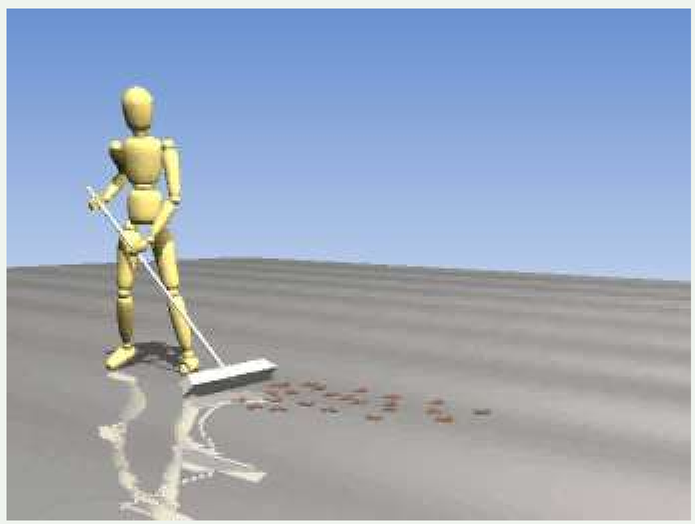
Animation



History

Photography has beeb around since the 19th century. Realistic animation began in 1872 when Eadweard Muybridge settled a bet about a flying horse

History

Muybridge did many studies of human and animal motion, taking photographs against grid backgrounds

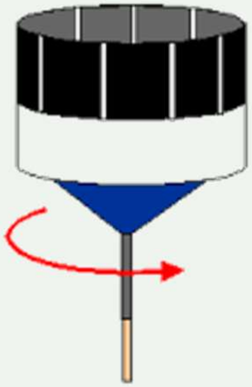
He then displayed the photographs one after another, using a zoopraxiscope

The appearance of moving images was the first early animation

- cartoon-like animation already existed

Early viewers

Many, including flipbook, zoetrope



How it works

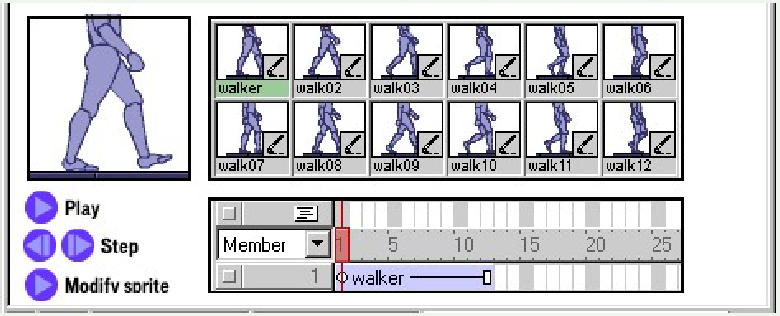
Still images are rapidly displayed in sequence

- If the positions of shapes on the images change, human perception sees this as the shapes moving
- Need at least 10 frames per second to give smooth flicker-free motion but more are needed in practice, where there are “blanks” in between images
- Cine projectors use one or more “blades” to cover the projector for a small period of time whilst the film advances
- Cine films generally have 24 frames per second

Frame-by-Frame Animation

How do we produce the individual frames?

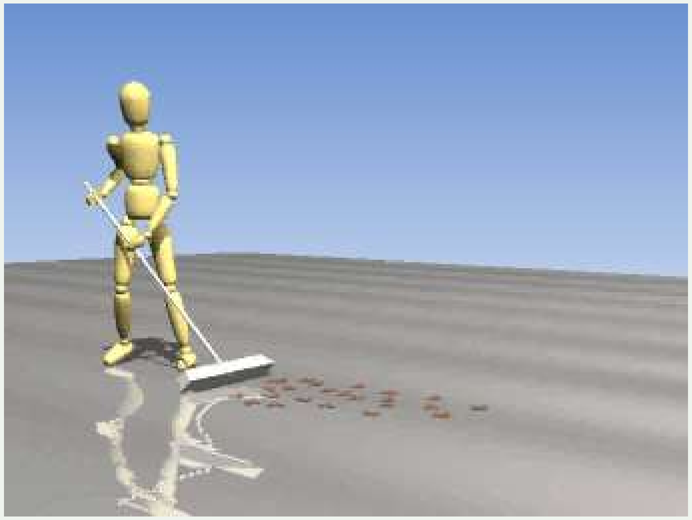
- By drawing each frame by hand, and photographing them (e.g. Walt Disney)
- By creating real scenes, using plasticine figures, and making a series of very small changes, photographing each time (e.g. Aardman's Wallace and Gromit) – stop motion
- Using computer software ...



