

Creating Particles (by MrBigzy)

This tutorial will describe all the functions in the particle editor.

****General****

= General =

Name: Name of the particle, you can add many particles to one file.

Max Particles: Maximum amount of particles that can spawn.

Particles Per Sec: How many particles spawn per second. (This will seem to be more than actual value if dead particles respawn).

Start Time Offset(s): Offset time before particles spawn after particle entity is created.

Respawn Dead Particles: Whether particles should respawn after they disappear. Used for looping particle entities.

Warmup time(s): ?

Warmup Steps/Sec: ?

= Pause =

Length(s): Min and max pause time for particle respawn?

Interval(s): Seems like same thing?

= Offset =

Position(m): Offset position of the particles on the coordinate system.

Angles(deg): Offset angle of the particles on the coordinate system.

****Start ****

= General =

Start Type - Box: Uses the coordinate system as basis for positioning.

Start Type - Sphere: Uses a sphere as basis for positioning.

= Box Start =

Min Position(m): Variable min offset of particle creation.

Max Position(m): Variable max offset of particle creation.

= Sphere Start =

Min Angles(deg): Minimum angle of sphere to contain particles.

Max Angles(deg): Maximum angle of sphere to contain particles.

Radius(m): Variable min and max radius of sphere to create particles.

Example: If the min and max angles for x and y are both 0, 180 and 0, 180, respectively, and the radius min and max is 2 and 3, particles will be created in a semi-sphere between 2 and 3 meters away from the center of the sphere.

****Movement****

= General =

Velocity Type - Box: Uses the coordinate system for velocity.

Velocity Type - Sphere: Uses a sphere template for velocity.

Note: Start Type and Velocity Type do not have to be the same!

Uses Direction: ?

= Gravity =

Note: This applies to both box and sphere type velocity.

Gravity Type - Vector: Uses a box type gravity template.

Gravity Type - Center: Uses a sphere type gravity template.

Acceleration(m/s²): Amount of acceleration in xyz directions in meters per second squared.

= Box Velocity =

Min velocity(m/s): Min variable initial velocity of particles with box type velocity.

Max velocity(m/s): Max variable velocity of particles with box type velocity. Cannot accelerate past this velocity.

= Sphere Velocity =

Min Angles(deg): Min angle of direction for particles to move with sphere type velocity.

Max Angles(deg): Max angles of direction for particles to move with sphere type velocity.

Speed(m/s): Min and max variable speed of particles.

= Acceleration =

Note: This applies to both box and sphere type velocity. It is also better to use this in place of the vector gravity type as it is more detailed, but does not cover sphere based acceleration.

Min Acceleration(m/s²): Min amount of acceleration in the xyz directions.

Speed Multiplier: Min and max speed multiplier. Can be used as a second order acceleration variable. (ie: vary acceleration two times)

Max Acceleration(m/s²): Max amount of acceleration in the xyz directions.

Maximum Speed: Variable maximum speed that can be achieved.

Note: The speed of box and sphere velocity is only the initial velocity which lasts about a 10th of a second. The maximum speed of acceleration is the speed that is achieved afterward. If the initial speed is much higher than the acceleration max speed, the particle will start out very quick and then slow down instantaneously to the accel max speed, which can be used as a good effect.

Rendering

= General =

Draw Type - Point: The particles acts as a point and always faces your direction.

Draw Type - Line: The particle will pan based on your vision, but only at a max of 180 degrees, preventing sight of the invisible back of the particle.

Draw Type - Axis: The particle will sit on a fixed axis, without panning based on your angle of vision. Will be invisible if looked at on the wrong side.

Multiply RGB With Alpha: ?

= Size =

Min Size(m): Min variable size of a created particle.

Max Size(m): Max variable size of a created particle.

= Size Fading =

Note: 1 is default size and/or length of life.

Start Relative Size: Mul size of particle when it is created.

Middle Relative Size: Mul size of particle after it reaches its middle state based on Middle Relative Start.

End Relative Size: Mul size of particle at its destruction.

Middle Relative Start: Mul of life length when middle state begins. (ie: if it is 0.3, the middle state will begin at 3/10ths of its life span.)

Middle Relative Length: Mul of life length of middle state. (ie: if it is 0.5, the middle state will last for 1/2 of its life span, and then start expanding/shrinking toward its end state, assuming there is enough

life span left for it to happen)

Color

= General =

Min Color: Variable min color of particle.

Max Color: Variable max color of particle.

= Fading =

Start Relative Color: Color at creation of particle. Make it black to fade particle in (assuming Middle Relative Start is >0).

Middle Relative Color: Color at middle state of particle.

End Relative Color: Color at destruction of particle.

Middle Relative Start: Mul of life length that the middle state color will be active.

Middle Relative Length: Mul of life length for middle state color to be active.

Life

= Life =

Life Span(s): Min and max life span of particles in seconds.

= Death =

Death Type - Age: The only death type unfortunately. Simply makes the particle die when it reaches the end of its life span.

PS Created At Death: Can be set to make another particle be created at the death of the particle. Could be used to loop particle creations.

Material

Material File: The most important function; it tells the particle what material and texture it will show!

Number Of Files: ?

Animation Length(s): ?

= Sub Division =

Amount: Specifies the amount of UV subdivision of the texture.

Type - Random: Chooses a random subdivision for a particle (if there is more than 1).

Type - Animation: Animates through the different subdivisions for a particle (if there is more than 1).

Rotation

= General =

Use Particle Spinning: Whether the particle will spin around its own depth axis.

Spin Type - Constant: Not sure the difference between the two.

Spin Type - Movement: Not sure the difference between the two.

Particle Spin Velocity Range (rad/s): Variable min and max velocity of particle spin in radians per second (6.283 radians is a single spin).

Use Revolution: Whether the particle will revolve around a center point.

Min: Minimum speed of the revolution in xyz directions.

Max: Maximum speed of the revolution in xyz direction.

Collision

= General =

Use Collisions: Whether particles will collide or not.

Max Collisions: Variable min and max amount of collisions that can happen at a given time.

Update Rate: Rate in seconds that the particles are checked for collisions.

Bounce Amount: Variable min and max amount of bounce between two colliding particles. Values higher than 1 can cause particles to bounce off their tangent.

Beam Specific

No functions exist here, probably a scrapped functionality.

That's the end of my tutorial, hope it makes it easier to create particles!

From:
<https://wiki.frictionalgames.com/> - **Frictional Game Wiki**

Permanent link:
https://wiki.frictionalgames.com/hpl2/tutorials/particle_editor/tutorial_1?rev=1305466883

Last update: **2011/05/15 14:41**



