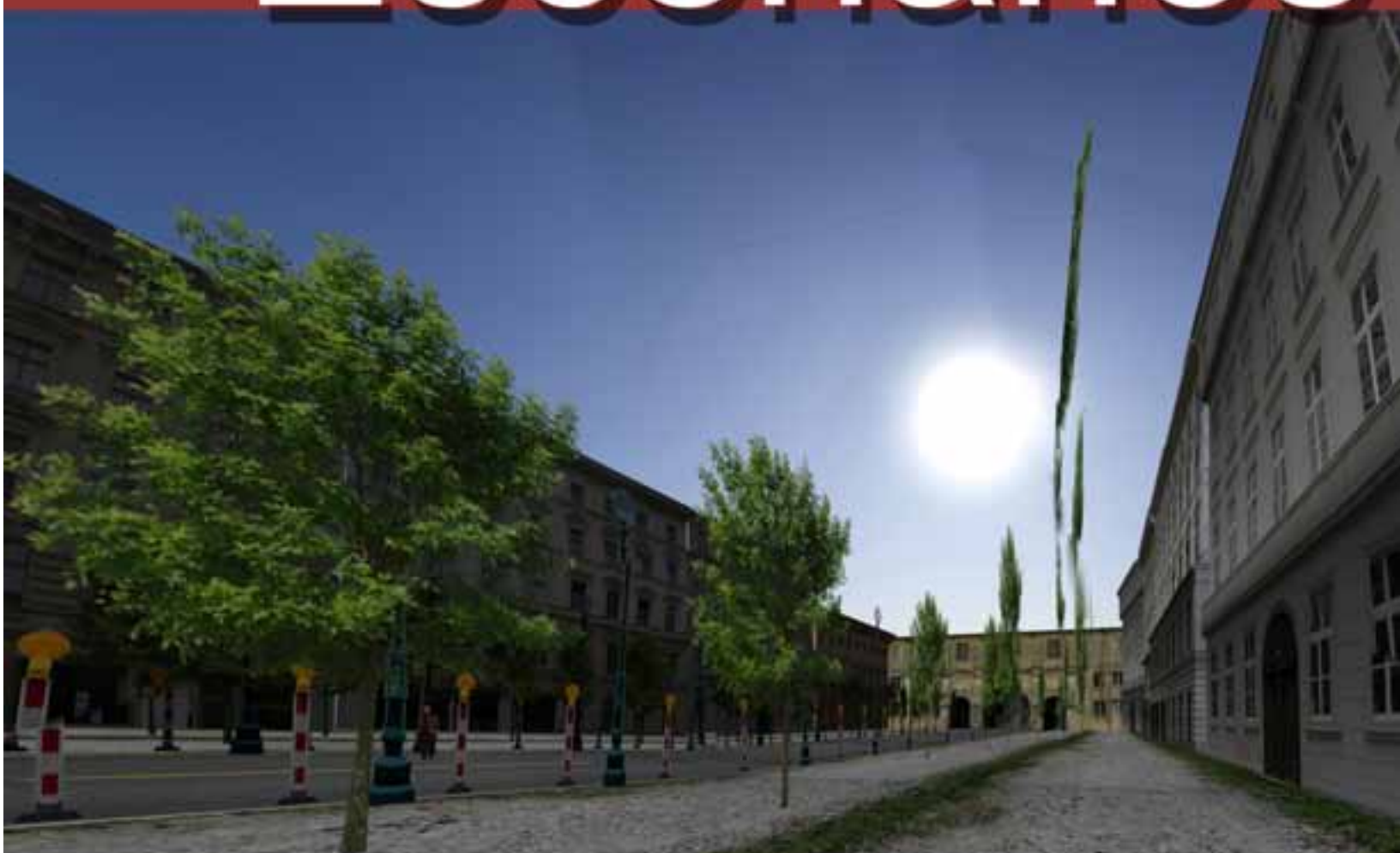


Curso Escenarios con 3D Max

# Escenarios

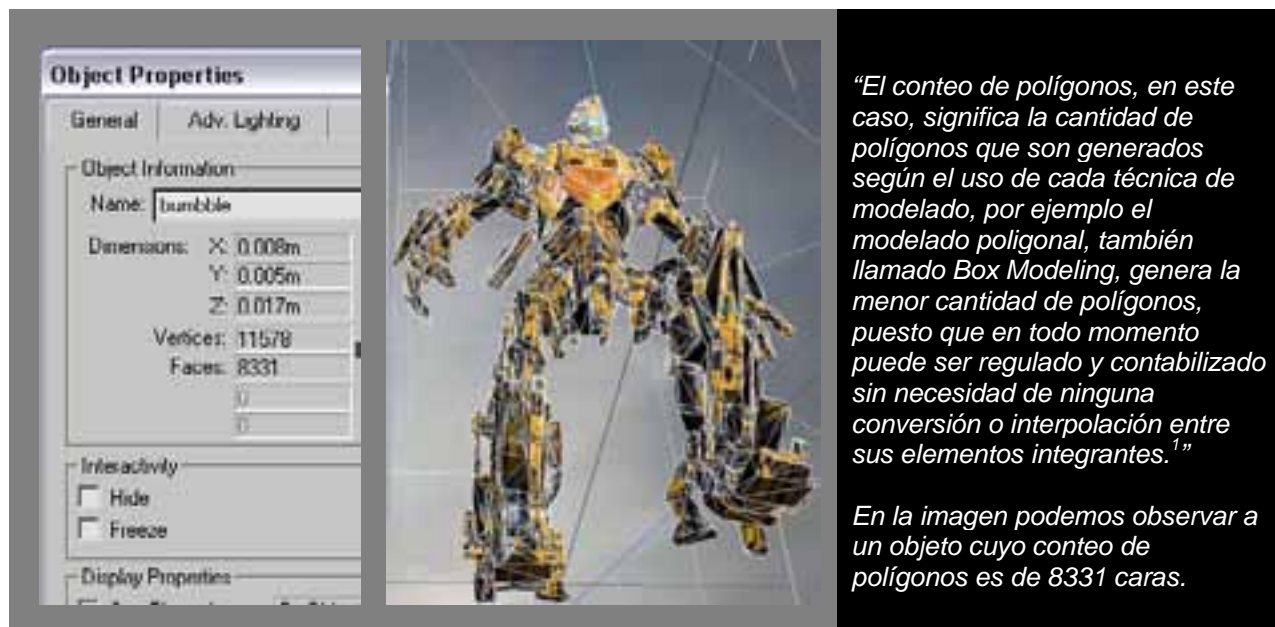


## Exportación de escenarios con 3D Studio MAX

D.C.V. Victor Hugo Franco Serrano  
Dept. de Realidad Virtual, DGSCA UNAM  
[vhfranco@unam.mx](mailto:vhfranco@unam.mx)

El objetivo de este curso es el aprendizaje de los límites y las capacidades que el alumno enfrentará al momento de generar contenido gráfico tridimensional para tiempo real, además de formar el criterio del modelador tridimensional y aprender formas de optimizar al sistema en la parte gráfica.

### 1.- Conteo de polígonos



La necesidad de control y conocimiento del conteo de polígonos de los objetos de la escena radica en la necesidad de optimización del sistema global, o en otras palabras mientras más alta sea la cantidad de polígonos de un objeto más complicado será calcularlo por el sistema.