Bitmap versus Vector

Same advantages/disadvantages of bitmaps and vector-based ways of representing moving images, only more so

- The shapes/objects/figures need to be represented as such to meaningfully move them around
- Again, same common practice:
  - Use vector-based file formats or metafiles to store the animations
  - Export as a bitmapped version (typically compressed)



Animation Creation

Model Design

- Uses a modelling tool to produce the objects to be animated

Animation Design

- Motion sequences planned
- Lighting, action, interaction
- Animation package uses all these to produce the animation “script”

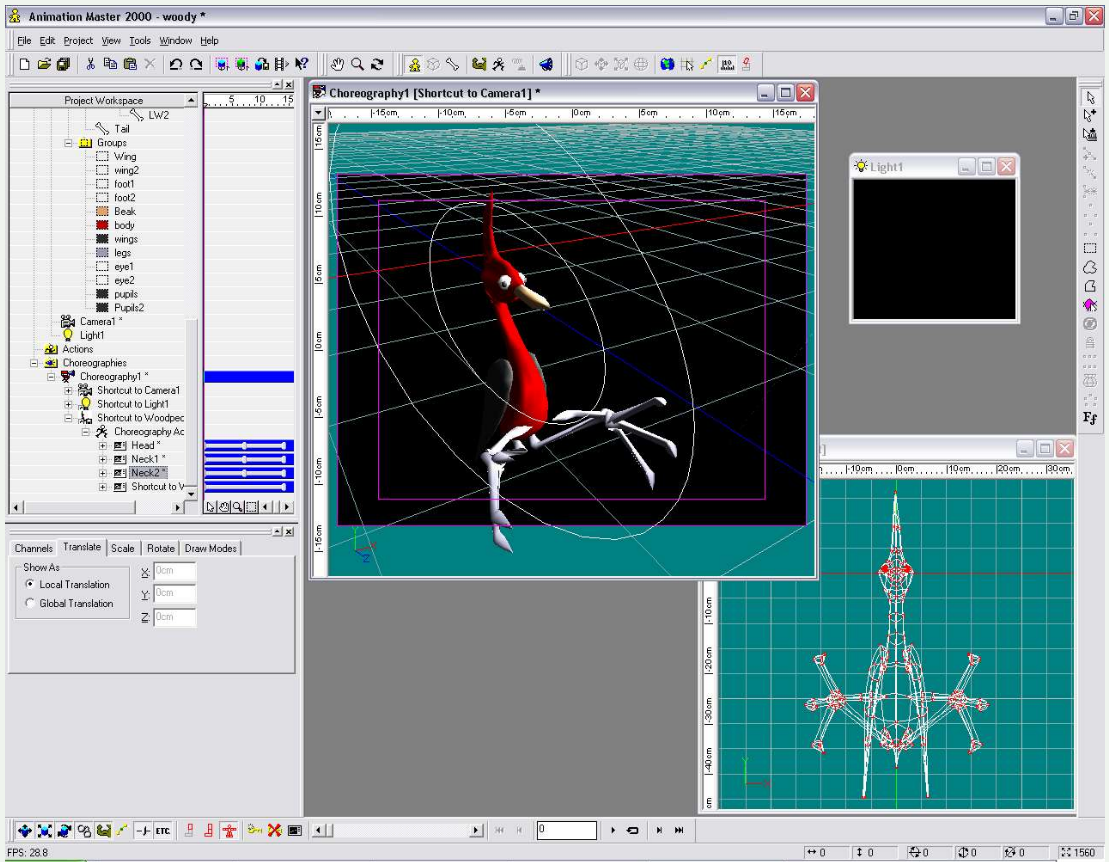
Rendering

- Takes the script, and generates the images (i.e. vector to bitmap conversion)

Post-production

The more of these steps that happen in one software package, the more convenient for the animator!

