

Cómo añadir elementos a la escena

- ❑ Se añaden en el método **IG2App::setupScene(void)** que es invocado al terminar **IG2App::setup(void)**
- ❑ Los añade el gestor de la escena **mSM** que es un atributo de **IG2App** que toma valor en el **setup()**:

```
mSM = mRoot->createSceneManager();
```

- ❑ Recuerda que el gestor de la escena no es **Singleton** y puede no ser único

Cómo añadir elementos a la escena

- ❑ Todos los elementos se añaden a la escena incluyéndolos en un nodo.
- ❑ Son elementos que se pueden añadir a nodos de la escena los denominados **MovableObjects** entre los que se encuentran las cámaras, las luces y las entidades
- ❑ Para añadir una **entidad** a la escena se realiza el siguiente proceso:
 1. El gestor de la escena crea la entidad a partir de una malla, si la malla existe

```
Ogre::Entity* ent = mSM->createEntity("sphere.mesh");
```

Ojo, las entidades no se comparten. Si quiero añadir dos esferas a la escena es necesario crear dos entidades con la malla de la esfera, aunque se puede reutilizar el nombre **ent**. Si no se hace así, se produce un error y la escena no se renderiza.

Recuerda que **resources.cfg** debe permitir el acceso a la malla y que una manera recomendable de hacerlo es copiar **sphere.mesh** desde **media/models** a **media/IG2App**.

Cómo añadir elementos a la escena

2. El gestor de la escena crea un nodo vacío, a partir de un nodo que ya exista, dándole un nombre que sea único entre todos los nombres de todos los nodos de la escena. El nodo será hijo del nodo que lo creó. En el siguiente ejemplo, el nuevo nodo es hijo de la raíz

```
Ogre::SceneNode* mSceneNewNode = mSM->getRootSceneNode()  
->createChildSceneNode("esfera");
```

Si se repite un nombre, se produce un error y la escena no se renderiza. Si se quieren crear nodos con nombre, y se hace dentro de un bucle cuyo contador es **i**, una forma de garantizar la unicidad del nombre es usar el contador como cadena de caracteres mediante **std::to_string(i)**

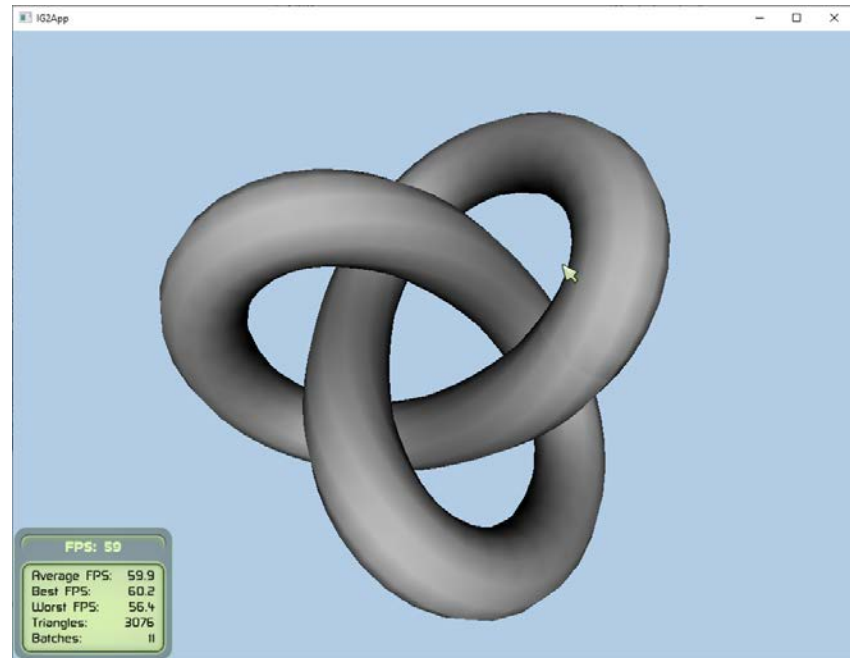
También es posible crear nodos sin nombre con **createChildSceneNode()** sin parámetros, pero entonces esos nodos no se pueden recuperar, si fuera necesario, por el nombre

