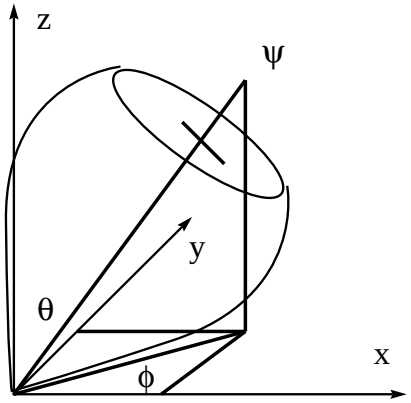
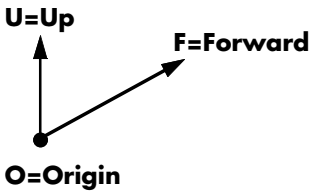
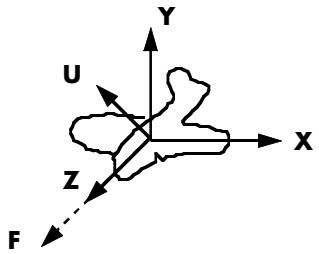
Combining Rotations: Euler Angles



Copyright © Pat Hanrahan

Orientation Matrix





W = O

Z = F

X = U x Z

Y = Z x X

T=(X Y Z W)

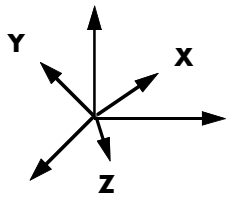
Copyright © Pat Hanrahan

Rotation Matrix = Orthogonal Matrix

$$R^T R = \begin{pmatrix} \hat{X} \\ \hat{Y} \\ \hat{Z} \end{pmatrix} \begin{pmatrix} \hat{X} & \hat{Y} & \hat{Z} \end{pmatrix} = \begin{pmatrix} \hat{X}\bullet\hat{X} & \hat{X}\bullet\hat{Y} & \hat{X}\bullet\hat{Z} \\ \hat{Y}\bullet\hat{X} & \hat{Y}\bullet\hat{Y} & \hat{Y}\bullet\hat{Z} \\ \hat{Z}\bullet\hat{X} & \hat{Z}\bullet\hat{Y} & \hat{Z}\bullet\hat{Z} \end{pmatrix}$$

$$R = \begin{pmatrix} \hat{X} & \hat{Y} & \hat{Z} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = I$$



$$R^{-1} = R^T$$

Copyright © Pat Hanrahan

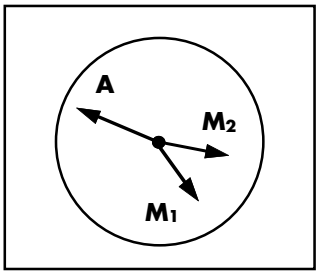
Rolling Ball

Control rotation by mouse difficult

mouse x -> pan

mouse y -> tilt

Rolling Ball metaphor



$$\vec{M} = \begin{pmatrix} x & y & \sqrt{1-x^2-y^2} \end{pmatrix}$$

$$\vec{A} = \vec{M}_1 \times \vec{M}_2$$

$$R_1 = \begin{pmatrix} \hat{M}_1 & \hat{A}\times\hat{M}_1 & \hat{A} \end{pmatrix}$$

$$R_2 = \begin{pmatrix} \hat{M}_2 & \hat{A}\times\hat{M}_2 & \hat{A} \end{pmatrix}$$

$$R = R_2 R_1^T$$

Copyright © Pat Hanrahan

Composition of Transformations

- **Sequence of transforms = single transform**

  $P' = A (B (C (D (P))))$

  $= (A B C D) P$

  $= T P$

- **Inverse of a sequence**

  $P = (D^{-1} C^{-1} B^{-1} A^{-1}) (A B C D) P$

  Therefore

  $(A B C D)^{-1} = D^{-1} C^{-1} B^{-1} A^{-1}$

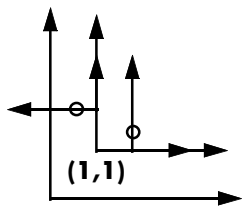
  1. Apply inverses in the reverse order

  2. Concatenate and numerically invert matrix

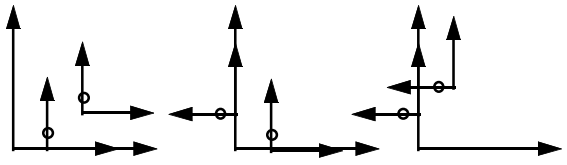
Copyright © Pat Hanrahan

Transforming Transformations

Apply a transformation in another coordinate system



R by 90 about 1,1



T(-1,-1) R(90) T(1,1)

More generally: $R_T = T R T^{-1}$

Copyright © Pat Hanrahan

Current Transformation Matrix (CTM)

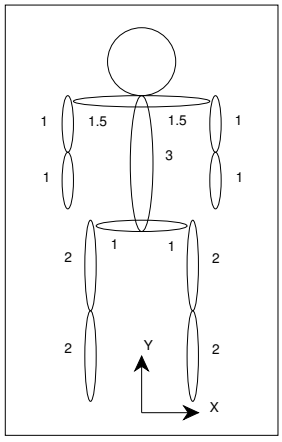
- **Graphics systems maintain a *current transformation matrix (CTM).* All geometry is transformed by the CTM.**
- **The CTM defines the current or *object* or *local* coordinate system. All geometry is defined in the current coordinate system.**
- **Transformation commands are concatenated onto the ctm. Note: The last transformation specified is the first to be performed.**

  CTM′ = CTM * T

- **The CTM may be pushed onto a transformation stack, and latter popped to restore a previous state.**

Copyright © Pat Hanrahan

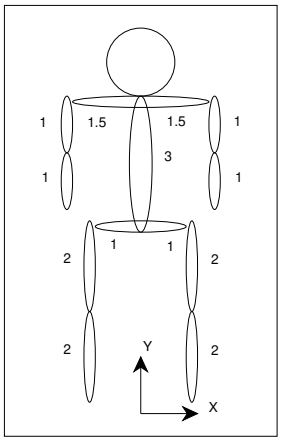
Skeleton




```
body
  torso
    head
    shoulder
      larm
        upperarm
        lowerarm
        hand
      rarm
        upperarm
        lowerarm
        hand
    hips
      lleg
        upperleg
        lowerleg
        foot
      rleg
        upperleg
        lowerleg
        foot
```


Copyright © Pat Hanrahan

Skeleton




```
translate(0,4,0)
torso();
pushmatrix();
    translate(0,3,0);
    shoulder();
    pushmatrix();
      rotatey(necky);
      rotatex(neckx);
      head();
    popmatrix();
    pushmatrix();
      translate(1.5,0,0);
      rotatex(lshoulderx);
      upperarm();
      pushmatrix();
        translate(0,-1,0);
        rotatex(lelbowx);
        lowerarm();
        ...
      popmatrix();
    popmatrix();
...
```


Copyright © Pat Hanrahan