### 2.- Importancia

*“De ella siempre dependerá la complejidad del mismo, los parámetros para definirla dependen de cada entorno pero siempre deberá ser realizada con la menor cantidad de polígonos posible.*

*Por ejemplo si el entorno es una visualización del diseño de un automóvil, entonces el automóvil deberá ser modelado con un mayor nivel de detalle, pero el entorno no, digamos un parque este en su mayoría deberá ser modelado con bajos polígonos o en caso de los árboles pueden ser sustituidos por planos auto orientables.*

*Depende de que tanta interacción se necesite con el entorno este puede prescindir de detalles innecesarios como por ejemplo postes, personas, animales, bardas o rejas, pueden ser sustituidos por*

<sup>1</sup> Modelado Tridimensional para la sala ixtli (Franco Serrano, Modelado tridimensional para la sala ixtli, 2008)

*planos, si esta labor se realiza bien cada integrante de todo el entorno podrá ser optimizado al máximo dando como resultado un entorno virtual mejor diseñado<sup>2</sup>.*"



### 3.- Identificación de las diferencia de visualizar en máquina de trabajo y en sala ixtli

Las diferencias más significativas entre la visualización en una maquina de escritorio y la sala ixtli son:

#### 3.1 – Numero de despliegues de pantalla

La mayoría de las computadoras de escritorio tienen solamente una salida de despliegue, por ende el proceso de cálculo de los gráficos de la escena se realiza solamente una vez.

El caso de la sala IXTLI difiere de la mayoría de las maquinas de trabajo pues posee 3 despliegues que se calculan independientemente y en ocasiones en conjunto.

#### 3.2 – Estereoscopia

En la mayoría de los casos las maquinas de trabajo no despliegan estéreo activo, esta cualidad de la sala IXTLI debe considerarse, pues a la larga influye en el desempeño del sistema.

<sup>2</sup> Modelado Tridimensional para la sala ixtli (Franco Serrano, Modelado tridimensional para la sala ixtli, 2008)

### 3.3 – Resolución de pantalla

Las resoluciones de pantalla de las computadoras de trabajo por lo general van desde 640 por 480 hasta 1280 por 1024 y en ocasiones un poco más alta, pero el despliegue de resolución de los juegos, programas que manejan gráficos tridimensionales y gráficos bidimensionales no manejan tanta resolución.

Un ejemplo de esto es visible en 3d max pues maneja viewports de baja resolución, también es el caso de virtools, es debido mencionar que aunque estas resoluciones pueden ser aumentadas tienden a alentar demasiado a la mayoría de los sistemas, por lo que en lo general las resoluciones de imagen con las que se trabajo no son mejoradas en su calidad.

¿Qué pasa con la sala IXTLI?, la sala IXTLI tiene 3 despliegues de pantalla, maneja estéreo activo y posee tres salidas de pantalla de resolución de 1260 por 1024 cada una con un total máximo de 3780 por 1024, aunque muchas de las aplicaciones no aprovechan al ciento por ciento estas resoluciones.

Analizando únicamente los tres elementos mencionados podemos visualizar que la sala, además de desplegar modelos 3d con sus texturas debe realizar todos estos cálculos inmediatamente, por eso como modeladores para tiempo real debemos realizar modelos y escenarios optimizados, pues, de lo contrario nuestros modelos y texturas harán al sistema más lento y difícil de ser representado.

### 3.4 – Arquitectura del hardware

Las arquitecturas de hardware varían, pero en lo general la máquina de trabajo es una máquina de escritorio con una salida de video, por lo regular posee una aceleradora grafica de buen desempeño, aunado a una buena cantidad de ram y un procesador mediano.

La sala IXTLI posee varias arquitecturas pues el despliegue a tres pantallas se basa en 3 maquinas y la diferencia fundamental es como calculan la información, por ejemplo la maquina Silicon Graphics calcula separadamente las texturas, las geometrías y la salida grafica.

### 3.5 – Software de modelado y software de visualización

La diferencia entre software de modelado y software de visualización consiste en que éste último puede agregar información programada para que los elementos del entorno interactúen entre sí, además de posibilitar el uso de programación de inteligencia artificial y controladores adicionales para el sistema, en cambio el programa de modelado por lo general no permite esto.

### 3.6 – Manejo de NURBS

Uno de los engaños más comunes al entender al IXTLI y compararlo con la máquina de trabajo es la comparación del desempeño que las aplicaciones que manejan NURBS (Rhino, MAYA, entre otros), pues estos programas son vistos en pantalla a través de software que presenta las interpolaciones suavizadas.

