

Overview

This article breaks down the parameter categories that make up the particle editor and are used to customize particles.

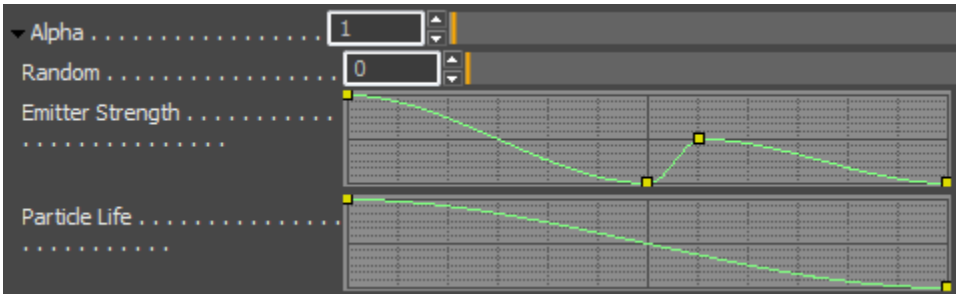
Before proceeding with this article, you should familiarize yourself with the [Particle Editor Overview](#) article which covers the basics of the Particle Editor interface.

For advanced techniques, please see [Particle Editor Advanced Techniques](#) and the [Particle Tutorials](#) section.

- [Common Functions](#)
- [Emitter](#)
- [Timing](#)
- [Location](#)
- [Angles](#)
- [Appearance](#)
- [Lighting](#)
- [Audio](#)
- [Size](#)
- [Movement](#)
- [Rotation](#)
- [Collision](#)
- [Visibility](#)
- [Advanced](#)
- [Configuration](#)

Common Functions

Control Curve

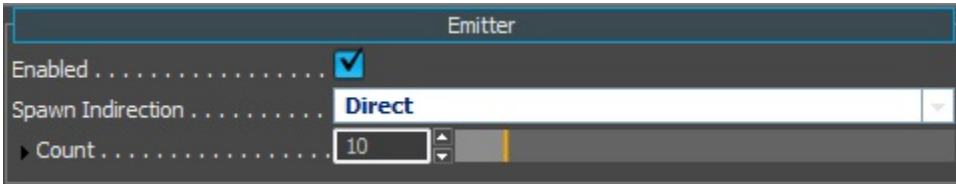


You have the following controls to edit control curves:

Function	How To
Adding a key	Double-click the spline graph.
Moving a key	Left-click a key and drag it around.
Changing a key's tangents	Select the key and press Space. This toggles the spline tangents at that key between continuous and non-continuous. Continuous tangents move more smoothly through a key (ease in/out), while non-continuous tangents are more abrupt. To make a given spline segment completely linear, set both end-keys to non-continuous.
Deleting a key	Double-click an existing key, or select it and press Delete .
Parameter	Description
Emitter Strength	Use this param to control the alpha strength (as per above example) over the lifetime of the particle. Only works with finite particles. If continuous is set, this has no effect. Above it starts out at the maximum value (1), drops to zero 1/2 way, creeps back up again then slides back to zero towards the end.
Particle Life	Use this to control the alpha (as per above example) over the individual particles life time. For example, use this to fade a smoke particle away to nothing, once its lifetime has finished. Depending on where you reduce the value to zero, the particle will fade out earlier / later.

Emitter

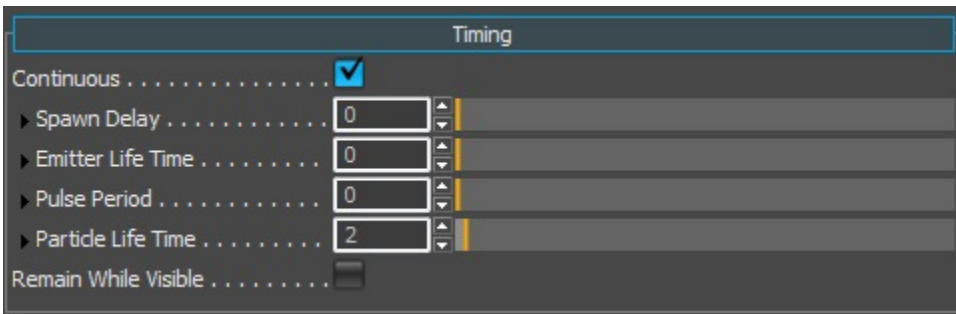
Parameters in this tab control the particle amount & spawning location of the particles.



