Particle Systems according to Reeves

- William T. Reeves, Particle Systems - A Technique for Modeling a Class of Fuzzy Objects", Computer Graphics 17:3 pp. 359-376, 1983 (SIGGRAPH 83).

Particle Systems

- An object is represented as a cloud of particles that defines its volume.
- They are not static, the particle system evolves.
- Non-deterministic.

Properties of Particle Systems

- Particles are simple (computationally efficient)
- Procedural
- LOD is easy.
- Good at complex objects.
- Good at amorphous objects.
- Good at complex behaviour.

What is a Particle System

- Made up of many particles.
- Controls the behaviour (change) of particles over time.

  $\forall$ frames

1. Generate new particles with initial attributes.
2. Kill off particles destined to die (lifespan, etc.)
3. Modify particle attributes: postion, color, etc.
4. Render remaining particles.

Particle Generation

- Generated stochastically.
- Generated in one of two methods.
  - ⋆ Designer designating mean and variance per frame.

$$NParts_f = MeanParts_f + Rand() \times VarParts_f$$

$$Rand() \in [-1, 1]$$

  - ⋆ Designer designating mean and variance per $pixels^2$ (LOD)

$$NParts_f = (MeanParts_{SAF} + Rand() \times VarParts_{SAF})$$

Modifying system over time

- $MeanParts_f = InitialMeanParts + DeltaMeanParts \times (f - f_0)$
- Variance constant.

Particle Attributes

Particle Attributes

- position

Particle Attributes

- position
- velocity (speed and direction)

Particle Attributes

- position
- velocity (speed and direction)
- size

Particle Attributes

- position
- velocity (speed and direction)
- size
- color

Particle Attributes

- position
- velocity (speed and direction)
- size
- color
- transparency

Particle Attributes

- position
- velocity (speed and direction)
- size
- color
- transparency
- shape

Particle Attributes

- position
- velocity (speed and direction)
- size
- color
- transparency
- shape
- lifetime

Particle System Attributes

Particle System Attributes

- Position

Particle System Attributes

- Position
- Orientation (two angles)

Particle System Attributes

- Position
- Orientation (two angles)
- Shape: Sphere, circle in plane, rectangle in plane.

Initial Attributes

- $Speed = MeanSpeed + Rand() \times VarSpeed$
  For spheres particles move away from origin, for planes away from plane with variation (orientation angles).
- $Color_{rgb} = MeanColor + Rand() \times VarColor$
- $Opacity_{rgb} = MeanOpacity + Rand() \times VarOpacity$
- $Size = MeanSize + Rand() \times VarSize$
- Particle shapes: spherical, rectangular, or streaked spherical (motion blur).

Particle Dynamics

- Over time a particle evolves (position, size, color, transparency)
- Velocity can be modified with a gravity force.
- Parameters for rate of change for color, size, opacity can be global or stochastic.

Particle Extinction

- Life expectancy.
- Becomes invisible.
- Moves too far away.

Particle Rendering

- What assumptions to improve rendering?

Particle Rendering

- What assumptions to improve rendering?
  - ⋆ Particles don’t intersect other objects.
  - ⋆ Particles can be rendered as light sources.
- How do these assumptions help?

Particle Rendering

- What assumptions to improve rendering?
  - ⋆ Particles don’t intersect other objects.
  - ⋆ Particles can be rendered as light sources.
- How do these assumptions help?
  - ⋆ Don’t need to sort particles.
  - ⋆ Render objs, particles separately then composite.
  - ⋆ No shadows to deal with.
- How do these assumptions fail?