Esto se ve muy bien, el problema llega al tener que visualizar estos modelos en la sala IXTLI, pues deben ser convertidos de NURBS a polígonos simples de tres puntos, al realizar esta conversión nuestro conteo de polígonos se va a las nubes, pues al realizar el cálculo en polígonos las interpolaciones suavizadas deben representarse con un gran número de polígonos y no es lo más conveniente.

Es cierto que existe la posibilidad de bajar el número de interpolaciones en las curvas NURBS pero la conversión a polígonos no es controlada por el usuario resultando configuraciones de geometrías tridimensionales demasiado complejas.

### 3.7 - Desempeño de un sistema

Los sistemas que integran entornos tridimensionales texturizados con interactividad y calculados en tiempo real (es el caso de los proyectos IXTLI), poseen un cierto nivel de desempeño durante su tiempo de ejecución, ósea mientras estos corren y despliegan al entorno tridimensional en pantalla, una falla en el desempeño ocasionara que el sistema refleje síntomas entre los cuales se encuentran:

#### a – Latencia

La latencia ocurre cuando el sistema está sobrecargado de cálculos y los despliegues son mostrados con lentitud, por ejemplo si un sistema deberá correr idealmente a 60 cuadros por segundo pero el desempeño esta disminuido y solo representa 12 cuadros por segundo entonces se crea el efecto de latencia.

#### b – Mal desempeño

Un mal desempeño del sistema puede deberse a varios factores, entre los más comunes se encuentran, una mala optimización de la programación de sistema, una arquitectura de hardware inadecuada o deficiente, limitaciones internas de los programas de visualización y por último, pero muy importante para nosotros los modeladores 3d, modelos, texturas y shaders mal optimizados, inadecuados o demasiado grandes.


#### c – Falta de credibilidad

El último de los síntomas refiere a la percepción humana, siendo esta la causante de la sensación de realidad podemos entender que un sistema con un mal desempeño, aunado a un mal uso de los recursos de modelado, texturización y precálculos, generara un entorno falso y el usuario no sentirá que está inmerso en el.


## 4.- Sintetización de formas y objetos

Los objetos en la realidad poseen características complejas de forma, color y textura, pero para tiempo real es necesario tener en mente la integración de objetos modelados dentro de entornos virtuales.

Para realizar el proceso de creación de objetos debemos aceptar el hecho de que lo que vamos a producir es una representación no realista del objeto que necesitamos representar, luego entonces podremos entender el por qué de un buen manejo de la síntesis de las formas de los objetos.



Los objetos orgánicos o que posean formas orgánicas, ser humano, automóviles deportivos, arboles entre otros, tienden a poseer muchas formas curvas lo que en tiempo real debe ser convertido en polígonos, al convertirse en polígonos las formas curvas se vuelven angulares pues sus aristas y ejes son evidentes.



La síntesis geométrica y la síntesis por textura son decididas mentalmente por el modelador, puesto que antes de comenzar a modelar es posible decidir qué elementos son geométricos y que elementos son textura, por ejemplo:

Vaso	Monitor	Raqueta	Silla
			
<p>El vaso puede solucionarse geoméricamente.</p>	<p>El monitor puede solucionarse usando geometría y textura.</p>	<p>Este ejemplo puede usar geometría para la carcasa general y una textura para la red interior.</p>	<p>La silla se soluciona geoméricamente pues puede ser un elemento con algo de importancia.</p>

### 5.- Técnica de modelado más adecuada

“Otro de los criterios de selección es la elección de la técnica más conveniente para realizar el modelado, estas pueden ser por tiempo y facilidad en el trabajo de modelado de una geometría.

Por ejemplo si se necesita hacer una llanta lo más probable es que el método de modelado por modificadores generatriz – directriz en su variante de rotomoldeo sea la mejor y la más rápida elección, además de que, si es bien llevado a cabo podrá controlarse la cantidad de polígonos generados y la llanta será optima<sup>3</sup>.”

