1 Explosion effects

The following types are specialized on providing particle effects for ship and weapon explosions.

1.1 Type »Ship«

1.1.1 Required fields

Option name	Argument type	Description
$Class	**ShipClass**	This specifies the class this definition targets.
+Effect	**Effect**	The created particles will have this effect. This gets overridden when *+Weapon* is specified after this entry.
+Weapon	**WeaponClass**	When specified a weapon of the specified weapon class is created instead of a particle. This gets overridden when *+Effect* is specified after this entry Either *+Weapon* or *+Effect* must be specified.
+Number	**Number**	Range of numbers specifying the number of particles created *per second*. The real value is a random value between the two given ones. Floating point numbers will be converted to integers. *(One or two arguments)*
+Speed	**Numberlist**	The maximum and minimum speed the particles will have. *(One or two arguments)*
+Size	**NumberList**	Maximum and minimum size the particles will have. Size is given in percent of the target ship class model radius *(one or two arguments)*
+Emitstate	**Number**	A number specifying the state in which the particles will emit. These states are engine-intern values that give the current explosion state. *(Valid arguments: 1, 2 and 3)*

1.1.2 Optional fields

Option name	Argument type	Description
+Time	**NumberList**	When one number is given this will be the duration the particles will spawn. If two numbers are given the first will be the delay from the beginning of the state, the second will be the duration of particle creation. *(one or two arguments)*
+Box Min	**Vector**	If both vectors are set these vectors create a box in which the particles will be created. If *+Creation raycast* is set this box is the boundary for the ray check.
+Box Max	**Vector**	See *+Box Min*

Option name	Argument type	Description
+Spewcone	**NumberList**	Each number describes a maximum spawning angle (from $0°$ to $90°$). This angle control the direction in which the particles will move after they are created *(up to three arguments)*
+Trail	**String**	The name of a trail definition. The names are resolved after parsing of the file so it doesn't matter if the trail definition is already present when this entry is created.
+Add Velocity	**Boolean**	A Boolean specifying whether the velocity of the parent object should be added to the created particles or not.
+Offset	**Vector**	A vector specifying an offset from the centre of the parent object.
+Velocity Factor	**Number**	A number specifying the factor the parent velocity will be multiplied with. 0 disables velocity adding, 1 is default. **Attention**: Using this will override the value given by *+Add Velocity*
+Velocity X Offset **+Velocity Y Offset** **+Velocity Z Offset**	**NumberList**	This option gives control over the direction the particles will move to. The values describe a »box« where the velocity vector will point into. The length of this vector will be normalized to be 1. *(one or two arguments)*
+FPS	**NumberList**	Allows to change the FPS of the effect to add some diversity. When two values are specified the real value will be randomized to be between those two values. *(one or two arguments)*
+Creation raycast	**Boolean**	Enables or disables the usage of ray casting. This feature uses rays to get intersects with the hull of the parent objects to achieve the effect of particles that emit directly from the hull of this object. When specified it uses *+Box Min/Max* as the boundary for the ray-check. **Attention:** The boundaries have to specify a volume that intersects the hull or all ray casts will fail! If *+Box Min/Max* are not specified the *+Spewcone* values are used.
+Raycast retries	**Number**	A value to specify the maximum number of retries the ray casting does if it can't find an intersect. Default is 5.
+Name	**String**	A string that identifies the effect. Currently unused, intended to be used by the FRED interface.
+Fixed Size	**Boolean**	Specifies if the size of the +Size field should be absolute or relative to the parent ship. YES means that the size is absolute.

Option name	Argument type	Description
+Emit Vector	**Vector**	A vector which specifies in which direction the particles will move
+Emit Variance	**Number**	Specifies the variance of the created particles. Only used when *+Emit Vector* is present.
+Emit from hull	**None**	Shorthand for setting *+Creation Raycast* to *YES* and setting the *Box Min/Max* values to the bounding box of the ship model. Causes all particles to spawn from the hull of the ship
+Move outwards	**None**	If specified the particle will always move away from the creation centre (either object centre or specified offset)
+Use Ray normal	**None**	The particles will move in the direction of the normal where the ray has hit. Only used when *+Creation raycast* is set

1.2 Type »Weapon«

1.2.1 Required Fields

Option name	Argument type	Description
+Size	**Number**	Specifies the *absolute* size of the particles

1.2.2 Optional Fields

Option name	Argument type	Description
+Use Normal	**Boolean**	If set to *YES* the normal of the point where the weapon has hit will be used. The normal is a vector which is perpendicular to the polygon that has been hit.
+Absolute Normal	**Vector**	A absolute vector overriding the impact normals for the weapon impact effects. *(Only used when* +Use Normal *is set to* YES*)*
+Reflect	**None**	Makes the particles move along the reflection vector of the weapon impact. *(Only works with* +Use Normal*)*
+Armed	**Boolean**	Specifies the armed state the weapon has to be in so this effect gets triggered *(By default this will be any armed state)*
+Emitstate	**String**	Specifies the state the weapon has to be in for this effect to get triggered. The possible states are listed below (case-insensitive): • *Impact*: Triggered when the weapon impacted somewhere • *Intercepted*: Triggered when the weapon got shot down by another weapon • *Self-destructed*: Triggered when the weapon gets removed from the game without having hit anything *(Default is* Impact*)*
+Particle collides	**Boolean**	Specifies if the particles created by this effect will be checked for collisions. For performance reasons only the target of the weapon will be checked for collisions. *(Default is* YES*)*