# Particle Effects Lab

## Table of Contents

# Lab Overview

Ah, particle systems. What a great invention. This lab will be a brief introduction to Unity's uber powerful particle system component. Although Unity's particle system is great because of all that it can do, beginners will often find it a bit scary. The goal for this lab is to introduce you to some of the ways you can use Unity's particle system to create effects you want, as well as give you some intuition as to how you can go about creating a particle system.

# Lab Instructions

Before we start creating a particle system, we should have some idea as to what we want to make. In this case, we want to make lightning, and with this goal in mind, we can actually start creating it!

Go ahead and create a particle system by right clicking the Hierarchy and selecting Effects>Particle System. Reset its Transform so that it has a position of (0, 5, 0), rotation of (0, 0, 0), and scale of (1, 1, 1). At this point, you should have something that looks like this:
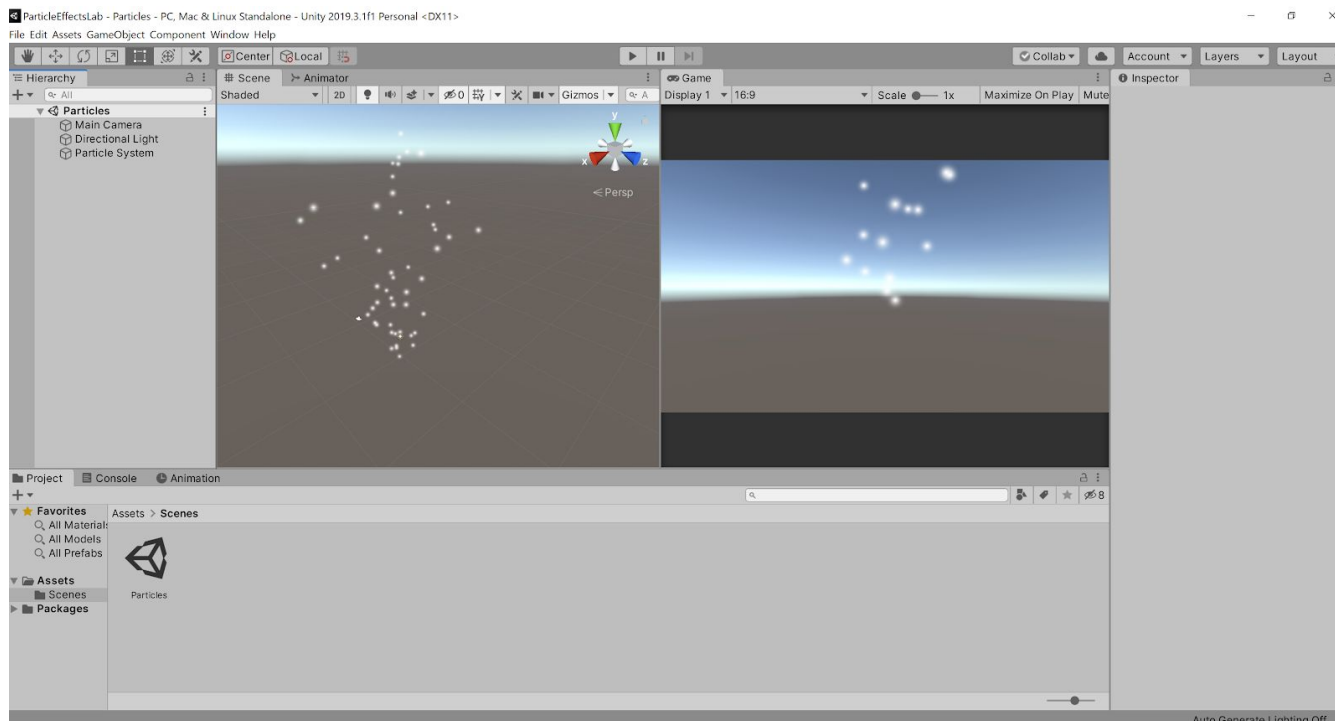
We're aiming for something that looks like this:



We have a long way to go…

Before we start, I just have to mention one thing. Randomness is your best friend when working with particle systems. You will see as we work through this that I use randomness nearly everywhere. Randomness makes particle systems feel a lot more realistic in general, and I've found that when you don't use it in certain places, it just looks unnatural.