Particle Rendering

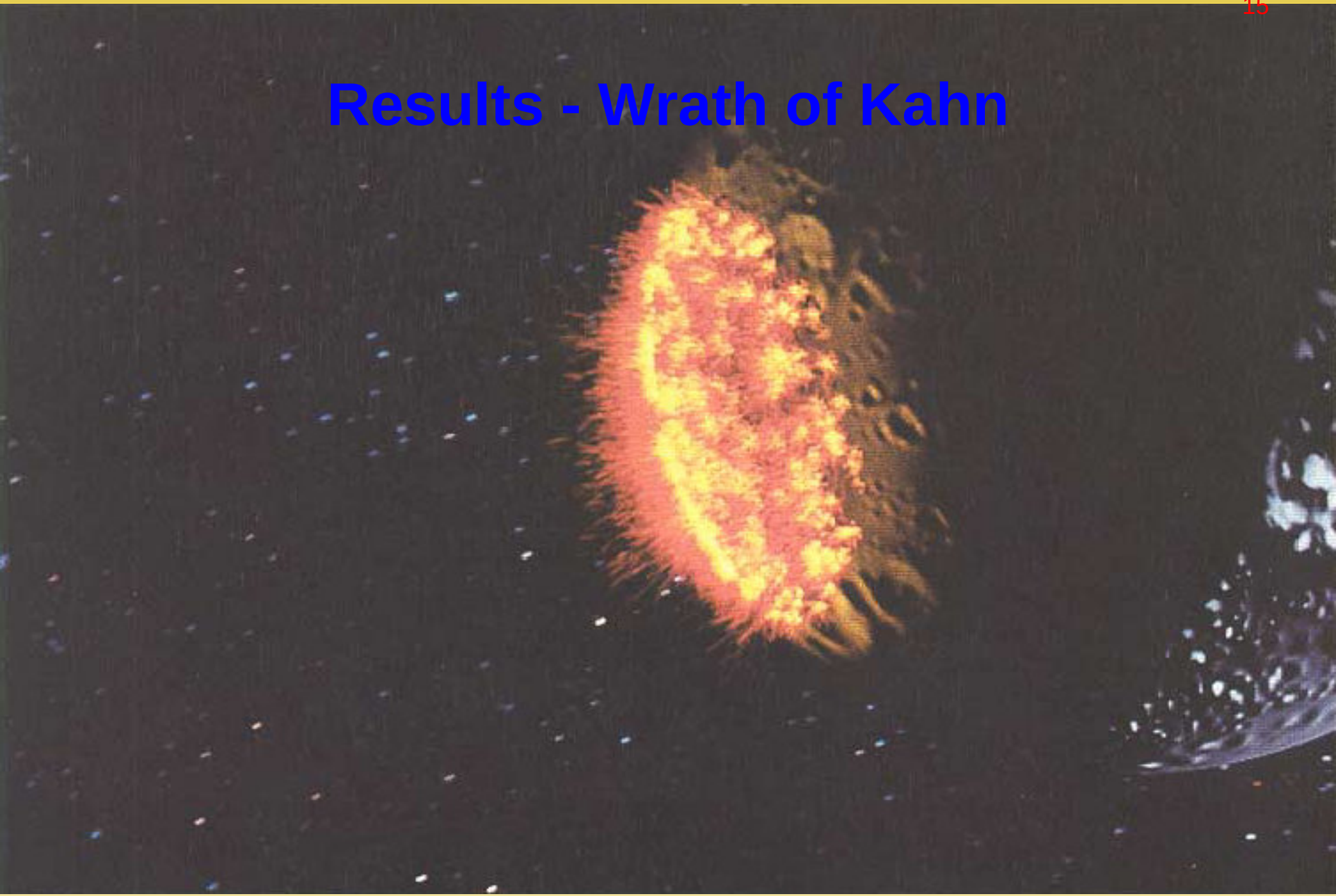
- What assumptions to improve rendering?
  - ⋆ Particles don’t intersect other objects.
  - ⋆ Particles can be rendered as light sources.
- How do these assumptions help?
  - ⋆ Don’t need to sort particles.
  - ⋆ Render objs, particles separately then composite.
  - ⋆ No shadows to deal with.
- How do these assumptions fail?
  - ⋆ Only works well for fire and explosions.
  - ⋆ For clouds, smoke, water less effective.

Hierarchy

- Particles can themselves be particle systems.
- This can allow for separate global and local control mechanisms.
- What would be an example where this would be useful?

