

FireBlade Features

This is a short overview of FireBlade's main features as well as an overview of its utilities, the effect manager and several emitters.

Particle Appearance:

- Support for GL_QUADS and Point Sprites
- Billboards
- Blending function is choosable
- Color fading
- Texture fading
- Two modes for resizing during a particles life
- Two modes for color, texture and size interpolation
- Normals for a plastic appearance
- Rotation
- Smear effect with adaptive texture shifting
- Fast concurrent depth sorting, which nearly always has bestcase conditions
- Frustum culling

Particle Physics:

- Two physics models, simple and realistic
- Movements calculated with Newtons laws
- Correct air resistance calculation
- Wind
- Fast, extensible collision detection with sphere, cube, cone, face and heightmap primitives built-in
- The collision trees are generated separately and can be used for several contexts
- Simple, overwriteable collision friction model
- Magnet points with 5 predefined modes, including a realistic mass gravitation model

Miscellaneous:

- Event-driven effect creation possible, events for destroying, calculating, gravitating (magnet points) and colliding a particle
- Vertex storage choosable between Vertex Arrays, Vertex Buffer Objects or user defined storage methods
- Several parts of the calculation (physics, appearance, vertex calculation) can be disabled if not necessary to increase the performance
- Each calculation can be modified or overwritten. full access up to the particles vertices. This guarantees a maximum of flexibility
- Preallocation of memory possible

Emitters:

- **General emitter functionality**
 - Compatible with the effect manager
 - Fill delay and emitter lifespan
 - Automatic refill
 - Level of Detail
 - Not statically connected to one context, can be used to fill several contexts
 - Two Velocity modes, point orientation and controllable randomized initial velocities
 - Several renderpasses possible (with different blending functions)
- **Sphere emitter**
 - Emits particles between an inner and outer radius
 - Spawning region controllable
- **Cube emitter**
 - Emits particles in a cube with variable width, height and depth
- **Cone emitter**
 - Emits particles in or on a cone with variable base radius, top radius and height
 - When spawning on a surface, base and/or top plane can be excluded from the spawn region
 - Presumption fix can be enabled for a homogenous spawning
- **Face emitter**
 - Spawns particles on a surface described by triangles
 - The triangles may be generated the same way as in OpenGL
 - Transformation of the initial velocity by the triangles matrix can be enabled
- **Font emitter**
 - Writes texts using particles
 - The font itself is generated separately from a bitmap (so called bitmap-fonts) and can be used for several font emitters
 - Mono- and Polyspace choosable
 - Character and Linegap variable

Effect Manager:

- comfortable handling for particle effects
- Global manipulation functions for registered effects
- Frustum culling
- Depth sorting
- Efficient time delays adjustable for updating effects, which can enormously increase the performance