

La escena completa de la aplicación es un solo grafo, como se puede ver en la Figura 1, donde cada objeto utiliza su pipeline asignado. Hay varios objetos que se mueven de manera relativa a otros, aprovechando de este modo, la modelación jerárquica.

## Solución Propuesta

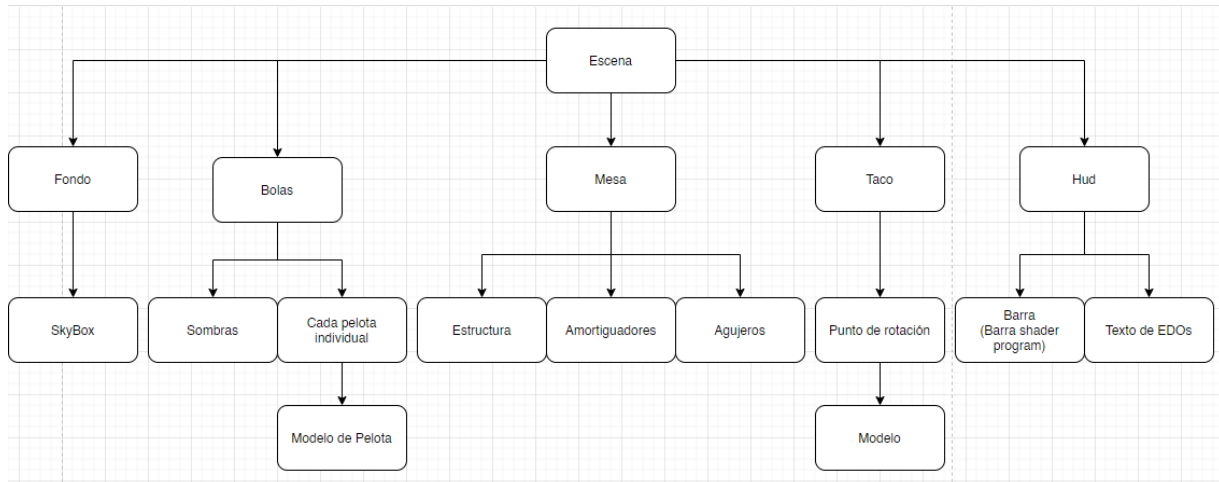


Figura 1: Grafo de la escena simplificado.

El movimiento de las bolas fue hecho en base a tres fenómenos físicos:

Simulación de roce para el desplazamiento, conservación del momento para las colisiones entre pelotas, y por último, se simuló la acción de la fuerza de gravedad para cuando las bolas caen en los bolsillos.

Hay tres vistas diferentes:

La primera es una vista aérea ortogonal que permite ver el estado de la mesa fácilmente. La segunda es una vista en primera persona para poder moverse alrededor de la mesa. La última vista te permite ver el punto de vista de la bola golpeada mientras esta esté en movimiento.

Cuando se está en vista aérea, aparece una barra que indica las unidades de fuerza (UF) con las que se golpeará a la bola objetivo. Además, aparecerán vectores debajo de todas las pelotas, indicando la dirección y la fuerza con las que serán empujadas si son golpeadas directamente en el centro.

La barra indicadora de fuerza es un conjunto de texturas que utilizan un fragment shader el cual elimina los píxeles arriba de cierta coordenada  $y$ , basada en la distancia del taco a la bola. Al golpear, la barra se mueve utilizando shearing.

La escena está rodeada por una *skybox* hecha previamente con iluminación global. Además, la mesa tiene una sombra pre-hecha aplicada en la textura del suelo y las bolas proyectan una sombra en tiempo real, la cual es una textura que se mueve, rota y escala dependiendo de su posición respecto a la luz.

Se puede elegir que método de resolución de EDOs se usará para el desplazamiento de las bolas. Las opciones son RK4, Euler, Euler Mejorado, Euler Modificado.

## Instrucciones de Ejecución

Para ejecutar el programa, se debe ingresar a la carpeta *PoolParty* desde una consola dentro de un entorno con las librerías del curso y OpenMesh. Luego, se debe escribir el comando *python pool\_party.py config.json*.

El *config.json* es el nombre del archivo de configuración con el valor del roce y coeficiente de restitución de las bolas. Se recomienda un roce de 0.01 y un coeficiente de 0.95.

Con la tecla **1**, se alterna entre la cámara aérea y la primera persona.

Con la tecla **2**, se cambia el método de resolución de Edo.

Con la tecla **3**, se puede elegir que bola golpear con el taco.

La cámara en primera persona se mueve utilizando las teclas **W** y **S** para moverse adelante y atrás, **Q** y **E** para moverse a la izquierda o a la derecha, y **A** y **D** para rotar la cámara a la izquierda o a la derecha.

Cuando todas las bolas estén detenidas se puede rotar el taco alrededor del objetivo con las teclas **Izquierda** y **Derecha**, además de que se puede elegir la fuerza alejando o acercando el taco con las teclas **Arriba** y **Abajo**. Al presionar **Espacio** se golpea la bola.

Con la tecla **4**, se alterna la opción de ponerse en la perspectiva de la bola golpeada. Esto solo funciona si hay bolas en movimiento.

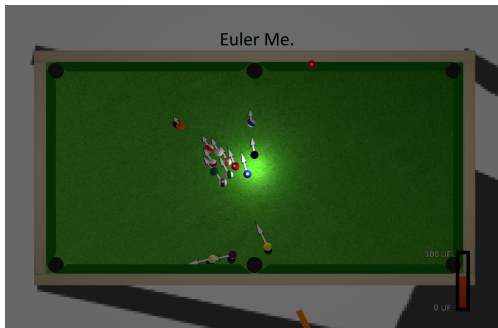
El programa se puede cerrar presionando **Escape**.

## Resultados

Al comenzar el programa el usuario se encuentra en un pedazo de hielo rodeado de icebergs y de noche. La única opción que tiene es jugar con la mesa de pool en el centro de la escena o admirar el paisaje.

El jugador puede golpear la bola que prefiera hasta haberlas encestado todas en los bolsillos (las bolas encestadas son colocadas en el borde de la mesa). También puede elegir cuanta fuerza usar en cada golpe y con que técnica de resolución de EDOs se moverán las pelotas.

El usuario puede elegir de que manera ver la mesa. En primera persona, vista aérea o desde el punto de vista de una pelota.



Vista aérea vs Vista en primera persona

## Autoevaluación

Criterio-Puntaje	0	1	2	3
Transformaciones				x
Modelación Jerárquica				x
Vistas y Proyecciones				x
Texturas				x
Iluminación Global				x
Visualización Científica				x
Diferencias Finitas				x
Colisiones y Física				x
Funcionalidades mecánicas o lógica de juego				x
Entradas o Control de usuario				x
Visualización de estado del programa			x	