3D Animation



Representation of Animation

Various approaches:

Keyframe animation

- 2D
- 3D

Track-based animation

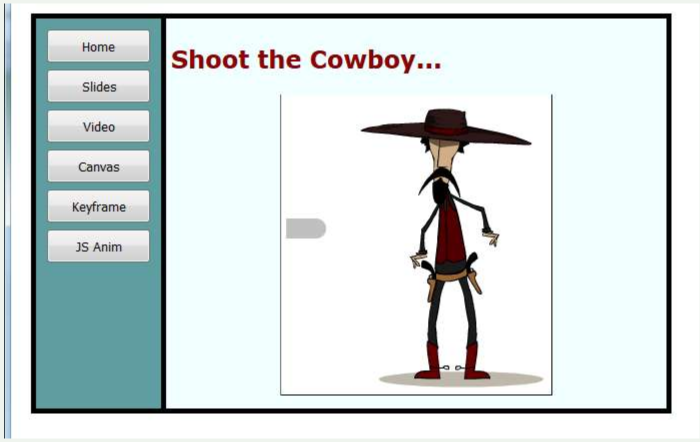
Other approaches

- Particle Systems
- Hierarchical Systems
- Flock Systems
- Autonomous Systems

Keyframe-based Animation

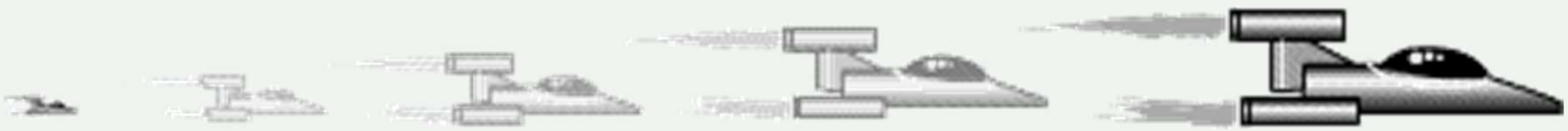
Animator creates key frames, and the software interpolates between them

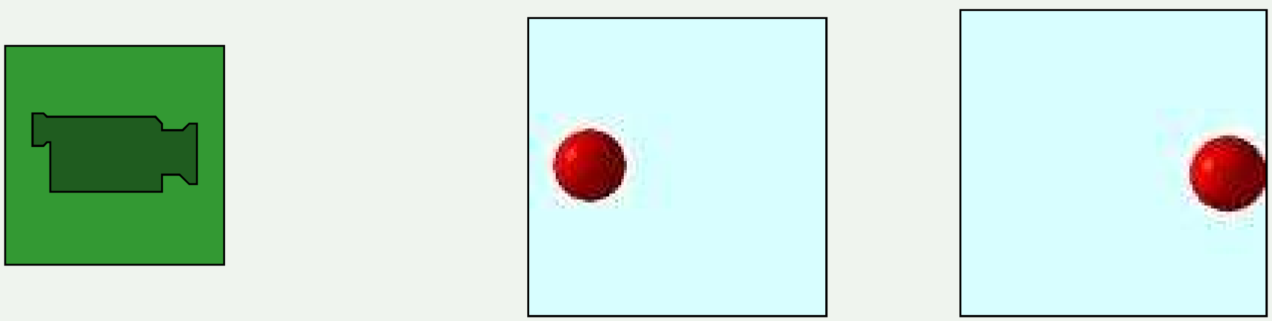
- Keyframes hold all information about the state of the animation at that point in time (more detail for 3D)
- The creation of intermediate images between keyframes is known as in-betweening (or just tweening)



Tweening

How should these frames be tweened?





