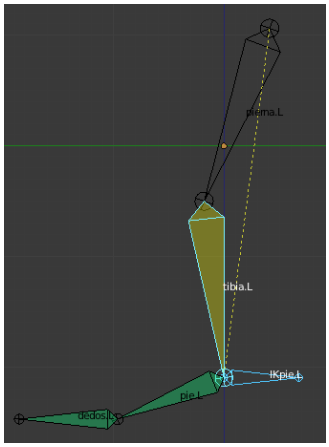


Título: Rigg (Constraints sistema IK PIE) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En la grafica se muestra el cuadro de propiedades para el Constraint llamado Inverse Kinematic IK, antes llamado también IK solver. Una vez que ya habíamos seleccionado al primer hueso de la cadena de influencia del hueso IK y haberle asignado

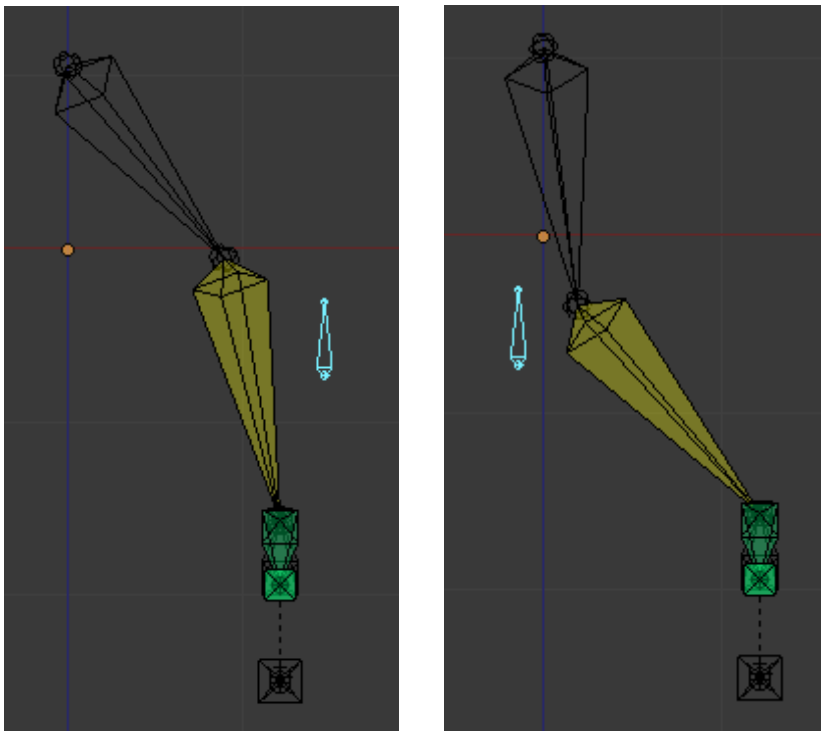
este constraint, se configura las propiedades del mismo y tenemos lo siguiente, como Target es la Armadura del presonaje en este caso se llama Armature que es un nombre por default, luego escogemos el hueso IK que va a ser el que tenga el dominio sobre la cadena de huesos con influencia IK.

Chain Lenght: Esta propiedad del constraint IK nos sirve para dar la influencia del hueso IK en una cadena de huesos, se denota con una línea entrecortada desde el inicio al final de la misma.



Título: Rigg (Constraints IK sistema huesos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Pole: en el caso de este tipo de controlador, es un hueso que no va conectado a la cadena de huesos IK y sirve principalmente para dar la dirección en una articulación que sea importante como en el caso de la rodilla del personaje, para no tener movimientos en las articulaciones que se vean anormales e irreales. El hueso Pole da una dirección en la cual la articulación generada entre el hueso pierna.L y tibia.L



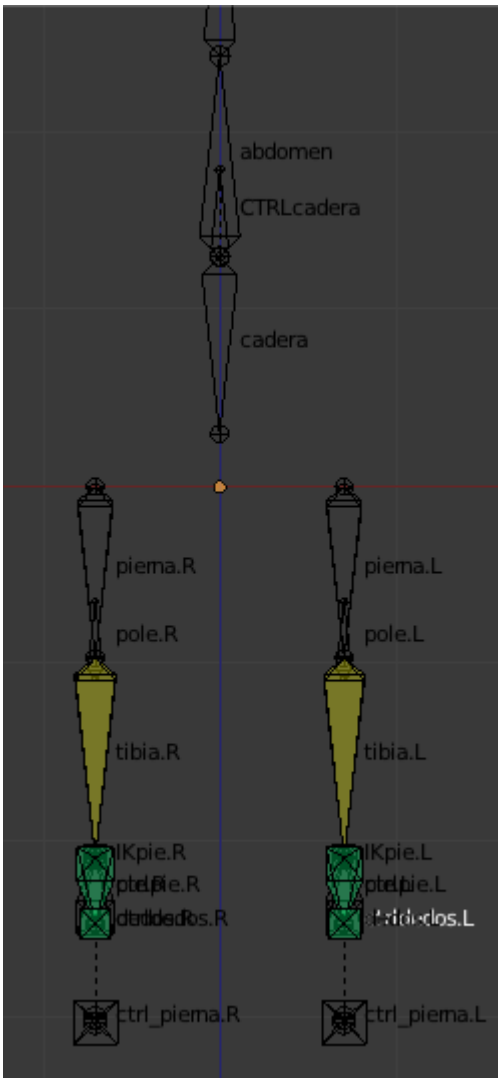
Título: Rigg (Constraints POLE sistema huesos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Como se aprecia en las figuras el hueso Pole ha sido desplazado de derecha a izquierda, de esta manera la articulación de la rodilla sigue su trayectoria para movimientos de una articulación con movimiento real.

Se utilizaron además de este otros controladores como en el caso del hueso dedos.L y ctrldedos.L en ese caso el único movimiento que se tiene es de rotación por lo que el constraint que se utilizará es el Copy Rotation.

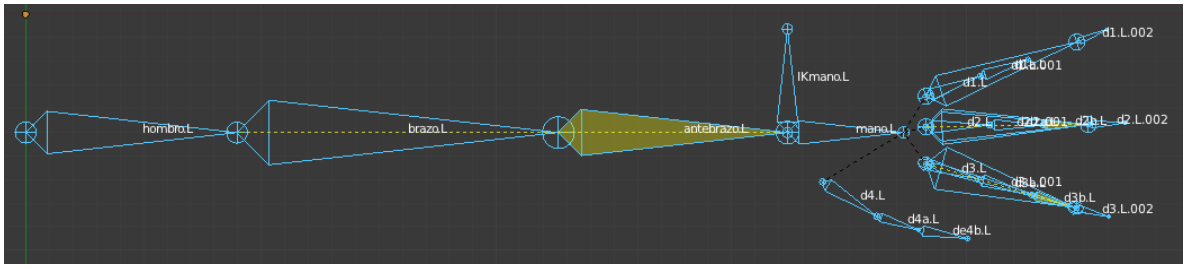
O en el caso de los huesos pie.L y ctrlpie.L para este caso se utilizo el constraint "Track to" que copy el movimiento dirigido o con target en el hueso principal seleccionado.

Todos las configuraciones de los huesos como de sus constraints son realizados solamente en en uno de sus lados en el caso de ser un bípedo como en el caso del uso de modificadores "mirror" para la generación de mallas, también los huesos pueden ser duplicados y con ellos todas sus configuraciones y constraints, solamente hay que tener cuidado con las extensiones .L y .R



Título: Rigg (Constraints sistema huesos PIERNAS) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En el caso del rigg del brazo tendríamos la siguiente configuración.



Título: Rigg (Constraints sistema huesos BRAZO) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En la configuración del brazo se tiene configuraciones similares en el uso de constraints, tanto para el brazo como de la mano y los dedos.

Como tenemos muchos huesos en una armadura y como hemos visto varios huesos están sobrepuestos, éstos también pueden ser trabajados por capas y ser ocultados cuando es necesario, por ejemplo los huesos pertenecientes a una cadena IK pueden estar ocultos ya que solamente es necesario tener visible el hueso IK para generar el movimiento deseado.



Título: Rigg (CAPAS sistema huesos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3.2.6 SKINING

Este es el proceso por el cual a una malla cualquiera le sea asignada una armadura o estructura de un sistema de huesos como el que se ha revizado en el capitulo anterior,

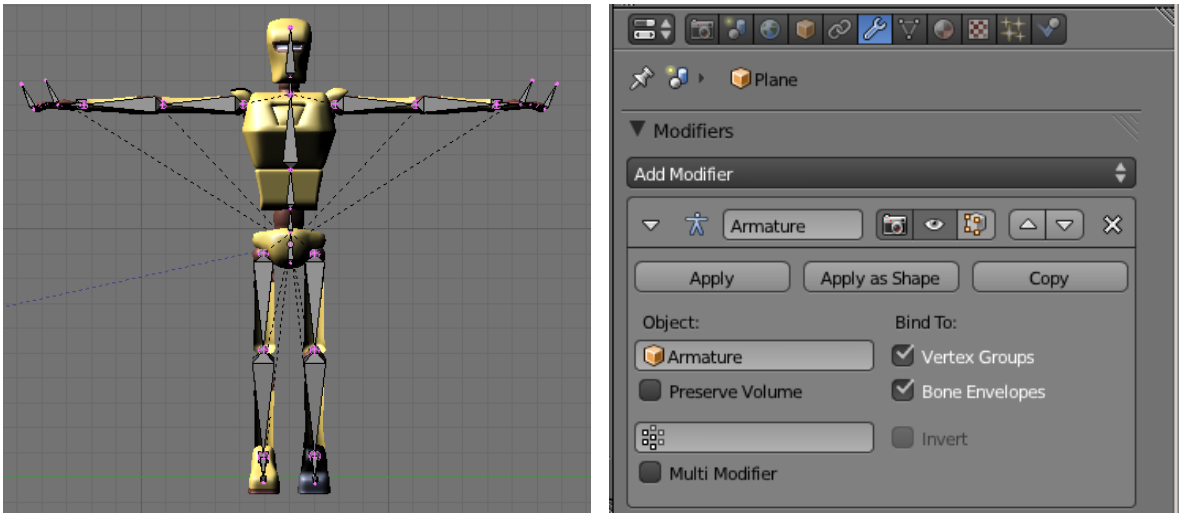
la idea es que los huesos queden totalmente cubiertos por la malla del personaje, como lo estarían los huesos cubiertos por carne en una persona real.

Para conseguir que una deformación en el esqueleto provoque una deformación proporcional en el cuerpo.

El Skinning asigna los vértices a los huesos de tal forma que los primeros están sujetos al movimiento de los últimos.

Para realizar el skining hay que asegurarse de que la malla y el esqueleto coinciden exactamente en orientación y en tamaño mediante CTRL+A (Apply Size/Rotation) Seleccionaremos la malla del personaje, y haciendo uso del modificador Armature del panel de propiedades del objeto en este caso de la malla del personaje.

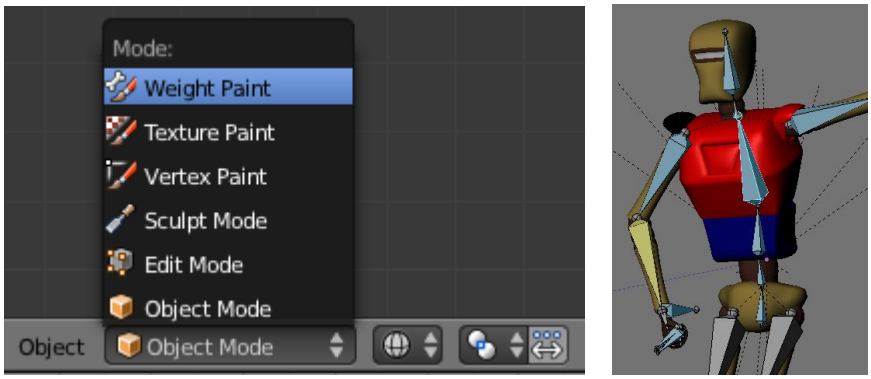
Estos botones indican de qué forma Blender calculará cuales son los vértices de la malla que cada hueso del armature deberá mover, si o hará mediante el uso de Envolventes, mediante Grupos de Vértices, o los dos.



Título: Rigg (Modificador Armature proceso skining) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

De esta forma queda emparentado el esqueleto o armadura de huesos con el modelado de la malla del personaje.

Para afinar el skinning en un personaje existe una herramienta que asegura que el hueso está asignado a cierto número de vértices para su control, esta herramienta se llama Weight Paint, en donde podemos seleccionar el hueso que deseamos examinar asi la malla que se encuentra alrededor cambia de gama de color dependiendo el nivel de pertenencia a la malla más próxima.



Título: Rigg (Skining - sistema de pesos por color) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Como se aprecia en la figura del personaje, el hueso del pecho tiene asignado toda la malla superior del torso, en la parte inferior del torso esta de color azul por lo que no tiene ningún tipo de unión con este. La gama de colores pueden y desde azul que significa una débil unión con el hueso, la gama va cambiando se tiene después de acuerdo a la fuerza de unión colores que van desde azul, amarillo, verde y rojo que es la más fuerte.

3.2.7 INTEGRACION ARCHIVOS

Una vez que tenemos elaborados los personajes como los props en 3D digital con sus texturas y sus respectivos Riggs en caso de que lo necesitaren, damos paso a la fase de producción, esta es la fase de animación.

Para realizar la animación se hace uso de las referencias ya realizadas anteriormente estas son el animátic, además se tiene en cuenta mucho el diseño de los personajes y el guión del cortometraje, todas estas son las herramientas que se le entregan a los animadores para su desarrollo en la fase de animación del proyecto.

Para el cortometraje se dividieron en 11 escenas, como ya se indicó anteriormente, se realizó la animación de cada una de estas por separado, el animatic como el diseño de personajes nos ayuda para que en la animación de cada escena los animadores puedan tener referencias de los personajes y asi tener una fluidez en el proceso, sin cambios que se puedan reflejar estos.

Para el caso de Leroy solo existió un animador por lo que el proceso fue realizado casi sin ningún cambio.