Results - Wrath of Kahn

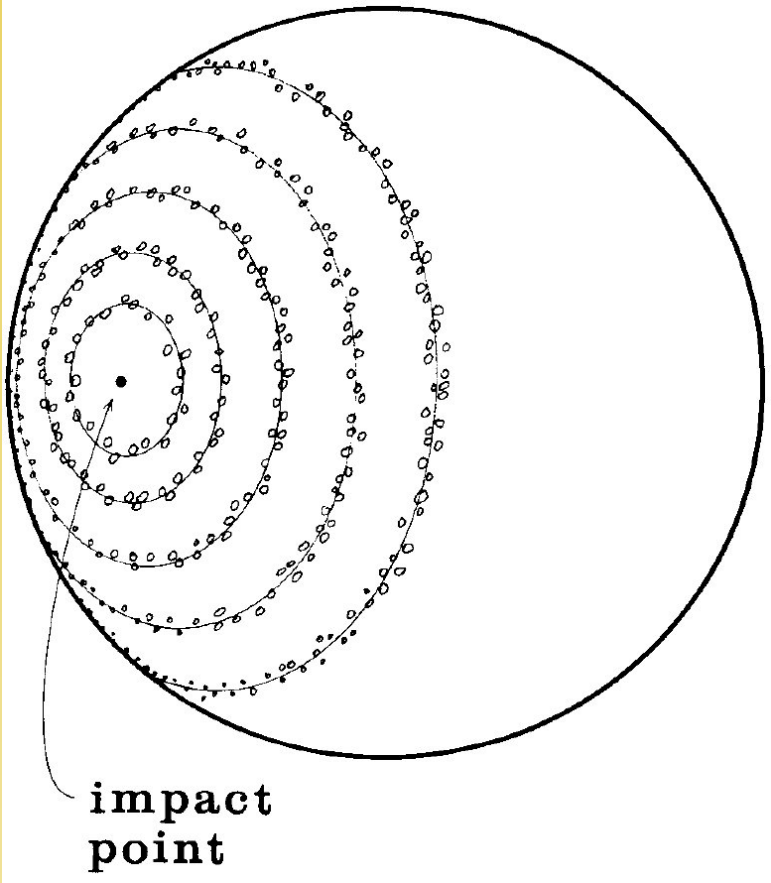
Results - Wrath of Kahn



Results - Wrath of Kahn

Results - Wrath of Kahn

Results - Wrath of Kahn

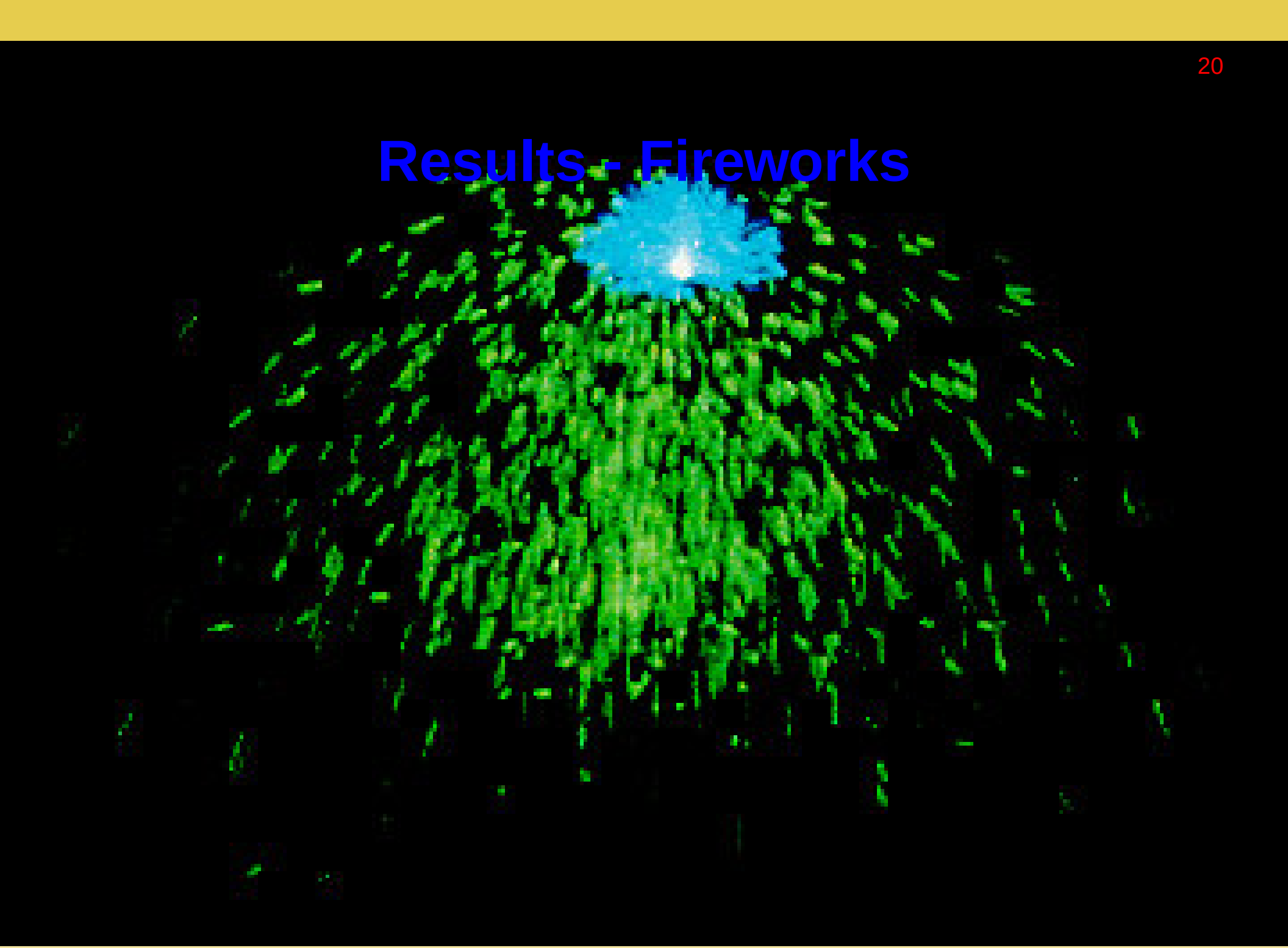


- Particle systems generated in concentric rings.
- # systems based on ring circumference and density parameter.
- # new fire particles based on distance from impact crater.

