

Auxiliar 4

Grafo de Escena

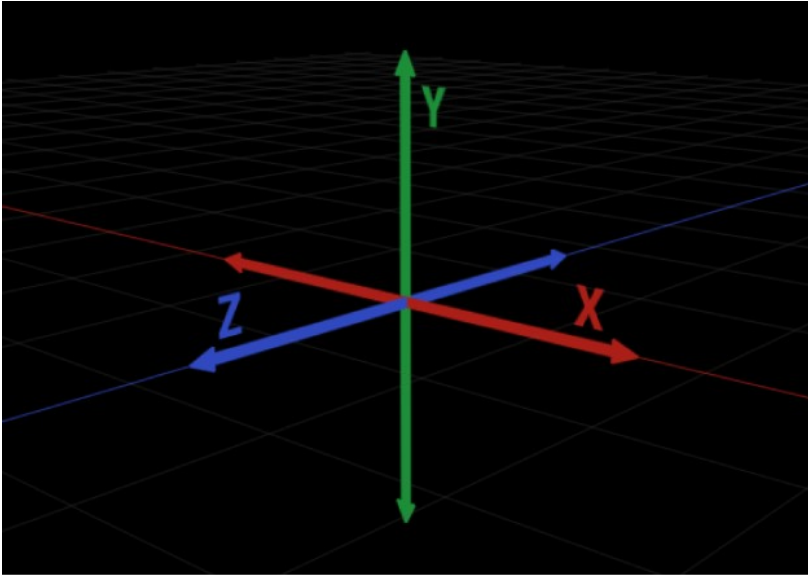
CC3501 Modelación y Computación Gráfica para Ingenieros

Primavera 2023

Profesor: Iván Sipiran

Auxiliar: Ariel Riveros

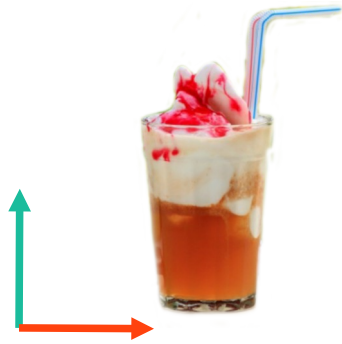
Espacios



World Space: Punto de referencia absoluto

Local Space: Punto de referencia relativa

Espacios



“Jugo”:
Posición Local = $[1, 0]$

Espacios



“Jugo”:

Posición Local = $[1, 0]$

Posición Global = $[1, 1]$ (Pongo el vaso sobre la mesa)

Mesa:

Posición Local = $[0, 0]$

Posición Global = $[0, 0]$

Espacios



“Jugo”:

Posición Local = [1, 0]

Posición Global = [4, 3]

Mesa:

Posición Local = [0, 0]

Posición Global = [3, 2]

Fonda:

Posición Local = [0, 0]

Posición Global = [0, 0]

Espacios



“Jugo”:

Posición Local = $[1, 0]$

Posición Global = $[4, 3]$

Rotación Global = π

Mesa:

Posición Local = $[0, 0]$

Posición Global = $[3, 2]$

Rotación Global = π

Fonda:

