Keyframing

based on hand animation techniques

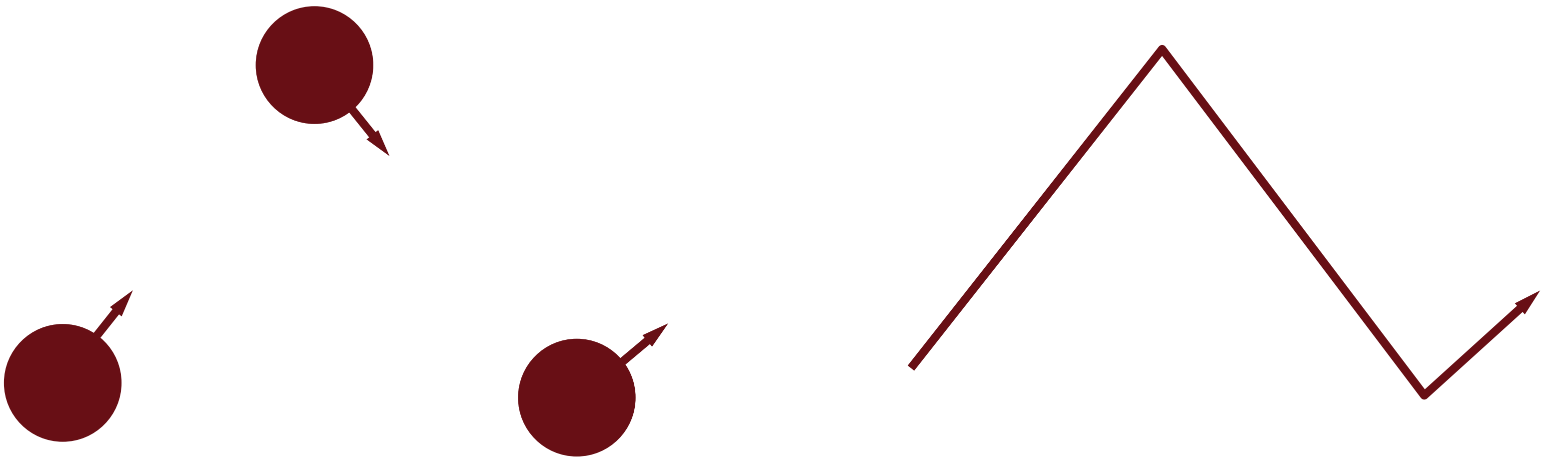
- animator –> inbetweener –> inker
- Snow White:
  $100/week –> $35/week –> $25/week

save time for the animator

force explicit thought about timing
[vs. straightahead animation]

simple interpolation !=
intelligence/judgment of inbetweener

Keyframing



ugly motion (as Alan showed!)

$F/m = a = d^2x(t)/dt^2$ => C^2 continuous
x(t) position of ball

Splines for interpolation

Rotation Representation

transformation matrices

$$\mathbf{Rot_x(\theta)} \quad \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\mathbf{Rot_y(\theta)} \quad \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

multiply to get arbitrary rotations

$$\mathbf{Rot_z(\theta)} \quad \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotation Interpolation

3x3 rigid body rotation matrix has rows and columns that are orthonormal (unit length and perpendicular)

simple interpolation won't preserve that property: object won't rotate rigidly

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotx(90) –>Rotx(–90) point in the middle is

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

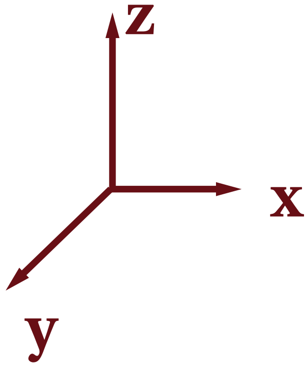
Rotation--Euler Angles

fixed order rotation about
x,y,z = (17,42,89)

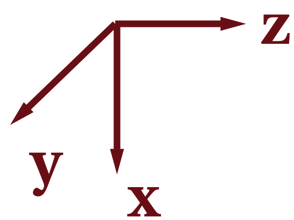
if axes become aligned then representation breaks down: gimbal lock

x,y,z = (0,90,0)

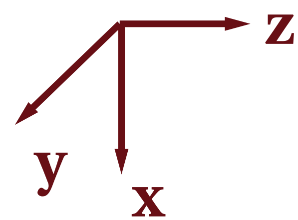
incremental changes in either x or z have the same effect on the system. A degree of freedom has been "lost."



Rotx(0)



Roty(90)



Rotz(0)

x,y,z = (90,90–Δ,90) will allow rotation by Δ about vertical axis.

Interpolation of Euler Angles:

(0,90,0) –> (90,45,90)
is just a rotation by 45 in x
but obvious interpolation
would go through (45, 67.5, 45) which
isn't the shortest path

Axis/Angle or Quaternions

angle and axis (x,y,z)

$[\sin(\theta/2)(x,y,z), \cos(\theta/2)]$