Parameter	Description
Enabled	Enables/disables current effect.
Comment	A free-form text field for any comment on the effect.
Inheritance	<p>Specifies the source of default (starting) parameters for the effect.</p> <ul style="list-style-type: none"> • Standard: Uses the built-in (coded) parameter defaults. These are zero or off for most parameters, except for a few which serve as multipliers, and for which 1 is the more natural default. • System: (default) Reads the "System.Default" effect to use as defaults. If no such effect exists, uses <i>Standard</i> defaults. • Parent: Uses the parent particle effect for defaults. One possible use for this is to create a parent effect with one set of parameters, then a variety of sub-effects which alter some of the parameters for variation. Sub-effects can be spawned on their own. Editing the parent effect will update the default values of all sub-effects. <p>The <i>Inheritance</i> source has the following consequences:</p> <ul style="list-style-type: none"> • When you create a new effect, it takes its default parameters from the Inheritance source, which by default is <i>System</i>. • The labels of all non-default parameters are highlighted in the Particle Editor. This allows you to quickly see which values you have actually changed. • If you change the Inheritance source, it does <i>not</i> change any other parameters. However, different parameters may be highlighted, as their defaults have changed. • To actually set parameters to their default values, press the Reset icon, or Reset item on the effect context menu. This can be done just after creating a new effect and changing its Inheritance value; or at any time during editing, to reset parameters to the selected Inheritance default. (Note: the Inheritance parameter itself is not changed by the Reset command.) • When effects are saved to XML libraries, only non-default values are saved. When they are loaded from XML, the current defaults for the effect's Inheritance are used as a base. • When you edit any parameters of a parent effect, the non-edited parameters of all children (and descendents) which have <i>Inheritance = Parent</i> will be instantly updated. • To customize the default effects for your game, create a System library, and a Default effect. • If you edit the System.Default effect, and then Save the System library, the non-edited parameters of all effects and emitters will be updated. • To customize the default effects for a specific Configuration, create a child of the System.Default effect, give it any name you want, set its Inheritance = Parent (not required, but helpful), set its Configuration parameters to a subset of possible configurations (e.g. VeryHigh only), and then edit its effects. When the engine looks for the default parameters to use, it looks for the deepest effect in the System.Default family which matches the current engine configuration.
Spawn Indirection	<p>Contains 4 options:</p> <ul style="list-style-type: none"> • Direct: Spawn in the normal way. • Parent Start: Spawn once the parent has spawned. • Parent Collide: Once the parent particle has collided with an object, this is the trigger to spawn a particle with this setting. • Parent Death: When the parent particle has lived out its lifetime, this is the trigger to spawn a particle with this setting.
Count	The total number of particles at any one time that will be active. Determines emission rate (Count / Particle Life Time).
Maintain Density	<p>Increase emission rate (and particle count) when emitter moves to maintain the same spatial density as when motionless. The increase can be scaled from 0 to 1.</p> <ul style="list-style-type: none"> • Reduce Alpha: When Maintain Density is active, this will reduce particle alpha correspondingly, to maintain the same overall emitter alpha. The reduction can be scaled from 0 to 1.

Timing

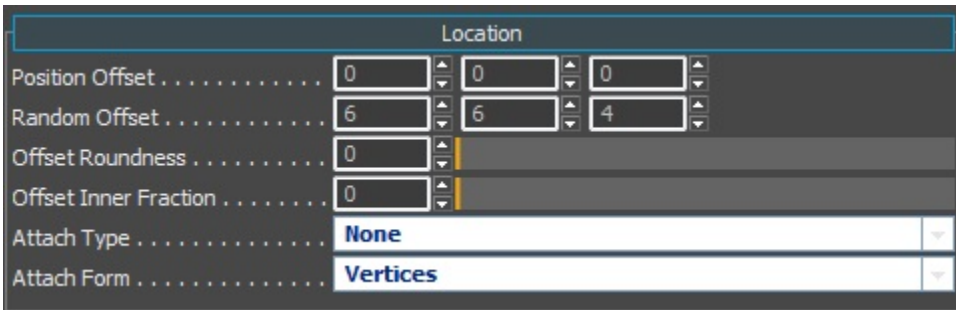
Parameters in this tab control the timing of the particles.



Parameter	Description
Continuous	If false, all particles are emitted at once, and the emitter then dies. If true, particles are emitted gradually over the Emitter Life Time. If true, and Emitter Life Time = 0, particles are emitted gradually, at a rate of Count / Particle Life Time per second, indefinitely.
Spawn Delay	Delays the start of the emitter for the specified time. Useful to delay sub-effects relative to the overall emitter creation time.
Emitter Life Time	If Continuous = true, specifies the life time of the emitter. 0 = infinite life time (default). Emitter Life Time does not apply to non-Continuous effects, which will always disappear as soon as they have emitted all of their particles.
Pulse Period	If >0, the emitter will restart repeatedly at this interval.
Particle Life Time	The life time of individual particles. Even after the emitter's life time has expired, spawned particles will live out their own life time.
Remain While Visible	Particles in the effect will not die until the entire emitter is out of view.

Location

Parameters in this tab control the spawning locations of the particles.

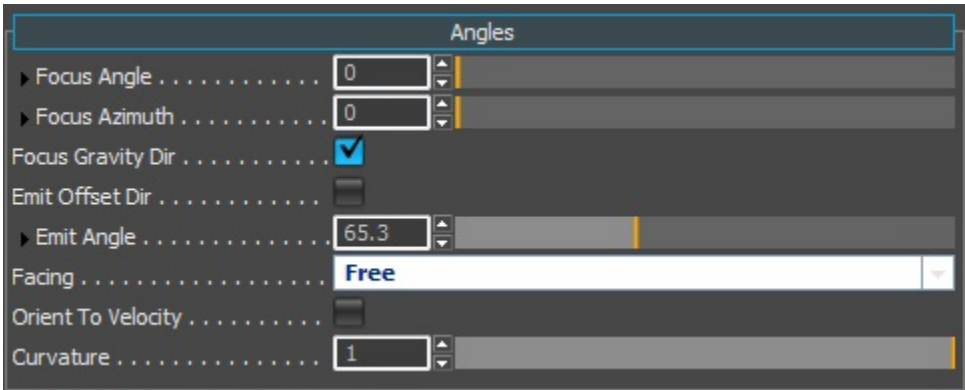


Parameter	Description
Position Offset	XYZ define a spawning position away from the emitter itself, in emitter space.
Random Offset	XYZ define the range of a random spawning box, in both directions, away from the position offset.
Offset Roundness	Fraction of spawning volume corners to round. 0 = box shape, 1 = ellipsoid shape.
Offset Inner Fraction	Ratio of inner to outer spawning volume. 0 = spawn within entire volume, 1 = spawn only at surface.
Attach Type	Specify the location of emission when the emitter is attached to geometry, or when the parent particle has geometry: <ul style="list-style-type: none"> None: Particles ignore geometry and emit from emitter center as normal. Bounding Box: <i>Deprecated</i> Physics: Particles emit from the geometry of the attached physics object (can be a mesh or simple primitive). Render: Particles emit from the full mesh of the render object (usually static or animated mesh). Generally more CPU-intensive than emitting from physics.