Posición Local = $[0, 0]$

Rotación Local = π

Posición Global = $[0, 0]$

Espacios



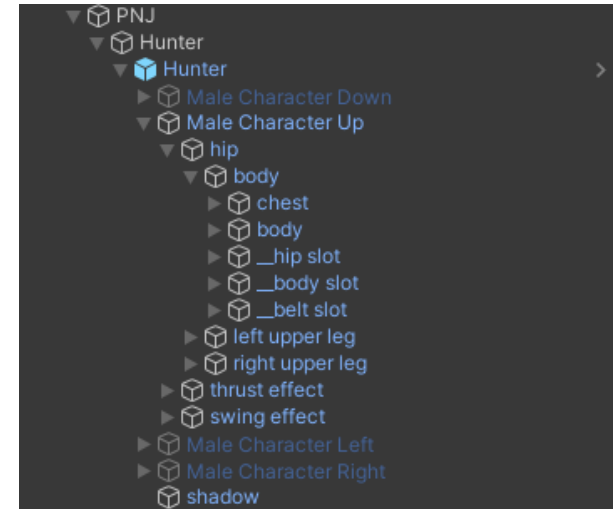
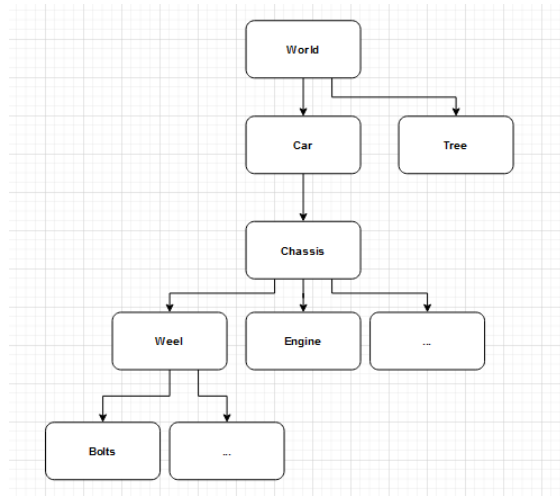
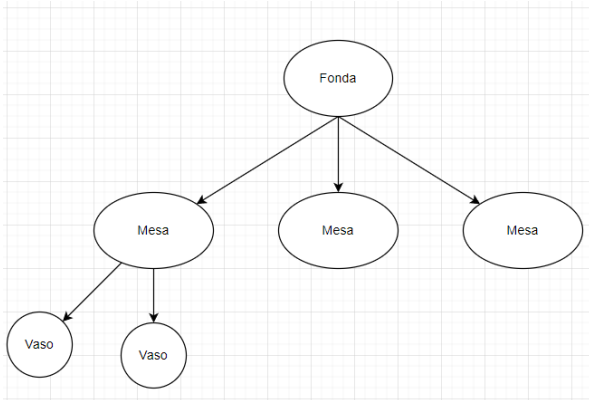
¿Qué le pasó a mi juguito :c?

Transformación del jugo = Transformación Fonda × Transformación Mesa × Transformación Local Jugo

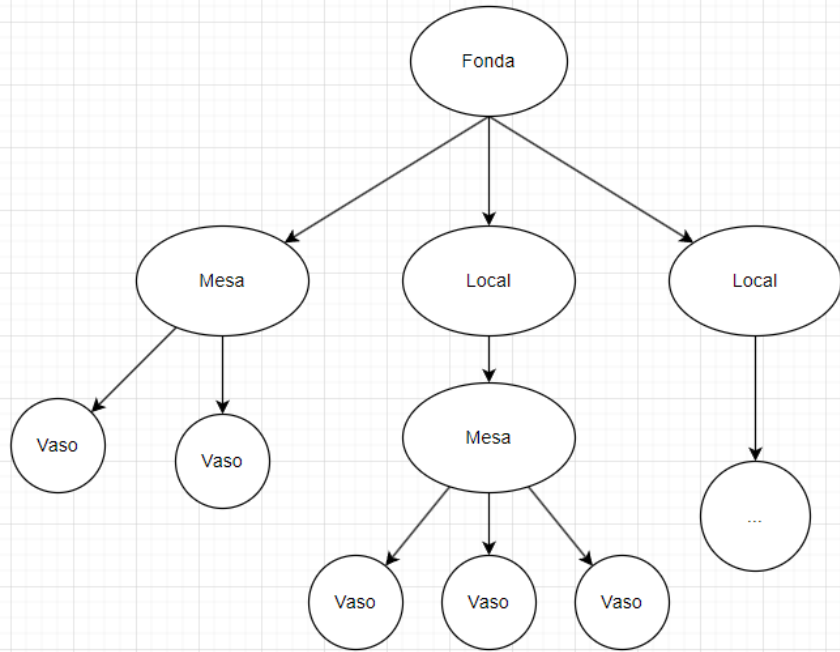
Grafos de Escena

¿Cómo representar una escena con transformaciones de manera jerárquica?

