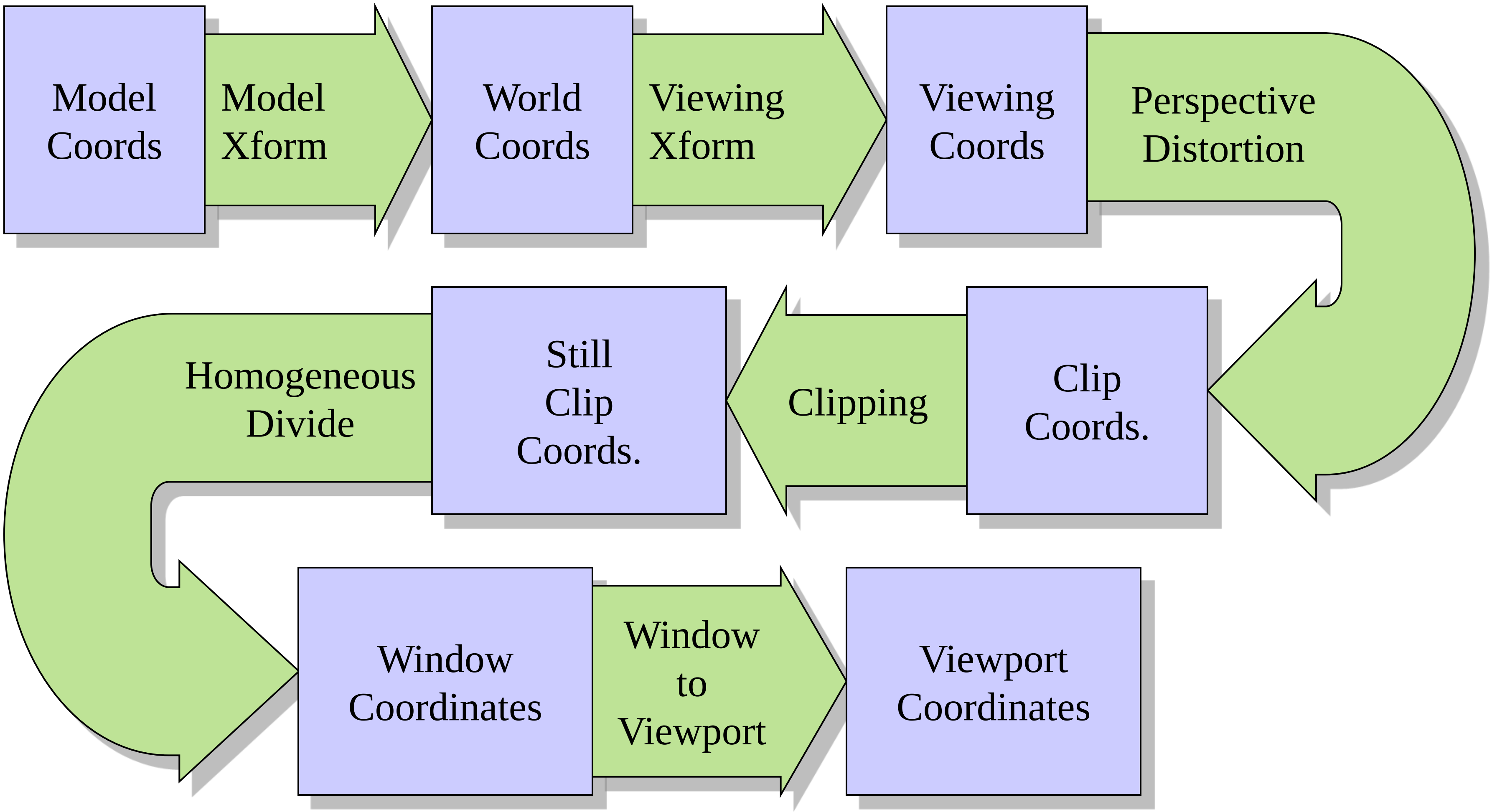
3-D Transformational Geometry

CS418 Computer Graphics

John C. Hart

Graphics Pipeline



3-D Affine Transformations

- General

$$\begin{bmatrix} d & e & f & a \\ g & h & i & b \\ j & k & l & c \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} dx + ey + fz + a \\ gx + hy + iz + b \\ jx + ky + lz + c \\ 1 \end{bmatrix}$$