Attach Form	<p>When Attach Type is not None, Specifies the elements of the geometry (box or mesh) that particles will emit from:</p> <ul style="list-style-type: none"> • Vertices: Emit randomly from the vertices of the geometry. Most efficient form of Mesh emission. • Edges: Emit randomly from the edges of the geometry. Useful for effects on breaking element pieces. • Surface: Emit randomly from the surface (faces) of the geometry. • Volume: Emit randomly inside the volume of the geometry.
--------------------	--

Angles

Parameters in this tab control the focus direction (which is by default the emitter's +Y axis) and rotation value of sprites.

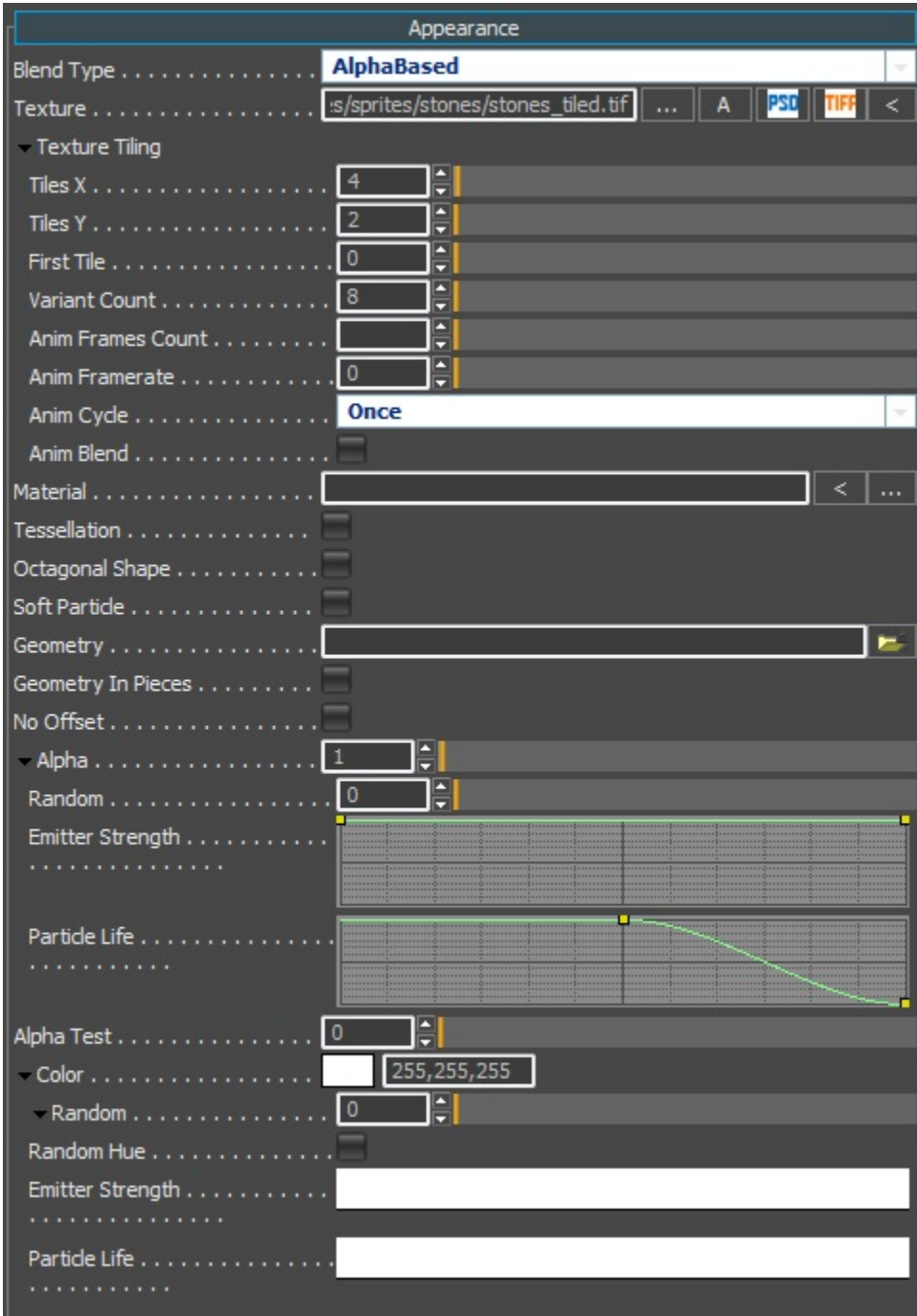


Parameter	Description
Focus Angle	The amount to rotate, in degrees from Y axis. (0 = straight up, 90 = Horizontal, 180 = straight down).
Focus Azimuth	The amount to rotate the new axis about the Y axis. (0/360 = North, 90 = West, 180 = South, 270).
Focus Camera Dir	Set focus direction to face camera. Amount can be scaled from 0 to 1.
Focus Gravity Dir	DEPRECATED (use Gravity Scale instead) Force the focus direction to world negative gravity direction (usually world +Z), ignoring emitter orientation.
Focus Rotates Emitter	If true, changing the focus direction with the above parameters also rotates the orientation of the Offset parameters. If false, Offset vectors remain in emitter orientation.
Emit Offset Dir	If true, change each particles emission direction to be aligned with its offset from the origin.
Emit Angle	The angle deviation of an emitted particle from the focus axis (default local +Y). (0 = +Y straight up, 90 = comes out horizontal to the emitter, 180 = -Y straight down). This is the maximum angle from the focus. The Random sub-parameter controls the minimum angle. <i>Examples:</i> <ul style="list-style-type: none"> • To emit in all directions, set Emit Angle = 180, Random = 1. • To emit in the top hemisphere, set Emit Angle = 90, Random = 1. • To emit in a horizontal circle, set Emit Angle = 90, Random = 0.
Facing	Applies only to 2D particles, Determine how the sprite is oriented in space. Texture orientation is further modified by rotational parameters: <ul style="list-style-type: none"> • Camera: Faces the viewer, texture X&Y aligned with screen X&Y. In this mode only, particles are assumed to represent spherical objects, are lit accordingly (see Curvature below). In all other modes, particles are lit as flat polygons. • CameraX: Rotates about local Y axis only, to face camera as much as possible. • Free: Rotates freely in 3D. (Remember to give it some rotation; the default orientation is equal to the emitter's). • Velocity: Faces direction of movement. • Water: Faces upward, moved and aligned to nearest water plane. • Terrain: Faces upward, moved and aligned to nearest terrain location. • Decal: Renders the particle as an actual deferred decal, projected onto the nearest surface. (The Thickness parameter controls the projection depth.) (Only works with Materials, does not work with textures)

Orient To Velocity	Forces Texture X axis aligned to direction of movement. Rotation parameters can rotate it further.
Curvature	For Facing = Camera particles, sets how far the vertex normals are bent into a spherical shape, which affects lighting. 0 = flat, lit like a polygon. 1 = bent into an (almost) hemispherical shape, lit like a sphere, which is the default.