So, let's start by setting the speed. Notice that right now we have a field called Start Speed set to 5. Let's go ahead and set it to 0 because we will actually use a special module to control the speed. What a module, you ask? Well, a module is one of these things attached to the particle system component. Some of these modules are Emission, Shape, Velocity over Lifetime, Color over Lifetime, Noise, Sub Emitters, and Renderer. If you want to know what each of them do, hover your cursor over the name and Unity provides a good explanation for the behavior of each field.

| Particle System | |
|---|---|
| ✓ Emission | |
| ✓ Shape | |
| ☐ Velocity over Lifetime | |
| ☐ Limit Velocity over Lifetime | |
| ☐ Inherit Velocity | |
| ☐ Force over Lifetime | |
| ☐ Color over Lifetime | |
| ☐ Color by Speed | |
| ☐ Size over Lifetime | |
| ☐ Size by Speed | |
| ☐ Rotation over Lifetime | |
| ☐ Rotation by Speed | |
| ☐ External Forces | |
| ☐ Noise | |
| ☐ Collision | |
| ☐ Triggers | |
| ☐ Sub Emitters | |
| ☐ Texture Sheet Animation | |
| ☐ Lights | |
| ☐ Trails | |
| ☐ Custom Data | |
| ✓ Renderer | |

Start Speed is a field in the Main module. Top open and close a module, just left click on it.

Go ahead and open the Velocity over Lifetime module. Everything should be greyed out at the moment, and that's because we haven't activated this module yet. In order to activate, click on the little circle to the left of the module name. Hopefully you now have something like this:
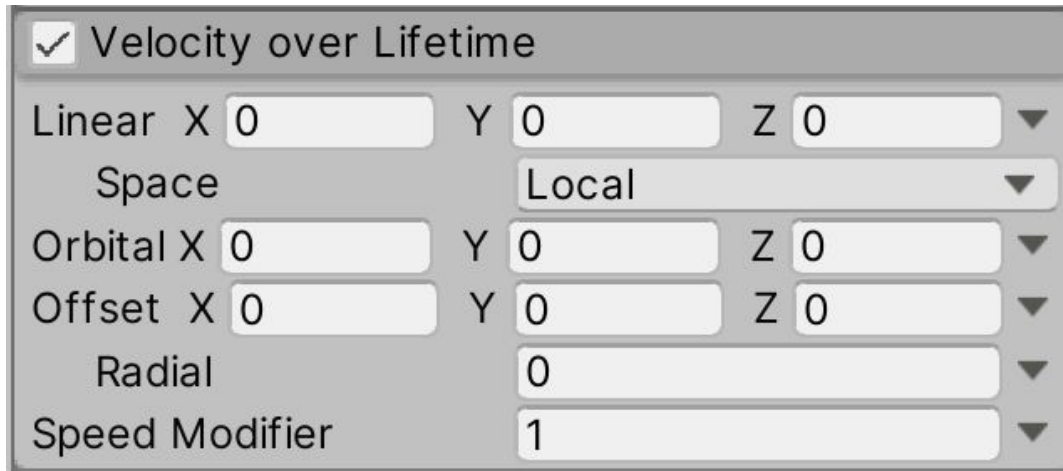


This might be a little bit intimidating but don't worry. The only field you should care about is the Linear field. Remember randomness? Well, let's begin using our new friend.Click on the down arrow at the right of the Linear Z field and then select Random Between Two Constants.