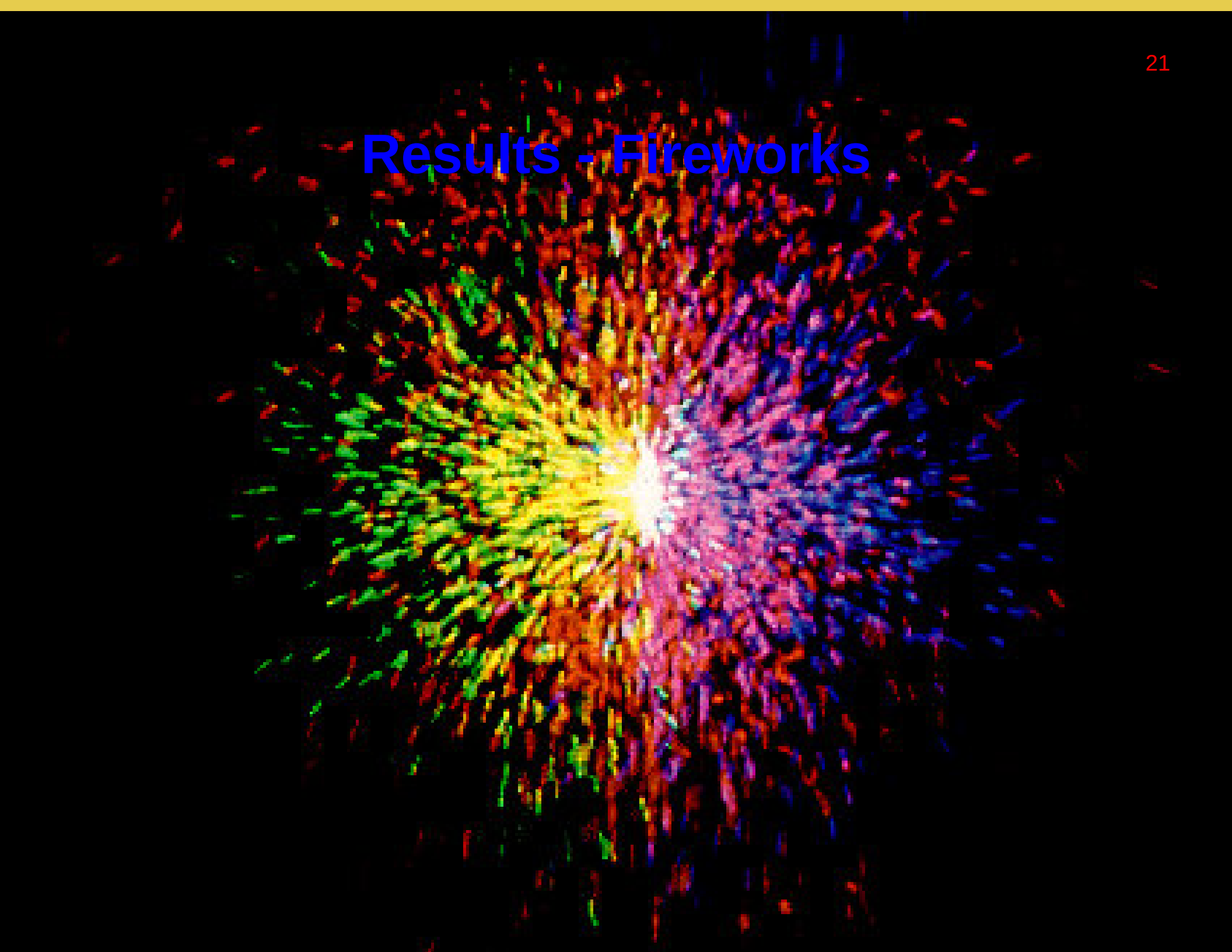
Results - Wrath of Kahn

VIDEO - genesisp.mpg

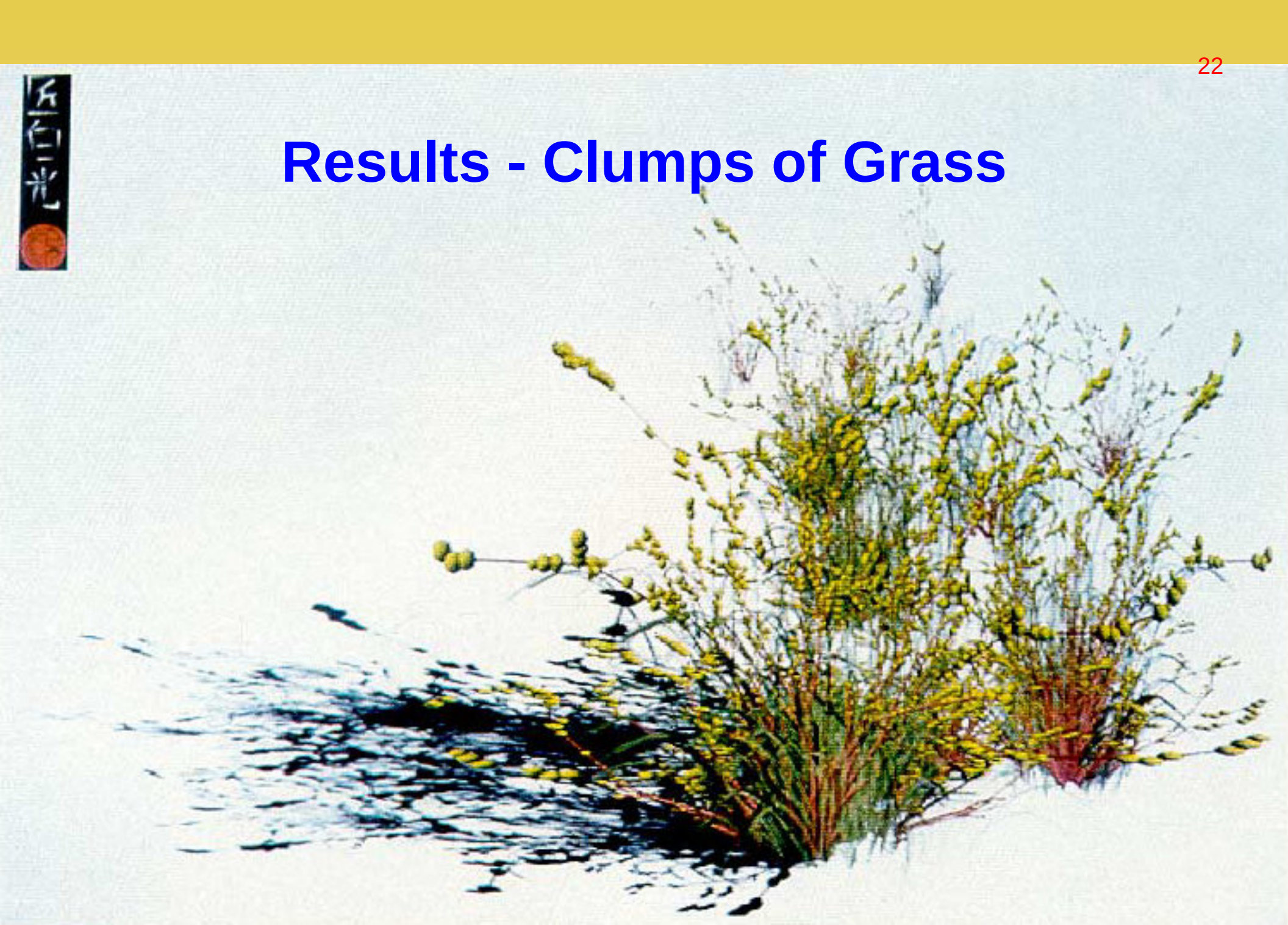
Results - Fireworks



Results - Fireworks



Results - Clumps of Grass



Results - Clumps of Grass

- A grass clump is a particle system.
- A particle is a blade of grass.
- Draw parabolic streak over entire lifetime.

Modifications

- How could we improve Reeves version of particles?
- How could we extend it?
- What could we use it for?

Karl Sims - Particle Dreams (1989)

VIDEO - particle75_1_89.mov

- Each particle has own virtual processor.
- Allows for complex physical phenomena.
- 3D face of particles decomposes and is recomposed via predetermined physical laws.

Karl Sims - Evolved Creatures

Reynolds Flocking

- Particles are now boids (Why boids?)
- Boids have orientation and geometry.
- Boids have more complex behavior
