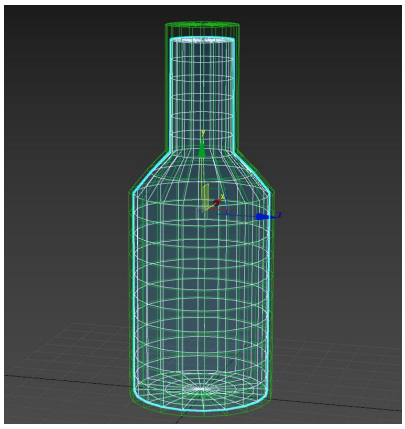
Liquid Shader

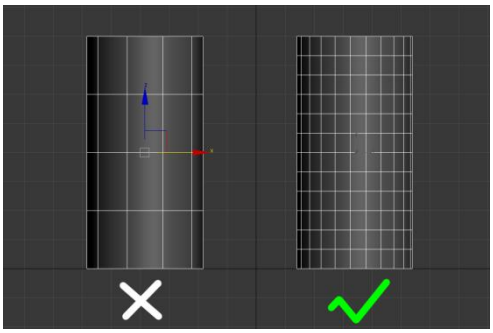
v1.0

Modeling

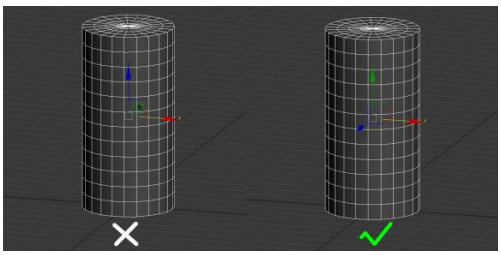
1. To make this work, you need to create a **container model** and a **liquid model**, The liquid model should be slightly smaller than the container model to present the shape of the liquid when it's full of the container. Both of them should have the **MeshFilter** and **MeshRenderer** compoents in the Unity. Also a **cork model** is **optional** to create.



2. The **liquid model shouldn't** have **too few vertexes**, and try to make the polygon spread as **evenly** as possible.



3. The **pivot** of the liquid should be set to the **center of the object**, and make sure **Y axis** is **point upwards**, Finally make sure the **scale** of both the liquid and the container are **(1,1,1)**



(Y axis point up)

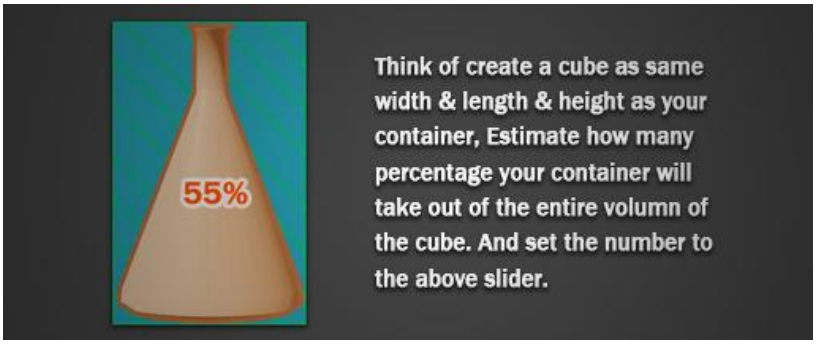
Basic Setup

1. The **liquid** and the **cork** gameobject should be set to the **children of the container** in the Unity:

2. First you need to create a **material** for your **liquid** and assign to its **MeshRenderer**, The **liquid** must use my **SoftKitty/Liquid** Shader, The **container** could be any shader. If you want to add particles inside the liquid, the **particles** must use my **SoftKitty/Particle** Shader.

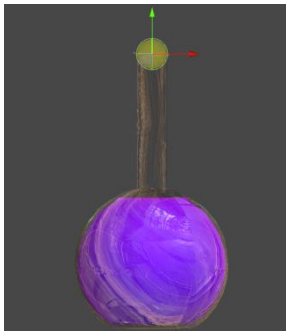
3. Then add my **LiquidControl.cs** compoent to your **container** gameobject which is the root of the whole thing. Click on the **"Auto find the liquid mesh"** red button, so it will auto setup everything for you. Once it's done, the red button should turn into a green check mark.

4. Setup the **"Volumn of cube"** Slider to a proper value:



Optional Setup

1. If you wish to have a **mouth for the container** so the liquid will be able to flow out from there, you can click the **"Create container mouth transform"** red button, it will auto add a transform to present the mouth of the container, use the **slider to adjust its size**, and **move the transform to position it** right. The size should be exactly the same as the hole of the container mouth.



2. If you have a **cork model** for the mouth of the container, drag it to the **"Cork Model"** slot of the **LiquidControl** script. In the runtime, when you move the **cork** gameObject away or dis-active it, the liquid will be able to flow out, Otherwise it will be locked inside the container.

3. It is recommended to add a **MeshCollider** to your **container mesh**, so the liquid from other other containers will not flow though it.

Physics Setting

I use **Layer 4 (Water Layer)** to detect the mouth of the containers, and it only trigger events if the collider is set to **"IsTrigger"**

Script

Useful Accessable Datas:

```
public float GetCurrentTotalVolumn()
//Return the total current volumn of the liquid

public float GetCurrentVolumnByColor(Color _colorTop, Color _colorBottom)
//Return the current volumn of the liquid with certain colors

public int GetTotalLiquidCount()
//Return how many liquid has been mixed into this container

public float GetLiquidSurfaceHeight()
//Return the liquid surface Y axis position in world space,  This is useful if you want to make something floating on the surface of the liquid

public bool isReachOpenning()
//Return if the the surface of the liquid is above the bottom of the container mouth, Meaning should the liquid flow out of the mouth if the cork is not set.
```

```
public bool isCorkSet()
// Check if the cork is set to prevent liquid flow out
```

Useful Public Funtions:

```
public void SetFlowSpeed(float _speed)
//Set the liquid flow out speed, default value is 1

public void SetWaterLine(float _amount)
//Directly change the percentage amount of the liquid (0-1)

public void FillInLiquid(float _amount, Color _colorTop, Color _colorBottom)
// Fill in x amount of liquid with certain colors

public void AddLiquid(float _volumn, Color _colorTop, Color _colorBottom, bool _addAll =
false)
//_volumn can be nagetive value
```

If you have any questions, feel free to send me a mail to: **1285744247@qq.com**