Lightning is fast and travels down. Therefore, let's set Y to -15 and -20 for now as they're large negative numbers. It doesn't matter what order you place the numbers, but it helps to be consistent so in my case, I'll try to always put the smaller number before the larger number.
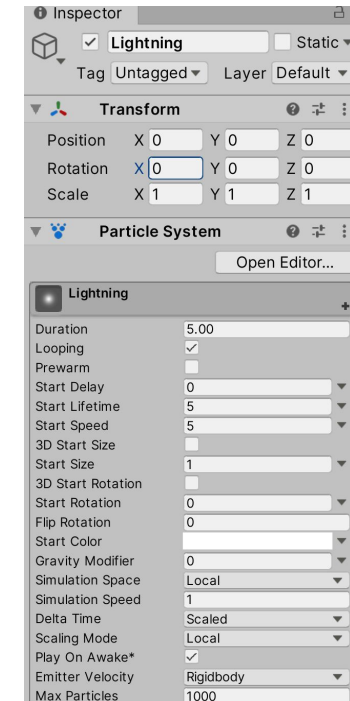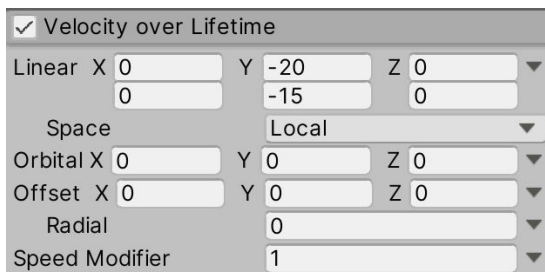
The next step I usually take is setting the Shape module. This module allows us to set the type of shape that the particles spawn from. By default it's a cone. This doesn't look very good though, so let's open the Shape module and set the Shape field to Hemisphere and its X Rotation to 90.

Now it's time to give the particle an actual lightning-like shape. This is going to be a bit more complicated than the previous parts because it requires tweaking a few modules instead of just one.

The modules we will work with are the Main module, Trails, and Renderer modules, so go ahead and open them up. Let's start with the Renderer. As it stands, we have white balls flying down. Let's get rid of this by setting the Render Mode field to None (by default it should be set to Billboard). Next, let's go ahead and set the Trail material to Default-Particle. We shouldn't see anything right now, but if we activate the Trail module, we can see the path that our particles take. The path is drawn using the Trail Material that we just selected, so if we choose a different material, then the trail will also look different. If there's nothing selected, a pink path would be displayed (that's the default for "no image" lines).
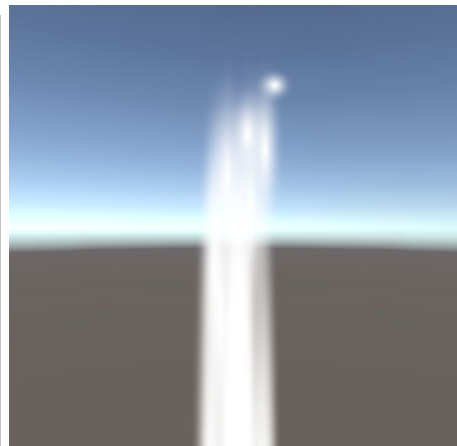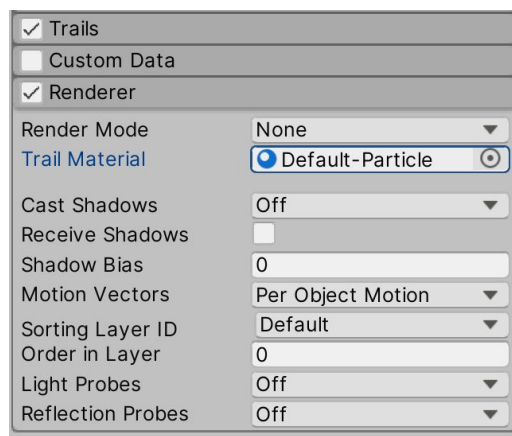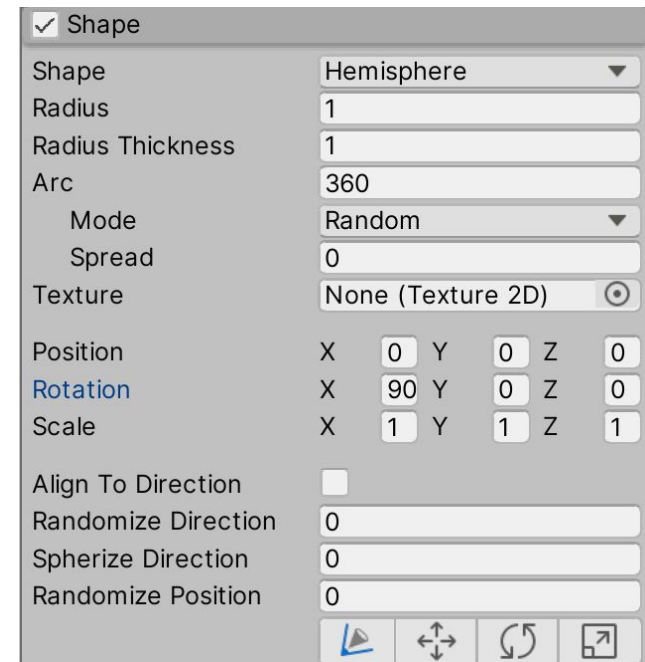
Now let's look at the Main module, and find the Start Size field. Like the Velocity over Lifetime module, make this into a Random Between Two Constants. Set the first field to 0.1 and the second to 0.2 to make our particles skinnier.