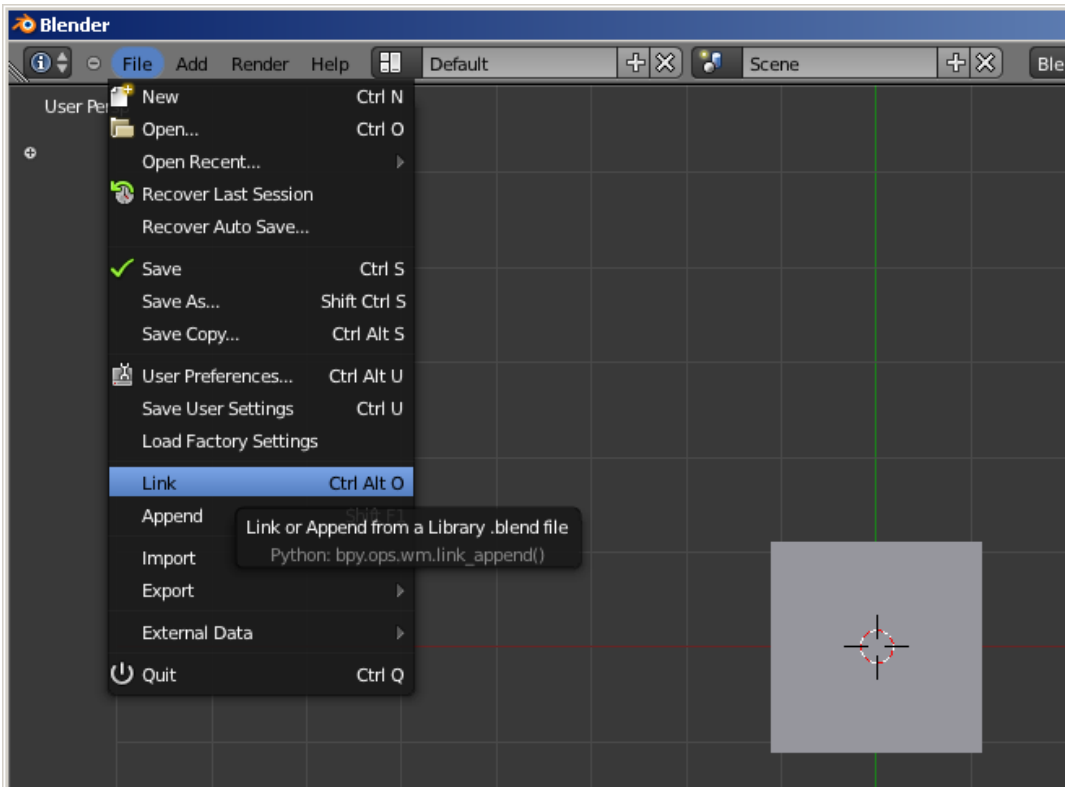
Proceso de animación Escena

El proceso de animación de una escena de Leroy es descrita a continuación:

Primeramente tenemos todos los elementos previamente guardados en las diferentes carpetas del proyecto Leroy, ahora lo que se debe hacer es "jalar" los objetos necesarios al archivo nuevo **"escenaX.blend".**

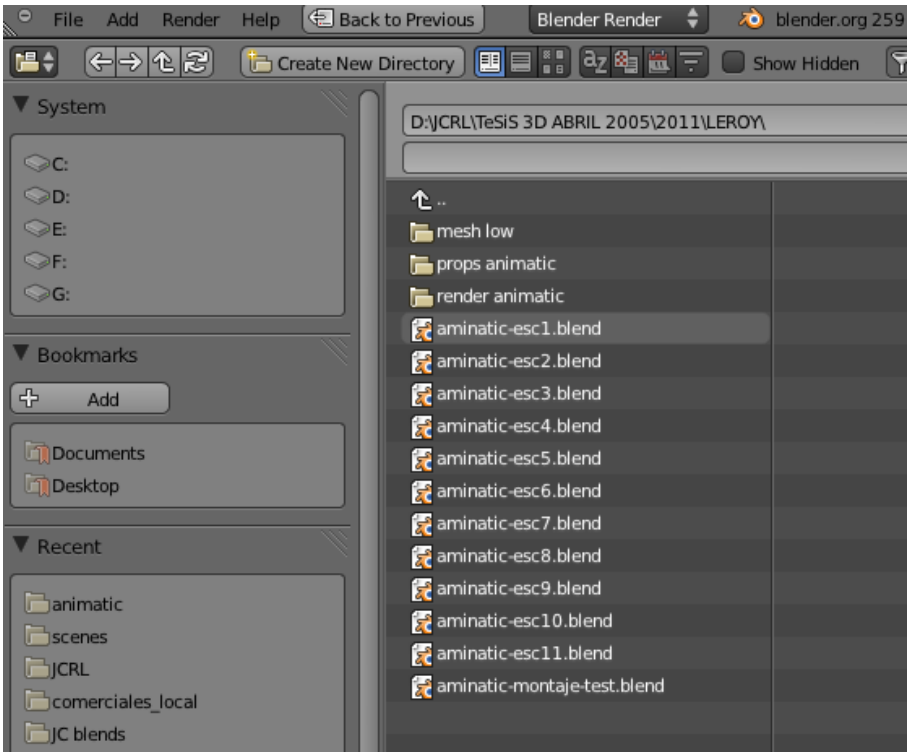
Primeramente se jala o "linkea" los objetos necesarios del animatic de la "escenaX" para tener la referencia de animación de los personajes de la escena.

En un archivo nuevo que ha sido creado en la carpeta SCENES procedemos a linkear objetos de referencia del animatic.



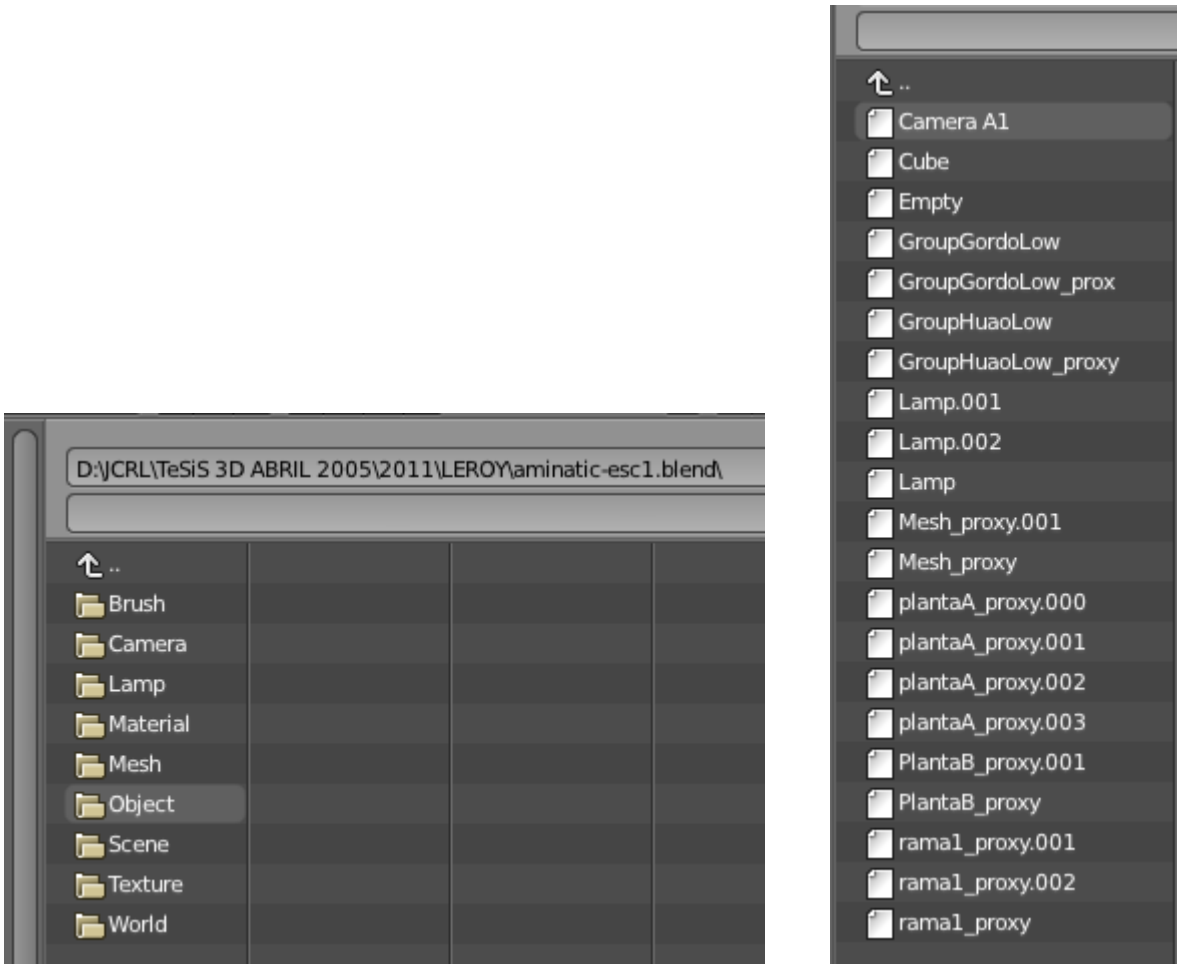
Título: Importación Objetos (link) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Mediante el exploardor interno de Blender nos dirijimos a la carpeta contenedora del proyecto Leroy y seleccionamos la escena a ser puesta en producción.



Título: Importación Objetos (link escena 1) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Como se ve en la imagen ha sido seleccionado el archivo animatic-esc1.blend, para después poder ingresar a las diferentes carpetas que contienen los objetos de UN solo archivo, en este caso el de animatic de la escena 1, podemos ver que existen las carpetas por default: Brush, Camera, Lamp, Material, Mesh, Object, Scene, Texture y World, cada una de estas contiene los objetos que se puede "linkear" del archivo animatic-esc1, cabe recalcar que cuando se linkea un objeto a otra escena como en este caso no se está copiando el objeto en sí, sino se está haciendo un llamado a la instancia específica, por lo tanto es de mucha ayuda para el desarrollo de un proyecto de este tipo, ya que mientras se está en el proceso de producción, por ejemplo en la animación, cuando los animadores están trabajando en un personaje, este puede estar con sus texturas no terminadas y las personas encargadas de realizar las texturas finales de este personaje pueden seguir trabajando en ello, sin que los animadores tengan problema alguno en su trabajo y sin interferir su trabajo, ellos pueden ir notando los cambios progresivos en las texturas del personaje.



Título: Importación Objetos (link escena 1) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Para el ejemplo se selecciona la carpeta llamada Object en donde están alojados los archivos pertenecientes a los modelos 3D realizados en lowPoly que están creados en el archivo animatic-esc1, de esta manera tenemos los objetos del aminatic en la escena a animar, debemos seleccionar todos ellos, ya que todos ellos forman parte de la referencia de la escena 1.

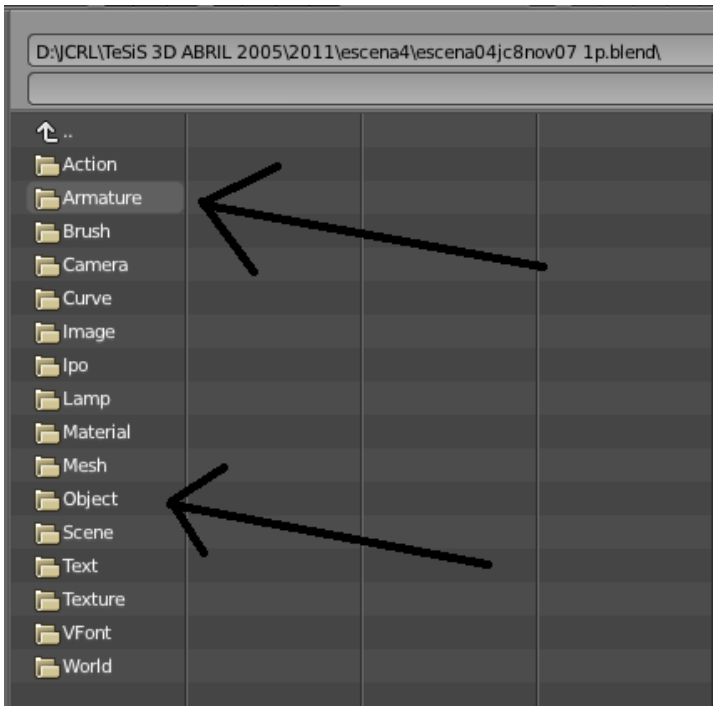


Título: sistema Capas - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Blender maneja capas en este caso localizamos al animatic de la escena 1 y lo movemos a la capa 11 para no confundir el animatic de la animación de la escena, mediante el shortcut "m".

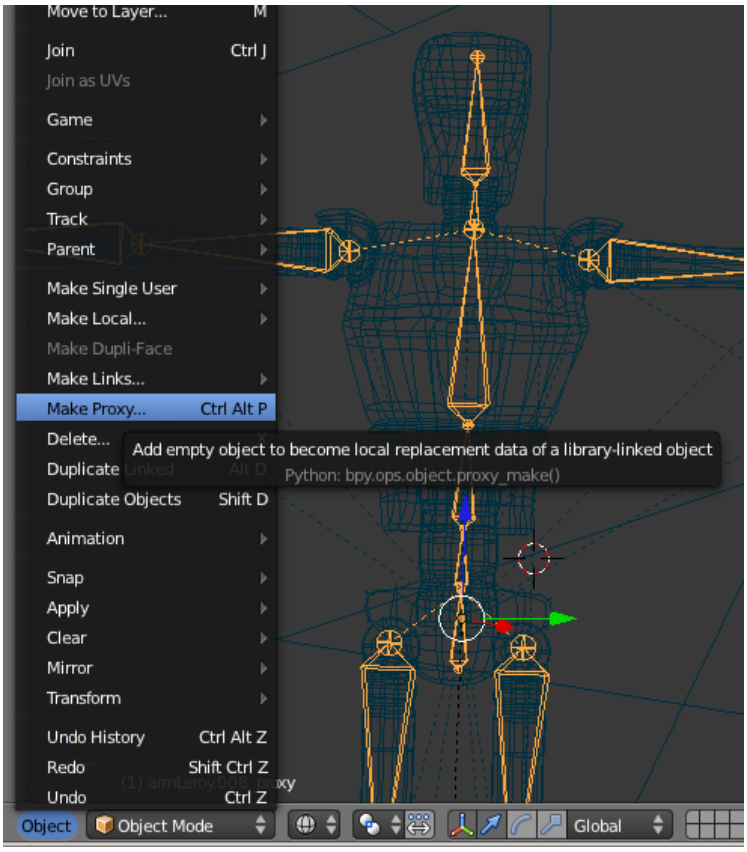
Una vez que se ha realizado el link de los objetos del animatic a la escena y puestos éstos en la capa 11, procedemo a realizar el LINK de los objetos que forman parte de la animación de la escena,

Por ejemplo en esta escena está presente el personaje LEROY con nave espacial, de la misma forma se procede a linkear de la carpeta del proyecto LEROY y de la carpeta chars al personaje Leroy. Tanto el modelo tridimensional como la armadura "RIGG" del mismo, estas se encuentran en las carpetas de la imagen a continuación.