3. Se adjunta la entidad al nodo

```
mSceneNewNode->attachObject(ent);
```

- ❑ Añadir el nudo a la escena

```
Ogre::Entity* ent = mSM->createEntity("knot.mesh");  
Ogre::SceneNode* knotNode =  
    mSM->getRootSceneNode()->createChildSceneNode("nudo");  
knotNode->attachObject(ent);
```



Cómo añadir elementos a la escena

1. Acciones opcionales posteriores que se pueden realizar sobre los nodos de la escena
 - Situar el nodo en un lugar determinado de la escena; por ejemplo, fijando una posición

```
mSceneNewNode->setPosition(400, 100, -300);
```

- Realizar transformaciones 3D (traslaciones, rotaciones y escalaciones) sobre el nodo

```
mSceneNewNode->setScale(5, 5, 5);
```

```
mSceneNewNode->yaw(Ogre::Degree(-45));
```

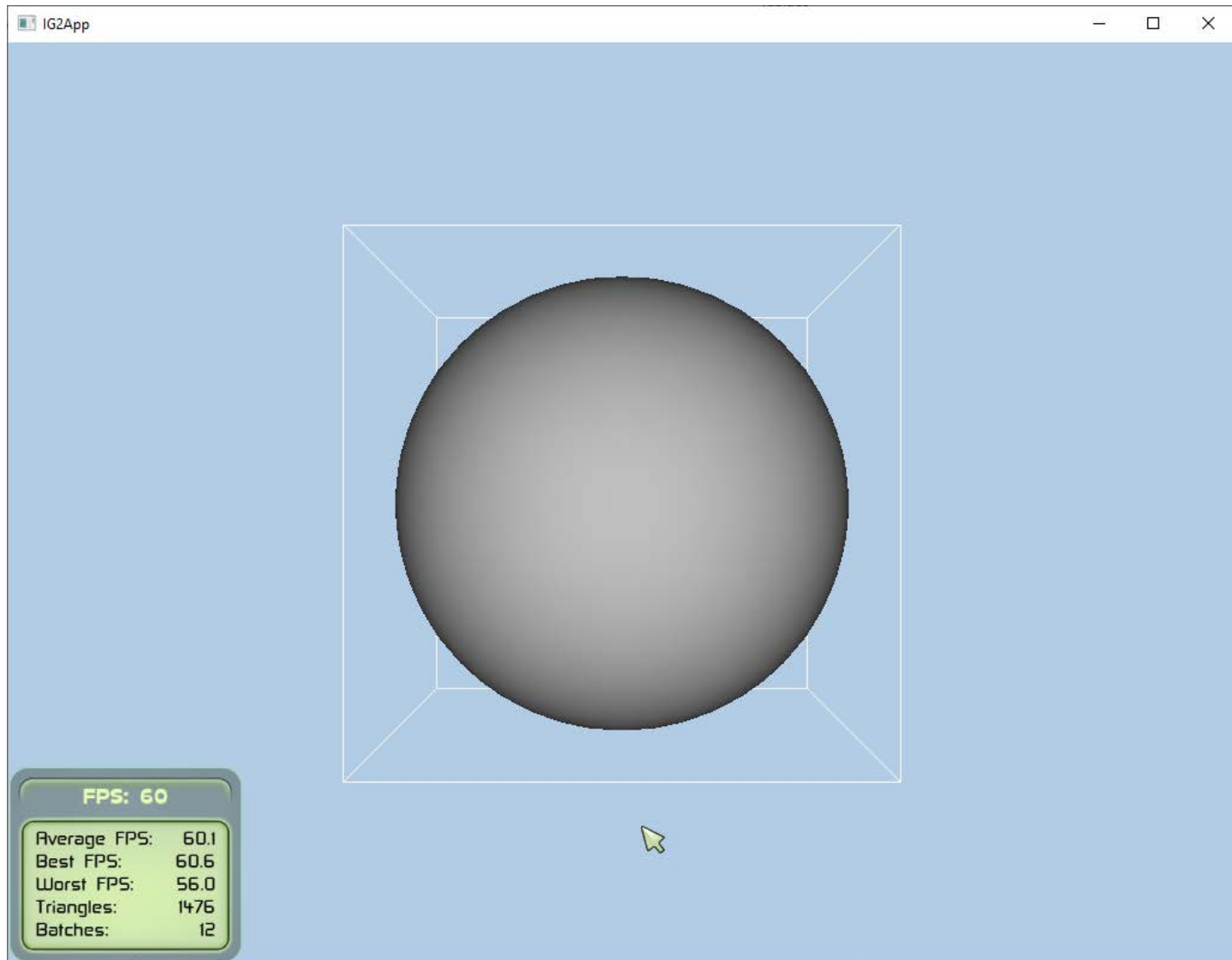
- Mostrar la caja limitadora que contiene el nodo

```
mSceneNewNode->showBoundingBox(true);
```