Appearance

Parameters in this tab control the basic appearance of the particle. It should be setup first, because this tab includes the Texture slot, generally used for most particles.



Particles must specify *one* of either a **Texture**, **Material**, or **Geometry** asset. Texture or Material are for 2D sprite particles (or decals). Geometry is for 3D mesh particles.

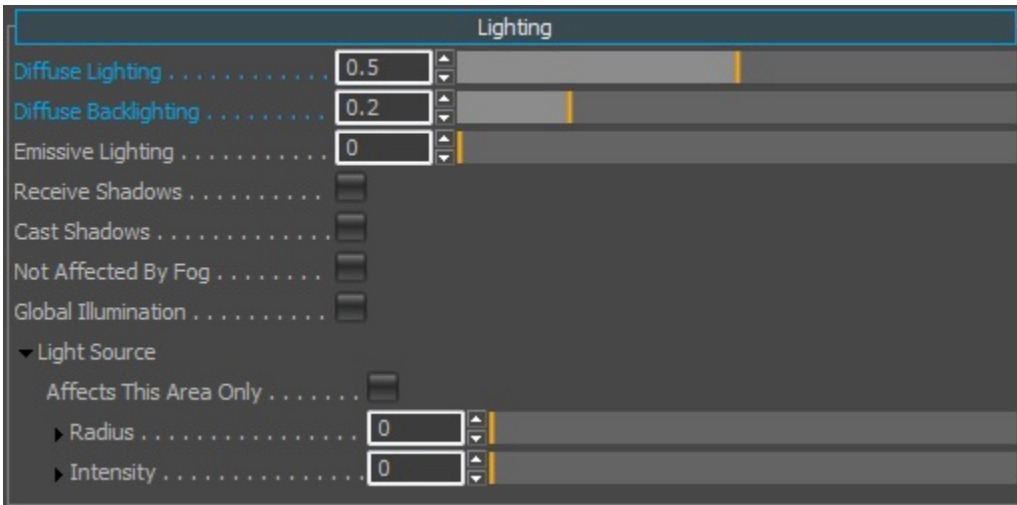
Parameter	Description
-----------	-------------

Blend Type	<p>Applies only to 2D particles. Determines how the sprite blends with the background.</p> <ul style="list-style-type: none"> • Alpha Based: Final Color = Particle Color * Particle Alpha + Background Color * (1 - Particle Alpha). • Additive: Final Color = Particle Color + Background Color. • Multiplicative: Final Color = Particle Color * 2 * Background Color.
Texture	Browse and choose the texture used for 2D sprite particles. Displays a preview of the texture when the mouse cursor is over the input box.
Texture Tiling	<p>Splits the texture into tiles, for variation and animation:</p> <ul style="list-style-type: none"> • Tiles X & Y: How many tiles the texture is split into. • First Tile: The first of the range of tiles used by this particle (numbered from 0). • Variant Count: How many consecutive tiles in the texture the particle will randomly select from. • Anims Frame Count: How many tiles make up an animation sequence. Variant Count and Anim Frames Count can be used together. For example, if Variant Count = 2 and Anim Frames Count = 8, then the particle will randomly choose between using tiles 0 through 7, or 8 through 15, as an animated sequence. • Anim Framerate: Frames per second for the animation. If 0, then the animation runs through one sequence in the particle lifetime. • Anim Cycle: If Once, the animation plays once, and holds on the last frame. If Loop, it cycles indefinitely. If Mirror, it alternates cycling forward and backward indefinitely. • Anim Blend: Render the particle blended between the two adjacent anim frames. <i>Note:</i> performance impact. • Flip Chance: Specifies the fraction of particles that will be rendered mirrored in texture X.
Material	Browse and choose the material used for 2D sprite particles.
Tessellation	<p>If supported by hardware, enables tessellation, rendering more vertices within the sprite. This is useful when Receive Shadows is set, increasing the resolution of shadows; or when Tail Length or Connection is set, creating smoother curves in connected particles. This also helps with regards to receiving light from point lights, as the lighting will be more accurate.</p>
Octagonal Shape	Renders sprites as octagons instead of quads, reducing pixel cost. Only to be used with textures that fit within an octagon, otherwise clipping will occur.
Soft Particle	Apply a rendering technique that softens the intersection between sprites and nearby objects, to prevent unnatural seams. Slightly more expensive, so use sparingly on particles that need it, eg: smoke.
Geometry	Browse and choose 3D object to use for the particles.
Geometry in Pieces	<p>If true, and the Geometry asset contains multiple sub-objects, then the geometry will be emitted in split-up pieces, one set per particle Count, originating at each piece's location in the asset. If false, each particle will use the entire geometry (all sub-objects together).</p>
No Offset	For Geometry particles, use the geometry pivot for centering.
Alpha	Controls the particle's opacity, for Blend Type = Alpha.. For Additive blending, serves as a multiplier. Normally 0 to 1, but you can push the alpha beyond 1 to achieve a stronger effect.
Alpha Clip	<p>A set of parameters to customize how the particle Alpha value controls opacity and/or alpha test values.</p> <p>Each parameter below has 2 values, corresponding to their values when particle Alpha = 0 and 1. They are interpolated for each particle with its Alpha value, and then used in the shader with the following equation:</p> $FinalOpacity = saturate((TextureAlpha - SourceMin) / \min(SourceWidth, 1 - SourceMin)) * Scale$ <ul style="list-style-type: none"> • Scale: A multiplier for the final alpha value. Defaults to (0, 1), so that particle Alpha directly scales final opacity. • Source Min: Specifies the minimum source (texture) alpha to be rendered (alpha test); values below become transparent. Defaults to (0, 0), corresponding to no alpha test. • Source Width: Specifies the feathering range of alpha clipping; 0 specifies hard-clipping, 1 soft-clipping. Defaults to (1, 1), corresponding to full utilization of texture alpha. <p><i>Examples:</i></p> <ul style="list-style-type: none"> • Default: Alpha controls opacity, no alpha clipping: Scale = (0, 1), Source Min = (0, 0), Source Width = (1, 1). • Hard clipping at texture alpha = C, no feathering: Scale = (1, 1), Source Min = (C, C), Source Width = (0, 0). • Hard clipping, controlled by particle alpha: Scale = (1, 1), Source Min = (0, 1), Source Width = (0, 0). • Feathered clipping, with width F, controlled by particle alpha: Scale = (1, 1), Source Min = (0, 1), Source Width = (F, F). • Soft clipping, test value controlled by particle alpha: Scale = (1, 1), Source Min = (0, 1), Source Width = (1, 1) • Clipping and opacity scale, controlled by particle alpha: Scale = (0, 1), Source Min = (0, 1), Source Width = (1, 1)

