An introduction to unity particle system

Content

Introduction to particle systems

* Introduction

Adding a Particle System

* Creating a particle system
* Preview Panel
* Particle effect View

Particle system properties

* Properties
* Toolbar
* Curve editor

Particle system Modules

* Emission module
* Shape module
* Velocity over lifetime module
* Limit velocity over lifetime module
* Colour over lifetime module
* Colour by speed module
* Size over lifetime module
* Size by speed module
* Rotation over lifetime module
* Rotation by speed module
* Collision module
* Sub emitters module
* Texture sheet animation module
* Renderer module

Particle system Scripting

Introduction to a particle system

Introduction

Particle effects are integral component to a game or graphic that when used correctly can add a subtle bit of realism that draws users/viewers into the content allowing them to feel and see what you are trying to convey but better.

A particle system is a technique in games or graphics that uses many graphical objects to simulate highly chaotic systems such as natural phenomena or processes caused by chemical reactions.

Some of these graphical objects include sprites and 3D models that can be simulated in a way that can replicate these chaotic systems much better that more conventional rendering techniques.

Particle systems are defined as a group of points in space guiding by a collection of rules defining behaviour and appearance.

In unity the term Particle system and Particle effect are interchangeable however these two terms do have different meanings

A particle system is a single particle system component that is attached to a GameObject whereas a particle effect refers to a hierarchical composition of GameObjects each with their own Particle system component. Basically, a combination of Particle systems which together compose a Particle effect.

Hopefully by following this document you will be able to utilize the unity particle system.

Adding a Particle system

Creating a particle system

You can add a particle system to your scene in multiple ways, you can

* Create a particle system GameObject
* Create a GameObject and add the Particle system component

To create a Particle system in your scene you can either create a new particle system GameObject by selecting the Particle system under GameObject > Create Other > Particle System from the menu on the top bar as seen in the picture below

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generatedThe other way you can create a Particle system is to create an empty GameObject and adding a Particle System component to it, you can do this by selecting the Particle System under Component > Effects > Particle System from the top bar as seen below

Preview Panel

When you select your GameObject that has a particle system attached to it the preview panel will appear in the scene view

Graphical user interface, text, application

Description automatically generated

This panel allows you to pause simulate and stop the particle effect simulation, while the effect is being simulated or paused you can also change the value of playback time which changes how long the particle effect will appear.

Particle effect view

You can open the particle effect view by selecting Window > Particle Effect from clicking the open editor button on in the particle system component in inspector or can also be accessed from the main menu as seen below

Graphical user interface, application

Description automatically generated

The particle effect view consists of three primary panels.

First the toolbar that provides buttons that can simulate, pause and stop the particle effect in the scene view

Secondly the system editor that provides access to the properties of the particle system

And finally, the curve editor that allows you to manipulate the values of the particle system properties using curves

Particle system properties

Properties

Graphical user interface, text

Description automatically generatedUsing either method to create a GameObject with a Particle system component attached to it. In the Inspector you will see the Properties below

Here we will break down what each option does:

Duration – The duration of the effect if looping isn’t checked

Looping – if checked the effect will loop

Prewarm – only works if the effect is looping and will simulate a full cycle of the effect before enabling the animation for instance it could mean the difference between a fire hydrant blasting water high into the air or building from a small fountain to a large one over time.

Start Delay – the delay in second it takes until it starts when triggered

Start Lifetime – the initial lifetime in seconds for the particles the particle is destroyed after this time is elapsed

Start Speed – the initial speed of the particles the greater the speed the more spread they will be

Start Size – the initial size of the particles

Start Rotation – the initial rotation of the particles

Start Colour – the initial Colour of the particle

Gravity Modifier – How much gravity will affect the particles, scales of the gravity value set in Unity physics manager

Inherit velocity – will apply a velocity once to the particles and any changes to this velocity will not affect that particles after it has been made

Simulation space – moves the particles in local space along with the particle system, world space particles move freely once emitted.

Play on Awake – Starts emitting immediately when enabled. If this is turned off you will have to manually activate the system with a script

Max Particles – the upper limit of how many particles can be alive at once for the system

Most of these properties you can change the value they are effected by to one of the following types:

Constant – a single numeric value that is used over the entire duration of the particle system simulation or over the lifetime of the particle

Curve – the value of the property will be determined by a value on a curve at the time during the particle system simulation or at a time during the lifetime of the particle

Random between two constants – chooses a random value from a min and max value that you have specified

Random between two curves – creates a random curve between a min and a max curves the current value will be chosen based on the current time of the system simulation or the lifetime of the particle.