Con grafos



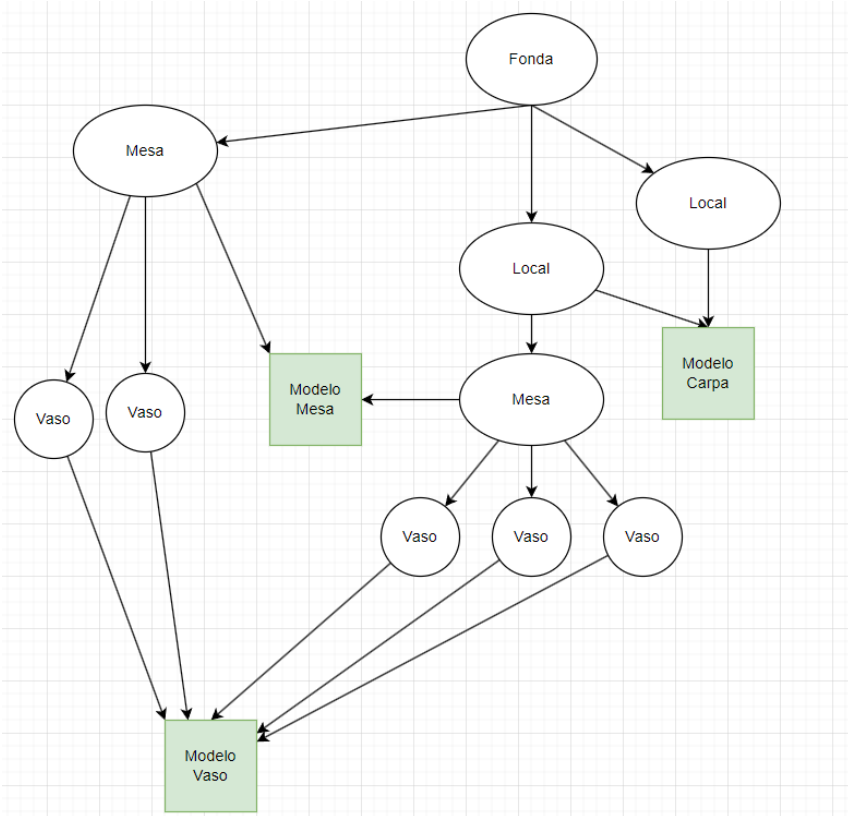
Grafos de Escena



Cada nodo tiene sus transformaciones y propiedades

Recursivamente se transforman los nodos hijos hasta llegar a las hojas, donde se almacena su *world space transformation*

Instancing



Lo que se dibuja son las hojas (cuadrados verdes)

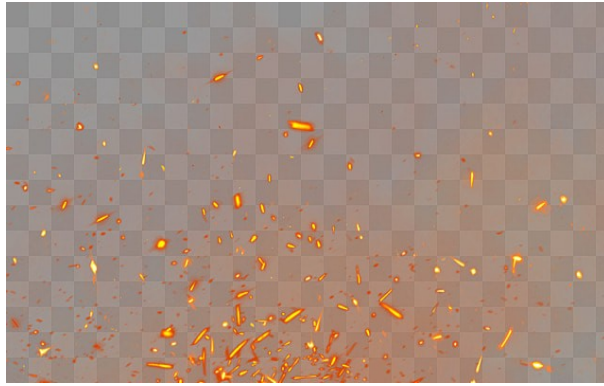
Utilizar el mismo modelo con distintas transformaciones o propiedades se llama *instancing*

