Title: README Unity Python Move Object

Document Number: 190a-5

CTI One Corporation

Table 1a. Document History

2022-02-21	Establish this document, document archive: /media/harry/easystore/backup-2020-2-15/CTI/3proejcts/3-8-smart-tech/3-8-4-CTI/3-8-4-6-products/AIV200/190-robots-health/190a-unity-simulation/190a-5-moving-objects	YY
2022-02-23	Remove OpenCV part	YY

Table 1b. Testing and Release Approval Form

2022-02-??	Tested by ??? and approved for release by ???	Pending for testing and approval

Table 2. References

Number	Name and URL	Note
1.	How to move objects in Unity (3 methods with examples) https://gamedevbeginner.com/how-to-move-objects-in-unity/	transform.position is used in this document
2.	NetMQ Request Responce	

	https://netmq.readthedocs.io/en/latest/request-response/	
3.	C# - Multithreading https://www.tutorialspoint.com/csharp/csharp_multithreading.htm	

Table 3. Prerequisite

Software Prerequisite No.	Description and Version	Note
1.	Ubuntu 18.04	
2.	Python version 3.7	On Ubuntu
3.	Unity 2020.3.26	On Ubuntu
4.	PyZMQ version 22.3.0	On Ubuntu
Hardware Prerequisite No.	Description and Version	
1.		

1. Setup Python Packages And Unity Project

1.1. Open a terminal and perfome the following commands;

pip3 install pyzmq

1.2. Open Unity Hub, click “ADD”, select the project folder “Python-Move-Object-2022-2-21” and click “OK”

2. Run the programs

2.1. Open the Unity project “Python-Move-Object-2022-2-21” on Unity hub

2.2. Push the “Play” button on Unity.

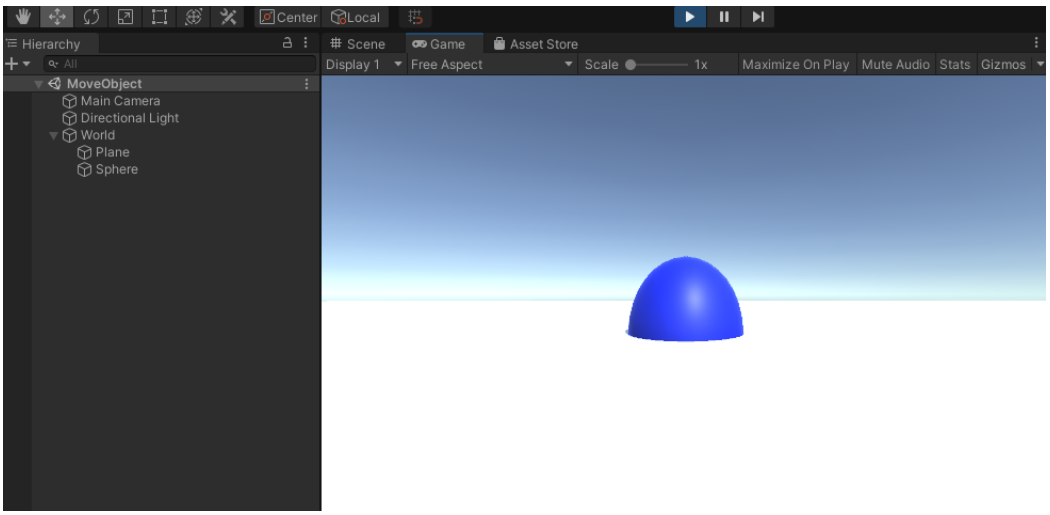

Figure 1. The result of Unity C# and Python Programs

2.3. Open a terminal and perfome the following commands;

python3 SendActionClient.py

The below message will be shown;

“Please enter the number; 1: Up, 2: Down”

Input 1 or 2 to move the object up or down on Unity

2.4. Stop the Python program

2.5. Push the “Play” button to stop the Unity program

3. Create the Python client program from scratch

3.1. Create a Python program file Name “SendActionClient.py”

3.2. Copy the following source code to SendActionClient.py

```
import zmq

context = zmq.Context()
socket = context.socket(zmq.REQ)
socket.connect("tcp://127.0.0.1:8555")

while True:
    actionMessage = input("Please enter the number; 1: Up, 2: Down\n")
    socket.send(actionMessage.encode('ascii'))   # ZeroMQ can send String in ASCII only. No Error Handling

    message = socket.recv()
    print("Received responce: ", message)
```