These can be used on the colour and numerical properties of the system

Toolbar

The tool bar is at the top of the particle effect view and allows you to control aspects of the particle effect some of these aspects are preview controls, show: all/selected, resimulate and wireframe.

Preview Controls:

The pause simulate and stop buttons on the toolbar allow you to start and stop the effect in the scene view and game view. These operate exactly the same way in the particle effect preview panel.

Show: all/selected

This button will toggle the view of the particle system shown in the particle effect view and in the scene view. With show: all selected the particle system components will be visible in the particle effect view and in scene view. With show: selected only the current selected effect will appear.

Resimulate:

Is enable the particle effect will be updated when changes to the properties are made and will immediately visible in the scene view. If not enables you will need to manually resimulate the effect to see changes.

Wireframe:

Allows the user to preview the particle effect in the scene view with wireframes turned on for the particles it will also show the screen space bounding box for each particle in scene view.

Curve Editor

The curve editor is used to manipulate values for properties of the particles, in the image below the start size property of the particle is selected and viewable in the curve editor and is selected to be random between two different curves using max and minimum value, you can see the two dots that you can move to alter the curves to make small adjustments for your effect.

Particle system modules

Emission module

The emission module defines the rate at which particles are emitted and allows you to define particle bursts when a large number of particles are emitted at certain intervals over the duration of the effect.

The module defines 2 properties rate and burst. Rate effects the number of particles emitted if the dropdown is set to time, then this parameter determines the number of particles that are emitted per second, if set to distance then it determines the number of particles that are emitted per world unit per meter.

Burst allows you to specify specific intervals at which extra particles should be emitted, time of the burst is measured in seconds and the particles property determines the number of particles emitted at a time, burst intervals can be added or removes with the + or – buttons.

Shape module

This module detemines the shape of the area that emits particles. The shape parameter can be one of the following values

First of is Sphere where the shape of the particle is a 3D sphere, secondly is Hemisphere where the shape of the particle is a half sphere where the hemisphere is always in the local positive Z axis relative to the GameObject

Third is a cone, the cone is always in the direction of the local positive z axis, next is the shape of a 3D box and finally the last is a Mesh where the particles will be emitted from either the vertex, edge, or triangles of a mesh. The mesh can either be an asset in the project folder or a mesh that had been placed in the scene

Velocity over Time Module

This module allows you to animate the velocity of the particle over its lifetime, this module has a few properties one is its coordinate values that simulate the direction of the velocity vector in the format of X,Y,Z and can is determined by numerical properties.

Secondly it has another property which is space which determines if the vector is in the Local or World space.

Limit Velocity over Lifetime Module

The Lvol module allows you to Limit the max velocity of the particle over its lifetime, it has 3 properties which are a toggle called separate axis which if enabled will then limit one of the x,y or z directions to certain speed depending on the users choice, the one property is the same as the module above with a coordinate vector and a speed property that limits the overall velocity of the particle.

Force over Lifetime Module

This module is used to apply a force to a particle that effects its velocity this can then be used to create a particle that can be influenced by other vectors becoming a turbulent particle. This particle has 3 properties the first one is another coordinate force vector then following is a space property that determines if its local or world space. Third there is a Randomize property that if enables will randomize the force vector on the particle between 2 values that the user has specified

Colour over Lifetime Module

This module allows you to specify the Colour of the particle over its lifetime by simply selecting a colour when clicking the drop down in the menu.

Colour by speed module

This Module will change the colour of the particle based on its speed. It has two properties first is colour which can be a gradient or random betwwen two gradients and secondly is speed range that maps the speed of the particle to colour

Size over lifetime

The size over lifeitme module allows you to control the scale of the particlles over their lifetime, this is defined by a single parameter and can be any of the methods in the numerical properties.

Size by speed

This module allows the user to specify the scaled size of the particle based on its speed, the module has 2 properties one is the size of the particle that it will scale with and secondly the speed range that maps the minmax size to a value on the speed curve.

Rotation over Lifetime

This module allows the user to control the angular velocity of the particle over its lifeimte, this module has only one property called angular velocity which controls the rotation of the particle in degrees per second. This value can be any of the numerical methods.