- Boids tend to interact more.
- Boids care about internal and external state.

Reynolds Flocking

- Boids are actors which have builtin behavior
- They are not animated, but directed.
- Class of behavioral animation.
- Allows for creation of more complex interaction.

Geometric Flight

- Motion tangent to a 3D curve.
- Dynamic, incremental, rigid geometrical transformation.
- Object is not rigid, free within its coordinate system.
- Motion path not known in advance.
- Herds are restricted in 3rd dimension.

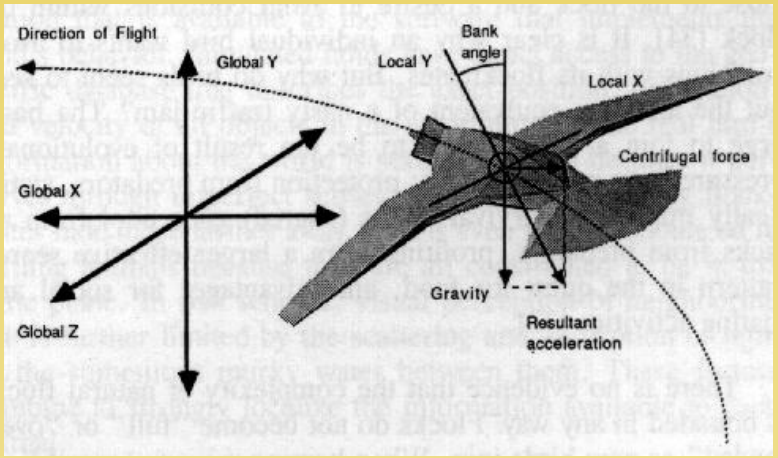
Geometric Flight

- A boid has a current forward direction.
- Incremental translation in this direction for motion.
- Steering (rotation about local X & Y) can also occur.
- Steering realigns the local Z-axis giving new forward.
- Local scale in terms of body length
- How often do we translate forward?