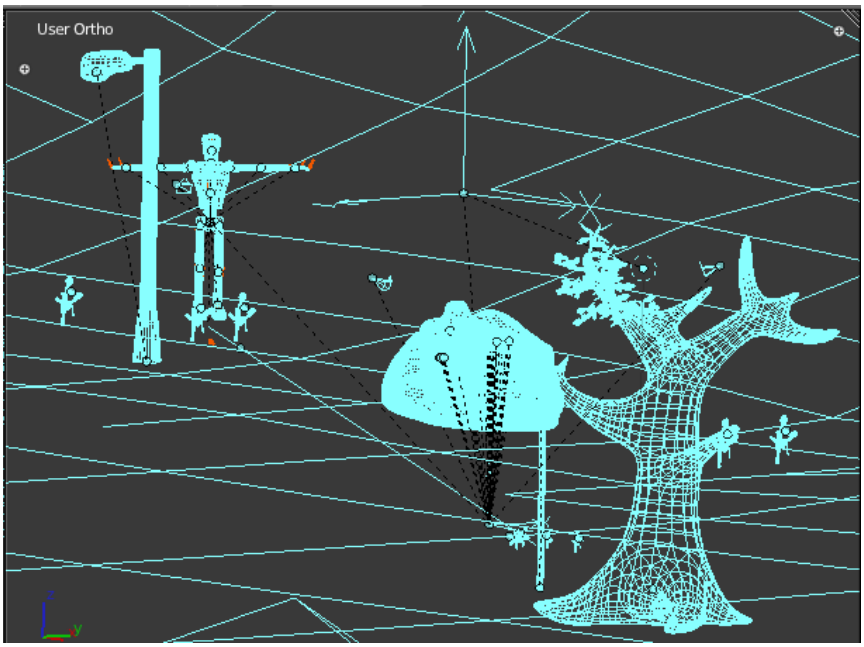
Título: Importación Objetos (link Armadura escena 1) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Después de efectuar el LINK del personaje con su malla y su armadura o RIGG para poder animar tenemos que hacer uso de la utilidad llamada PROXY.



Título: Proxy (Armadura) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

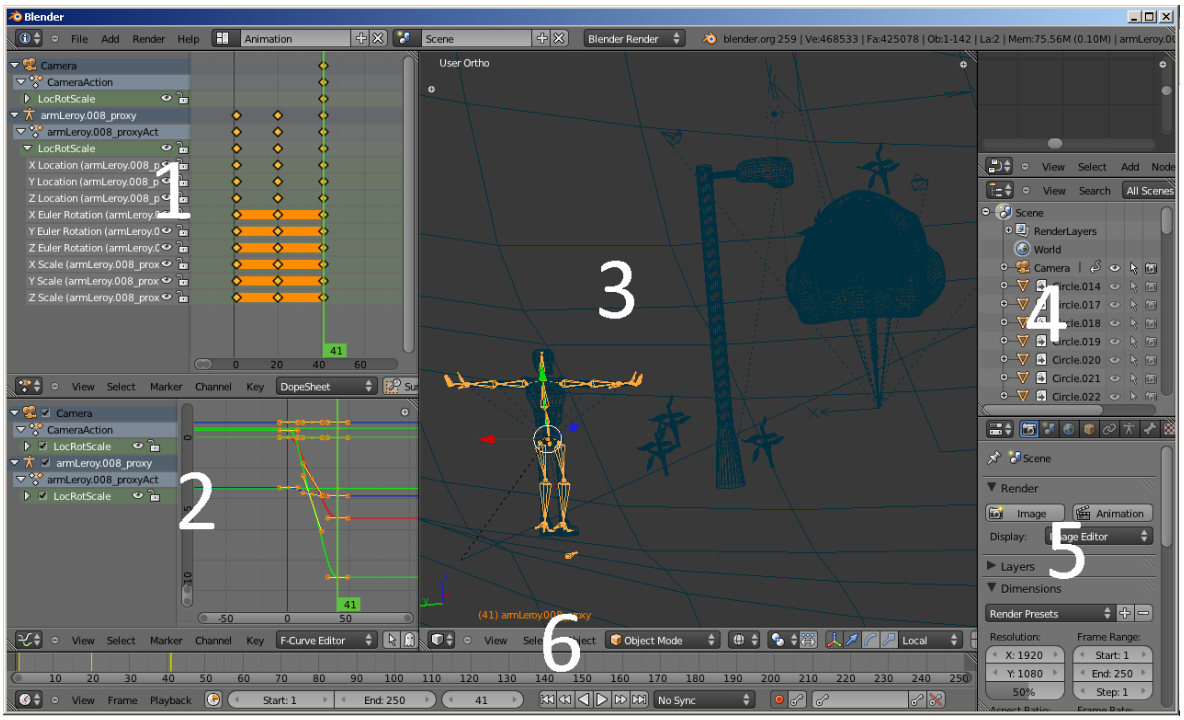
El proxy sirve para que la Armadura del personaje Leroy quede liberada, de esta manera se puede hacer uso de ésta, para la animación que se necesite en ese archivo en específico, en este caso escena_1.



Título: Importación Objetos (link Objetos escena 1) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Cuando los elementos son linkeados a la escena que se va a animar, éstos al ser importados cambian de color como en la imagen.

Setup para animación de Blender.



Título: Animación (intefaz UI) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Entrada en la fase de Producción se da paso a la animación de las escenas que forman parte del cortometraje.

Para eso se configuró a Blender con el setup de animación, en esta distribución tenemos a 6 ventanas principales.

En la imagen tenemos:

1. Dope Sheet – controlador de acciones.
2. Graph Editor - el editor de curvas IPO.
3. 3D View – visor de elementos 3D.
4. Outliner – explorador de objetos 3D.
5. Properties – propiedades del Objeto.
6. Time Line – Linea de tiempo de la animación de la escena.

Dope sheet.- El anteriormente llamado "Action Editor" fue sido extendido para convertirse en una completa Dope Sheet, este permite el control sobre multiples acciones a la vez, agrupando por tipo, y mejor acceso a los shape keys, es decir cada movimiento que se genere tanto en loccion, rotación o escala de uno de los huesos controladores de una armadura en un personaje es grabado en el Dope sheet.

Graph Editor.- El graph editor antes llamado "editor de curvas IPO" que sirve principalmente para controlar la variación de los atributos de los objetos a lo largo del tiempo. En realidad son curvas de Interpolación que, a partir de los valores que toman en los puntos de control (frames clave), interpolan los valores del parámetro en los frames intermedios.

Cada curva del interfaz se corresponde con una propiedad animable, como todos los objetos en Blender son animables podemos tener una idea de la ayuda que tiene esta herramienta.

"Curvas IPO (de InterPOlation)".

3D view.- Esta ventana es la comúnmente utilizada para la visualización de los objetos que se están utilizando al momento en Blender, usada para tener una visualización espacial en los ejes coordenados utilizados.

Outliner.- Esta ventana sirve para visualizar la distribución de los objetos utilizados en el archivo .blend, es decir que este es un tipo de explorador de objetos que forman parte de éste, en el cual tenemos los objetos que han sido importados "linkeados", como también los objetos creados en el mismo, las luces, las texturas, las armaduras, etc. Con la ayuda del Outliner podemos visualizar, seleccionar o denegar la renderizacion de un objeto en particular.

Properties.- Esta ventana como ya se ha mencionado anteriormente están todos diferentes parámetros que son provistos de configuración que pueden ser modificados, en estos podemos encontrar las configuraciones para editar: Obletos, Materiales y Texturas, Armaduras, Luces.

Time Line.- Esta ventana es de mucha ayuda al momento de animar personajes, ya que como su nombre lo indica es la línea de tiempo en la que una escena es animada, como se mencionó anteriormente un segundo de animación son 24 cuadros, entonces la

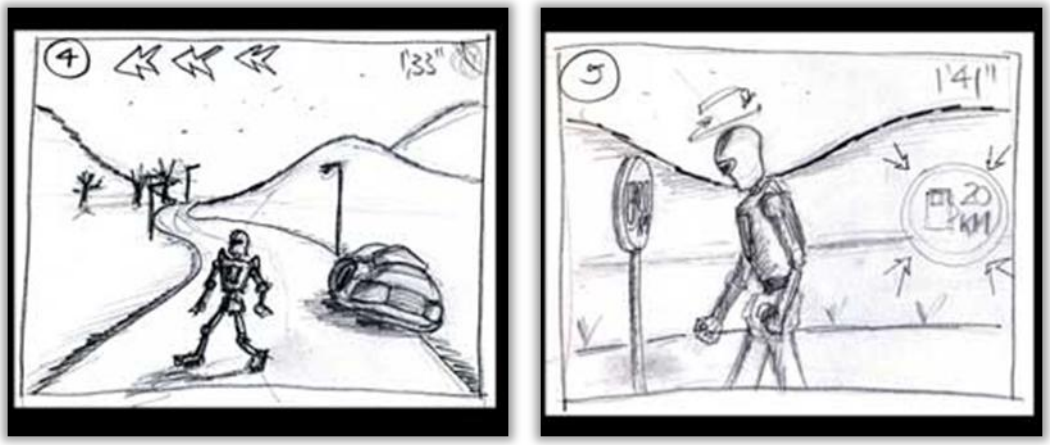
escena depende principalmente del trabajo de preproducción, el animátic en especifico donde ya tenemos registros los valores en tiempos de cada escena en tiempos y frames. En esta ventana podremos visualizar el tiempo representado en frames. También se tiene controladores de avance y retroceso del mismo, además de propiedades de sincronización para obtener una reproducción lo más cercana a la velocidad que se tendría finalmente realizado el render. También se tiene un botón de activación de Auto Keyframe que sirve principalmente para insertar keyframes automáticamente cuando se realiza algún movimiento en un hueso de una armadura, éste que registrado automáticamente en el dope sheet.

Cabe recalcar que la configuración de las ventanas es totalmente editable, este depende de las necesidades del animador y la necesidad de visualización del mismo, en versiones a partir de la 2.5x Blender trae una nueva utilidad para separar las ventanas para así poder trabajar en varios monitores, pero esto depende del Hardware que se tenga para efectos de usar esa nueva utilidad.

3.2.8 ANIMACION

El proceso de animación de este cortometraje fue dividido por escenas las mismas que fueron realizadas en el proceso del animatic así como también los tiempos y la composición que son las referencias para el desarrollo de la animación de la escena en la que se trabaje.

Por ejemplo en la escena 4, se tienen las dos viñetas 4 y 5, en donde Leroy se queda varado en medio de nada, y el camina hacia una señal para ver que decía ésta, quedando el muy frustrado al ver que la próxima estación de servicio está a 20Km del punto donde se encuentra.

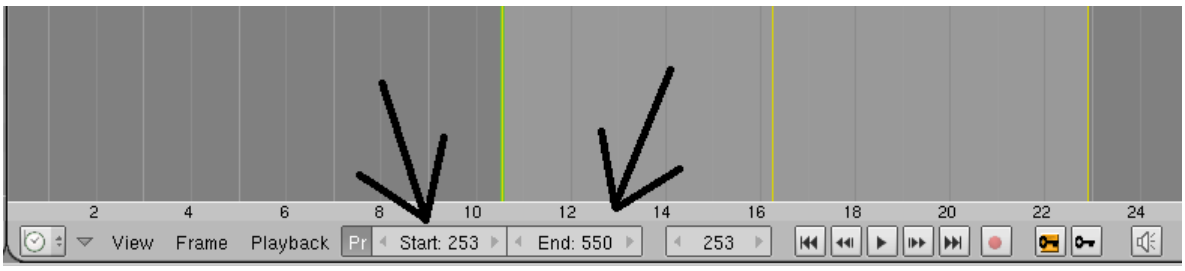


Título: Animación (Story escena 4) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Estas 2 viñetas corresponden a la escena 4, no confundir con el número que tienen cada una de éstas, en el animátic es en donde se asignan las respectivas escenas del cortometraje pero como a éste se lo tiene en formato de video, se ha mostrado lo mismo que representa pero con el storyboard.

Después de haber realizado el proceso del link de los objetos a animar en la escena incluido el animatic de referencia procedemos a realizar la animación.

Primeramente nos fijamos que en el animatic tenemos los siguientes valores en el rango de tiempo:



Título: Animación (Timeline escena 4) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En esta escena el primer frame comienza en el numero 253 y termina en el frame 550, como tenemos que un segundo tiene 30 frames tenemos 297 frames en total y en segundos son 9,9 segundos.

En esta escena No.4, el storyboard como el animatic muestra como Leroy después de verse varado en el desierto, camina con pocas esperanzas hacia un letrero que mira al otro lado del camino, para darse cuenta que la ayuda más cercana se encuentra a 20KM de distancia, como ya vimos el tiempo de duración es de 9,9 segundos es decir sería el timing de esta escena en frames serían 297.

Teniendo ya importados "linkeados" los personajes involucrados en esta escena así como los props, pasamos a la animación de los mismos- En esta escena tenemos en la que Leroy camina desde su nave hasta el letrero ubicado al otro lado del camino, se animó varias acciones como el caminar, el estar de pie, la asombro que le causa ver el letrero y la queja que viene después del mismo.