Location, Size, Colour, Linear or more complex equations

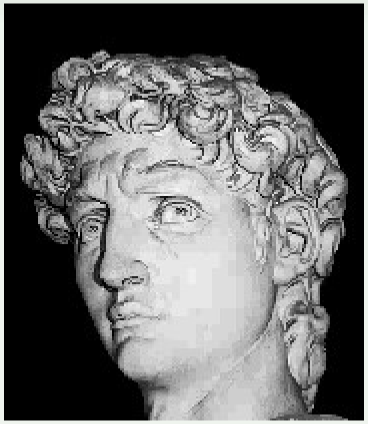
2D Animation

Animated image processing

- uses morphing technique

Cel animation

- flattened version of 3D
- uses layers (cels)



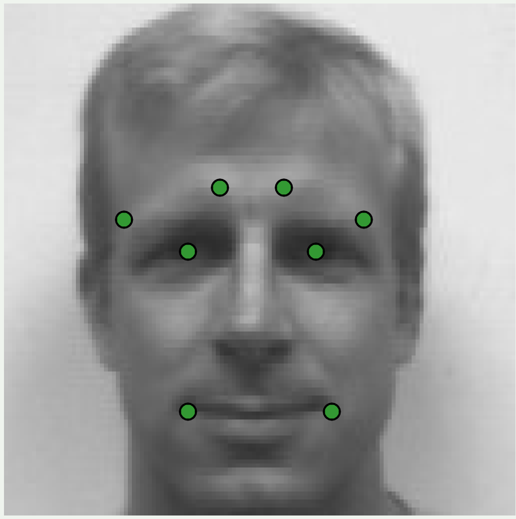
From F. Hofstetter, *Multimedia Literacy*, McGraw-Hill

From Chapman & Chapman, *Digital Multimedia*, Wiley

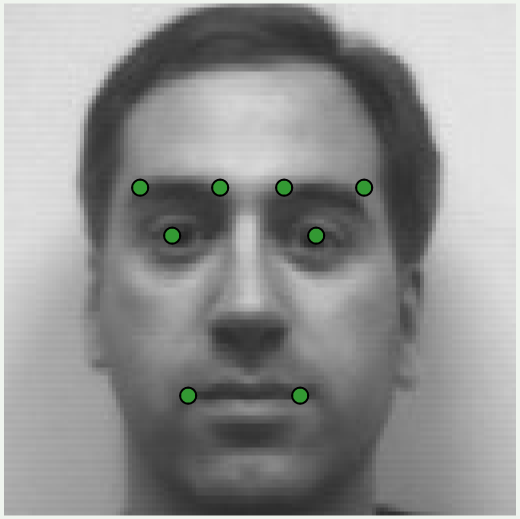
Morphing

Control points are used in order to stretch (and colour) the source image smoothly to the destination image

Source



Destination



Cel Animation

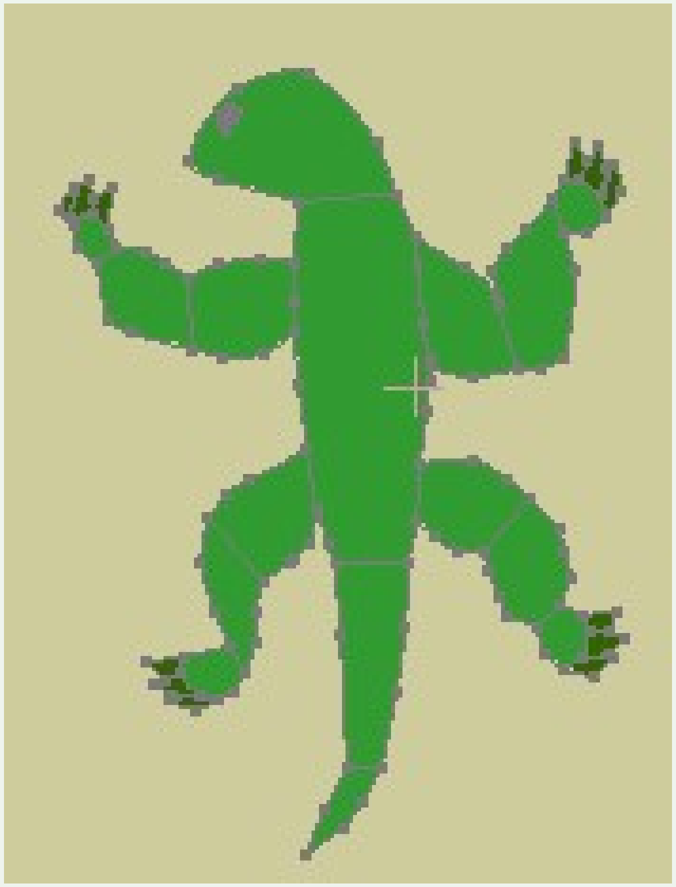
The image is separated out into layers (*cels*)