- Hacer no visible el nodo en la escena. Por defecto es visible

```
mSceneNewNode->setVisible(false);
```

Cómo añadir elementos a la escena



Cómo añadir eventos (de teclado) a la escena

- ❑ Se modifica el método **keyPressed()** de **IG2App**
- ❑ Los nodos que se modifiquen en el **keyPressed()** tienen que aparecer como atributos de la clase **IG2App**
- ❑ Por ejemplo

```
else if (evt.keysym.sym == SDLK_g) {  
    knotNode->roll(Ogre::Degree(3));  
}
```

hace girar el nudo alrededor de su eje **Z** cuando se pulsa la tecla **g**

- ❑ Qué se puede hacer: usar varias veces una variable de programa
 - ❑ Reutilizar la variable **ent** para referirse a entidades, posiblemente diferentes, en distintos lugares del código

```
Ogre::Entity* ent = mSM->createEntity("RomanBathLower.mesh");  
Ogre::SceneNode* mNode = mSM->getRootSceneNode()->createChildSceneNode();  
mNode->attachObject(ent);  
  
ent = mSM->createEntity("RomanBathUpper.mesh");  
Ogre::SceneNode* mNode1 = mSM->getRootSceneNode()->createChildSceneNode();  
mNode1->attachObject(ent);
```

- ❑ Reutilizar la variable **mNode** para referirse a nodos, posiblemente diferentes, en distintos lugares del código

```
Ogre::Entity* ent = mSM->createEntity("RomanBathLower.mesh");  
Ogre::SceneNode* mNode = mSM->getRootSceneNode()->createChildSceneNode();  
mNode->attachObject(ent);  
  
ent = mSM->createEntity("RomanBathUpper.mesh");  
mNode = mSM->getRootSceneNode()->createChildSceneNode();  
mNode->attachObject(ent);
```


❑ Qué no se puede hacer: adjuntar varias veces una misma malla de una entidad o repetir el nombre de un nodo

❑ Reutilizar la malla de una entidad

```
Entity* hour = mSM->createEntity("sphere.mesh");
Ogre::SceneNode* mHourNode =
    mSM->getRootSceneNode()->createChildSceneNode("Hour 1");
mHourNode->attachObject(hour);

mHourNode = mSM->getRootSceneNode()->createChildSceneNode("Hour 2");
mHourNode->attachObject(hour);    // ERROR
```

❑ Utilizar el mismo nombre para nodos diferentes

```
Entity* hour = mSM->createEntity("sphere.mesh");
Ogre::SceneNode* mHourNode =
    mSM->getRootSceneNode()->createChildSceneNode("Hour 1");
mHourNode->attachObject(hour);

hour = mSM->createEntity("sphere.mesh");
mHourNode = mSM->getRootSceneNode()->createChildSceneNode("Hour 1"); // ERROR
mHourNode->attachObject(hour);
```