

Johns Hopkins
Engineering for Professionals
605.767 Applied Computer Graphics

Brian Russin

Module 7E

Particle Systems

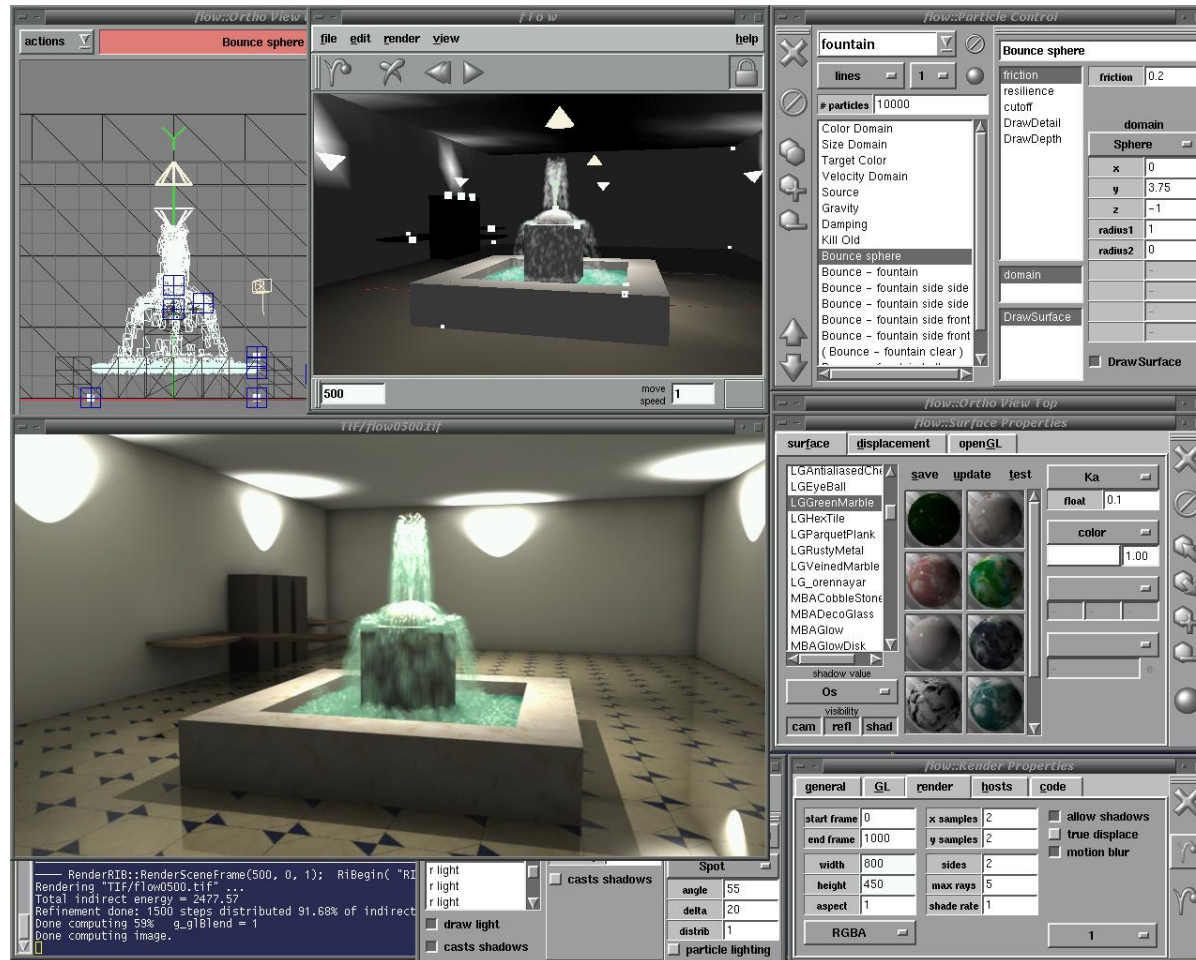


Particle Systems



<https://repository.gatech.edu/server/api/core/bitstreams/2716ad0e-99c9-4339-ae66-465ead34b8e0/content>