Instancing

instancing es muy efectivo dibujando el mismo objeto muchas veces



follaje

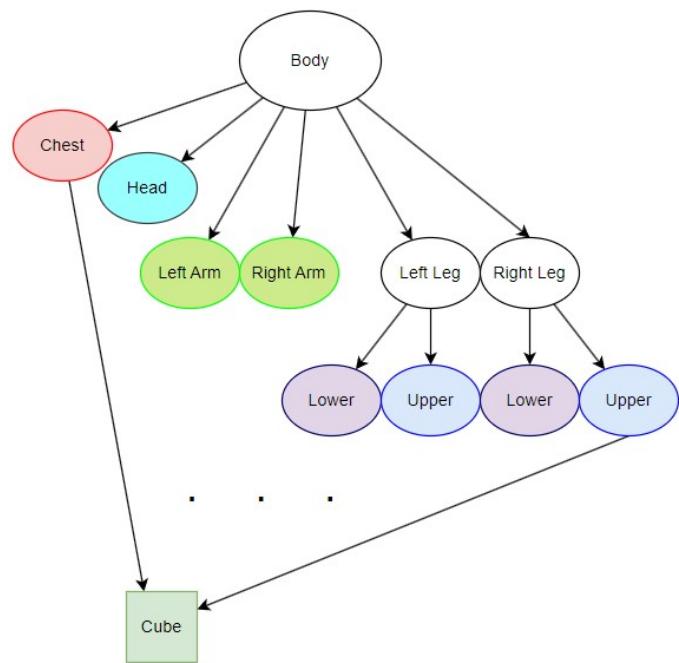


partículas



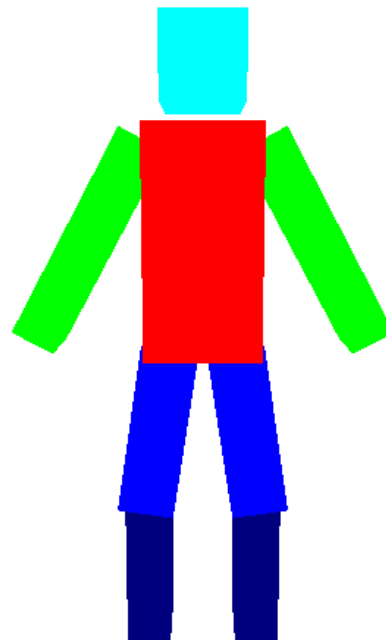
multitudes

Instancing

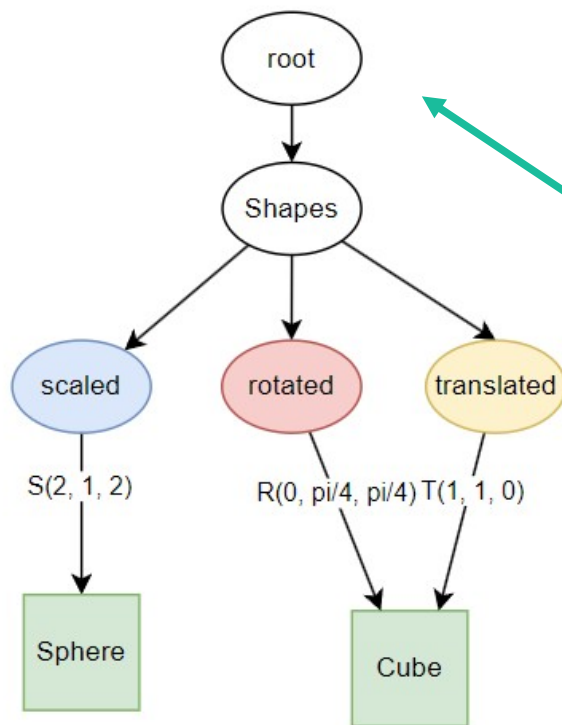


Ejemplo

Cada nodo transforma y colorea al mismo cubo de manera distinta



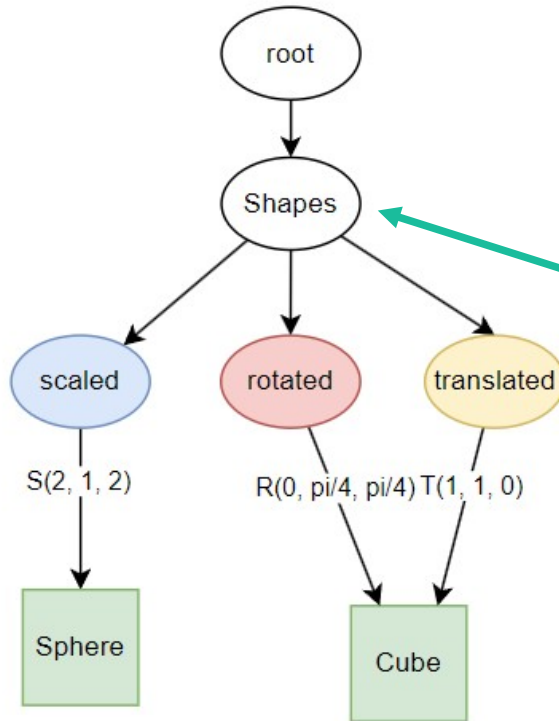
Cómo implementar una escena



Primero se define la raíz
camera es la cámara que se utiliza en la escena

```
graph = SceneGraph(camera)  
graph.add_node("shapes")
```

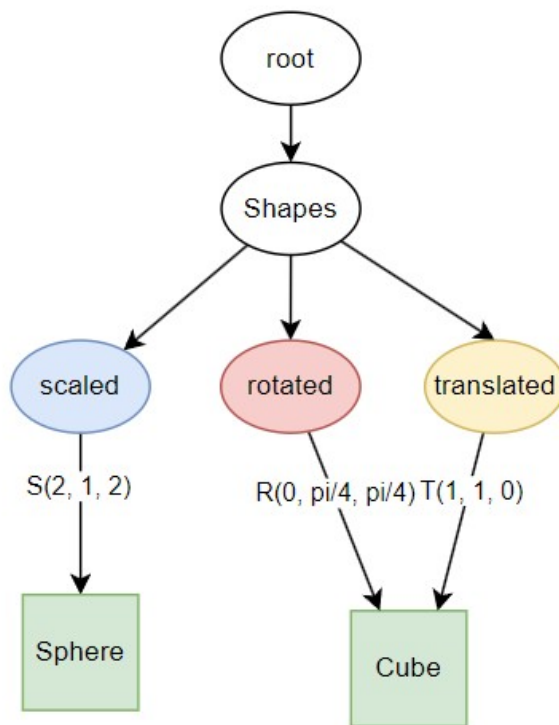
Cómo implementar una escena