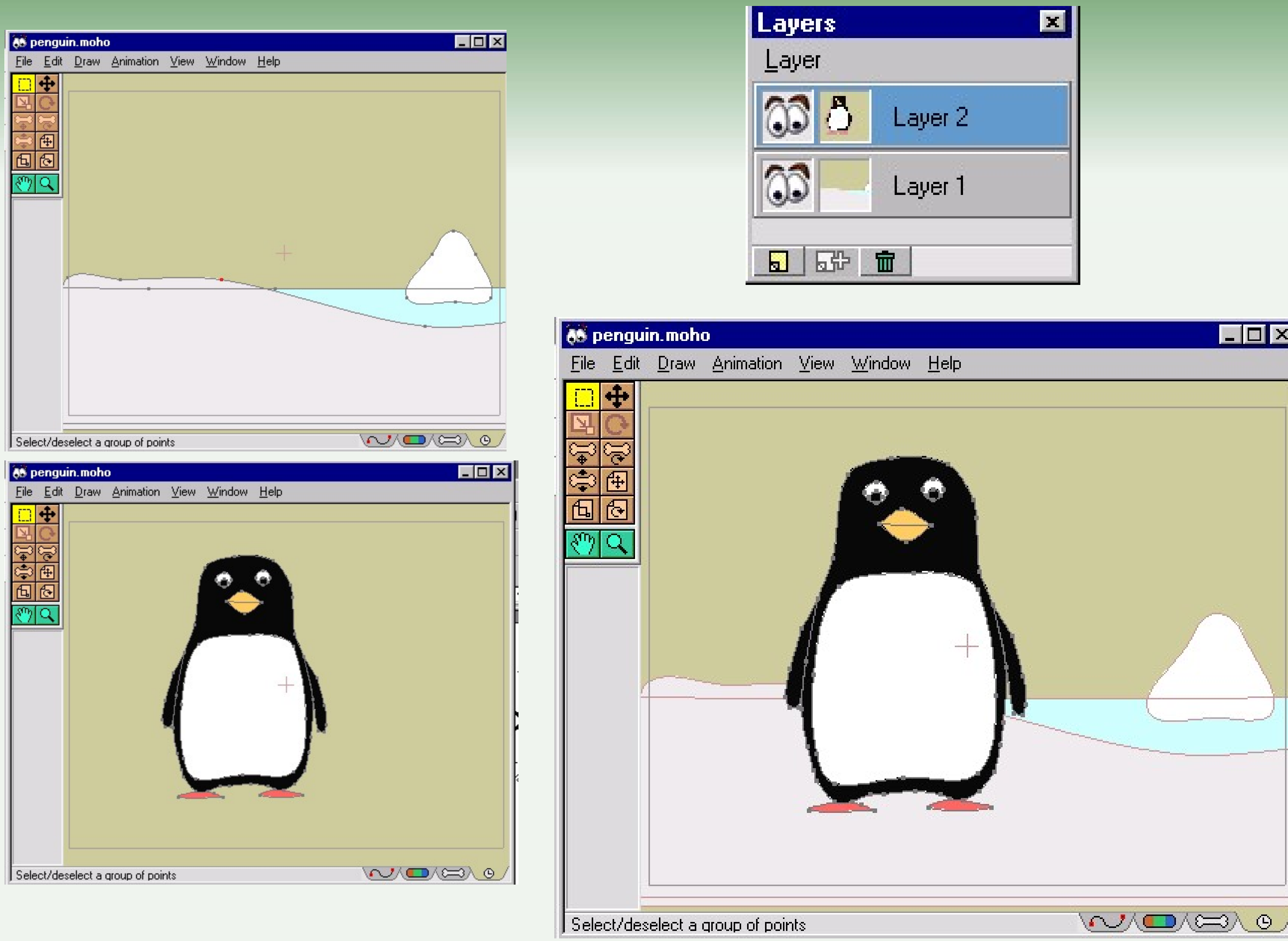
Historically these cels would have been separate sheets of celluloid