Entonces tenemos 4 acciones principales para este personaje en esta escena, estas acciones luego nos servirán para montar la escena cada acción que se use en el montaje de la escena se en el editor NLA se llamará strip, estas son:

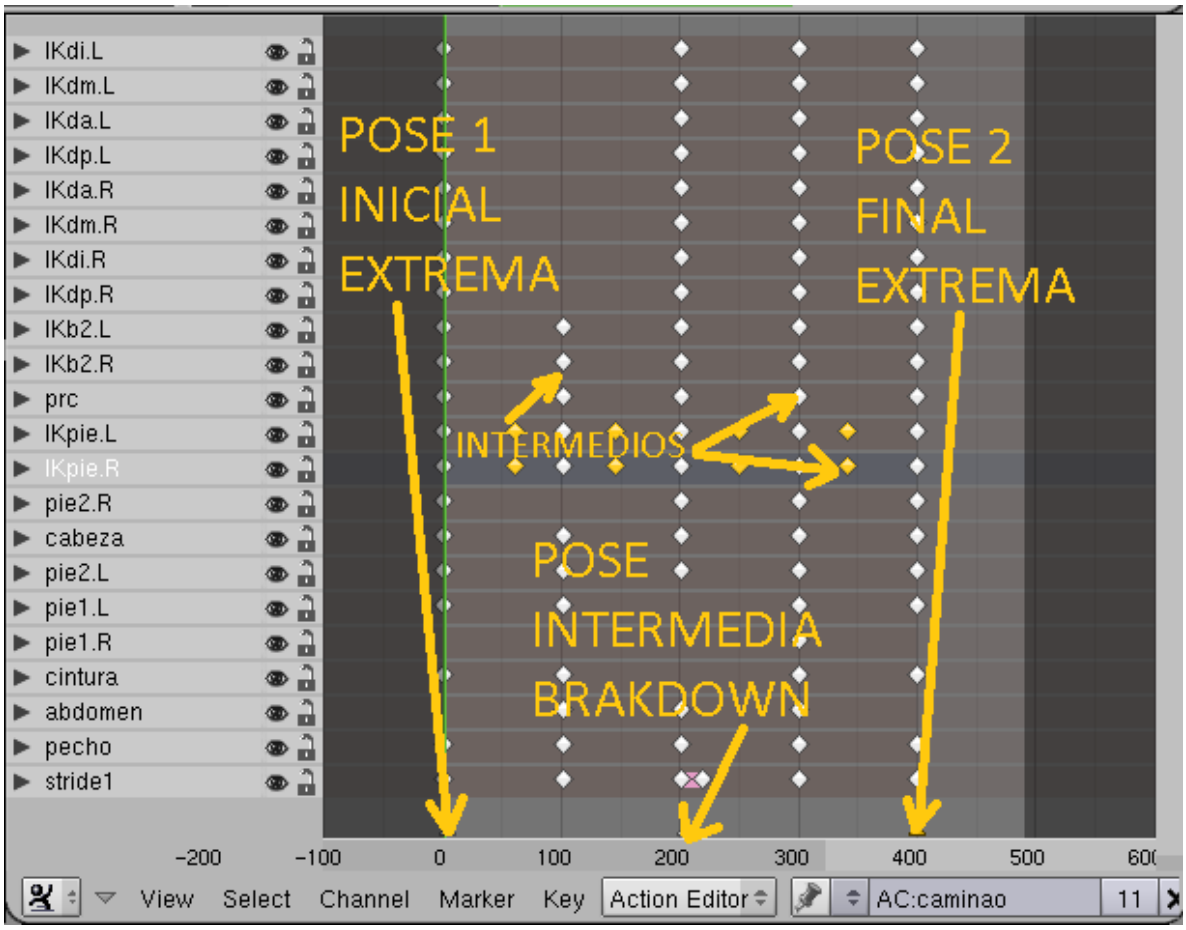
1. Caminado.
2. Estado de pie.
3. Asombro – Fijación.
4. Queja.

Primeramente se crearon los diferentes movimientos en la armadura de Leroy para crear sus respectivas acciones.

3.2.8.1 ACCIÓN CICLO CAMINAR.

Para crear las acciones por separado, nos valemos de una combinación de ventanas entre estas están: Ventana 3D View, Action Edition ahora llamada ventana Dope Sheet y la ventana Time Line.

En la mitad de la acción tenemos la pose 1 pero invertida a esta pose la denominamos Breakdown, entonces tenemos:

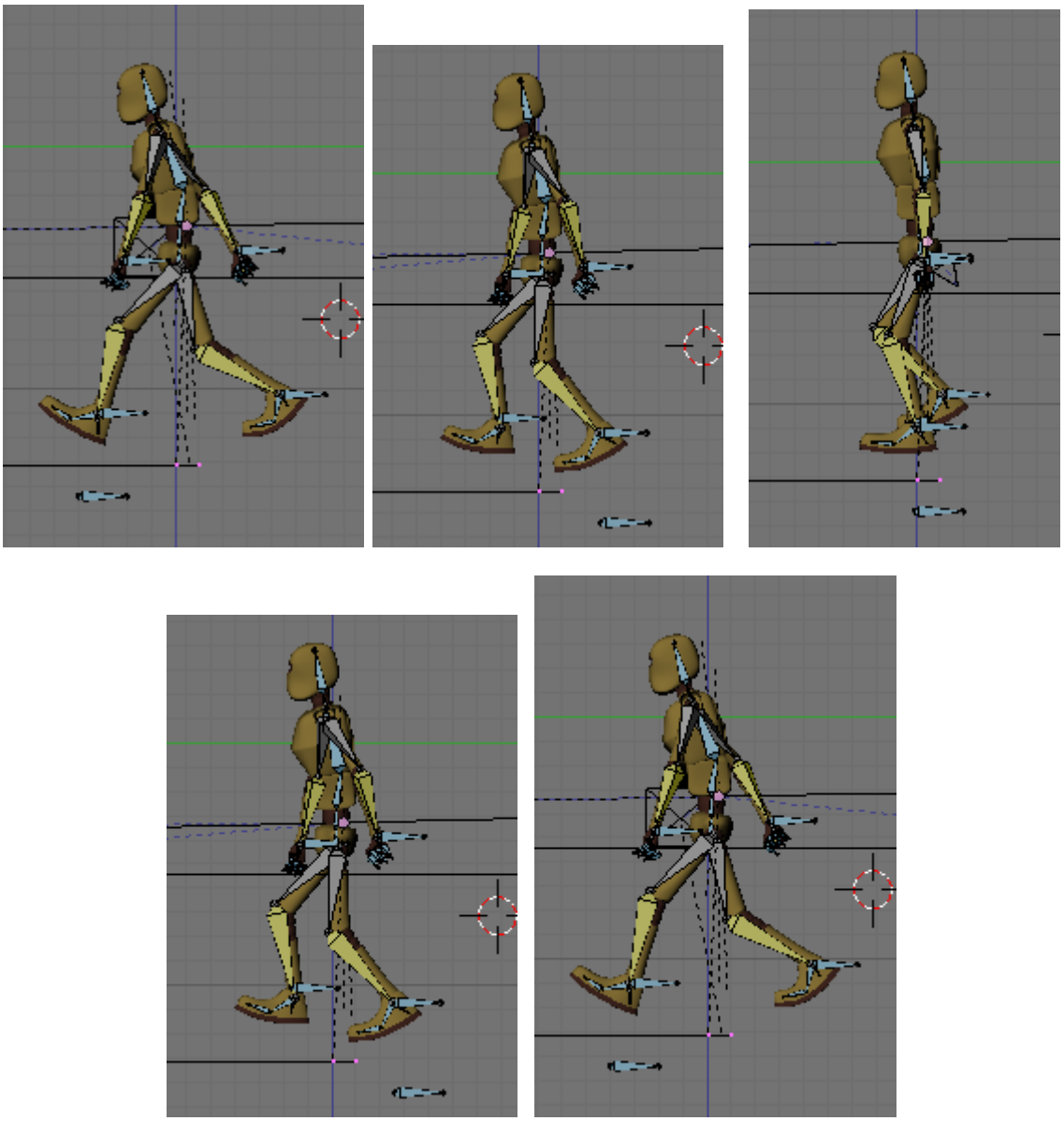


Título: Animación (Ciclo caminar POSES) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Entre la pose 1 y la pose Breakdown se han creado más poses llamadas intermedios o "in between" estas refuerzan las poses extremas y las breakdown para que exista una fluidez en la animación de la acción.

En el caso de la acción Caminado tenemos 6 poses "intermedias" o "In between", 3 de ellas se encuentran entre la Pose 1 y la Pose Breakdown y sus correspondientes invertidas se encuentran en el segundo intervalo del ciclo.

Para esta acción tememos las siguientes poses.



Título: Animación (ciclo caminar) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Para lograr las poses de las que tratamos anteriormente, hacemos uso de los huesos IK para las piernas ya que en éstas no hay sistema FK y para los brazos se desactiva el sistema IK para los brazos quedando solamente FK para su manipulación.

El uso de IK únicamente en las piernas es dado que es más fácil su control, a diferencia de las extremidades superiores tienen ambos tipos de control IK y FK.

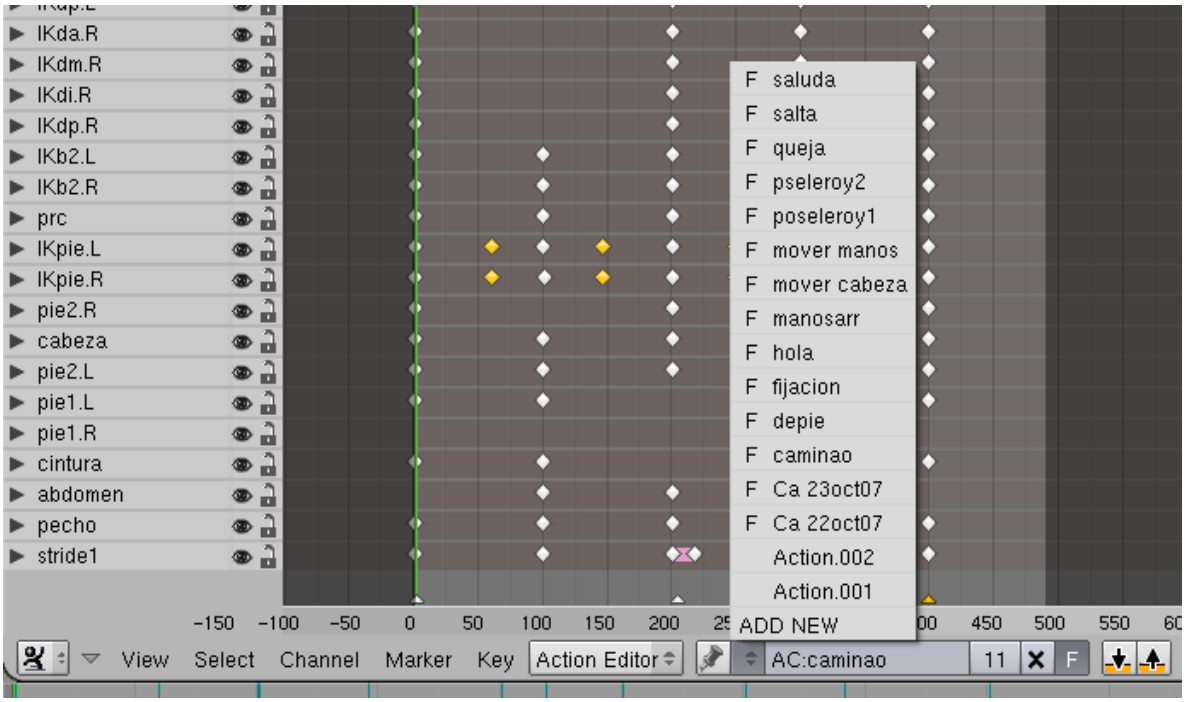
Para construir una pose trasladamos los huesos de control o la cadena de huesos "FK" hasta lograr la ubicación deseada de éstos. Cuando un hueso está en la posición deseada se tiene 2 opciones para fijar esta posición en la línea de tiempo, la primera es fijar el hueso mediante el shortcut "i - insert", al presionar esta tecla aparece un menú contextual "Insert Keyframe Menu" en el que se despliega varias opciones entre las cuales tenemos: Location, rotation, scaling, LocRot, LocScale, LocRotScale. Cada uno de estas sirve para guardar en ese keyframe en la línea de tiempo su Ubicación, su rotación, su escala o la combinación de éstas. Todo esto realizado desde la ventana 3D view.

Otra opción es la ventana Time Line, en donde se encuentra una utilidad muy necesaria al momento de crear poses, es un botón llamado "**Auto Keyframe Insertion**", la facilidad que presenta para el animador es que únicamente se debe ubicar al personaje en la línea de tiempo y realizar los cambios de los huesos de control, los mismo que son fijados automáticamente, es de gran utilidad pero se debe tener mucha atención ya que se pueden realizar cambios no deseados si no es desactivada cuando no se la usa.



Título: Animación (autosave keys, timeline) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Después de crear las diferentes acciones para cada uno de los personajes como en el caso de Leroy en la escena 4 son: Caminado, estado de pie, asombro y queja cada una de estas acciones son almacenadas en la librería de acciones de Blender, las mismas que se pueden editar deacuerdo a las necesidades del animador, estas se las puede encontrar en la ventana del Action editor ahora tambié llamada Dope sheet.



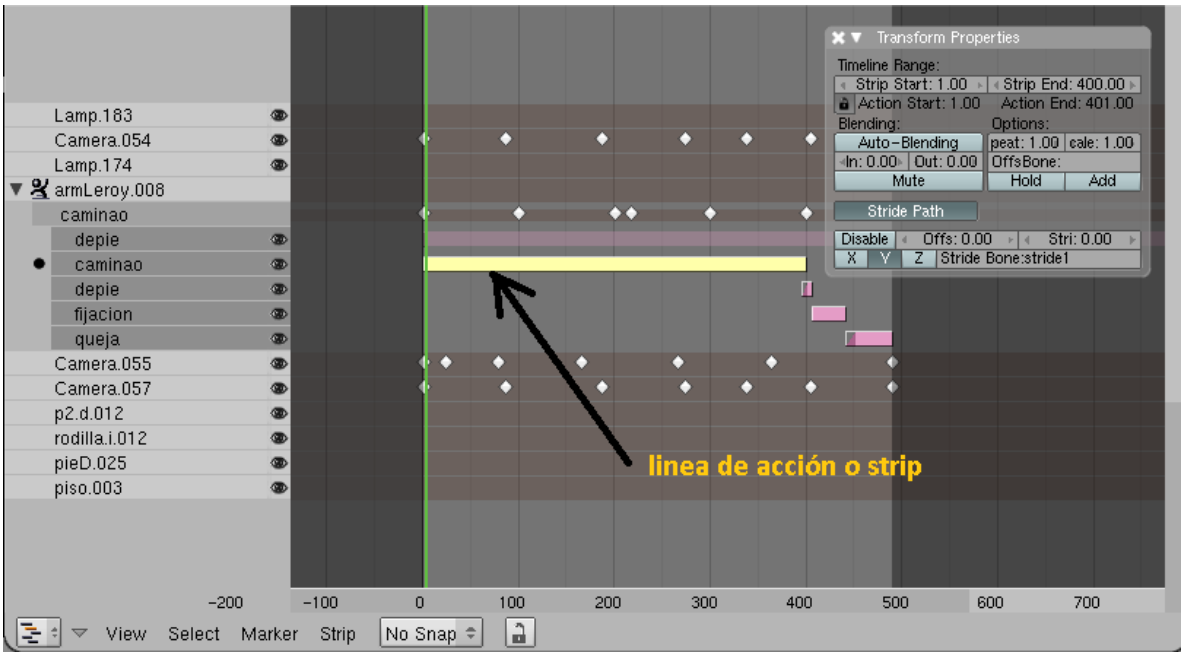
Título: Animación (Acciones predefinidas) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3.2.8.2 EDITOR NLA.

El editor NLA es un editor de video no lineal, esto quiere decir que no necesariamente se tiene que animar una acción del personaje en una sola línea de tiempo, sino que se han creado las acciones por separado y en el editor se adjuntan las líneas de acción o strip.