“**Tabla criterios y técnicas**<sup>4</sup> - Dentro de la tabla podemos observar que las técnicas de modelado (primera columna) se contraponen a los criterios de selección (primera fila) para lograr una decisión de uso y si es adecuada la técnica, siempre para lograr objetos tridimensionales para tiempo real.”

	Conteo de polígonos	Técnica más conveniente	Finalidad e importancia	Adaptación geométrica
Por polígono	Adecuado	Adecuado	Adecuado	Adecuado
Modificadores	Usable	Adecuado	Usable	No interviene
Clonación	Usable	Adecuado	Usable	No interviene
Metaobjetos	Alto conteo	Inadecuado	Inadecuado	Adecuado
Booleano	Usable	Adecuado	Usable	No interviene
NURBS	Alto conteo	Inadecuado	Inadecuado	Adecuado
Secciones cruzadas	Alto conteo	Adecuado	Usable	Adecuado
Fractal	Alto conteo	Inadecuado	Inadecuado	No interviene
Diferenciación dicromática	Usable	Adecuado	Usable	No interviene
Datos precisos	Alto conteo	Inadecuado	Inadecuado	No interviene
Fotografías	Usable	Adecuado	Usable	No interviene
Simulación física	Alto conteo	Adecuado	Usable	No interviene

NOTA: Esta tabla debe ser entendida como sugerencia, pues cada proyecto tiene diferentes posibilidades y necesidades de modelado.

### Optimizadores de sistema

“Las optimizaciones que un sistema puede necesitar se dividen según el rubro al que pertenecen, están las **optimizaciones geométricas**, que emplean recursos como la sustitución de geometría muy pesada por geometría ligera, también se encuentran las **optimizaciones de textura**, que emplean precálculos de la iluminación, texturas aplicadas a polígonos planos y shaders para sustituir geometría de la escena.

Por último se encuentran las **optimizaciones de cámara**, estas ayudan al desempeño del sistema al no representar en pantalla, ni calcular en el proceso de síntesis de la imagen geometrías innecesarias. Dentro de la tabla se enlistan las tres subdivisiones de optimizadores de sistema con sus respectivos elementos<sup>5</sup>”:

<sup>3</sup> Modelado Tridimensional para la sala ixtli (Franco Serrano, Modelado tridimensional para la sala ixtli, 2008)

<sup>4</sup> Modelado Tridimensional para la sala ixtli (Franco Serrano, Modelado tridimensional para la sala ixtli, 2008)

<sup>5</sup> Modelado Tridimensional para la sala ixtli (Franco Serrano, Modelado tridimensional para la sala ixtli, 2008)

Optimizadores del sistema		
Geométricas	Por medio de imágenes	Por medio de la cámara
Nivel de detalle	Billboards Precálculo de la iluminación Precálculo de los reflejos Shaders	Clipping Z-buffer Culling Eliminación de caras ocultas

Por lo general los optimizadores de cámara son usados por los programadores para optimizar el sistema, así que como modeladores para tiempo real solo debemos preocuparnos por tomar en cuenta los optimizadores geométricos y de textura.

## 6.- Optimización geométrica

Las optimizaciones geométricas pueden realizarse de varias maneras, algunas de ellas son a través de la máquina y otras se realizan por el modelador.

La optimización geométrica por distancia o nivel de detalle por distancia es usada por el sistema para reducir el cálculo de polígonos de la escena haciendo a los objetos más lejanos más ligeros al sustituirlos con objetos que los representan pero que están bastante reducidos en la cantidad de polígonos.

La optimización geométrica por importancia es controlada por el modelador, quien basado en su criterio decidirá qué elementos de la escena necesitan un mayor trabajo de detalle durante su modelado, por ejemplo si el proyecto se llamara el auto del mañana lo más importante por ende es el auto y los demás elementos de la escena pueden irse degradando en su importancia hasta llegar a ser solo planos.