Le sigue el nodo “shapes”
add_node recibe un argumento attach_to
que por defecto se adjunta a “root”

```
graph = SceneGraph(camera)
graph.add_node("shapes")
```

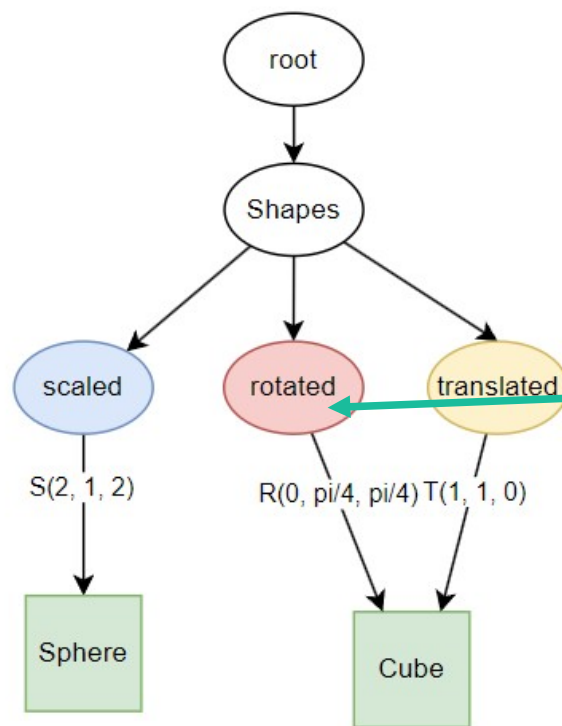
Cómo implementar una escena



Translated: se adjunta a shapes con `attach_to="shapes"`
Se le asigna una hoja, es decir, un modelo que se dibuja en `mesh=cube`
Se le asigna un color amarillo y una transformación de traslación

```
graph.add_node("translated",  
               attach_to="shapes",  
               mesh=cube,  
               color=shapes.YELLOW,  
               transform=tr.translate(1, 1, 0))
```