Color	<p>Pick the color to apply to the particle:</p> <ul style="list-style-type: none"> • Random: How much a particle's initial color will vary (downward) from the default. 0 = no variation, 1 = random black to default. • Random Hue: Causes the Random color variation to occur separately in the 3 color channels. If false, variation is in luminance only. • Emitter Strength: Define the color of the particle over the emitter's life time. Add a keyframe color by double clicking inside the panel and assigning a color. You can add multiple keyframes into the timeline by repeated double clicks. • Particle Life: Define the color of the particle over the particle's life time. Add a keyframe color by double clicking inside the panel and assigning a color. You can add multiple keyframes into the timeline by repeated double clicks.
--------------	--

Lighting

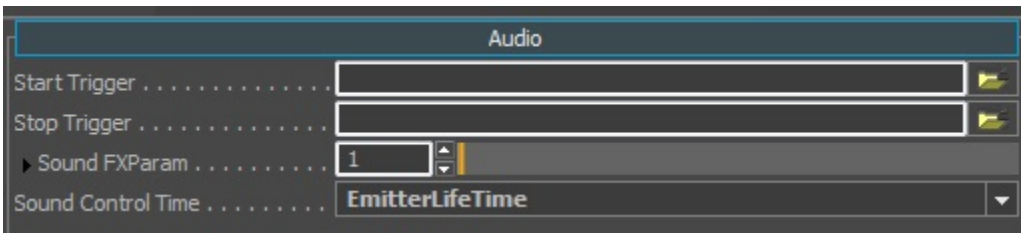
Parameters in this tab control the lighting of the particle.



Parameter	Description
Diffuse Lighting	Multiplier to the particle color for dynamic (diffuse) lighting.
Diffuse Backlighting	Fraction of diffuse lighting that is applied to unlit particle directions. <ul style="list-style-type: none"> • 0 = standard diffuse lighting, where the normals facing the light are lit the most. • 1 = omnidirectional diffuse lighting, where lights affects all normals equally.
Emissive Lighting	Multiplier to the particle color for constant emissive lighting. This can make a glowish appearance to the particle by adding some value.
Receive Shadows	Allows shadows to be cast on the particles.
Cast Shadows	Allows particles to cast shadows. This is only for Geometry particles.
Not Affected By Fog	Causes particles to ignores scene fog.
Global Illumination	Allows the particle to receive global illumination from the environment.
Light Source	Causes each particle to create a deferred light. Color is equal to the Color value. <ul style="list-style-type: none"> • Affects This Area Only: For use with Clip Volumes. When enabled, the particle lights do not exceed the volume boundary. • Radius: Radius of the light. • Intensity: Intensity of the light.

Audio

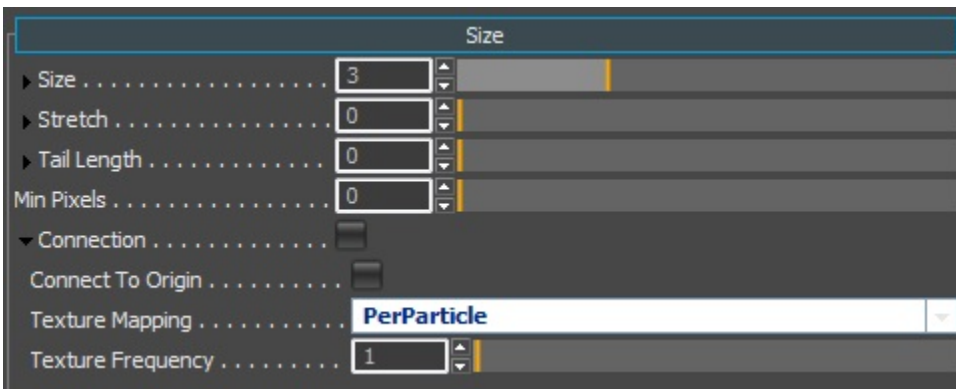
Parameters in this tab browse and control the sound files.



Parameter	Description
Start Trigger	Browse and choose the sound asset to play with the emitter.
Stop Trigger	Browse and choose the sound asset to play with the emitter.
SoundFX Param	Modulate value to apply to the sound. Its effect depends on how the individual sound's "particlefx" parameter is defined. Depending on the sound, this value might affect volume, pitch, or other attributes.
Sound Control Time	Control how long the sound plays: <ul style="list-style-type: none"> • Emitter Life Time: Plays for the length of the emitter's life time. • Emitter Extended Life Time: Plays for the length of the emitter's life time plus all particle's life times; i.e. until all particles die. • Emitter Pulse Period: Plays for the length of the pulse period.

