Thank you for purchasing 2D Volumetric Lights for Unity3D. In this short document I will walk you through setting up a few basic lighting systems!

Tutorial 1 : Basic Radial Light

Inside of a new scene click “GameObject -> Create Other -> 2D Lights -> Radial Light”

This will add a new radial light with the basic material “PlanerMaterial” set to a color of white. Lets change the color.

Select the light and look in the inspector. There is a variable called “Color”. Click there and set the color of the light to whatever you desire. Notice how the game view updates to show a preview of the light.

Note: Due to the way the lights are rendered you will only be able to preview them in the game view.

Now lets say you wanted to make the light to behave more like a spotlight. This is done by simply changing the “Cone” of the light from 360 degrees to something like 50 degrees. Go ahead and give that a try!

When you want to scale the light you will notice you will not be able to do so with the traditional scale inside of the unity editor. Instead you must use the “Size” variable to change the size of the light!

When you parent your light to another object please keep in mind that you must not parent more than 1 node deep as it will mess up the scale of the light and cause undesired effects. I hope to have this fixed in future versions!

Tutorial 2 : Casting Shadows

If you would like to cast shadows on specific objects, it is best to create a layer for the objects you want to cast shadows. By default the layer on the lights is set to “Everything” so that everything with a collider will cast a shadow.

Select the object you want to cast a shadow. Click the “Layer” dropdown in the upper right hand side of the inspector. Click “Add Layer…” and call your new layer “Cast2DShadow”. Now click the object again and assign your newly created layer to the object.

Next go to the light you would like to cast the shadow with and click “Shadow Mask List” button. Select “Nothing” to de-select all of the items, and then click it again to select the “Cast2DShadow” layer. This will make the light only cast shadows against other objects on that layer!

Tutorial 3 : Multiple Cameras

Okay, so now you feel pretty comfortable with setting up simple lights. Now lets say you have multiple cameras that you only want specific lights displayed in. To do this I provided a list of available cameras! Assign cameras by clicking the button that says “Used On (X) Camera(s)”. This will display a dropdown menu where you can select the cameras you want to render the current light. You may chose more than one camera if you like.

Tutorial 4 : Events

There might be a time when you will need your lights to send an event to the object its casting a shadow off of. For instance, if you have a game where your player is avoiding the light! With my system this is very simple to do! I provide 3 event delegates which you can assign your script to called ‘OnBeamEnter’, ‘OnBeamStay’, and ‘OnBeamExit’. The delegate returns void and broadcasts 2 variables, (GameObject obj, Light2DEmitter emitter).

Inside of your script write a function that matches the same footprint as the above delegate:

```
Void DoSomethingOnEnter(GameObject obj, Light2DEmitter emitter)
{
  If(obj == gameObject)
  {
    // Do some special code with the objects
  }
}
```

Then register the function to the even by calling:

```
Void Start()
{
  Light2DEmitter.OnBeamEnter += DoSomethingOnEnter;
}
```

Then inside of the light you want to send events check the “Enable Events” checkbox!

If you would like specific functions for specific objects, I provided the “Event Filter” string variable to pass unique identifiers through our event. Retrieve those by calling “emitter.eventFilter” inside of the example function above.

For more questions please email me at jake@reverieinteractive.com or visit my website www.reverieinteractive.com