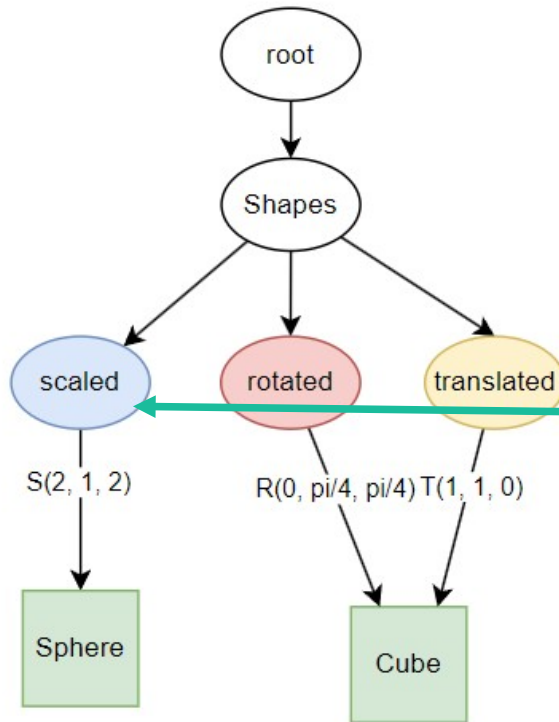

Cómo implementar una escena



Rotated, análogo a translated

```
graph.add_node("rotated",  
               attach_to="shapes",  
               mesh=cube,  
               color=shapes.RED,  
               transform=tr.rotationY(np.pi / 4) @ tr.rotationZ(np.pi / 4))
```

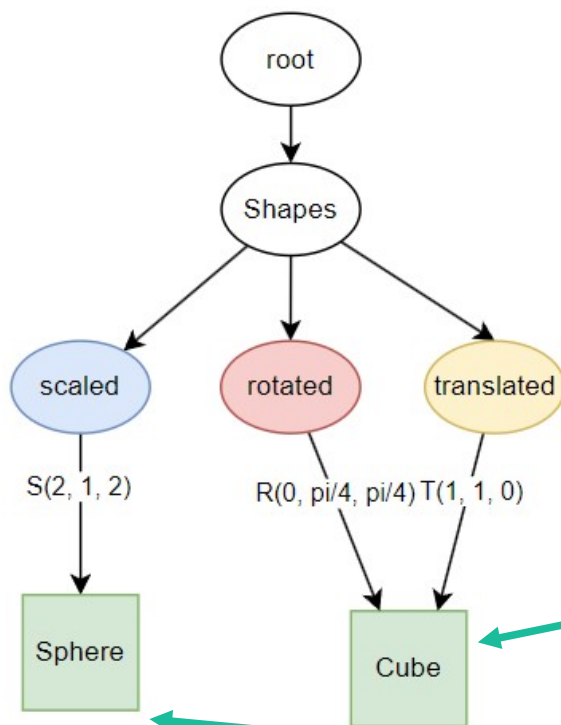
Cómo implementar una escena



Scaled análogo a Translated
pero asignando sphere en mesh

```
graph.add_node("scaled",  
               attach_to="shapes",  
               mesh=sphere,  
               color=shapes.BLUE,  
               transform=tr.scale(2, 1, 2))
```