En esta escena se utilizó estas cuatro acciones y haciendo uso del Editor NLA (Editor de video No Lineal) se incluyo las cuatro acciones en el editor y se las dispuso en éste, de acuerdo al orden en que van sucediendo dichas acciones.

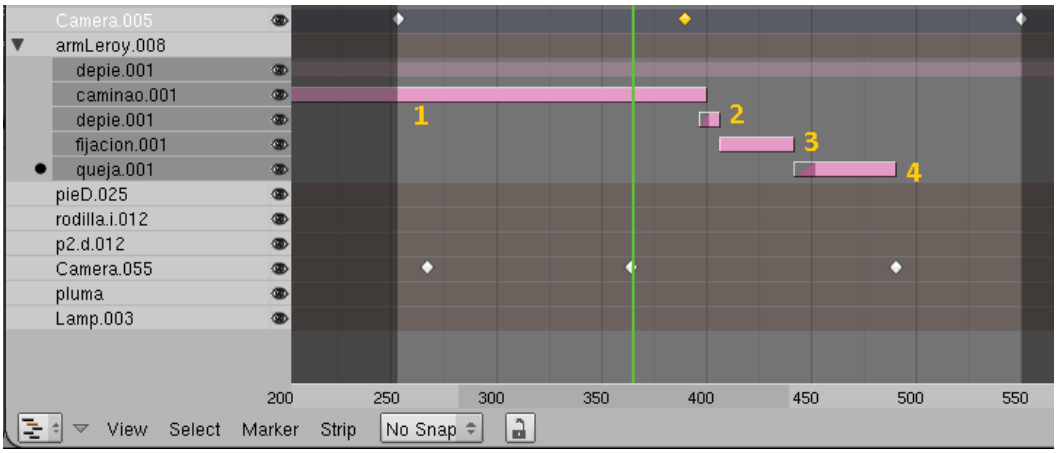
Cuando se realiza una animación desde el editor NLA, se puede incluir variar acciones que pueden interactuar , en la line de tiempo, por ejemplo en la escena 4 tenemos que Leroy camina, después se queda parado y finalmente después de ver el letrero en el camino tiene la acción de estar molesto.



Título: Animación (Editor NLA) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En el editor NLA podemos agregar varias acciones para la realización de la escena con la combinación de teclas Shift+A, las mismas que pueden ser trasladadas presionando la tecla G, como se ha visto a lo largo del proyecto los mismos comandos como Traslación, rotación y escalado son usados en todas las ventanas con casi la misma utilidad.

Como se aprecia se agregaron para la armadura Leroy, las 4 acciones de la escena.



Título: Animación (Editor NLA) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Las acciones fueron puestas en la línea de tiempo en nuestra escena, en el orden que va ocurriendo.

Estas acciones pueden ser tratadas en la línea de tiempo, para que sean usadas las veces que sean necesarias y en otros casos por el tiempo que uno desee.

En el caso de ciclo de caminado, se realiza las poses para construir un ciclo completo de caminado, para después usar la ventana de propiedades "Transform Properties"que es accionada mediante la tecla "N" en el editor NLA,



Título: Animación (Editor NLA) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En esta ventana podemos realizar varios seteos, como el rango de tiempo en el que la acción estará presente durante la línea de tiempo, también la opción para efectuar una repetición de la acción, como se mencionó anteriormente, se creó un ciclo completo de caminado para después con esta opción repetirla como sea necesario, a más repeticiones más ciclos y por ende mas pasos.

Otra opción puede ser en el caso de la acción de estar de pie en este caso la acción requiere de una sola pose, la misma que esta a lo largo de unos segundos, eso traducido a frames son los segundos x 24 frames. Para que esa acción se de a lo largo de la línea de tiempo, después de incluirla en el editor NLA como un Strip, a esta se la alarga por el tiempo que anteriormente calculamos es decir simplemente la escalamos usando la tecla "s".

Otra función importante es el Blending que se lo usa para realizar transiciones entre acciones, es decir para que los cambios entre una acción en la que el personaje está caminando y una acción en donde este está parado no sea abrupta sino que sea una transición fluida.

Esos son los principales seteos que se realizan con la ventana de propiedades del Editor NLA.

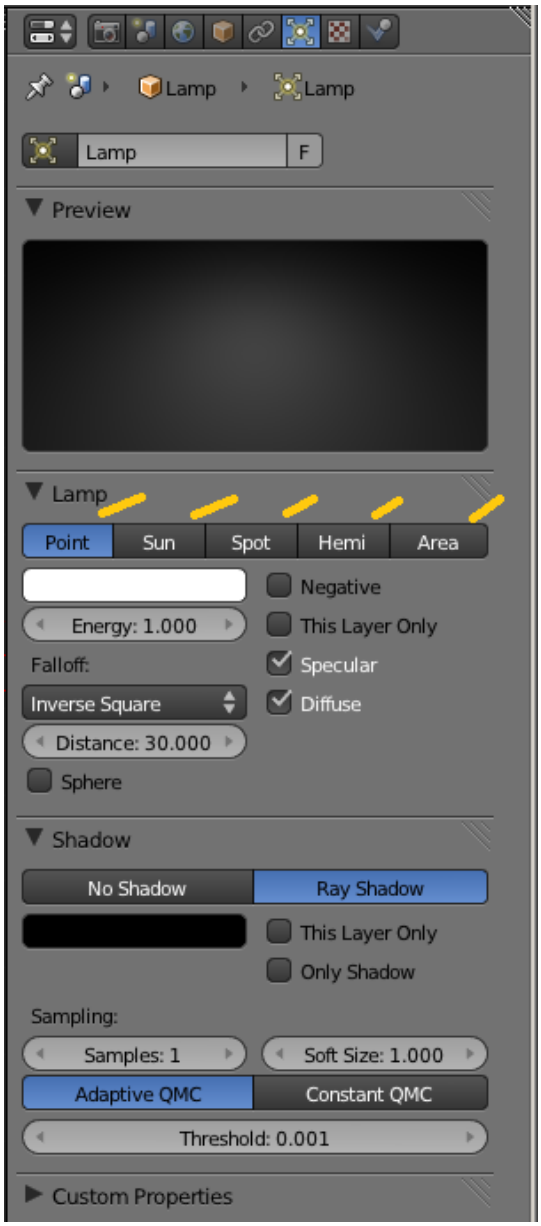
De esta forma se ha construido la animación de todas las escenas del cortometraje, haciendo uso de las herramientas Action Edition, TimeLine, 3D View y el Editor NLA, cada archivo de cada escena de igual forma es guardado en la carpeta local "Scenes" del proyecto para después actualizar las carpetas en nuestro repositorio.

3.3 POST-PRODUCCION

3.3.1 ILUMINACION

De acuerdo como se tiene la historia y el guión, la historia es desarrollada entrada la noche por lo que la iluminación que se eligió es en tonos azules, El tinte azul está presente para dar la ilusión de noche.

Para esto Blender tiene cinco tipos de luces entre los que tenemos:



Título: Iluminación (ventana propiedades LUCES) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En la imagen tenemos señalado varios tipos de luces como son: Point, Sun, Spot, Hemi, Area, cada una de estas tiene su aplicación especial.

Como se puede observar en la imagen, en estos tipos de luces podemos configurar el color que emiten, además de su posición, dirección y la energía y falloff o limite para la caída de iluminación.

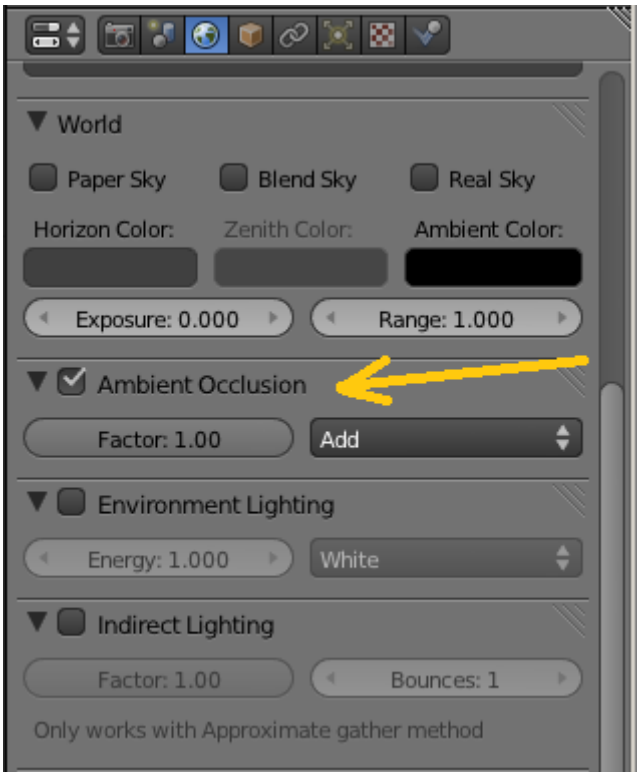
Para el proyecto Leroy se aplicaron varias de estas Luces para iluminar las escenas en especial las luces Point, spot y Sun.

Además se utilizó el AO Ambient Oclussion, que es un sofisticado truco ambiental, es método de sombreado o shading que se utiliza en 3D para darle realismo a la reflexión de los modelos.

La oclusión ambiental es una técnica para producir sombras rápidas, suaves y difusas en espacios abiertos mediante el uso de raytracing. Esta técnica es útil cuando se desea un efecto de pseudo-iluminación global sin los largos tiempo de renderización que requiere la iluminación global (GI). La oclusión ambiental es el más habitual cuando una escena tiene luz uniforme. Efecto similar al utilizando por una luz de entorno (environment light), en el caso de Leroy se requirió de una iluminación global y casi uniforme.

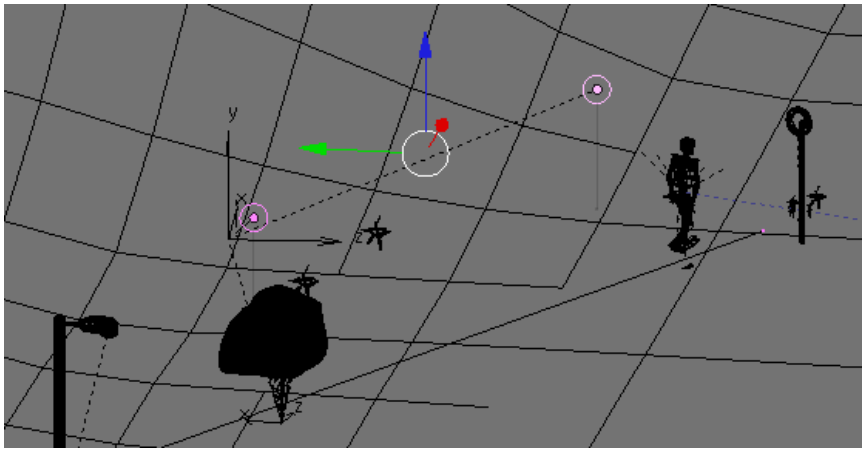
Para la escena que se ha venido desarrollando el proceso de producción, se ha dispuesto los siguientes tipos de luces.

1. Se utilizó AO en esta escena dado que es desarrollada en exteriores y se necesito de esta para producir una iluminación global.



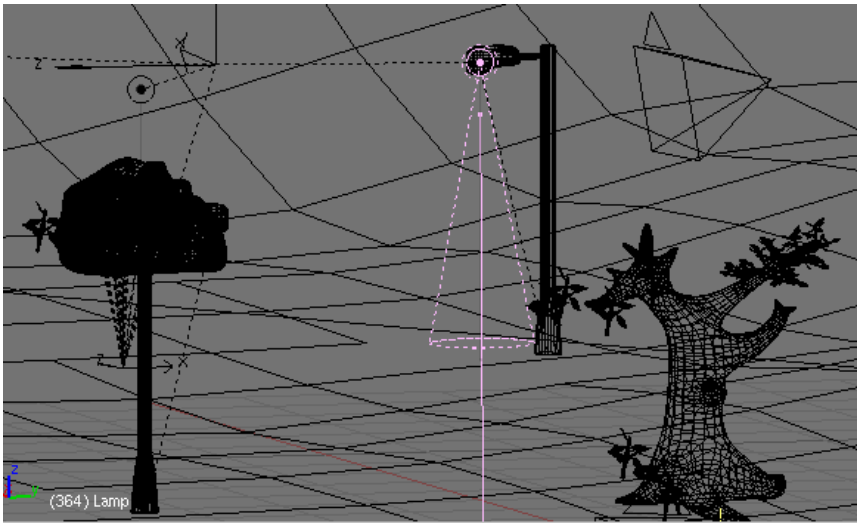
Título: Iluminación (sistema AO) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

2. Luces tipo Point que fueron ubicadas en varios sitios puntuales que debían ser resaltados con más claridad.



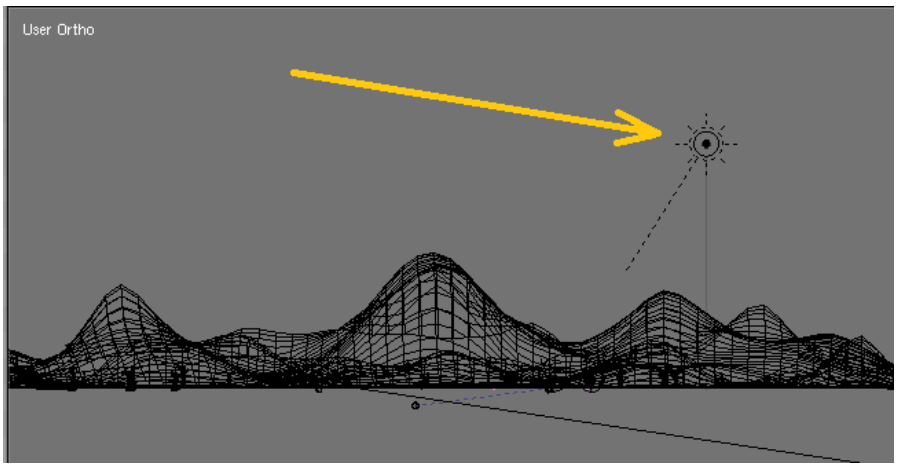
Título: Iluminación (lámparas Point) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3. Luces tipo Spot fueron colocadas en los vehículos como en los postes de la carretera.