Size

Parameters in this tab control the size and shape of the sprite.

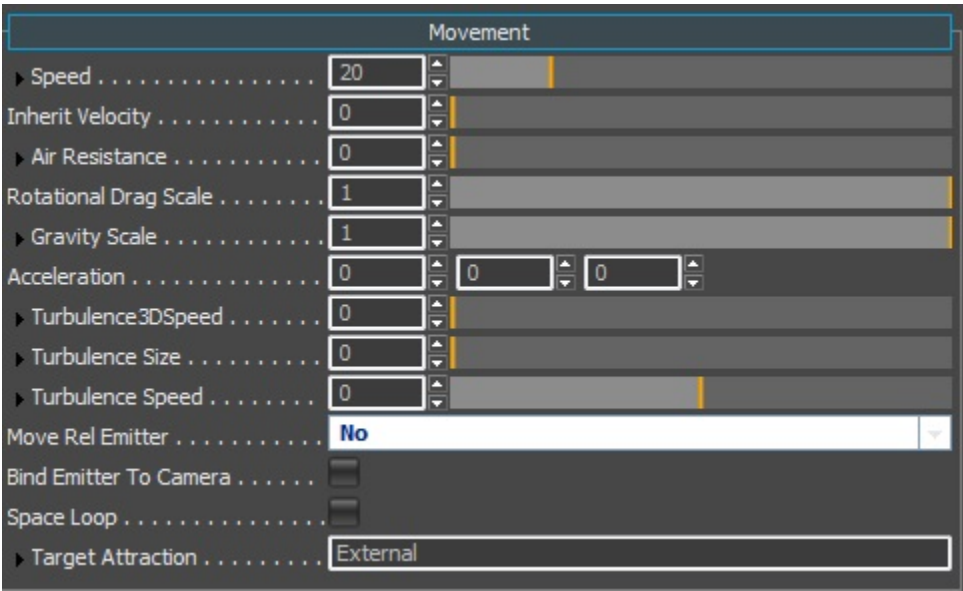


Parameter	Description
Size	For 2D particles, the world sprite radius. For 3D particles, the scale applied to the geometry.
Aspect	The X-to-Y axis ratio. Default = 1; >1 stretches in X direction, <1 in Y direction.
PivotX, PivotY	Move the pivot point of the sprite. Defaults = 0 (texture center); range +/- 1.
Stretch	The amount of stretch applied to the particle in the direction of travel, in seconds (based on current velocity). Stretches in both directions by default. <ul style="list-style-type: none"> • Offset Ratio: Adjusts the center of stretching. 0 = stretch both directions, 1 = stretch backward only, -1 = stretch forward only.
Tail Length	Length of particle's tail in seconds. Particle texture will be stretched out through the tail. <ul style="list-style-type: none"> • Tail Steps: Number of segments for tail. A higher number will produce smoother tail curves for non-linear-moving particles. If Tessellation is set, the tail will be rendered as a smoothly interpolated curve between tail segments.
Min Pixels	Adds this many pixels to particles true size when rendering. This is useful for important effects that should always be visible even at distance.

Connection	<p>Causes all particles to be rendered in a connected line, in sequence. Emission sequences separated by a Pulse Period produce separate polygons. Indirect child effects produce a separate polygon for each parent particle.</p> <ul style="list-style-type: none"> • Connect To Origin: Additionally connect the newest particle to the emitter origin, with the parameters of a particle of age 0. • Texture Mapping: This and the next parameter specify how textures are repeated over the stream. <ul style="list-style-type: none"> • Per Particle sets a default frequency of one texture per particle. • Per Stream sets a default frequency of one texture stretched over the whole stream. • Texture Mirror: Option which causes adjacent texture tiles to alternate direction; if false, they wrap at each repetition. Default = true. • Texture Frequency: Multiplies the texture repeating frequency specified above. Can be >0 or <0, which determines texture direction. <p>If Tessellation is set, the stream will be rendered as a smoothly interpolated curve between particles.</p>
-------------------	--

Movement

Parameters in this tab control the movement of the sprite.



Parameter	Description
Speed	The initial speed of particles.
Inherit Velocity	What fraction of initial velocity is inherited from the particle's parent. For indirect particles, the parent particle's velocity is inherited. For direct particles, the emitter's velocity is inherited.
Air Resistance	How much to react against the global wind speed. Units are 1/seconds. The higher the number, the faster particles will reach wind speed (or rest if no wind).
Rotational Drag Scale	Fraction of Air Resistance applied to slow down particle rotation. If 0 (default), only particle velocity is slowed, not rotation.
Wind Scale	Multiplier to environment wind, to use when computing particle drag.
Gravity Scale	Multiple of world gravity to apply to particles. Most physicalized particles should be set to 1 (use Air Resistance to provide drag). Set to a negative value for buoyant particles such as smoke.
Acceleration	The constant acceleration applied to particles, in world space (X,Y,Z).
Turbulence 3D Speed	Adds a 3D random turbulent movement to the particle, with the specified average speed.
Turbulence Size	Adds a spiral movement to the particles, with the specified radius. The axis of the spiral is set from the particle's velocity.
Turbulence Speed	When Turbulence Size > 0, the angular speed, in degrees/second, of the spiral motion.