Next, we want to give the lightning its signature look, and we'll use the Noise module to do that. This one essentially modifies particle fields of each particle based on the values provided. Right now, the particles are wavy. Let's see if we can make them a bit more angular. We can do this by increasing the Strength field to 6.

Now, let's go back into the Trails module and change the Minimum Vertex Distance to 0.8 so that the space between each stored vertex is longer. Values closer to 0 will make the trail smoother, while larger values will make the trail more angular.

If you look very carefully, you'll notice that all of the particles are actually taking the same path. If you deactivate the Shape module, this will be a lot more apparent. To fix that, let's change the Scroll Speed field in our Noise module to 0.7, and the Frequency to 3.
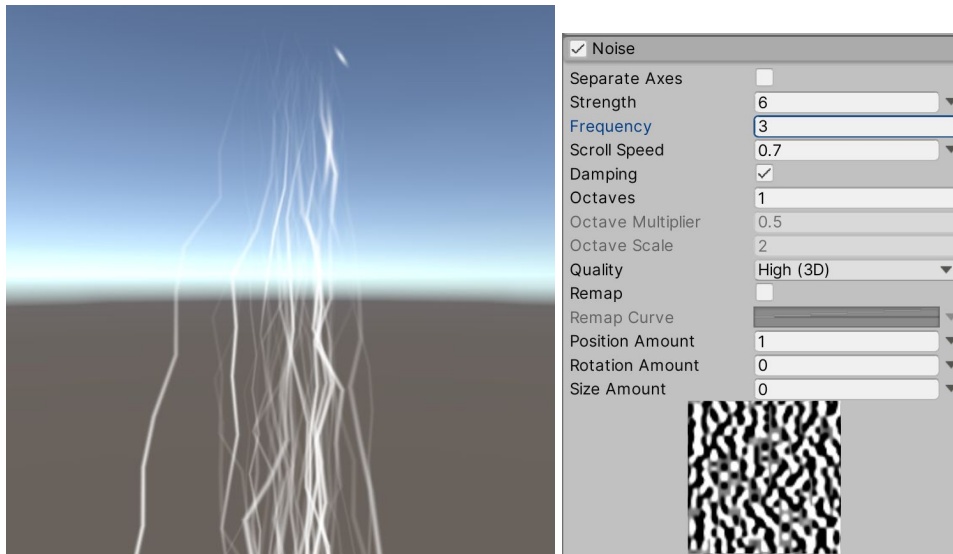


If you zoom out, you'll see that the lightning extends way down to the bottom. To fix this, let's set the Start Lifetime in the Main module to 1.5 so that every particle dies in 1.5 seconds instead of the default 5.

The next bit we'll look at is the color. The lightning in our image is a bit purplish blue, so go into the Main module, and to the right of the Start Color field, select the dropdown and choose Random Between Two Colors. You can use the color codes EAA5F1 and 9DD5E9, but feel free to use your own.