Purpose of cels is to avoid unnecessary redrawing

- e.g. when the background remains the same

Layers with characters on have control points

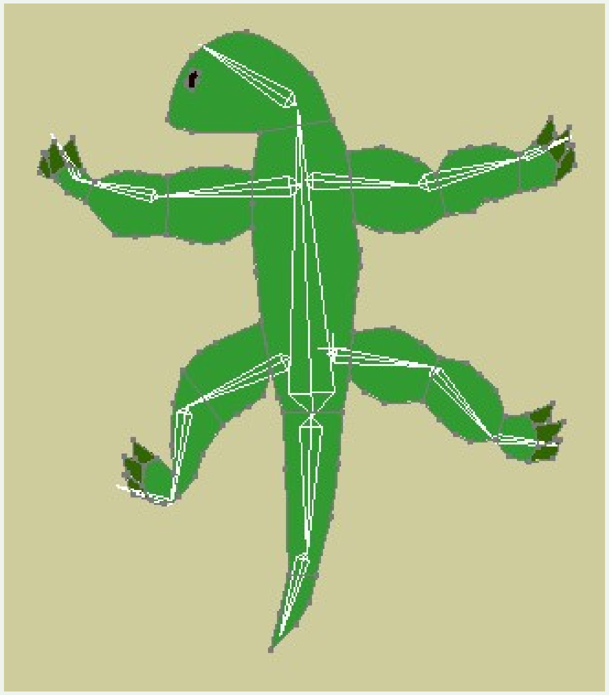



penguin.moho
File Edit Draw Animation View Window Help
Select/deselect a group of points
Layers
Layer
Layer 2
Layer 1

Movement: Bones

Control points are not sufficient to produce reasonable simulation of movement

- Skeletons are used to keep different parts of the body together (both in biology and in animation)
- Animation software allows the animator to define, connect, and move bones

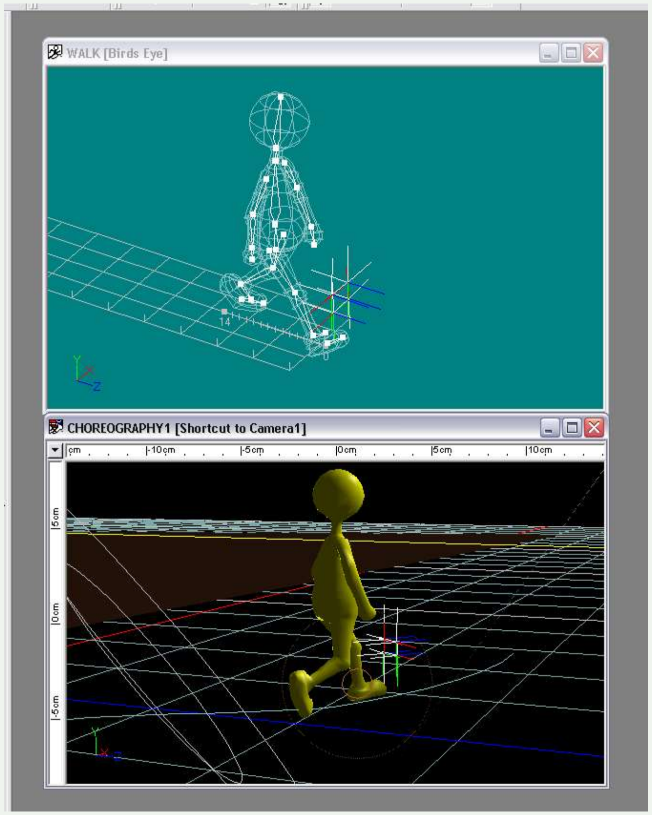


Forward and Inverse Kinematics

Forward kinematics involves setting all joint angles manually

Inverse kinematics allows positioning an end effector and the software calculates intermediate joint angles