Move Rel Emitter	<p>Forces particle motion in emitter space; particles will move with its emitter.</p> <ul style="list-style-type: none"> • No = particle moves in world space. • Yes = particle body moves with emitter, tail segments remain in world space. • Yes with Tail = both particle body and tail segments move with emitter.
Bind Emitter to Camera	<p>Forces the emitter to relocate to the main camera's position. Useful (with Space Loop) for making a rain or snow effect, which the player cannot pass by.</p>
Space Loop	<p>Particles loop within a region around the camera, defined by Camera Min/Max Distance (under the visibility tab). This is useful to make rain or snow effect, which has an effective infinite spawning area.</p>
Target Attraction	<p>Specifies if particles are attracted to a target. This causes them to orbit toward the specified target position, as controlled by the sub-parameters below.</p> <ul style="list-style-type: none"> • Target <ul style="list-style-type: none"> • External = particles attracted to a target entity, if the emitter is linked to one (default). • OwnEmitter = particles are attracted to their own emitter's origin. • Ignore = particles ignore any external attractor. • Extend Speed: Particles speed up to reach the target in their lifetime. Otherwise, they move at natural speed toward the target, and may not reach it. • Shrink: Particles shrink as they approach the target. • Orbit: Particles orbit around target when reached. Otherwise, they disappear into the target. • Radius: Distance from the target that particles either orbit around, or disappear.

Rotation

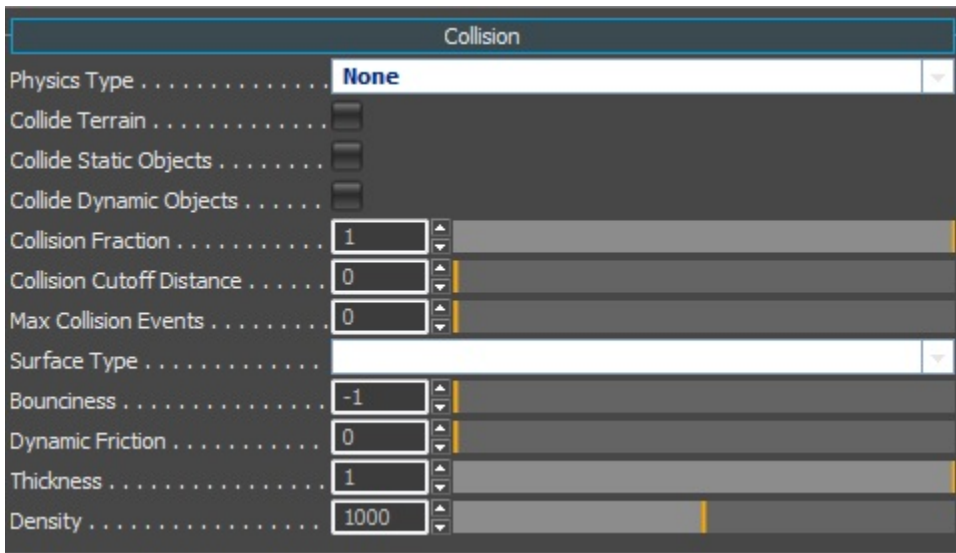
Parameters in this tab control the focus direction (which is by default the emitter's +Y axis) and rotation value of sprites.



Parameter	Description
Init Angles	The initial angle applied to the particles upon spawning, in degrees. For Facing = Camera particles, only the Y axis is used, and it refers to rotation in screen space. For 3D particles, all 3 axes are used, and refer to emitter local space.
Random Angles	Random variation (bidirectional) to Init Angles , in degrees.
Rotation Rate	Constant particle rotation, in degrees/second. The axes are the same as for Init Angles.
Random Rotation Rate	Random variation (bidirectional) to Rotation Rate , in degrees/second.

Collision

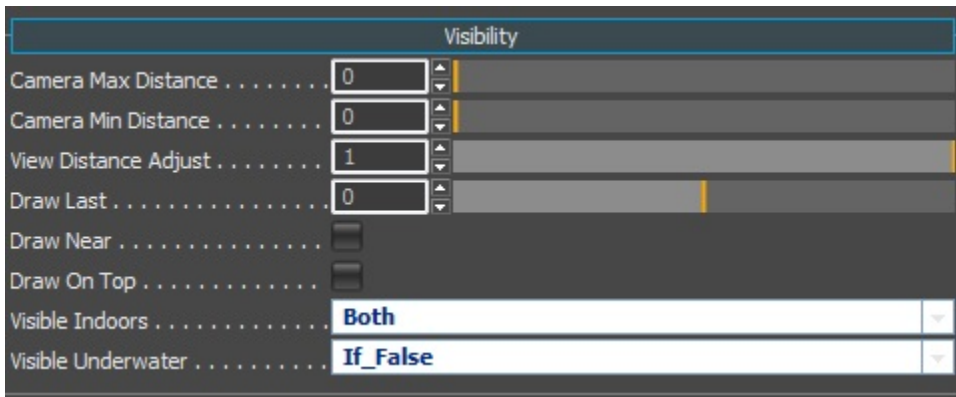
Parameters in this tab control the physical setup for the particles.



Parameter	Description
Physics Type	Select the type of physics to apply to the particle: <ul style="list-style-type: none"> • None: No collisions or other physics. • Simple Collision: Particle collides with the static environment using simple physics. This is most simple mode. • Simple Physics: Particle created as an entity in the physics system, and collides using a spherical particle model. • Rigid Body: Particle created as an entity in the physics system, and collides using the full geometry. A geometry asset must be set to the physicalized model in engine for this particle. This is most expensive mode.
Collide Terrain	Includes terrain in particle collisions.
Collide Static Objects	Includes non-terrain, static objects in particle collisions. This is expensive.
Collide Dynamic Objects	Includes non-terrain, dynamic objects in particle collisions. This is very expensive.
Collision Fraction	Fraction of emitted particles that actually perform collisions.
Collision Cutoff Distance	Maximum distance from camera at which collisions are performed (0 = infinite).
Max Collision Events	Limits the number of collisions the particle can have in its physics simulation. Only affects particles that have their Physics Type set to Rigid Body .
Surface Type	Determine surface material type for collision behavior. If set, overrides Bounciness and Dynamic Friction below.
Bounciness	Control the elasticity for collision response. Overridden by Surface Type if set. (Special value : if -1, particle dies on first collision). Only affects particles that have their Physics Type set to Simple Collision .
Dynamic Friction	The coefficient of dynamic friction. Overridden by Surface Type if set. Only affects particles that have their Physics Type set to Simple Collision .
Thickness	Control the fraction of the particle's visible radius to use for the physical radius. Only affects particles that have their Physics Type set to Simple Physics .
Density	Control the density of particle, in kg/m ³ . An example of a value which is physically correct, is for Water = 1000. Only affects particles that have their Physics Type set to Simple Physics or Rigid Body .