Por ejemplo en la escena cerrada florentina:

Automovil	Fuente	Edificio	Arbol
			
Conteo de polígonos: 1260	Conteo de polígonos: 209	Conteo de polígonos: 19	Conteo de polígonos: 2

## 7.- Optimización de textura

Las texturas pueden ser usadas para eliminar geometría innecesaria en una escena, por lo general se emplean texturas cuyas dimensiones y resolución no sea demasiado elevada, como por ejemplo una textura de 512 por 512 pixeles con un resolución de 72 dpi, es más que suficiente para su uso en tiempo real.

Otro de los parámetros que debemos tomar en cuenta al hacer uso de texturas es su importancia, si por ejemplo tenemos que texturizar una escena en la cual se hace uso de edificios y elementos muy lejanos que nunca podremos ver a detalle entonces las texturas que podremos usar para ellos pueden descender a tamaños como 128 por 128 o 64 por 64 pero conservar su resolución a 72 dpi.

Para decidir si se debe usar la optimización por textura debemos plantearnos la importancia del objeto y a su vez su distancia en relación a la cámara, una vez contemplado esto podremos decidir si es necesaria la optimización por medio de textura.



Los precálculos<sup>6</sup> son métodos para acelerar el despliegue de efectos adicionales, como sombras, brillos o reflejos de los objetos, como su nombre lo indica son efectos que se calculan previamente y que no influyen de sobremanera en el cálculo en tiempo real.

<sup>6</sup> Precálculos (Franco Serrano, IXTLI, Observatorio de Visualización de la UNAM, 2008)

## 8.- Optimización por shaders

Según la Wikipedia<sup>7</sup> “Shader es un tipo de lenguaje de programación destinado a programar el procesado de elementos de cualquier interfaz 3D o tarjeta gráfica (píxeles67, polígonos, texels68, voxels69, etc.). Los programas generados tomarán como entrada estos elementos y los procesarán, de tal manera que modifique la forma con la que lo perciba el ojo humano.

Los shaders englobarían el conjunto de procedimientos que dotan de propiedades a un punto arista o polígono y nos permiten determinar que un cubo dibujado por pantalla será de madera, con lo que reflejará la luz de tal manera, brillará o no, será transparente o no.

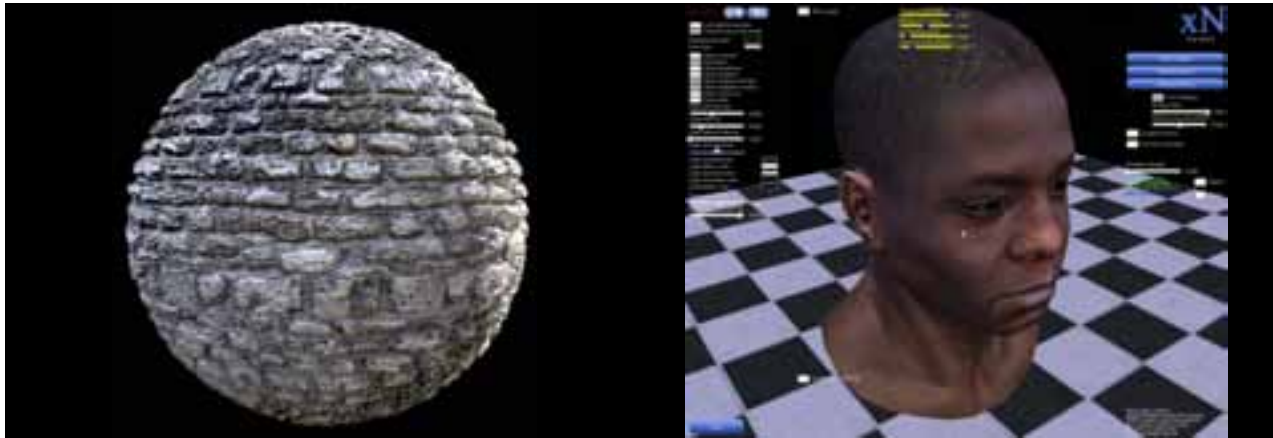
Cada vez las tarjetas gráficas son más programables lo que permite definir mejor la apariencia de objetos 3d, además de la incorporación de nuevas técnicas y fórmulas matemático-físicas. Todo esto constituye actualizaciones tan conocidas como el shader 3.0 o 2.0, que no son otra cosa que la incorporación de nuevos efectos aplicables en la definición de un objeto.