- Hard and underconstrained mathematical problem



3D Animation

Morphing can be used

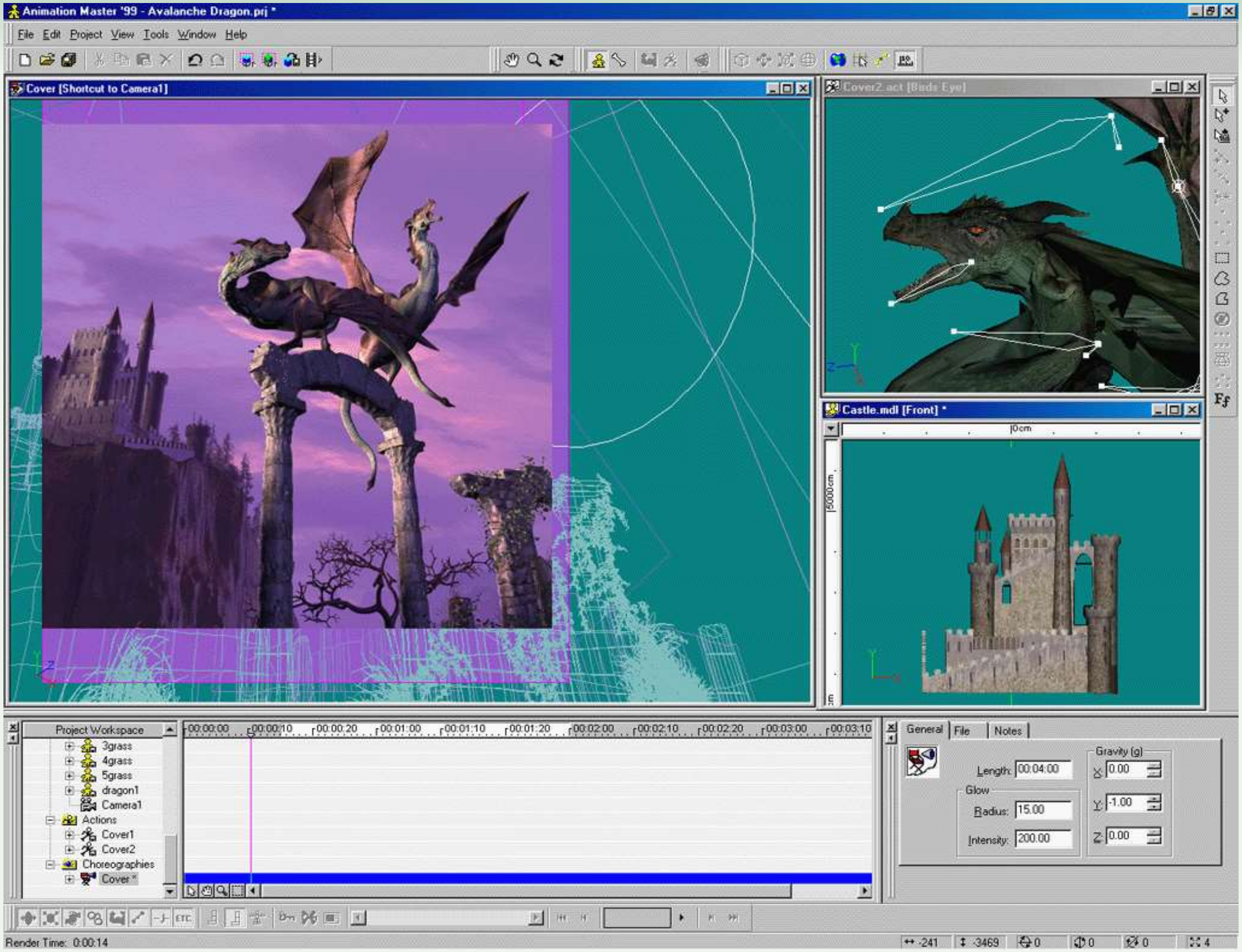
- however, it is very much more computationally expensive to morph in 3D

The 3D equivalent of layers is to have scenes

Scenes contain a lot of state information, including

- Background features
- Lighting (number, nature and brightness, position)
- Camera position
- Objects (size, shape, texture, position)

Each keyframe must record the state of all these objects for that keyframe


Animation Master '99 - Avalanche Dragon.prj *
File Edit Project View Tools Window Help
Cover [Shortcut to Camera1]
Castle.mdl [Front] *
Project Workspace
3grass
4grass
5grass
dragon1
Camera1
Actions
Cover1
Cover2
Choreographies
Cover *
General
File
Notes
Length: 00:04:00
Gravity (g)
Glow
Radius: 15.00
Intensity: 200.00
Render Time: 0:00:14

Track-based Animation

These generalize keyframe systems. With keyframes, each frame has the state information for the entire scene.