Finally, let's make the lightning spark when it collides with something. Activate the Collision and Sub Emitters modules. Since we want the lightning to collide with everything that has a collider, we'll set the Type field to World. We get this weird behavior where the lighting is bouncing upwards, so let's fix that by killing the particles soon after a collision and greatly decreasing how much it bounces. To kill it, set Lifetime Loss to 0.6; this removes 60% of the particle's total life. In our case, 0.6 * 1.5 = 0.9 seconds. Let's decrease the Bounce to 0.1 and increase Dampen to 0.95.

Now for our Sub Emitter module. Switch Birth to Collision, and click the plus icon to the right.
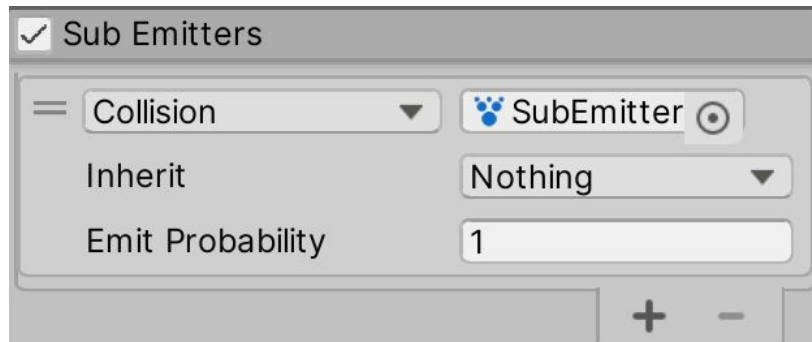


We just created a new particle system as a child of the lightning GameObject! Go into that new particle system and decrease the Start Lifetime to something small like 0.2 and 0.5 (using Random Between Two Constants). Let's also set the Start Size to be between 0.1 and 0.2.

Finally, go to the Color over Lifetime module. This one controls the color of each particle throughout its lifetime. Activate the module and then click on the color to select a new one. You'll notice that this is not the usual color picker window we get; this is actually the gradient window. On the left is the color and the alpha that the particle starts with, and the right is how it will look at the end of its life.
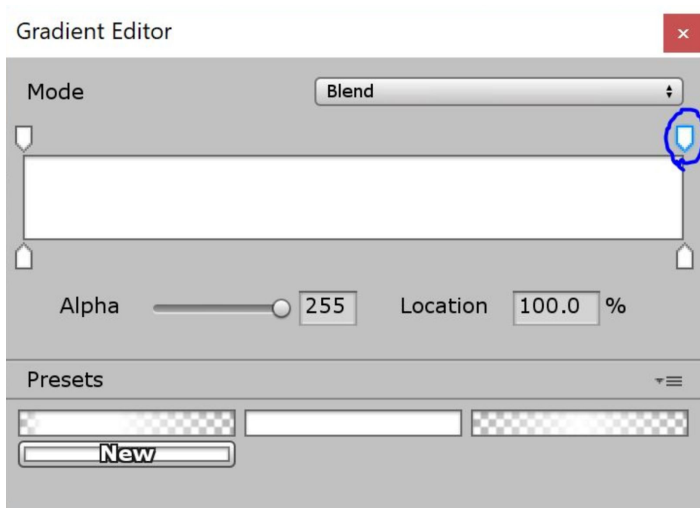
Click on the right pointer, and set the Alpha field to 0. This way, the particles from the explosion will gradually become more transparent until they are completely gone. This makes it seem more natural. In the Gradient Editor, the top pointers tell the alpha value of the particle at that point in its life while the bottom pointers tell the color value of a particle at that point in its life.

And we're done!

# Lab Summary

Unity's particle system may seem a bit scary because it has so much to it. The best way to approach creating a particle system is to begin by having a good image of the end product in your mind, and then figure out which modules to activate and what fields to modify to reach your end goal.

# Checkoff

- Make a rain effect

    - You are not allowed to set Render Mode to None (but you can use Trails)

- Create your own effect

    - Remember to have an idea of your end goal, so that you know what steps and what modules to use

# Challenge

Make a fireball that uses at least 4 particle systems!