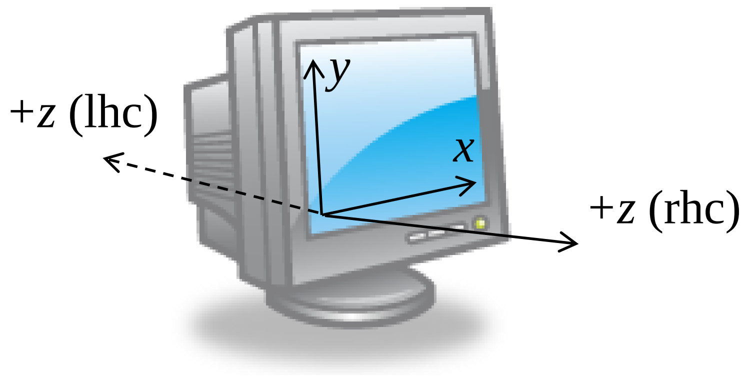
- Translation

$$\begin{bmatrix} 1 & & & a \\ & 1 & & b \\ & & 1 & c \\ & & & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + a \\ y + b \\ z + c \\ 1 \end{bmatrix}$$

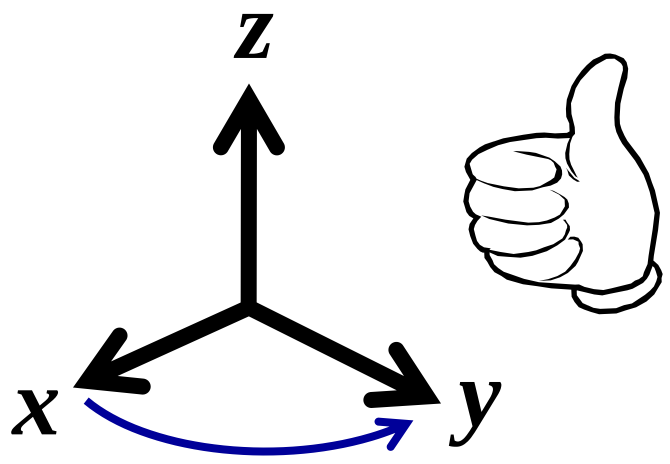
3-D Coordinates

- Points represented by 4-vectors
- Need to decide orientation of coordinate axes

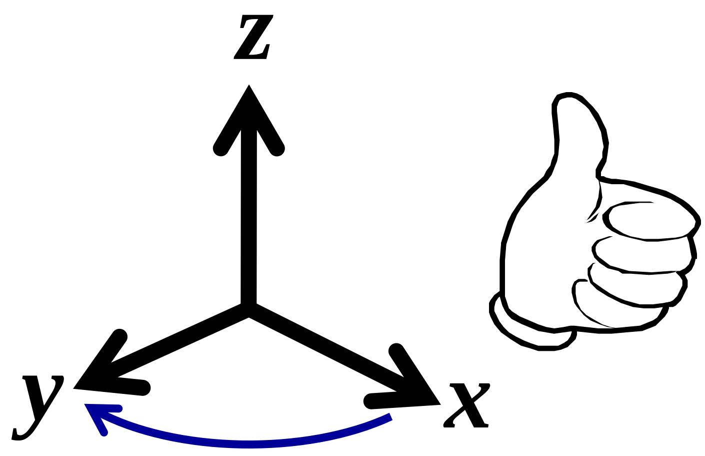
$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$



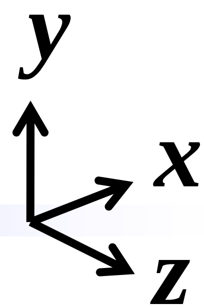
Right Handed Coord. Sys.



Left Handed Coord. Sys.

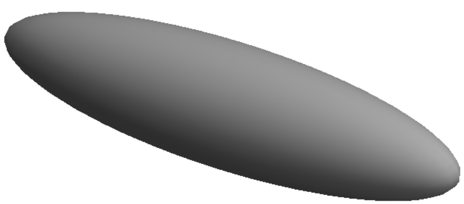


Scale



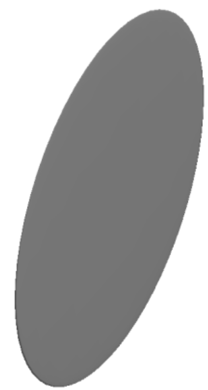
$$\begin{bmatrix} a & & & \\ & b & & \\ & & c & \\ & & & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} ax \\ by \\ cz \\ 1 \end{bmatrix}$$