Geometric Flight

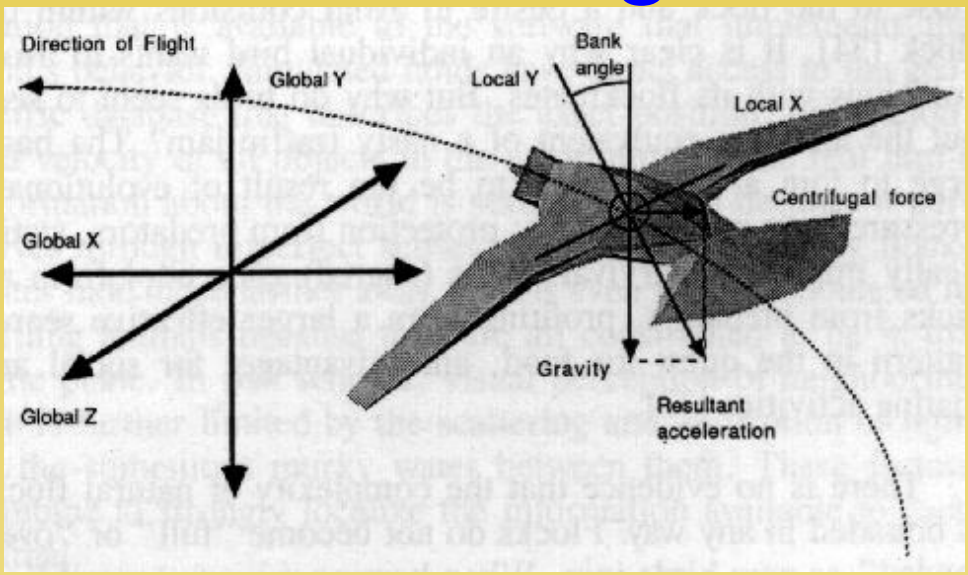
- Viscous damping.
- Maximum velocity.
- Minimum velocity (zero).
- Maximum acceleration.
- Gravity implemented procedurally, accelerate down, decelerate up, stall.
- Buoyancy - counteracts gravity
- Lift in direction of local up.

Banking



- Roll to align local Y with local XY component of acceleration.
- Why?
- Keep coffee in cup. Most efficient.
- Orientation is simpler.
- Looks more natural (but can/may change this)

Banking



- Roll to align local Y with the acceleration.
- Why? - Keep coffee in cup. Most efficient. Orientation is simpler.
- Looks more natural (but can/may change this)

The Flock

- Doesn’t ever get full
- Join for survival reasons.
- Better searching for food.
- Protection from predators.
- Protection from the environment.
- Drafting.
- Bird’s thinking is independent of the flock.

Boids Awareness

- Itself.
- Nearest few neighbors.
- Rest of flock.

Simulated Flocks

- Start with boids that model geometric flight.
- Add rules to avoid collisions and give the urge to join flock.

1. Collision avoidance - don’t run into other boids
2. Velocity Matching - match with neighbors.
3. Flock Centering - Attempt to stay close to neighbors.

Prioritizing Behaviors.

- Weighted sums works okay, but fails at times.
- Instead accumulate acceleration in priority order until max acceleration met.
- Boid behavior is deterministic.

Localized Perception

- Boids must have localized (limited) perception.
- This is not only realistic, but required.
- Reynolds uses a spherical neighborhood. Using an ellipse facing forward and based on velocity may be better.
- Neighbor attraction/repulsion is $\frac{1}{d^2}$