Las librerías gráficas de DirectX70 y OpenGL71 usan tres tipos de shaders:

- **Shader de vértice** afecta solo a una serie de vértices y esto alterara la forma del objeto. Los vértices calculados por este shader son comúnmente convertidos a shaders de geometría.

-**Shader de geometría** es usado para combinar una serie de vértices en un solo objeto que puede ser alterado por el shader de píxel.

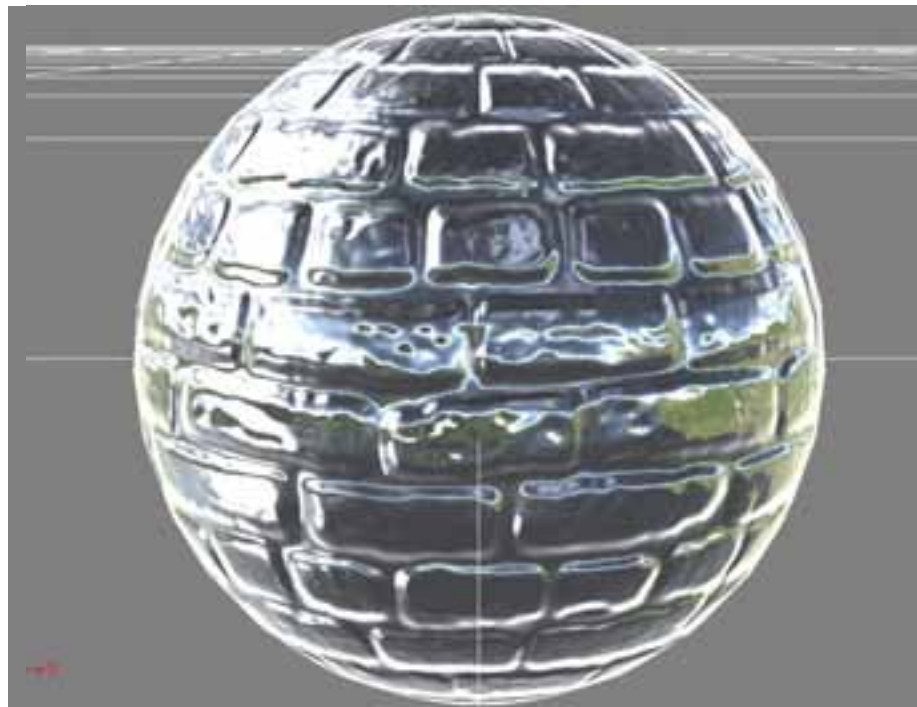
- **Shader de píxel** afecta a píxeles individuales de una geometría existente y le aplica texturas, mapas de relieve, efectos de niebla, entre otros.”



Su uso debe ser moderado pues aunque por un lado eliminan polígonos innecesarios de la escena, por el otro carga fuertemente al procesamiento gráfico realizado por la tarjeta gráfica. Como modelador tridimensional tu labor será la de planear su uso conscientemente y no engolosinarse por el posible aumento de calidad gráfica a costa del desempeño del sistema.