Rotation by speed

This module controls the angular velocity of the particle based on its speed. This mopdule has 2 properties which is the angular velocity and the speed range which maps the min and max speeds to a value in the angular velocity.

Collision Module

This module allows you to specify a set of transform nodes that will be used to perform collision with the particles in this particle system.

This module has six different properties. Its first property is called Planes thich is an array of a max of 6 tranform nodes that are used to define the position and orientation of invisible planes in the scene. The local y axis of the transform node is used to determine the normal of the plane and you can add up to 6 planes to this life by clicking the + and remove them by clicking the – buttons on the panel.

Second property is called the dampen which allows you to control how much of the speed is reduced when a collision occurs from 0-1, next is called bounce which will affect how much the particle will bounce of a collision and also has a value range of 0 to 1.

Another property is called Lifetime loss which determines how much of the particles lifetime is lost when a collision occurs so if you want a particle to die when it collided then this parameter is set to 1.

The last two properties are called visualization and Scale Plane, Visulization parameter will draw planes in the scene view and game view to help you determine where the planes are located relative to the particle system. The other one, Scale plane, if used to determine the scale of the virtual plane.

Sub Emitters Module

The sub emiiters module will allow you to spawn additional particles when a particle hits a certain check, for example if the particle is emmited, dies or collides with a plane.

This module has 3 properties all linked to each other which allow the user to specify what to do if the particle hits and of those checks

Texture Sheet Animation Module

The Texture sheet animation module will allow you to animate the texture coords of the particle. This module uses the texture size assigned to the marterial property of the renderer module.

This then allows you to create a sprite sheet of animated sprites for animated textures for the particles to use.

This module has 6 properties, first is tiles which is the number of tiles in the x and y directions for the texture, second is animation to determine if its animated as a whole sheet or a single row.

Third is random row which if checked will select a random row in the texture to be used. Next is Row and if random row isnt checked it will determine which row will be used for the particle, another property is Frame over time which determines if the frame to use over the lfetime of the particle.

Finally the last property is Cycles and determine how many times the particle will loop a texture over its lifetime.

Renderer Module

Finally we have the last module the renderer module that determins how the particles are rendered on the screen.

This render has a lot of properties so well go through them one by one

First is Render mode that has 5 sub properties billboard, streched billboard horizontal billboard, Verticle Billboard and Mesh. Billboard means the particle is rendered as a screen alligned quad meaning the particle will always face the viewer.

Streched bill board means the particle will be screen aligned and will be stretched according to 3 properties. Camera scale determines how much of the speed of the camera is taken into consideration whne stretching the particle. Speed scale is the scaling of thae particle along the particles velocity and finally the length scale determines the the length of the particle relative to its width.

Horizontal and verticle billboard means the particle will be alligned to a Horizontal/Verticle plane in the World space.

Mesh renders each particle as a mesh. Another property of the module is called material which determines which materila the render used to render the particle.

Sort mode sorts the particles based on 4 choices, None meaning they arent sorted, Distance means the particles closer will be drawn after tparticles further away, youngest first meaning older particles will be drawn over younger ones and vice verse for Oldest First.

Sorting Fudge will determine the order of particle systems drawn first.

Cast shadows will enable particles to emit shadows on other objects. Receive shadows does the oposite meaning shadows can be cast onto the particles this is escpecailly helpful when using a mesh particle.

Max particle size if the last property which sets the maximum size a Particle can be.

Particle System scripting

Our final topic is the scripting that can be done along side the particle system and allows users to have another level of customization when it comes to the particle system.

When using scripts you can access properties through ParticleSystem.main and to acces module properties you can user ParticleSystem.Module where module is the module you would like to edit.

Every module has special multiplier properties that allow you to change the effect of a curve without having to edit the curve itself for instance ParticleSystem.emission.rateMutiplier controls the effect of the emission rate of the given particle system.

When scripting for particle systems unity supports shorthand for constant values meaning you can shorthand your code and unity will fill in the rest.

For example instead of:



You can write:



Table

Description automatically generatedBelow is a list of coding systems scripting uses:

Text, email

Description automatically generatedGraphical user interface, text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

These screen grabs are all from the Unity documentation on Scripts:

https://docs.unity3d.com/ScriptReference/ParticleSystem.html