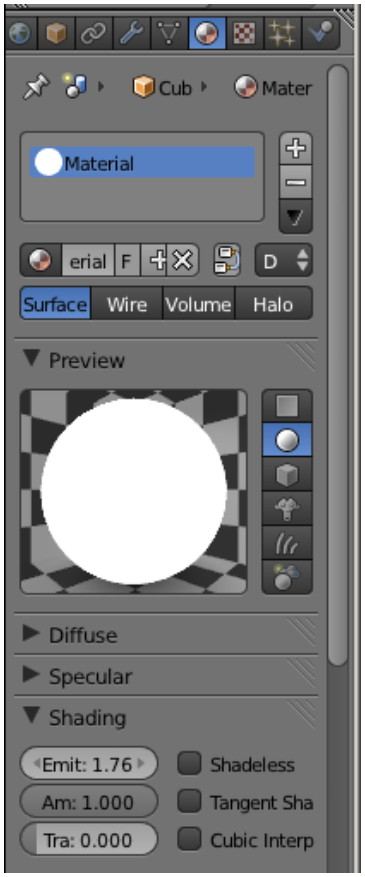
Título: Iluminación (lámparas spot) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

4. La luz principal es una tipo sun, que emite una fuente de luz constante, es decir no se degrada a la distancia y en un sola dirección de esa forma se puede dar las características de una iluminación en la mañana como en la tarde, además de tener todas las propiedades de las que se habló en las propiedades de las luces. Normalmente se utiliza como fuente de luz principal en escenas de exterior.



Título: Iluminación (lámparas sun) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Varios materiales fueron usados con propiedades de emitencia "emit" en varias escenas, esto fue dado que estas propiedades simulan un tipo de iluminación sin necesidad de ser un objeto tipo lámpara, es muy útil al momento de simular focos. Que no se encuentran en primer plano.



Título: Iluminación (ventana propiedades LUCES) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Dado que el cortometraje fue desarrollado en la misma locación la iluminación en todas las escenas es la misma.

En esta etapa final del desarrollo de este proyecto se realizara la fase de iluminación, el proceso de render de cada escena, la corrección de color de éstas y el montaje final de todas las escenas para terminar el cortometraje.

Realizado ya todo el proceso de producción tenemos las escenas cada una ya animada, los personajes listos con sus respectivos materiales y texturas, así como también se trabajo en todos los props necesarios para cada escena y se trabajo en el escenario que es único en este cortometraje, cada escena fue animada en las que se utilizaron archivos a través de un link y hacer uso de los mismos para su uso en la animación de cada escena como fase final de la producción se usó los diferentes tipos de luces y ambientes para completar el trabajo de cada escena.

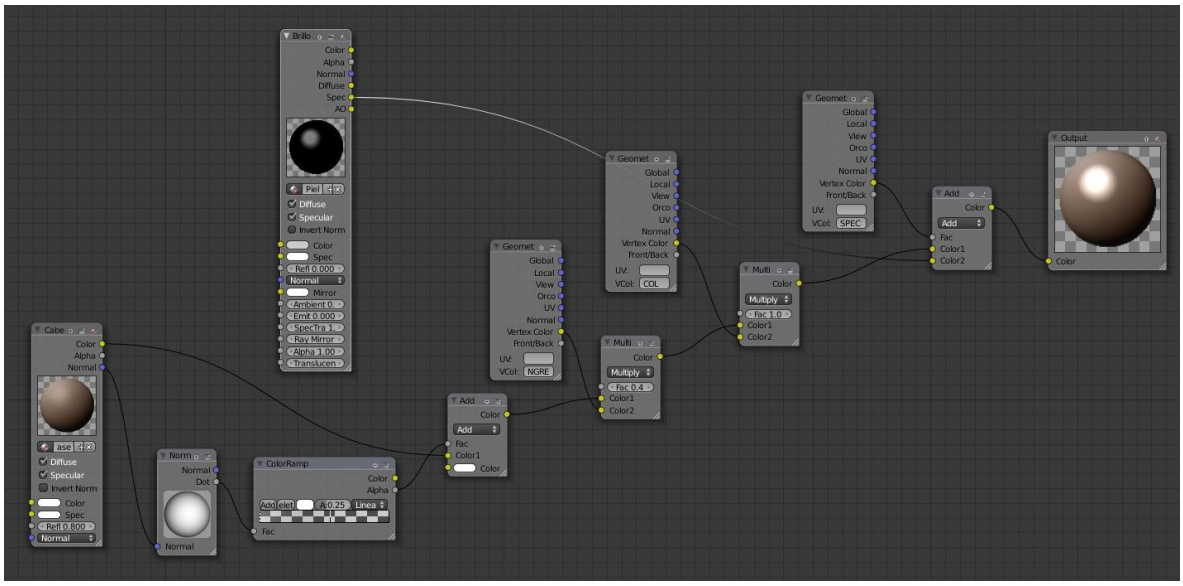
Teniendo lista esta fase de producción el siguiente paso es la composición por nodos de cada escena para dar paso a renderizarlas.

3.3.2 COMPOSICION POR NODOS.

El compositor es una manera de añadir efectos, hacer ajustes en los elementos en Blender, además sirve para crear una imagen más brillante. Realizar una degradación del color, desenfoque de movimiento, distorsión de la lente y la profundidad de campo, son sólo algunas de las cosas que se puede realizar con el compositor.

El compositor trabaja en un sistema basado en nodos, lo que significa que el aspecto final de la imagen se controla mediante la conexión de las cajas que se unen con líneas onduladas.

La mejor manera de entender la composición es de imaginarla como una gran cinta transportadora en donde se pueden ir agregando mas componentes cada uno crea un nuevo efecto, la combinación de estos componentes o nodos pueden lograr la composición deseada, como se muestra en la siguiente imagen.



Título: Composición (ventana sistema NODOS) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

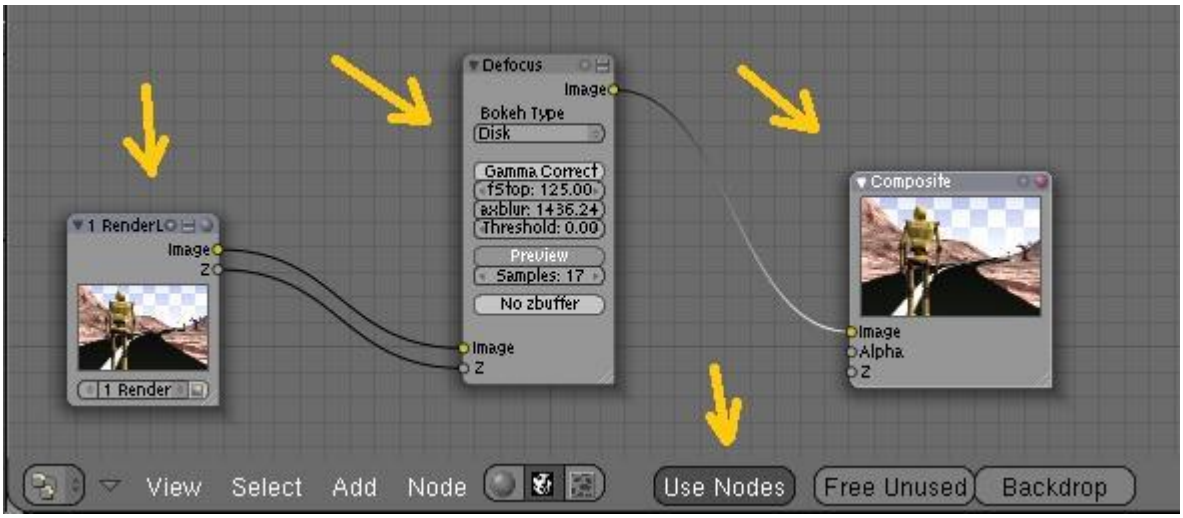
La técnica llamada Defocus en la cual los elementos que se muestran en el primer plano se muestran con más nitidez y los elementos que quedan en segundo plano se muestran desenfocados para dar una sensación de profundidad como en la imagen a continuación:



Título: Composición (sistema DoF)
Elaboración: Juan Carlos Reascos Laverdi.

Esta técnica de Defocus se puede aplicar a Blender usando el sistema de Nodos, ésta es uno de los tipos de nodos que tiene blender, se lo realizó en el cortometraje dado que la Locación fue en exteriores y fue necesaria para que se tenga un sensación de profundidad.

Para configurar este sistema de nodos es necesario activar la ventana para poder manejar este sistema y en ésta incluir los siguientes nodos.



Título: Composición (ventana sistema Nodos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

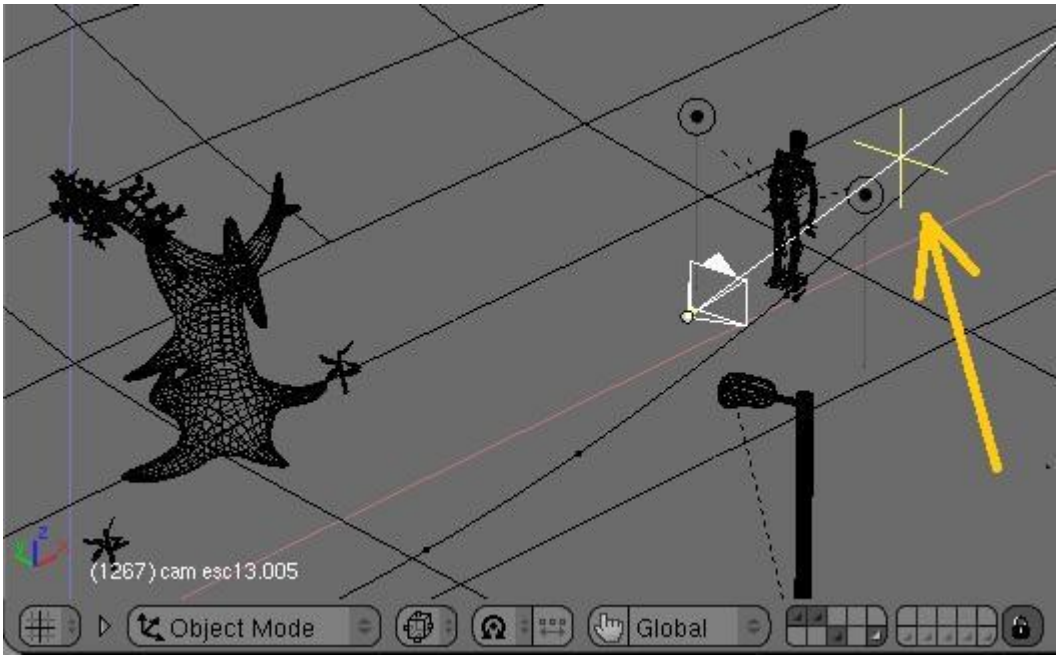
En este sistema de Nodos se ha incluido 3 tipos de nodos, el nodo de la izquierda representa la fuente de donde blender toma la imagen a procesar, el nodo central es el nodo que contiene el efecto Defocus, en este nodo se configuró el tipo de degradado que se tendrá con los elementos que están en segundo plano el tipo de opacidad o blur, la intensidad y la calidad que tendrá la imagen, Por último se tiene el Nodo llamado Defocus que es el resultado del sistema de nodos utilizado.

Además se debe configurar la cámara activa.



Título: Composición (ventana configuración sistema NODOS) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

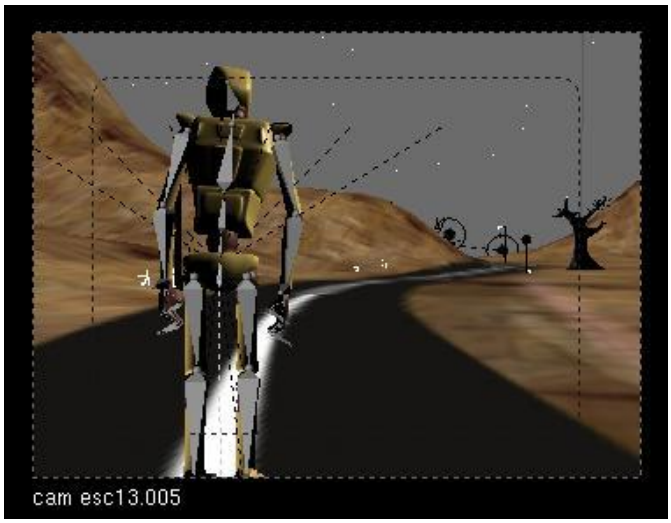
La cámara en la que se tiene este sistema de nodos es casi como una cámara normal sino que a esta se incluye un setting llamado DoF que es la distancia que la cámara toma como límite, es decir los elementos son imágenes nítidas, para empezar un degradado o blur de las imágenes que se verán en segundo plano.



Título: Composición (configuración cámara sistema de nodos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

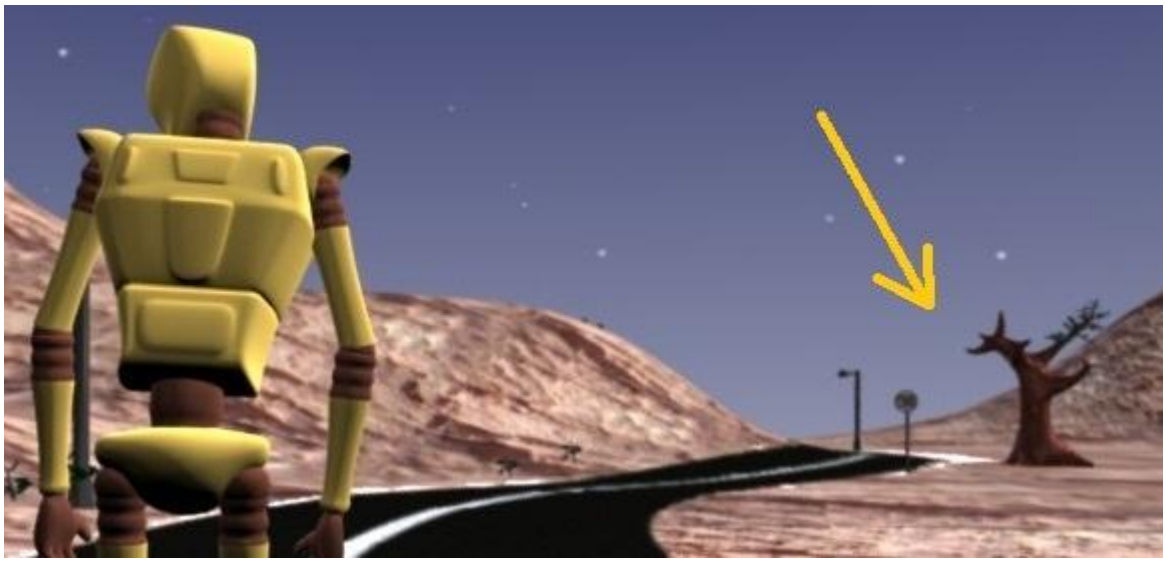
El valor de Dof está representado por la cruz en la imagen, por lo que todos los elementos posteriores a ese símbolo serán las que están en segundo plano.