Particle Systems



Flow: Particle Animation Project: [link no longer active]

Particle Systems

- Particle modeling system for modeling objects with fluid like properties
 - First described by Reeves (1983)
 - Useful process for animating objects that change over time
 - Effects like flowing, billowing, expanding, etc.
 - Used for the "Genesis bomb" sequence in Star Trek II: moving a wall of fire over the planet's surface
- Object is represented as a set (cloud) of elementary particles
 - Each is born, evolves in space, and dies
 - Particle shapes can be small spheres, boxes, etc.
 - Particles move and change attributes as functions of time
 - Color, size, transparency, etc.
 - Particle systems are essentially a method of animation
- Global behavior of the particle cloud and behavior of individual particles are derived from functions
 - Consider fireworks: parabolic trajectory due to effects of gravity
 - Physics



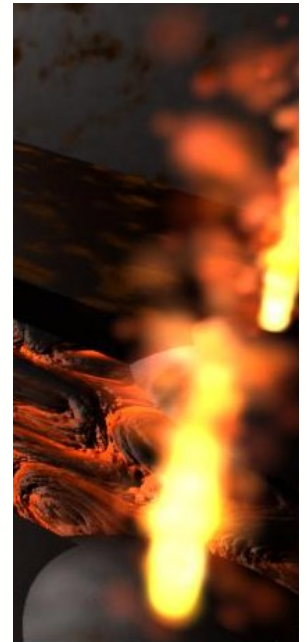
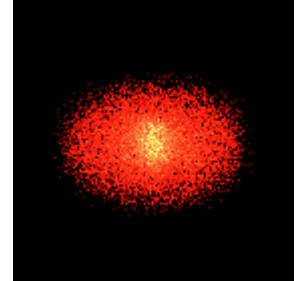
Example Uses of Particle Systems

- Early video games: bullets or exploding spaceships
 - Deterministic and placed individually
- Modeling fireworks
 - Particles traveling radially outward from the center of a sphere
 - Center of the sphere traveling along a trajectory
 - Color changes, lifetime of particles important for the effect
 - Secondary 'explosions' can be scripted
- Clumps of grass
 - Firing particles upward within a tapered cylinder
 - Particle paths are affected by gravity: become parabolas
- Waterfall and other water flow
 - Particles fall, some deflect from obstacles along the downward path
 - Particles splash - deflected upwards - when they hit the ground
- Wide variety of methods for modeling fire, smoke, explosions, and other natural phenomena