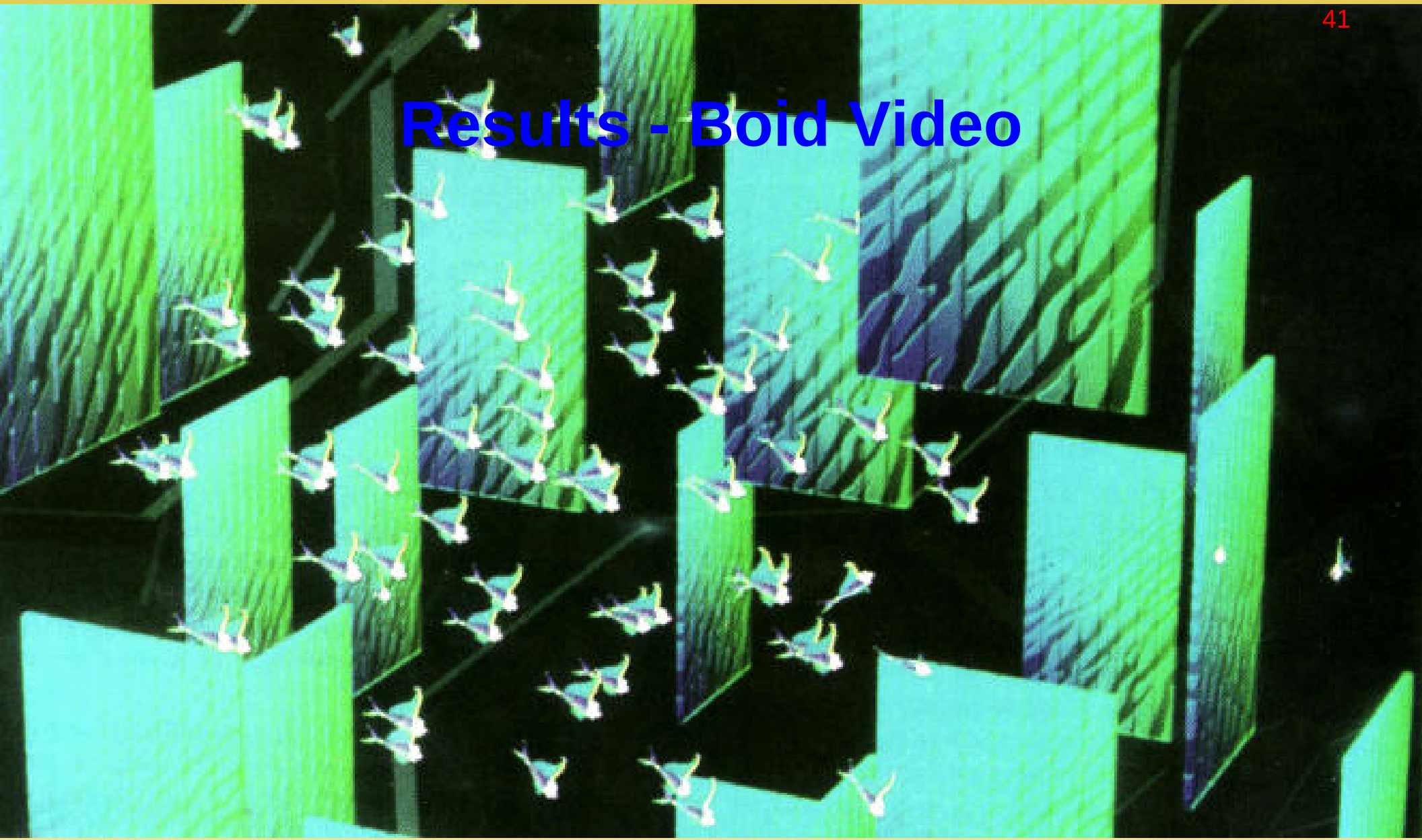
Controlling the Flock.

- Add a migratory urge: a global direction or location.
- The urge can be changed dynamically.
- How the boids get the updated value is procedural, can be instantaneous or based on boid location.

Environmental Obstacles

- Force-field
  - ⋆ Easy to implement.
  - ⋆ Obstacle given repulsive force vector field.
  - ⋆ If the boid heads directly towards obstacle the repulsive force may only slow and not turn the boid.
  - ⋆ Too strong up close and too weak far away.
- Steer-to-avoid
  - ⋆ If an obstacle in front, find closest silhouette edge and head that way.
  - ⋆ Better results.

Results - Boid Video



How Could Boids be Improved?

- Animate the individual boids. Are there issues here?
- More complex perception (vision model, other sensors).
- Make non-deterministic behavior model.
- Give boids emotion (hunger, fear, tired).
- What language was Reynolds software written in?

Example Implementation?