En el frame de prueba tenemos un plano en donde Leroy está de espaldas con un plano abierto de la carretera, aquí se puede mostrar el uso de sistema de nodos con Defocus en las siguientes imágenes.

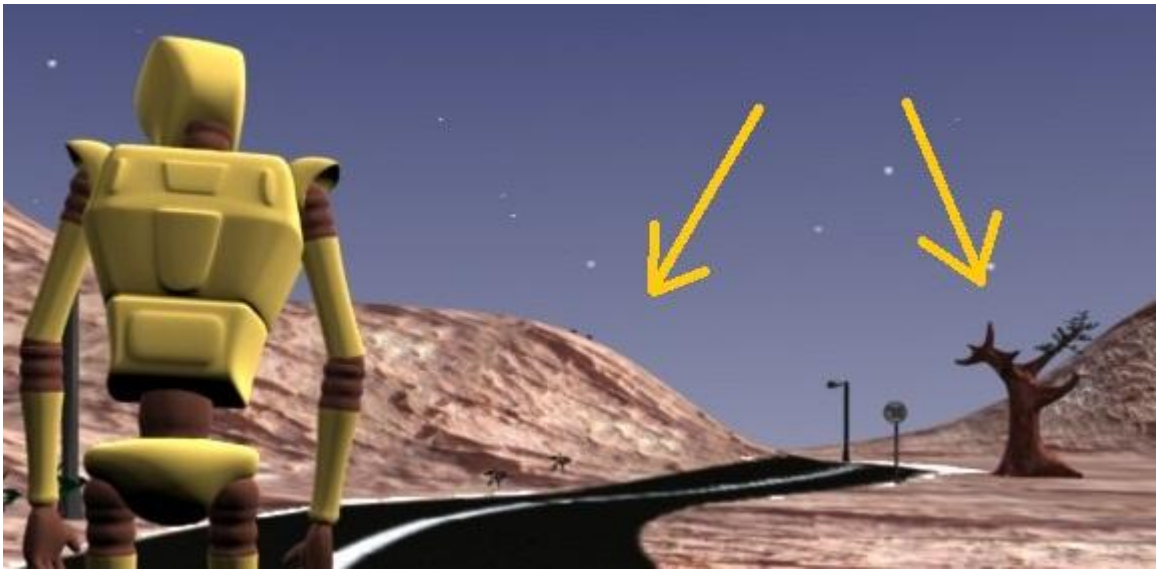


Título: Composición (configuración cámara sistema de nodos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En las imágenes a continuación tenemos la misma escena a la izquerda está la imagen sin la aplicación del sistema de nodos y a la derecha tenemos la imagen con el sistema de nodos.



Título: Composición (configuración cámara SIN sistema de nodos) - Leroy.



Título: Composición (configuración cámara CON sistema de nodos) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

3.3.3 HARDWARE Y RENDER

Los requerimientos de Hardware para la nueva versión de Blender son los siguientes:

Operating Systems

- Windows XP, Vista or 7.
- Mac OS X 10.5 o superior.
- Linux (existen distribuciones preinstalado Blender como Ubuntu studio, etc)
- FreeBSD

Mínimos requerimientos de Hardware

- 1 GHZ Single Core CPU
- 512 MB RAM
- 1024 x 768 px Display with 16 bit color

- 3 Button Mouse
- Open GL Graphics Card with 64 MB RAM

Requerimientos óptimos de Hardware

- 2 GHZ Dual Core CPU
- 2 GB RAM
- 1920 x 1200 px Display with 24 bit color
- 3 Button Mouse
- Open GL Graphics Card with 256 or 512 MB RAM

Production specs for Hardware (producción y render)

- 64 bits, Multi Core CPU
- 8-16 GB RAM
- Two times 1920 x 1200 px Display with 24 bit color
- 3 Button Mouse + tablet
- Open GL Graphics Card with 1 GB RAM, ATI FireGL or Nvidia Quadro

Blender se ejecuta en todas las tarjetas compatibles con Open GL, aunque hay algunos problemas con las tarjetas baratas integradas(Intel, Via). Por lo tanto, oficialmente no puede soportar del todo y garantizar que Blender funcione bien en esos sistemas.

http://wiki.blender.org/index.php/Doc:ES/Manual/Introduction/Installing_Blender

Para la realización de este proyecto se utilizaron los siguientes equipos

1. **PC Desktop**

Características

CPU: AMD ATHLON 64 X2 dual core Processor 4600,

RAM: 1,00 GB.

GPU: Nvidia GeForce 7300 SE/7200 GS.

2. **PC Laptop**

Características

CPU: Pentium Dual core CPU 4200 , 2,00Ghz

RAM: 1,00 GB.

GPU: Mobile Intel(R) 4 Series Express Chipset Family.

En esta PC se tuvo problemas ya que la tarjeta de video es integrada.

3.3.3.1 OpenGL y Blender

OpenGL se usa en Blender para todo el dibujado, incluyendo menús y botones. Esta dependencia hace que la instalación, en el sistema, de la librería OpenGL sea necesaria. No todos los fabricantes de tarjetas gráficas 3D dan soporte a OpenGL, especialmente para las tarjetas más baratas destinadas al mercado de videojuegos.

3.3.4 RENDIMIENTO

El rendimiento de los computadoras durante la producción y post-producción de este cortometraje fue exigente, pese a que la realización de este proyecto no se utilizó ninguna otra locación y los personajes fueron realizados lo más livianos posibles en cuanto a vértices, los modelos más complejos y que contenían mayor cantidad de vértices fueron los props, es decir las naves espaciales de los personajes, llegando a tener en una escena un promedio de 500.000,00 vértices por modelo, teniendo varios de estos según la escena, trabajando así la GPU y la CPU de las máquinas casi al 100%, y de esta manera se tuvo que reiniciar el trabajo de renderizado varias veces ya que hubieron renders en donde las maquinas colapsaron.

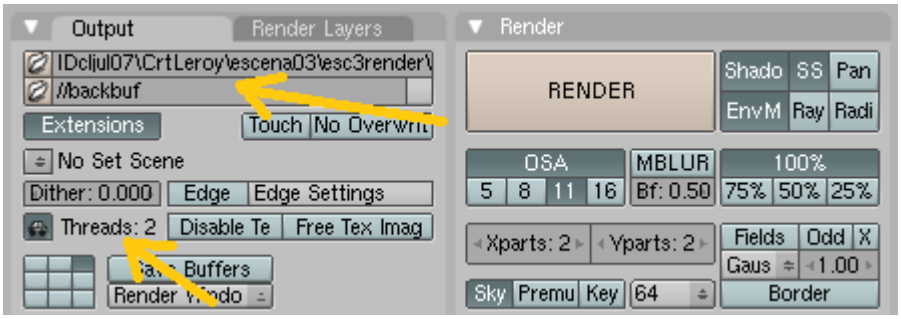
3.3.4.1 RENDER

Renderizado (render en inglés) es un término usado en informática para referirse al proceso de generar una imagen desde un modelo, más específicamente en 3D, la renderización es un proceso de cálculo complejo desarrollado por un computador destinado a generar una imagen 2D a partir de una escena 3D.

Este proceso se desarrolla con el fin de imitar un espacio 3D formado por estructuras poligonales, comportamiento de luces, texturas, materiales (agua, madera, metal, plástico, tela, etcétera) y animación, simulando ambientes y estructuras físicas verosímiles.

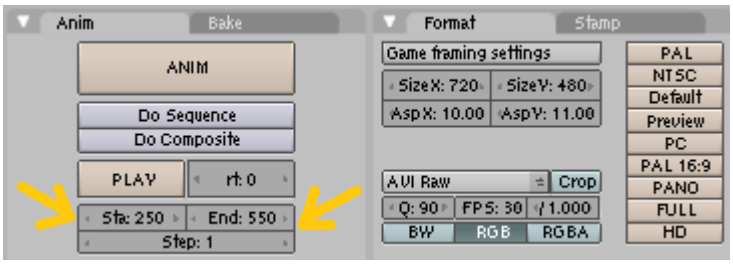
El proceso de render en el proyecto Leroy fue tratado de varias maneras a manera de experimento. En varias escenas se realizo este proceso

Para realizar el proceso de render se tiene en la ventana propiedades de Blender los diferentes seteos, en donde se puede definir el directorio en donde van a alojarse los archivos del render de la escena, en la misma ventana se puede observar el número de núcleos que posee el equipo en donde se está trabajando ese momento



Título: Render (ventana configuración RENDER) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

En este proyecto se utilizaron los 2 equipos para este procedimiento, con el método de renderizado par e impar para cada escena, es decir la misma escena fue cargada en cada computador:



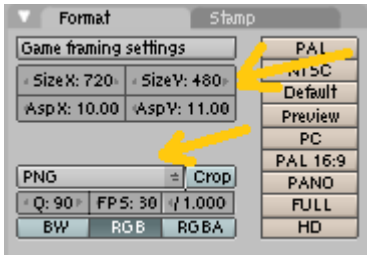
Título: Render (ventana configuración RENDER) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Como se muestra en la imagen en la escena que tenemos de demostración, la animación va a empezar en el frame 250 y terminar en la 550 con un step de 1 frame, eso quiere decir que realizará el render secuencialmente de 1 en 1 frame. Para trabajar con los 2 equipos se puede dosificar el poder de computo para el renderizaje haciendo que tanto un equipo realice el trabajo con los frames pares y el otro equipo con los frames impares.

En el caso de los pares empezando desde el frame 250 hasta el 550 con un step de 2, y para el caso de los impares empezando en el frame 251 hasta el 550 con un step de 2.

El tamaño que se utilizó para realizar el render en cada frame fue de 720 x 480 que es la resolución nativa de los DVD en el formato NTSC y el formato de la imagen que se obtenía fue PNG ya que la información en este tipo de archivo es suficiente para la su futura tratamiento y este formato es libre.

Se realiza este tipo de proceso para generar archivos de imagen en este caso PNG en carpetas definidas, para que en caso de que la máquina colapsara no se tenga que volver a empezar a renderizar desde el primer frame sino que se utilizaría el intervalo de frames que quedaron pendientes, dado que el render hasta el final es muy traicionero.



Título: Render (ventana configuración RENDER) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

El tiempo de renderización de cada frame sin el tratamiento posterior del sistema de nodos fue de 7 minutos promedio y con el sistema de nodos llegó a los 10 minutos.

Desde que los 'renders' se hacen en Blender con el motor de renderizado "internal render" utilizando la memoria principal y el procesador del computador, una tarjeta gráfica con aceleración por hardware no supone ninguna diferencia en el tiempo de renderizado, esta supone un mejor performance en la producción en si de un proyecto.

Ahora se está perfeccionando un nuevo sistema de render en blender se llama Cycles en donde trabajarían a la par CPU y GPU, este es un sistema de renderizado en tiempo real podemos obtener resultados al momento de realizar los diferentes cambios en las materiales de los objetos y en la iluminación dispuesta ese momento.

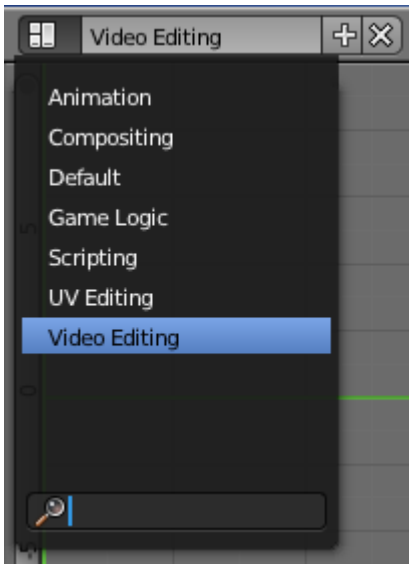
Los renderizadores externos son raytracers especializados, tenemos varios que están disponibles como software libre estos son: Yafaray, LuxRender y Aqsis Renderer, los mismos que son muy utilizados para fotorealismo y producir stills de animación o publicidad, no son muy usados para render en animación ya que la capacidad de computo que se debe tener es muy alta, por ejemplo, un frame que se realizó con Yafaray para Leroy usando las características de iluminación, materiales transparentes, translúcidos y reflectivos del personaje como de su nave tardo 1 hora 15 minutos en renderizar.

3.3.4.2 Administración y Editor de Video.

Dentro de la carpeta Scenes (escenas en español) se alojan los directorios con los archivos generados o renderizados, en el caso de la escena de demostración entre los dos equipos se obtuvieron 300 archivos de imagen PNG para la escena, sumados todos los archivos en carpetas obtuvimos 6.076 archivos de imagen.

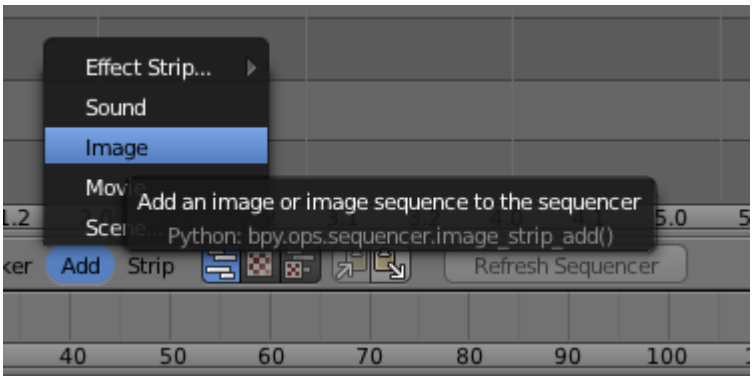
Ahora tenemos secuencias de imágenes que fueron el resultado del render por cada escena, ejemplo en la escena 4 tenemos la secuencia de imágenes que obtuvimos de los 2 computadores en donde se realizaron el proceso de render, al unir estos archivos en un directorio único tenemos la secuencia de la escena completa, tanto la del equipo de escritorio en donde se realizaron el render de los frames pares como también los archivos de imagen de los frames que se realizaron en la máquina portable.