Cómo implementar una escena

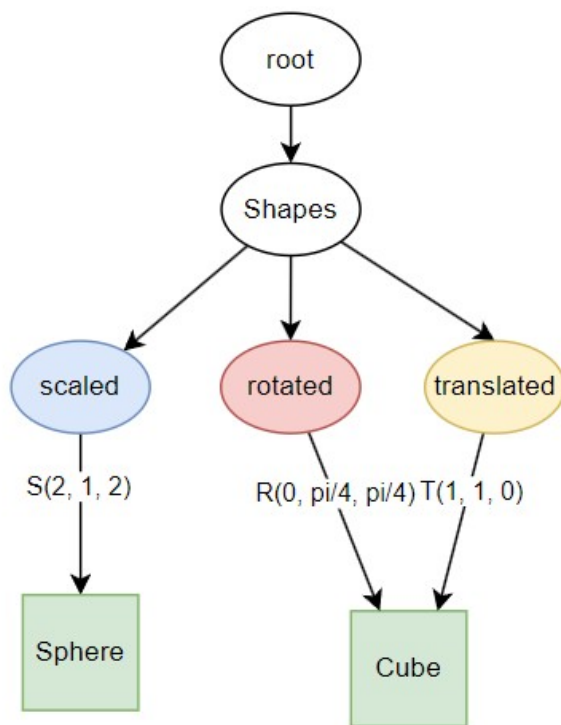


Los modelos se definen como siempre
Sin embargo, sus transformaciones se manejan en el grafo

```
cube = Model(shapes.Cube["position"], index_data=shapes.Cube["indices"])
cube.init_gpu_data(mesh_pipeline)

sphere = Mesh("assets/sphere.off")
sphere.init_gpu_data(mesh_pipeline)
```

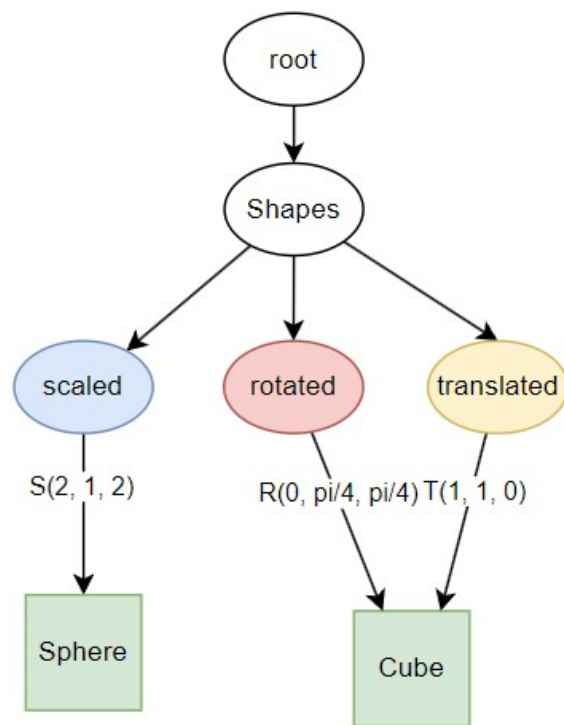
Cómo implementar una escena



Para dibujar la escena simplemente llamar la función `draw()`

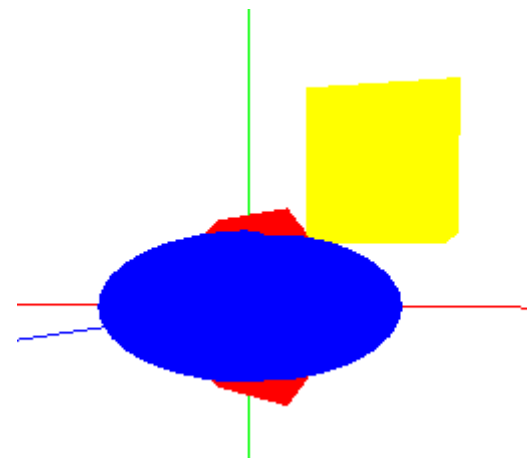
```
# draw loop
@controller.event
def on_draw():
    controller.clear()
    graph.draw()
```