4. Preparation for creating the Unity project

4.1. Clone the repository using the below command;

```
git clone https://github.com/off99555/Unity3D-Python-Communication.git
```

5. Create the Unity project from scratch

5.1. Open Unity Hub, push “New” button, choose temlate “3D”, project name: Python-Move-Object-2022-2-21

5.2. Rename the Scene as “MoveObject”.

On “Project” window, choose “Assets” → “Scenes” → “SampleScene”. right click -> “Rename”. Rename the Scene to “MoveObject”.

5.3. Create GameObjects

5.3.1 Create an Empty Object; “GameObject” → “Create Empty”. Name “World”

5.3.2 Create a Plane object under “World” object; Select “World” object and right click and choose “3D Object” → “Plain”. On “Inspecter” window, change the scale X: 10, Y:1, Z:10

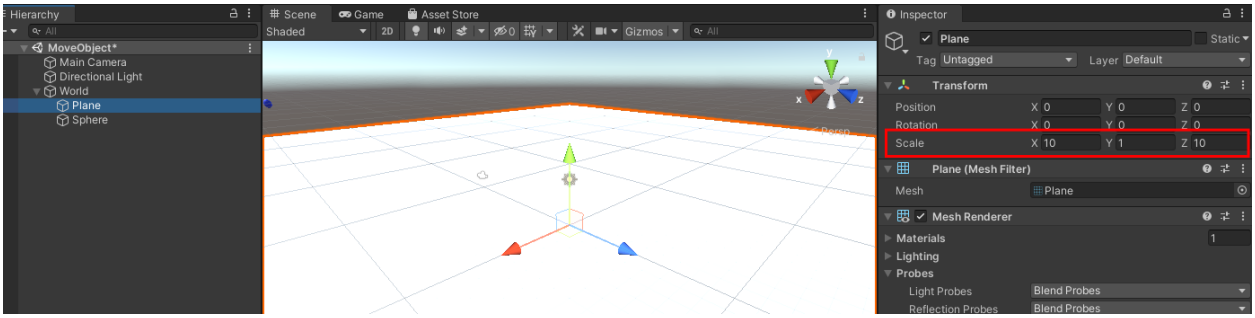

Figure 2: Change Scale

5.3.3 Create a Sphere object under “World” object; Select “World” object and right click and choose “3D Object” → “Sphere”. On “Inspecter” Window, change the scale X: 3, Y:3, Z:3

5.4. Change the objects' color

5.4.1 Create a material; On "Project" window, choose "Assets" folder, right click and "Create" → "Material". Name "SphereMaterial". Change the Material color to blue on "Inspector" window

