

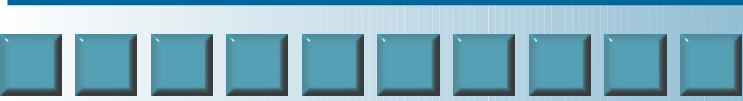
# ***Consideraciones generales para el diseño de juegos 3D***

---



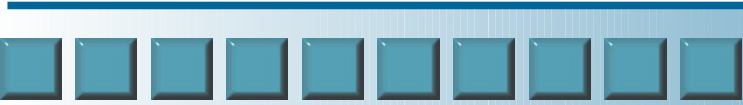
Adriana Rozo, Cesar Pachón

# *Orden del día/temas*



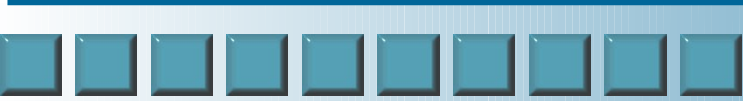
- Herramientas de programación
- Visualización 3D
- Detección de Colisiones
- Arte
- Formatos de Archivos
- Inteligencia Artificial

# *Herramientas de programación*



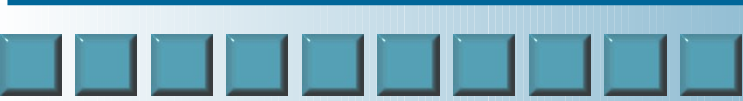
- Lenguajes de programación de propósito general (C++)
- Bibliotecas de acceso al Hardware: API's de sistema operativo. (SDL)
- API's 3D IM (Modo Inmediato) (OpenGL)
- API's 3D RM (Modo Retenido) (Lib. OBPlus)
- Engines (Cristal Space)
- Metalenguajes (VRML)

# Visualización 3D



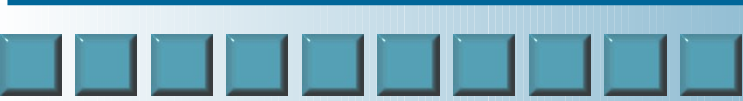
- Objetivo: “minimizar el número de polígonos enviados a través de la tubería de vista”
- Árboles BSP
- Quadtrees
- Octrees
- Casos especiales de geometrías Convexas: sectores y portales.

# *Detección de Colisiones*



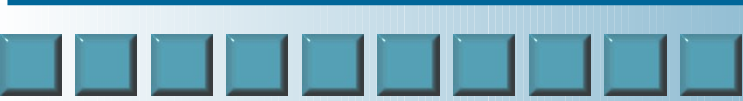
- Unir el Sistema de Detección de Colisiones a la eliminación de geometría no visible.
- Reconocer qué técnica aplica mejor al tipo de objeto: Esferas, AABB, OBB, comparación de polígonos , etc..
- Utilizar jerarquías de sistemas DC de orden de complejidad creciente (eliminación trivial)

# *Detección de Colisiones (II)*



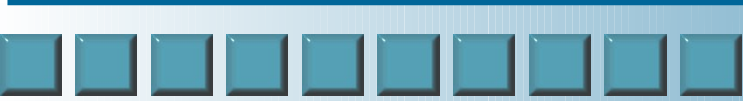
- Distinguir entre colisiones de terreno, objetos estáticos y objetos dinámicos.
- Considerar el uso de **Mallas de Navegación** como alternativa a la colisión contra escenarios estáticos.

# *Arte: Herramientas*



- Modeladores 3D: 3D Studio Max, Maya, Rhino, **Blender**...
- Editores gráficos 2D (Texturas): Corel Photo Paint, Photoshop, **Gimp**, **paint**...
- Herramientas para Animación.
- Editores de Audio, software de composición de música.

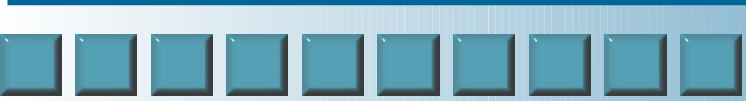
# *Arte: Modelado y Animación*



- Modelos tridimensionales de baja poligonalización.
- Utilización de mallas poligonales como objeto final. Otras técnicas sólo durante el modelado en sí.
- Conocer las características del sistema de visualización y optimizar el modelo para el.
- Sistema de animación: **jerarquías de objetos**, mallas deformables, **Keyframing**, simulación física.

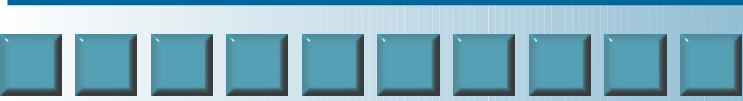


# Arte: Texturas



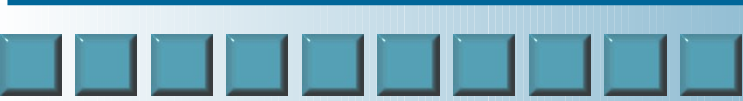
- Minimizar el tamaño de las texturas.
- Aprovechar la textura como medio para simular características no disponibles en la visualización de tiempo real: **sombreado**, efectos de animación especiales.
- Conocer las características del sistema de visualización y optimizar las texturas para el.

# *Formatos de archivos*



- Usar formatos estándar o **formatos propios**?
- Usar **formatos binarios** o **formatos ASCII**?
- Consideraciones a tener en cuenta: disponibilidad de exportadores / importadores, editores, visualizadores, módulos de carga, desempeño, privacidad de los datos.

# *Inteligencia Artificial*



- Regla: “los datos van en archivos externos, la lógica va en el código del juego”.
- Percepción del ambiente: Polling vs Paso de Mensajes.
- Técnicas: **Máquinas de Estado Finito**, Redes neuronales, lógica difusa.