Cómo implementar una escena



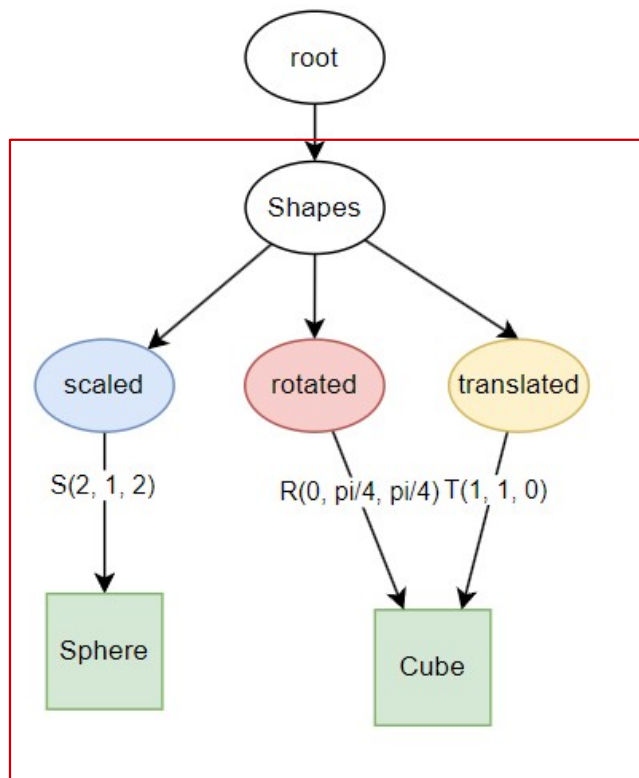
Para dibujar la escena simplemente llamar la función `draw()`

```
# draw loop
@controller.event
def on_draw():
    controller.clear()
    graph.draw()
```



Arte abstracto :o

Cómo implementar una escena

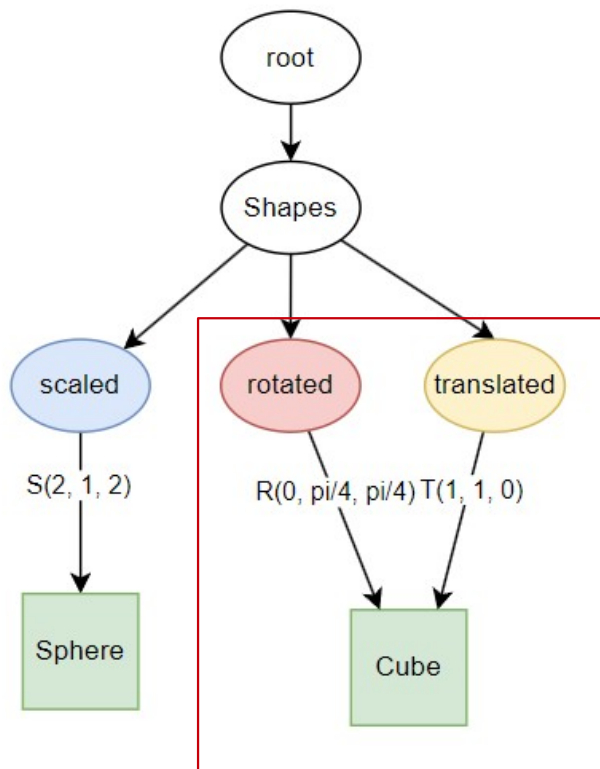


Podemos acceder a cualquier nodo del grafo y aplicar más transformaciones

```
def update(dt):  
    graph["shapes"]["rotation"][1] += dt
```

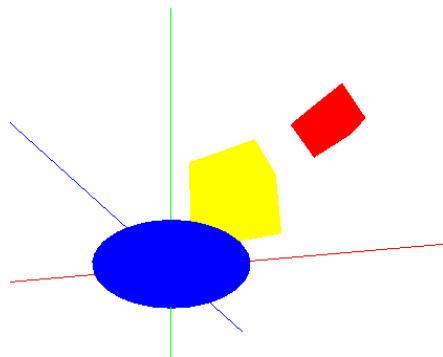
Esto aplicará rotación en el eje Y a todos los nodos hijos

Cómo implementar una escena



Podemos acceder a cualquier nodo del grafo y aplicar más transformaciones

```
def update(dt):  
    graph["rotated"]["position"][0] += dt  
    graph["translated"]["rotation"][1] += dt
```



Cómo implementar una escena

Motivación:

Estudiar las clases SolarSystem y Person

Revisen qué transformación está haciendo qué

Y cómo puedo controlar cada nodo independientemente

¿Puedo controlar a un person, trasladarlo y qué tan rápido se traslada
su animación de caminar sea más rápida? (sí, si puede)



Auxiliar 4

Grafo de Escena

CC3501 Modelación y Computación Gráfica para Ingenieros

Primavera 2023

Profesor: Iván Sipiran

Auxiliar: Ariel Riveros