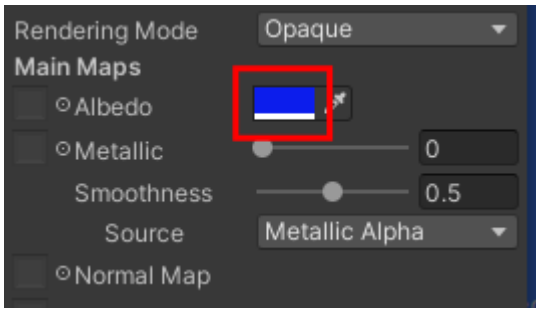

Figure 3: Change Color

5.4.2 Choose Sphere and drag "SphereMaterial" and drop to "Sphere" on "Hierarchy" window

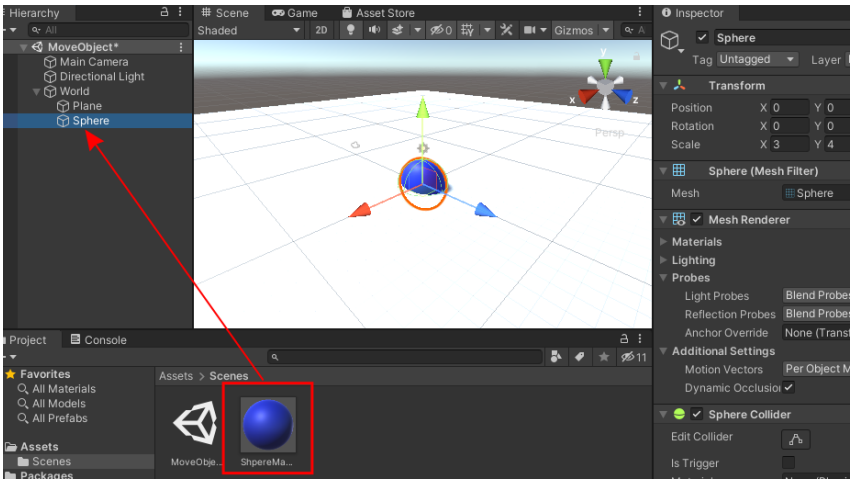

Figure 4: Drag and Drop the Sphere Material

5.5. Import Plugins and C# Script

5.5.1. Create a "Plugins" folder in "Assets" folder on "Project" window; right click and "Create" → "Folder". Name "Plugins"

5.5.2. Open “Plugins” folder. On file manager of OS, open “Unity3D-Python-Communication/UnityProject/Assets/NetMQExample/Plugins” folder cloned at Step 4.1. Drag “AsyncIO.dll” and “NetMQ.dll” and drop to “Plugins” folder on Unity Preject window

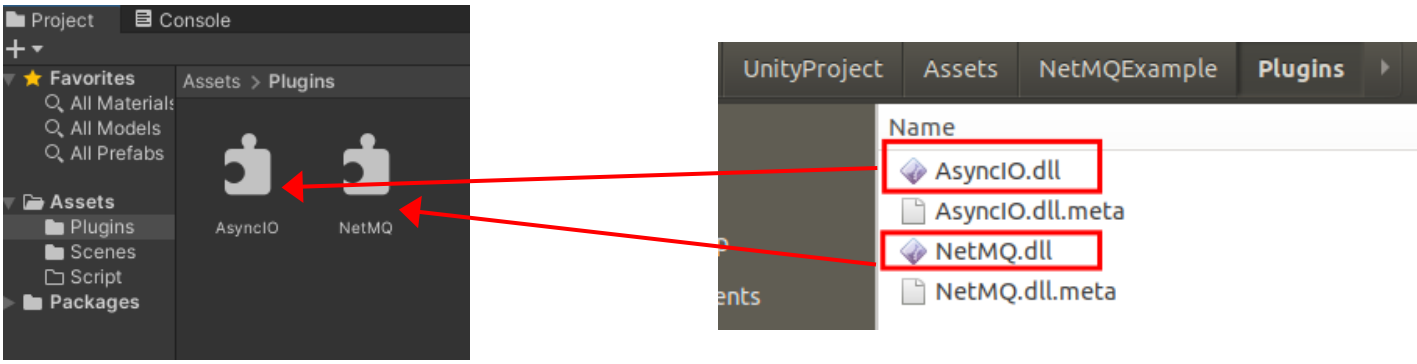

Figure 5: Import NetMQ Plugins To Unity

5.6. Create C# script

5.6.1. Create a “Script” folder in “Assets” folder on “Project” window; right click and “Create” → “Folder”. Name “Script”

5.6.2. Choose “Assets” → “Script” folder on “Project” window and create C# script; right click → “Create” → “C# Script”. Name “MoveObject”

5.6.3. Open “MoveObject” C# Script, and write the following code;

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.Threading;
using UnityEngine;
using AsyncIO;
using NetMQ;
using NetMQ.Sockets;

public class MoveObject : MonoBehaviour
{
    // Vector3 for objects moving up, down
    static Vector3 MOVE_VECTOR3_UP = new Vector3(0, 2, 0);
    static Vector3 MOVE_VECTOR3_DOWN = new Vector3(0, -2, 0);

    const int ACTION_STAY = 0;
    const int ACTION_UP = 1;
    const int ACTION_DOWN = 2;

    ResponseSocket responseSocket;

    // 0: Stay, 1: Up, 2: Down
    int actionNumber = 0;
    Thread serverThread;
```