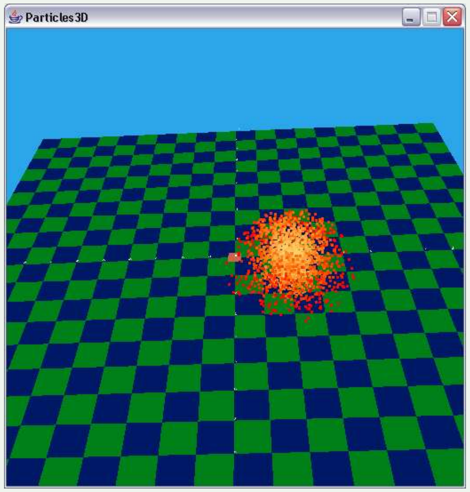
- In track-based systems each piece of state information has its own "track".
- In a track, only values controlling interpolation are specified at any one time
  - position of an object
  - brightness of a light
- This reduces the amount of information to be recorded, and focuses the interpolation just on the pieces of state information that are important to the animator.

Particle Systems

These consist of a set (possibly varying in number) of particles, and the rules of interaction between the particles

Particles have their own state. This may typically be:

- position
- velocity
- shape parameters
- colour
- transparency
- lifetime

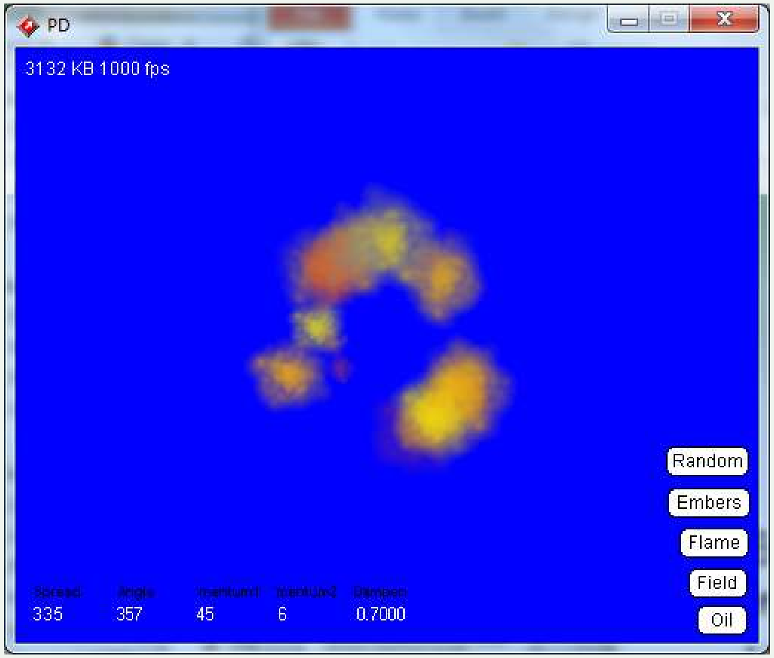


Particle Systems

Each frame is generated from the previous one and the interaction rules - no keyframes

Useful for natural phenomena such as:

- smoke
- rain
- fire

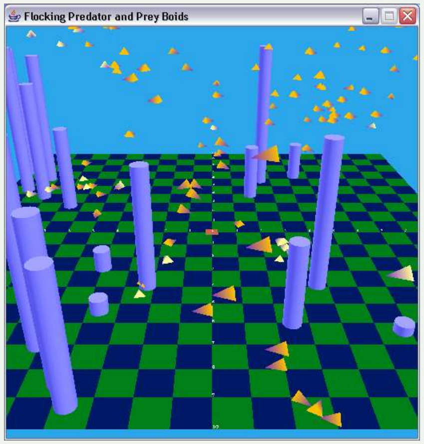


Flocking Systems

Like particles, but with rules like "follow the leader"

- Typically fewer objects involved

Saves effort, as the animator just needs to keyframe the leader, instead of all the members of the flock



Other approaches

Hierarchical Animation

- where objects relate to each other in a specific hierarchical way
- e.g. the solar system, with moons revolving around planets revolving around the sun

Autonomous Behaviour

- "Intelligent" behaviour of individual characters
- May use artificial intelligence techniques
- Perception, reasoning, memory
- Important in many computer games