

Practice 2: Hierarchical Modeling and Transformations

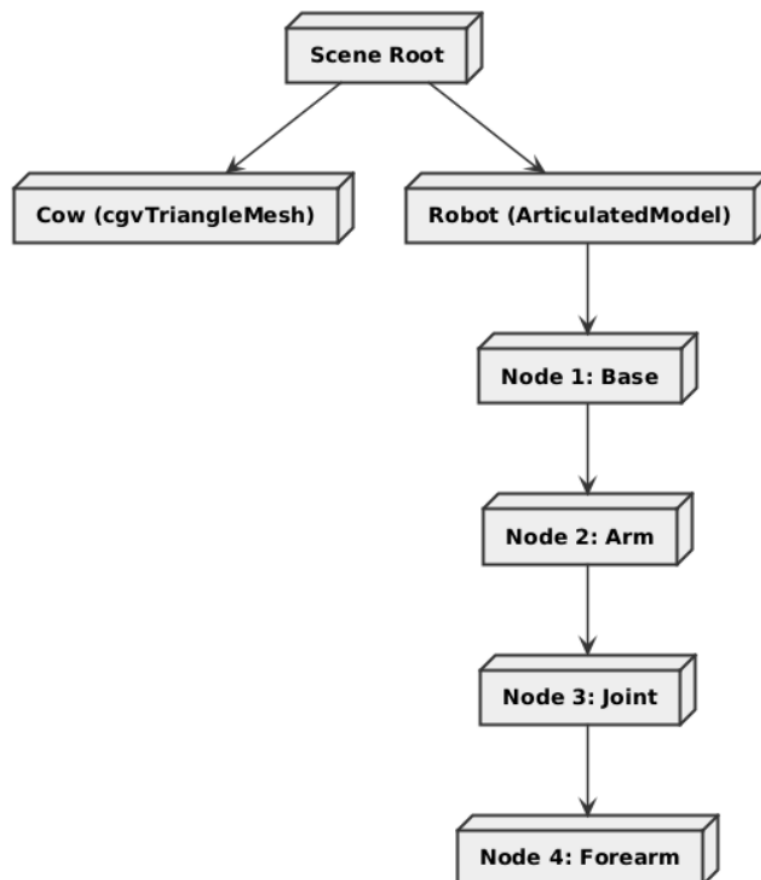
This document outlines the scene graph, transformations, and primitives used in the application.

1. Scene Graph Overview

The scene contains two root-level objects:

- Cow (cgvTriangleMesh): A non-articulated mesh loaded from an OBJ file.
- Robot (ArticulatedModel): An articulated model composed of several primitive shapes.

The transformation order for all objects is Translate -> Rotate -> Scale, as the `rstMode` is set to false by default in the `Object3D` constructor, meaning transformations are applied sequentially as they are called.



2. Object Details

2.1. Cow (cgvTriangleMesh)

- Parent: Scene Root
- Initial Transformations:
 - Translate(-5, 0, 0)
- Primitives:
 - A mesh of triangles loaded from objFiles/cow.obj.

2.2. Robot (ArticulatedModel)

- Parent: Scene Root
- Initial Transformations:
 - Translate(5, 0, 0)
- Hierarchy and Transformations: The robot is composed of several parts in a hierarchical structure. The transformations are applied in order from parent to child.
 - Node 1: Base
 - Parent: Robot Root
 - Transformations:
 - Rotate(dof[0], 0, 1, 0) (Yaw rotation around its Y-axis)
 - Scale(1.5, 1.0, 1.5)
 - Primitive: glutSolidCube(1.0)
 - Node 2: Arm
 - Parent: Base
 - Transformations:
 - Translate(0, 0.5, 0) (Positioned on top of the base)
 - Rotate(dof[1], 1, 0, 0) (Pitch rotation on its X-axis)
 - Primitives:
 - Joint: glutSolidSphere(0.4, 20, 20)

- Arm: A cylinder created by `draw_cylinder(0.25, 2.0, 20)`, which is internally transformed by `Rotate(-90, 1, 0, 0)` to orient it along the Y-axis.
- Node 3: Joint
 - Parent: Arm
 - Transformations:
 - `Translate(0, 2.0, 0)` (Positioned at the end of Arm)
 - `Rotate(dof[2], 1, 0, 0)` (Pitch rotation on its X-axis)
 - Primitives:
 - Joint: `glutSolidSphere(0.3, 20, 20)`
 - Arm: A cylinder created by `draw_cylinder(0.2, 2.0, 20)`, also internally rotated by `Rotate(-90, 1, 0, 0)`.
- Node 4: Forearm
 - Parent: Joint
 - Transformations:
 - `Translate(0, 2.0, 0)` (Positioned at the end of Joint)
 - Primitive: `glutSolidSphere(0.5, 20, 20)`

3. Degrees of freedom (DoF)

The articulated robot has 3 Degrees of Freedom, which correspond to the rotations of its joints:

- `dof[0]`: Controls the Y-axis rotation of the Base.
- `dof[1]`: Controls the X-axis rotation of Arm .
- `dof[2]`: Controls the X-axis rotation of Joint .

These DoFs can be manipulated via keyboard controls or mouse picking, which targets the specific joint to modify its corresponding DoF value.