- http://www.red3d.com/cwr/boids/applet/

Results - Lion King

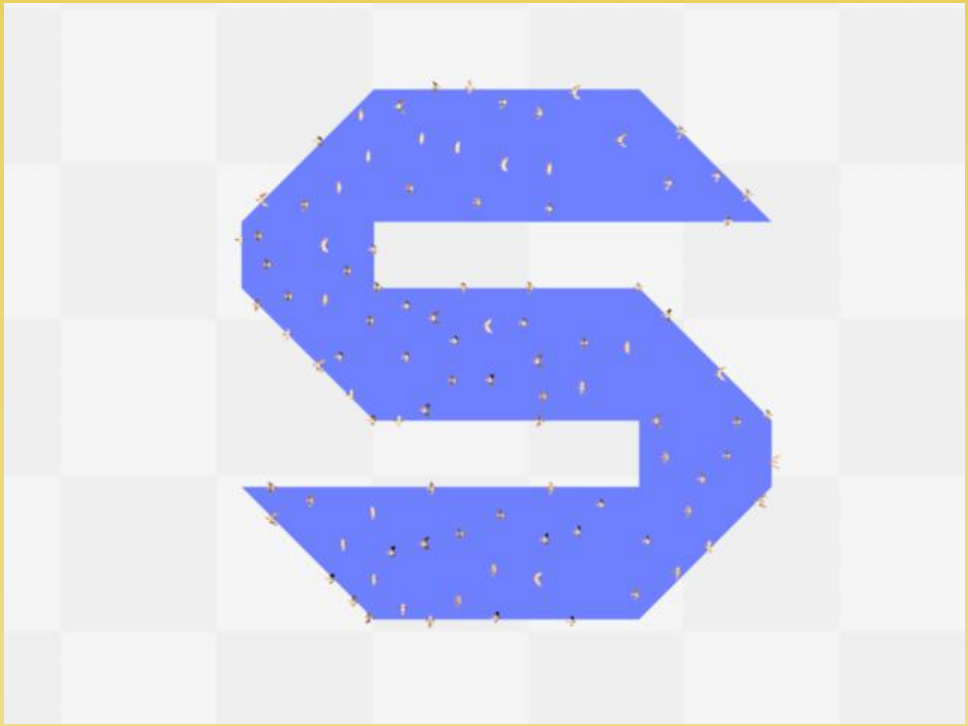
Pigeon Park

Reynolds, C. W. (2000) “Interaction with Groups of Autonomous Characters”, in the proceedings of Game Developers Conference 2000, CMP Game Media Group, San Francisco, CA, pp 449-460.

Pigeon Park

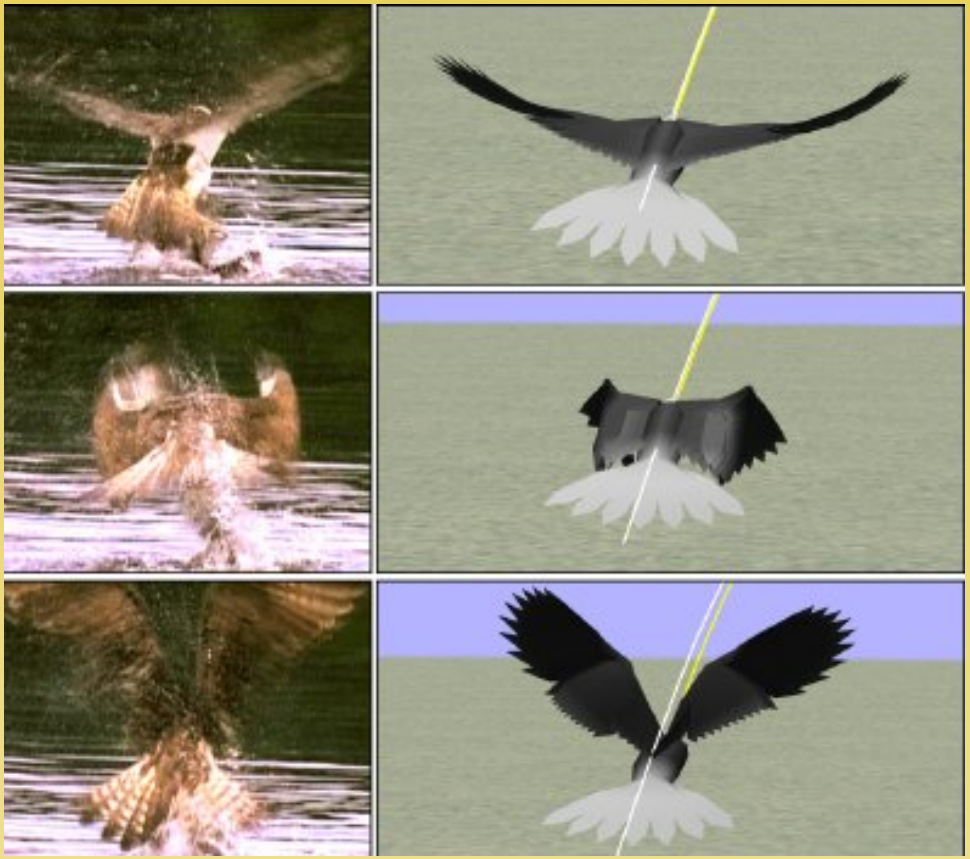
- Methodology for constructing large groups of autonomous characters that respond to user, each other and environment in real-time.
- Steering controllers of simple mental model which mediates between conflicting goals.
- Use library of animated motions; choreographed by controllers.
- Video

Constrained Flocking



M Anderson, E McDaniel and S Chenney, Constrained Animation of Flocks". Eurographics/ SIGGRAPH Symposium on Computer Animation 2003.

Constrained Flocking



Jia-chi Wu, Zoran Popović, “Realistic Modeling of Bird Flight Animations”, SIGGRAPH 2003.