**Black Hole VFX Tutorial**

Hi, my name is Joe, and I am going to walk you through a quick tutorial on how to make the black hole effect from my current project Space Time using Unity’s Shuriken particle system. I am using Unity version 2021.3.6 for this tutorial.

**Setting up your project**

1. Step 1 is to make a new project in Unity using the Universal Render Pipeline as a template. This is not essential, but it will give us access to some cool post-processing effect by default. You could always use an earlier version of Unity and add the post-processing package after you create the project.
2. Once you have your new project created, we need to do a little house keeping. First, we need to rename our sample scene to Black\_Hole. I always like to create a file structure at the start of a new project to keep everything tidy from the start. It can be edited and extended later but it is good practice to start as you mean to go on.
3. We are going to need a folder for materials, prefabs, models and 2D textures.
4. Now inside your Materials folder right click and select Create – Material. For the Shader select Skybox – Procedural from the drop-down menu.
5. Next set the values as follows.
6. Simply drag the material into the scene view to apply it to the game world.

**Setting up post-processing**

1. Now we need to set up our post-processing volume. Right click in the hierarchy and select Volume - Global Volume. Rename this to “Post\_Processing\_Volume”.
2. Next up we need to create a new Profile and call it Post\_Processing\_Profile.
3. The global volume will apply our post-processing effects to the whole scene and the profile will store our post-processing components and their settings.
4. The last thing we need to do it check the post-processing box in the camera rendering settings.
5. Finally, we need to test our post-processing is working correctly. To do this we will right click in the hierarchy and create a new 3D object in the form of a plane and also a point light. Position the point light slightly above the plane and adjust the intensity so you can see the light reflecting off the plane.
6. Now go to your post-processing volume and click the add override button at the bottom of the volume settings. Select Post-processing – Bloom. By enabling the check boxes next to threshold and intensity and setting them to 0.3 and 1.5 respectively you should be able to see the bloom taking effect in your scene.

**Create the blackhole effect**

**Black Hole Inner**

1. The first thing we need to do is create an empty game object to hold our effect by right clicking in the hierarchy and selecting Create Empty. Rename this empty game object to Black\_Hole\_Effect. This empty game object will hold all the components of our effect, making it easy to move and rotate the whole effect, as well as giving us a place to attach scripts in the future should we need to. Finally, we need to reset the transform of our empty by left clicking the ellipsis on the right-hand side of the transform component.
2. Now we can right click our Black\_Hole\_Effect empty and select Effects – Particle System. Rename this particle system to Black\_Hole\_Inner.
3. Next set values as follows.

**Inner accretion disk**

1. Next, we will right click our Black\_Hole\_Effect empty and create a new particle system, as before. Rename this one to Inner\_Accretion\_Disk.
2. Now enter the following values.

**Outer accretion disk**

1. We can no duplicate the Inner\_Accreiton\_Disk effect and edit the values to be as follows.

**Sprite mask**

1. In order to stop the particles from the inside of the black hole being visible from certain angles we can add a sprite mask to hide them. Using any art software you like, (I am using Affinity Designer) create a square image of 512x512 pixels.
2. Fill the background layer in black so we can use “Alpha source from grey scale” in Unity for our transparency.
3. Using the ellipse tool create a circle that fills the page and select white as the fill colour. Then create a smaller circle within it and select black as the fill colour.
4. Export this image as a .png file and import it into your Textures folder in Unity.
5. Select the sprite image and in the inspector, set the “Texture Type” to “Sprite (2D and UI)” and set the “Alpha Source” to “From Grey Scale”. Now click “Apply” at the bottom of the menu.
6. Drag this sprite into your Black\_Hole\_Effect empty and align it so that the central hole is tight to the event horizon of your black hole. This will hide the particle from being viewed from certain angles.

**Inner cone**

1. For this next step we will need to use some kind of 3D modeling software to create a simple cone shape. I am using Blender, as it is awesome and free!
2. Create a new project and delete everything in the scene.
3. Now pressed Shift + A to add a new object and select a cone shape. Left click the cone and press Tab to enter edit mode.
4. Now choose the face select option using 3. Left click the circular base of the cone and press X to delete the face.
5. Now go to File – Export - .FBX.
6. Import this mesh into Unity and we can add it to our effect empty.
7. Rescale the cone so the open end fits tightly around the event horizon of the black hole and is directed into the effect, containing the inner effect.
8. Finally, we just need to set up a material that will work well with our inner cone and create the illusion of the inside of a black hole.
9. Right click in your Materials folder and create a new material.
10. Enter the following values in the inspector. Ensure that in the Surface Inputs the Base Map and the Specular Map are both set to black.
11. Lastly drag and drop the material onto the cone in the scene view to apply the new material to it.

**Post-processing**

1. One final thing to really make this effect pop is to add some post-processing to the scene.
2. Select your post-processing volume and set the values as follows.

**Final thoughts**

This effect was tricky to make and took a lot of trial and error to figure out. I am happy with the stylized design and while it does not follow the laws of physics, it does create an interesting effect that looks great in the game.