<sup>7</sup> Shader (Wikipedia, 2008)

Existen varios tipos de soluciones de shaders, entre ellos se encuentran:



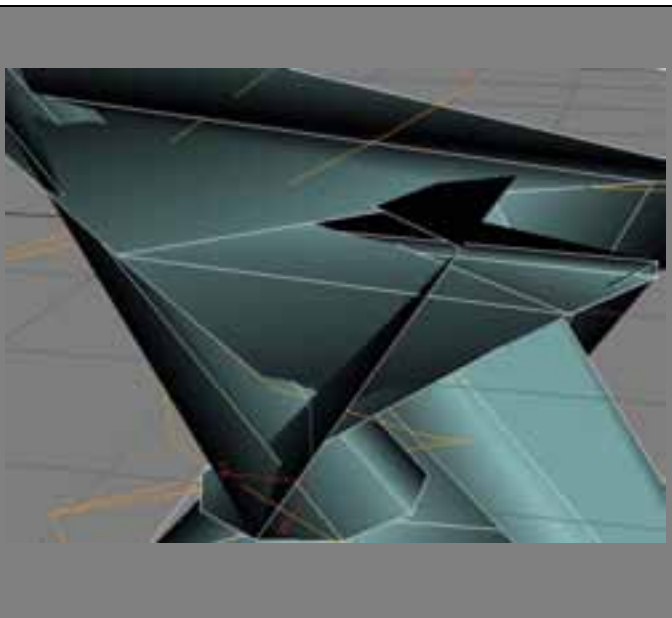
Aunque se pueden realizar una variedad bastante grande de shaders, lo más recomendable es plantear si realmente debe llevar shaders el sistema de realidad virtual.

Además de entender que cada shader debe ser programado individualmente o en algunos casos puede ser construido por medio de programación visual.

## 9.- Adaptación geométrica (animación)

“Es la necesidad de cada objeto de interactuar con los demás elementos, si al estar animado necesita una adaptación en la geometría del objeto o requiere de una entrada especial de información como un esqueleto o un sistema bípedo de 3d Studio Max, si se necesitara alguna de las anteriores entonces se deberá preparar al modelo para poder transformarse.

Puede emplearse cualquiera de las técnicas de modelado, de preferencia la que genere menos polígonos y poner especial énfasis en las zonas de articulaciones, por ejemplo los codos, muñecas, cuello, rodillas y tobillos en el modelado de un ser humano, estos segmentos deberán poder doblarse de tal forma que se evite la ruptura de los polígonos que lo integran<sup>8</sup>.”



### Bibliografía

Franco Serrano, V. H. (19 de Junio de 2008). *IXTLI, Observatorio de Visualización de la UNAM*. Recuperado el 3 de Octubre de 2008, de Iluminación Global para ambientes virtuales en tiempo real: <http://www.ixtli.unam.mx/media/tutoriales/manualgi.pdf>

Franco Serrano, V. H. (2008). *Modelado tridimensional para la sala ixtli*. DF.

Wikipedia. (1 de Octubre de 2008). *Shader - Wikipedia*. Recuperado el 2 de 10 de 2008, de <http://en.wikipedia.org/wiki/Shader>

### Índice

Exportación de escenarios con 3D Studio MAX .....	2
1.- Conteo de polígonos .....	2
2.- Importancia .....	2
3.- Identificación de las diferencia de visualizar en máquina de trabajo y en sala ixtli .....	3
3.1 – Numero de despliegues de pantalla .....	3
3.2 – Estereoscopia .....	3
3.3 – Resolución de pantalla .....	4

<sup>8</sup> Modelado Tridimensional para la sala ixtli (Franco Serrano, Modelado tridimensional para la sala ixtli, 2008)

3.4 – Arquitectura del hardware.....	4
3.5 – Software de modelado y software de visualización.....	4
3.6 – Manejo de NURBS .....	4
3.7 - Desempeño de un sistema.....	5
a – Latencia .....	5
b – Mal desempeño .....	5
c – Falta de credibilidad .....	5
4.- Sintetización de formas y objetos.....	5
5.- Técnica de modelado más adecuada.....	7
6.- Optimización geométrica.....	8
7.- Optimización de textura .....	9
8.- Optimización por shaders .....	10
9.- Adaptación geométrica (animación) .....	12
Bibliografía .....	12
Índice.....	12