Visibility

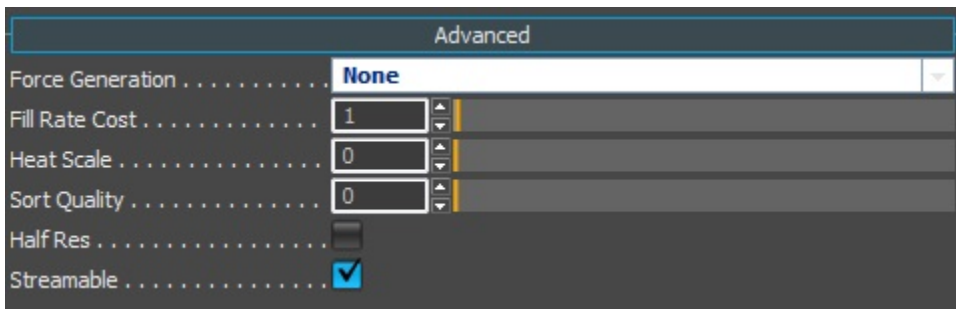
Parameters in this tab control the visibility of the particles.



Parameter	Description
View Distance Adjust	Multiplier to the automatically computed fade-out camera distance.
Camera Max /Min Distance	The camera range that particles render in. Defaults are 0, specifying unlimited range.
Camera Distance Offset	Offset the emitter away from the camera.
Sort Offset	Bias the distance used for sorting. Can be used to customise the sort order within an emitter tree: by default sub-emitters render in the order they are listed in the effect. A bias of 0.01 or greater will override that order. Larger biases can be used to adjust the sorting order with respect to other transparent objects in the level.
Draw Near	Render particles in a near 1st-person space (with weapons etc).
Draw on Top	Render particles on top of everything (no depth test).
Visible Indoors	For use in Vis Areas : <ul style="list-style-type: none"> • If_True: Hides particles when outdoors. • If_False: Hides particles when indoors. • Both: Show particles always.
Visible Underwater	For use with the Ocean / Water Volumes: <ul style="list-style-type: none"> • If_True: Hides particles when above water. • If_False: Hides particles when under water. • Both: Show particles always.

Advanced

Parameters in this tab Includes advanced appearance, movement and optimization settings.

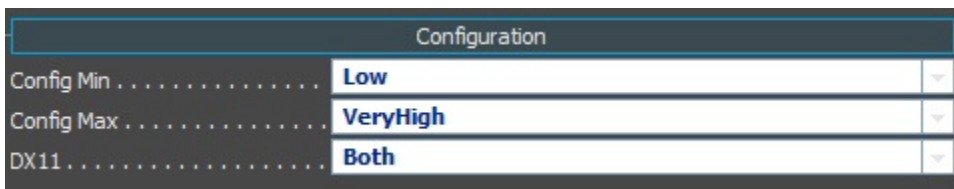


Parameter	Description
-----------	-------------

Force Generation	<p>Adds an additional force generated by the emitter:</p> <ul style="list-style-type: none"> • None: Doesn't add any additional force. • Wind: Creates a physical wind force, approximately following the velocity, direction, volume, and timing of the emitter's particles. This wind affects all particles and objects in its region, except particles in the emitter group. Setting the emitter's Speed to negative will create the wind force in the opposite direction, which can be used to create a vacuum force. • Gravity: Creates a physical gravity force, similar to the wind, but creates a gravitational acceleration force, instead of wind velocity.
Fill Rate Cost	Multiplier to this emitter's contribution to total fill rate, which affects automatic culling of large particles when the global limit is reached. Set this > 1 if this effect is relatively expensive or unimportant. Set this < 1, or 0, if the effect is an important one which should not experience automatic culling.
Heat Scale	Multiplier to thermal vision.
Sort Quality	<p>Specifies more accurate sorting of new particles into emitter's list. Particles are never re-sorted after emission, to avoid popping resulting from changing render particle order. They are sorted only when emitted, based on the current main camera's position, as follows:</p> <ul style="list-style-type: none"> • 0 (default, fastest): Particle is placed at either the front or back of the list, depending on its position relative to the emitter bounding box center. • 1 (medium slow): Existing particles are sorted into a temporary list, and new particles do a quick binary search to find an approximate position. • 2 (slow): Existing particles are sorted into a temporary list, and new particles do a full linear search to find the position of least sort error.
Half Res	Render particles in separate half-res pass, reducing rendering cost.
Streamable	Texture or geometry assets are allowed to stream from storage, as normal.

Configuration

Parameters in this tab control advanced configurations. These settings limit an effect to only be enabled on certain platform configurations. This allows you to create variant effects for different configurations.



Parameter	Description
Config Min	The minimum system configuration level for the effect. If the config is low than what is set here, the item will not be displayed.
Config Max	The maximum system configuration level for the effect. IF the config is higher than what is set here the item will not be displayed.
DX11	<p>Define if the effect should be used with the DirectX 11 renderer:</p> <ul style="list-style-type: none"> • If_True: Enables the effect only on DX11. • If_False: Enables the effect only on pre-DX11 (DX10). • Both: Enables the effect to display on both DX versions.

A practical use for these settings would be if you have the same sub-effect duplicated (but have a cheap & expensive variation) you can force either particle to play on the desired config spec, so the low spec will use the cheaper version, and the high spec will use the expensive one.

So for example, reserve a complex & crazy physics (**Rigid Body**) based sub-effect for the **Config Min = Very High**. Then have a cheaper physicalized version (**simple Collision**) based sub-effect that has a **Config Max = High**.