```
    void Awake()
    {
        ForceDotNet.Force(); // this line is needed to prevent unity freeze after one use, not sure why yet

        // Create Response(Server) socket
        responseSocket = new ResponseSocket("@tcp://localhost:8555");

        // Create a server thread for receiving messages from Python code
        ThreadStart serverThreadStart = new ThreadStart(GetRequest);
        serverThread = new Thread(serverThreadStart);
        serverThread.IsBackground = true;
        serverThread.Start();
    }

    // Update() is called once per frame
    void Update()
    {
        // Move the object up and down
        if(actionNumber == ACTION_UP){
            transform.position += MOVE_VECTOR3_UP;
            actionNumber = ACTION_STAY;

        } else if(actionNumber == ACTION_DOWN) {
            transform.position += MOVE_VECTOR3_DOWN;
            actionNumber = ACTION_STAY;
        }
    }

    // Child thread for communicate with Python
    void GetRequest(){
        while(true){
            // Receive a meesage and send it back
            var message = responseSocket.ReceiveFrameString();
            responseSocket.SendFrame(message);

            // Set the Action Numer, no error handling
            actionNumber = Int32.Parse(message);
        }
    }
    // OnDisable() is called when the behaviour becomes disabled.
    void OnDisable(){
        try{
            // Stop receiving thread
            serverThread.Abort();
            // Close the server socket
            responseSocket.Close();
            NetMQConfig.Cleanup(false);
        }catch(Exception exp){
             if (exp.Source != null){
                Console.WriteLine("Exeption source: {0}", exp.Source);
            }
        }
    }
}
```

5.6.4. Choose “Sphere” on “Hierarchy” window. Drag “MoveObject” C# Script and drop to “Add Component” button on “Inspector” window

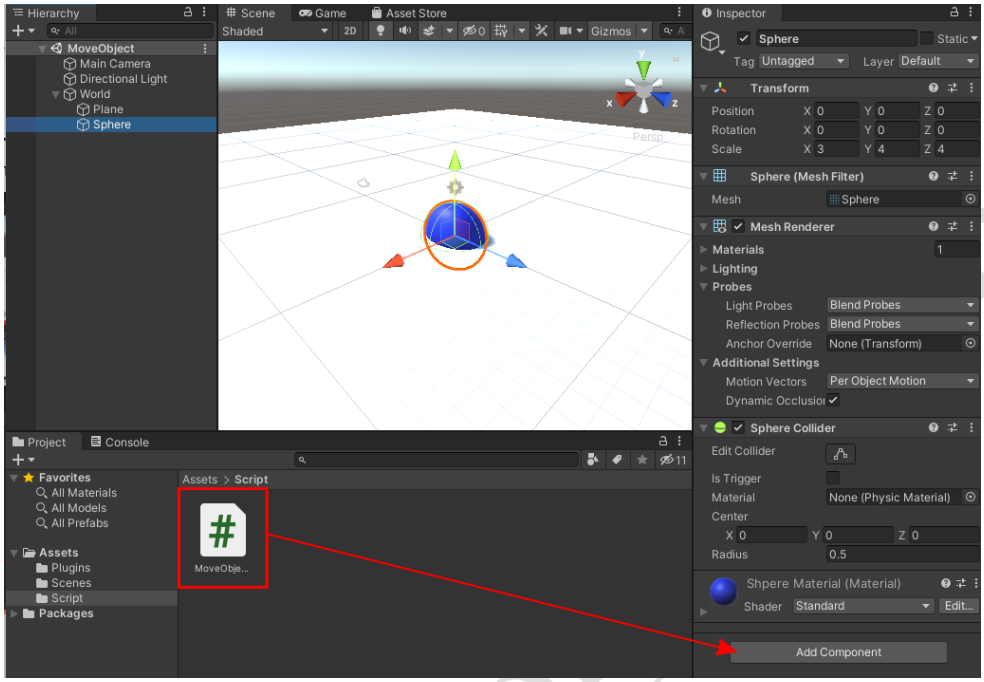

Figure 6: Set MoveObject.cs to Sphere Object

(END)

CTI Plus Co...