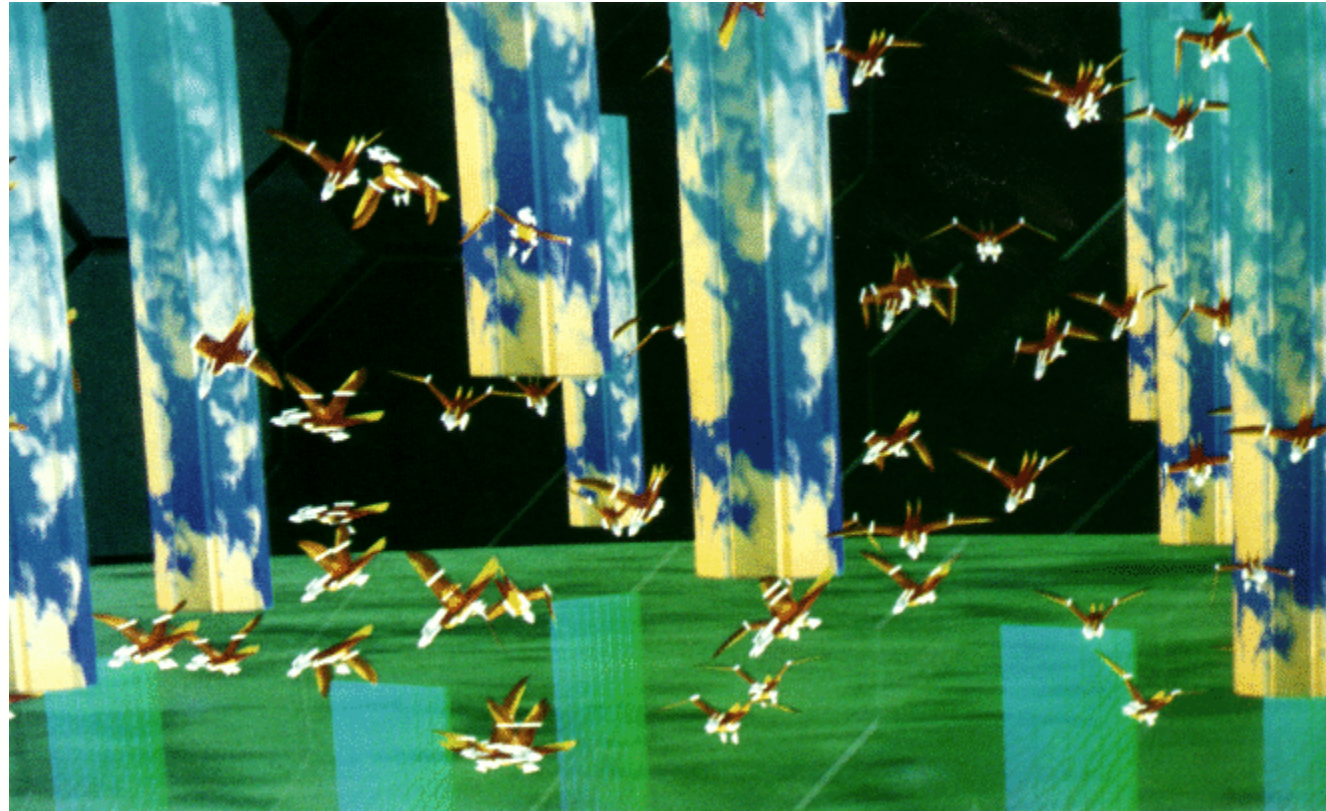


More Examples of Particle Systems

- OpenGL GLUT particle systems demo program
- On line examples
 - <http://www.cs.wpi.edu/~matt/courses/cs563/talks/psys.html>
 - Reeves method: single pixel particles, translucent particles, color of multiple particles map to same pixel is additive
 - Also discusses: modeling flocking birds, trees
 - David McAllister: Particle System API
 - <https://github.com/davemc0/Particle>
 - Torches, water fountain
- On-line articles
 - Gamasutra: (on-line content from Game Developer)
 - <https://www.gamedeveloper.com/programming/building-an-advanced-particle-system>
 - John van der Burg



Particle Systems



Flocks, Herds, and Schools: A Distributed Behavioral Model

Craig W. Reynolds

Symbolics Graphics Division

<http://www.cs.toronto.edu/~dt/siggraph97-course/cwr87/>

Implementing Particle Systems within Scene Graph

- Recommend a special node type: e.g., ParticleSystemNode
- Constructor sets initial conditions
- Draw method renders the particles
 - Remember – these should be simple primitives
 - Small triangles, points, lines
 - Would not want to use glutSphere (too expensive!)
- Update method updates particles
 - Particles are born, move, die
 - Perhaps change attributes during lifetime
 - Perform physically based modeling
 - Text discusses in more detail
 - Simple trajectories with effects of gravity work for fountains, fireworks
- Collision detection is problematic
 - Large number of particles



Implementing Particle Systems (cont.)

- Usually drawn last in the last to ensure translucency
 - Blend with other objects and other particles
- Enable depth testing; disable depth write
 - `glDepthMask(GL_FALSE)`
 - Will remove particles behind opaque objects, but not obscure other particles