Blender tiene entre sus fortalezas que también incorpora un editor de video en el cual ya se ha realizado el trabajo previo del animatic, pero en este caso sirve de editor para



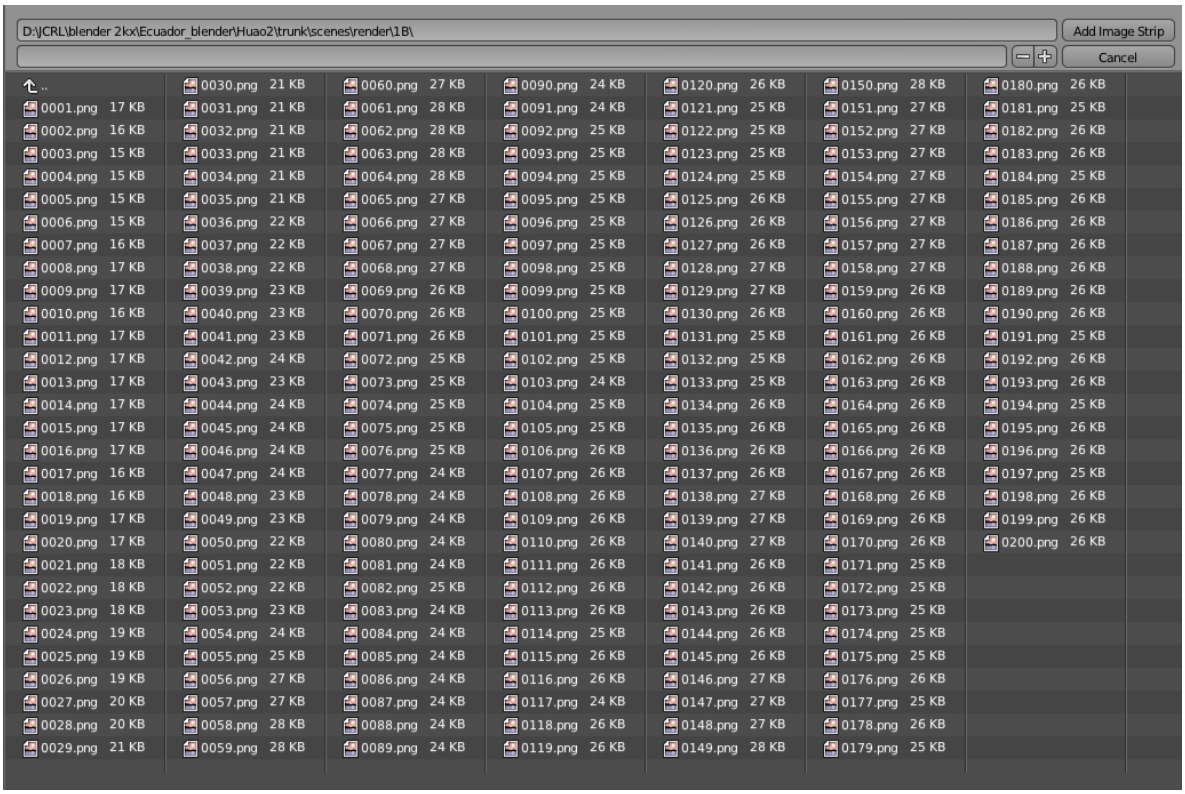
Título: Edición (ventana sistema Video Editing) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

la versión final del cortometraje, en éste se añade las imágenes de la cabecera del mismo y las imágenes generadas en cada escena, gracias a que en este editor de video se puede agregar imágenes estáticas, secuencias de imágenes, videos en las que pueden estar animaciones complementarias para el proyecto en el que se esté trabajando



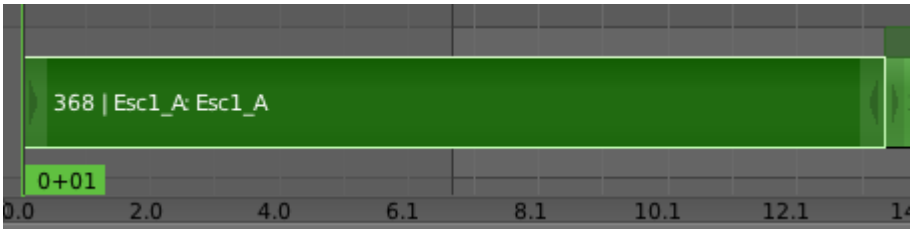
Título: Edición (ventana sistema Video Editing) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Se agregó las imágenes que están ubicadas en la carpeta Render dentro del directorio Scenes para cadu una de las escenas, en este caso se agregó las imágenes de la escena 4.



Título: Edición (ventana sistema Video Editing) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Luego de agregar la todas la imágenes de la escena son agregadas al editor de video de blender como un Strip en la línea de tiempo final.



Título: Edición (ventana sistema Video Editing) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

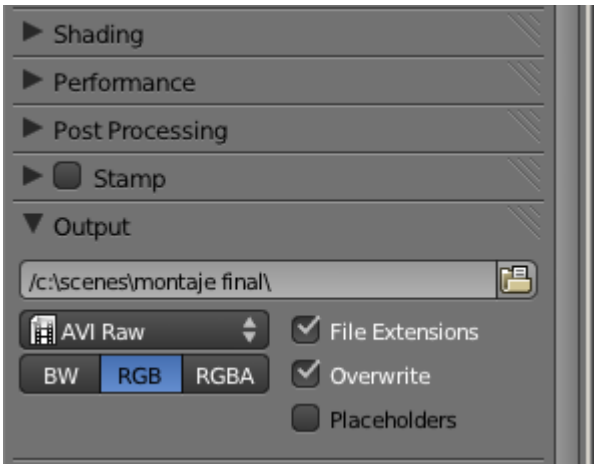
Al momento de agregar todas las escenas tenemos el montaje final, en el que tenemos el proyecto finalizado.



Título: Edición (ventana sistema Video Editing) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

Además de strip de imágenes, se agrego audio de la misma forma como se manejo la inserción de imágenes.

La exportación para el render final tendría los siguientes settings:



Título: Edición (ventana sistema Video Editing) - Leroy.
Elaboración: Juan Carlos Reascos Laverdi.

El resultado final se lo podrá apreciar en el dvd entregado adjunto a este documento.

CONCLUSIONES

En esta última fase del estudio se trata de condensar los puntos más significativos de lo tratado. La exposición se ha llevado a cabo desde una perspectiva doble: en primer término se registran los aspectos más relevantes de lo descrito a lo largo de todo el trabajo y; en segundo término las conclusiones que en estricto sentido se derivan de los hallazgos obtenidos.

Para facilitar la intelección de lo que aquí se propone, se presentan en forma de numerales:

1. La relación entre Arte y tecnología es tan importante que se han realizado contribuciones en diferentes áreas, con las múltiples aplicaciones desarrolladas que han favorecido la integración de los medios audiovisuales a la producción artística generando nuevas formas de expresivas bajo diferentes denominaciones de género: videoarte, video teatro, video danza, vídeo de autor, arte electrónico, arte digital y net art, arte multimedia, arte interactivo, arte virtual, cine animado, instalaciones y performances.

2. El desarrollo de software libre debe ser considerado como parte importante en la formación de profesionales, de esta forma se daría libertad a éstos para poder expandir sus conocimientos, que no deben estar arraigados a una corriente que vincule soluciones de un solo tipo y de esa forma se obtendría más alternativas que generen valor agregado a éstos.

3. El vertiginoso crecimiento que ha tenido herramientas como Blender 3D a lo largo de estos 2 últimos años, en donde se éste ha rediseñado casi en su totalidad su interfaz y se han desarrollado nuevos módulos, sumado a esto los nuevos proyectos desarrollados por la fundación Blender, el extenso material educativo que se ha

desarrollado en torno a éstos, su permanente presencia en el SIGGRAPH, Muestran que software libre como blender 3D tiene ya su sitial entre los programas más usados y de mejor desempeño en la creación de producciones audiovisuales y cinematográficas.

4. Los proyectos enfocados al desarrollo de software libre pueden ser auto sostenibles por ejemplo, desde la creación de la Fundación Blender como también el Instituto Blender, en los que se han desarrollado diferentes proyectos como Conferencias, el "Blender Conference" además del Google summer of code y presentaciones para la Siggraph y para el Instituto Blender tenemos a todos los proyectos relacionados con los cortometrajes y con éstos se auto financian lo que puede ser de gran ayuda para poder autogestionar un proyecto local, entre las diferentes formas de auto financiación están:

- Campañas de preventa de los DVD´s de los cortometrajes, donaciones por ejemplo via Paypal
- Ayuda Gubernamental, fondos Culturales Blender Institute es apoyado por la Unión Europea.
- Workshops, DVD de entrenamiento, seminarios, Documentación, la educación es un importante mercado para los Open Projects.
- Apoyo empresa privada.

5. Las plataformas más estables en las que se vienen realizando producciones son las basadas en Software libre. El proyecto de esta tesis en su realización se utilizaron en plataformas Windows, y Osx, demostrando que Blender 3D es estable con cualquiera de ellas.

6. La utilización de software libre para el trabajo en producciones con altas exigencias ya ha sido demostrado, el mito de que este tipo de software no es confiable, que no tiene soporte, que no tiene actualizaciones se ha quedado sin fundamentos, el uso de software libre en aplicaciones empresariales, gubernamentales, privadas, públicas y a nivel académico es un hecho, el software libre es el futuro.

7. En el desarrollo de esta tesis fue muy importante la vinculación de varias áreas, una de ellas muy importante la artística, de esta forma se pudo llegar a la estructuración de éste y que fueron una guía para su construcción. De ahí que se entendió que se debe formar equipos multidisciplinarios para poder realizar proyectos similares.

RECOMENDACIONES

Luego de la realización de este proyecto, he considerado pertinente, aportar con breves recomendaciones.

1. La vinculación del software libre como herramienta para la enseñanza no solo para desarrollo, para software de servidores y como utilidades como TIC´s sino también para desarrollo de productos multimedia.

2. Dar una apertura en el curriculum de Sistemas para su enseñanza mediante el uso de Herramientas libres, pues éstas son una alternativa que no es costosa, no depende de licenciamientos, es de acceso libre y de actualización inmediata y constante, además de fomentar su filosofía y principios.

3. Vincular trabajos de ingeniería con trabajos artísticos como se lo ha venido desarrollando con Arduino, Processing o blender3D, que son hardware y software libre.

4. Creando proyectos con estudiantes de otras carreras como arquitectura, ingeniería civil, diseño, artes visuales. En proyectos de este tipo de específicamente los de animación, requieren grupos de trabajo, ya que existen varios y muy específicos los que son dirigidos a fortalecer las destrezas y talento de cada uno de ellos, su vinculación repotenciaría proyectos de este tipo.

5. Crear clubes o promover proyectos estudiantiles como se ha visto pueden llegar a ser auto sostenibles dado ya el modelo de negocio de software libre.

BIBLIOGRAFIA

Libros electrónicos y Multimedia.

Reynish, William. ***Character Animation***. Holanda, Blender open movie workshop **2009**.

Revoy, David, ***Blend and paint***. Holanda, Blender open movie workshop **2010**.

Vazquez, Pablo, ***Venom Lab***. Holanda, Blender open movie workshop **2009**.

Pixar, ***Shortfilms Collection***. USA, **2009**

En Internet:

- Animación. **Internet**. http://es.wikipedia.org/wiki/Animaci%C3%B3n (dic 2011).

- Autodesk Maya. **Internet**. http://es.wikipedia.org/wiki/Autodesk_Maya. (dic 2011).

- Autodesk 3Dmax http://es.wikipedia.org/wiki/Autodesk_3ds_Max (dic 2011).

- Autodesk Softimage. **Internet**.

http://es.wikipedia.org/wiki/Autodesk_Softimage (dic 2011).

- Blender 3D- **Internet**. http://www.blender.org (dic 2011).

- Blender Historia. **Internet.** http://es.wikipedia.org/wiki/Blender#Historia (dic 2011)

- Blender Orange Project. **Internet**. http://orange.blender.org/ (dic 2011)

- Blender Bick Buck bunny Project. **Internet.** http://www.bigbuckbunny.org/ (dic 2011)

- Blender Manual. **Internet.** http://wiki.blender.org/index.php/Doc:ES/Manual (dic 2011)

http://www.blender.org/development/release-logs/blender-256-beta/updated-gui/

http://download.blender.org/documentation/bc2008/evolution_of_blenders_ui.pdf

http://wiki.blender.org/index.php/BlenderDev/Blender2.5/WM_Architecture

http://wiki.blender.org/index.php/BlenderDev/Blender2.5/AreaManager

- Blender YoFrankie Project. **Internet.** http://www.yofrankie.org/ (dic 2011)

- Blender Sintel Project. **Internet.** http://www.sintel.org/ (dic 2011)

- Blender Release logs. **Internet.** http://www.blender.org/development/release-logs/ (dic 2011)

- Cine de Animación. **Internet**. http://es.wikipedia.org/wiki/Cine_de_animaci%C3%B3n

- CGI. Internet. **Internet**. http://www.tech-faq.com/es/lo-que-es-cgi-animacion.html (dic 2011).

- CopyLeft. **Internet**. http://www.gnu.org/copyleft/copyleft.es.html (dic 2011).

- Creative Commons. **Internet**. http://creativecommons.org/ (dic 2011).

- Estudios Pixar. **Internet**. http://es.wikipedia.org/wiki/Pixar (dic 2011).

- Estudios Dreamworks. **Internet**. http://www.dreamworksanimation.com/ (dic 2011).

- Estudios Bluesky. **Internet**. http:// www.blueskystudios.com (dic 2011).

- Gerty. **internet**. http://68revoluciones.com/?p=72 (dic 2011).

- GNU. **Internet**. http:// www.gnu.org/ (dic 2011).

- Graphicall plumíferos. **Internet**. http://www.graphicall.org/builds/plumiblender.php (dic 2011).

http://www.graphicall.org/builds/builds/showbuild.php?action=show&id=1855

http://www.graphicall.org/builds/

- Invenciones . **Internet**. http://www.esacademic.com/dic.nsf/eswiki/729179: (dic 2011).

- Linterna mágica . **Internet**. http://books.google.com.ec (ene 2012).

- Lightwave. **Internet**. http://es.wikipedia.org/wiki/LightWave (dic 2011).

- Linux. **Internet**. http://www.gnu.org/gnu/linux-and-gnu.es.html (dic 2011).

- Marionette. **Internet**.

http://www.worldlingo.com/ma/enwiki/es/Marionette_%28software%29. (dic 2011).

- Perspectiva. **Internet**.

http://www.infor.uva.es/~descuder/proyectos/animacion/perspect.htm (dic 2011).

http://usa.autodesk.com/adsk/servlet/pc/index?siteID=123112&id=13571168 (dic 2011)

- Python & Blender. Internet.

http://wiki.blender.org/index.php/Release_Notes/Notes246/Python_API (dic 2011)

- Plumíferos. **Internet.** http://www.plumiferos.com/ (dic 2011)

- Principios de la Animación. **Internet**.

-Synfig. **Internet**. http:// www.synfig.org (dic 2011).

- Software Pencil. **Internet**. http://www.pencil-animation.org (dic 2011).

- Walt Disney. **Internet**. http://es.wikipedia.org/wiki/Dibujo_animado. (dic 2011).

http://es.wikipedia.org/wiki/Doce_principios_%28animaci%C3%B3n%29 (dic 2011).

- wikipedia. Gene Deitch. **Internet**. http://es.wikipedia.org/wiki/Gene_Deitch: (sep 2011).