Uniform Scale
a = *b* = *c* = ¼

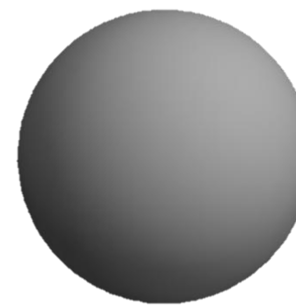


Stretch
a = *b* = 1, *c* = 4

Squash
a = *b* = 1, *c* = ¼



Project
a = *b* = 1, *c* = 0



Invert
a = *b* = 1, *c* = -1

3-D Rotations

- About x-axis
  - rotates $y \rightarrow z$

$$\begin{bmatrix} 1 & & & \\ & \cos\theta & -\sin\theta & \\ & \sin\theta & \cos\theta & \\ & & & 1 \end{bmatrix}$$

- About y-axis
  - rotates $z \rightarrow x$

$$\begin{bmatrix} \cos\theta & & \sin\theta & \\ & 1 & & \\ -\sin\theta & & \cos\theta & \\ & & & 1 \end{bmatrix}$$

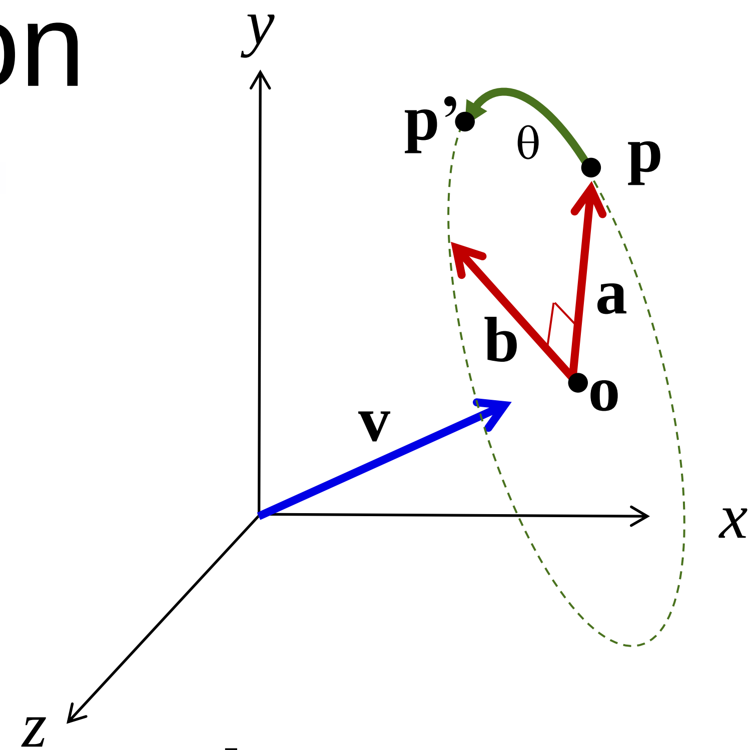
- About z-axis
  - rotates $x \rightarrow y$

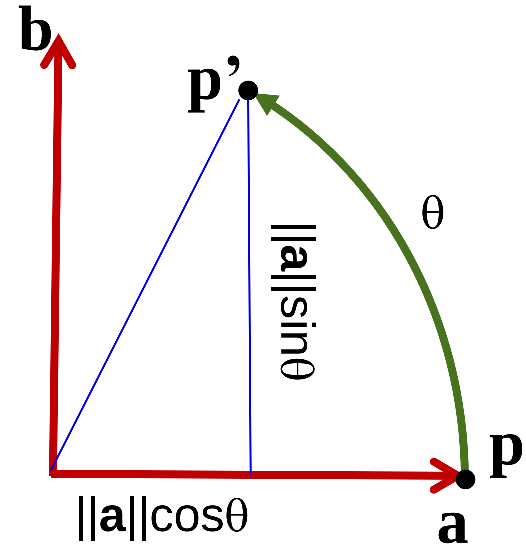
$$\begin{bmatrix} \cos\theta & -\sin\theta & & \\ \sin\theta & \cos\theta & & \\ & & 1 & \\ & & & 1 \end{bmatrix}$$

- Rotations do not commute!

Arbitrary Axis Rotation

- Rotations about x, y and z axes
- Rotation x rotation = rotation
- Can rotate about any axis direction
- Can do simply with vector algebra
  - Ensure $\|\mathbf{v}\| = 1$
  - Let $\mathbf{o} = (\mathbf{p} \cdot \mathbf{v})\mathbf{v}$
  - Let $\mathbf{a} = \mathbf{p} - \mathbf{o}$
  - Let $\mathbf{b} = \mathbf{v} \times \mathbf{a}$, (note that $\|b\|=\|a\|$)
  - Then $\mathbf{p}' = \mathbf{o} + \mathbf{a} \cos \theta + \mathbf{b} \sin \theta$
- Simple solution to rotate a single point
- Difficult to generate a rotation matrix to rotate all vertices in a meshed model

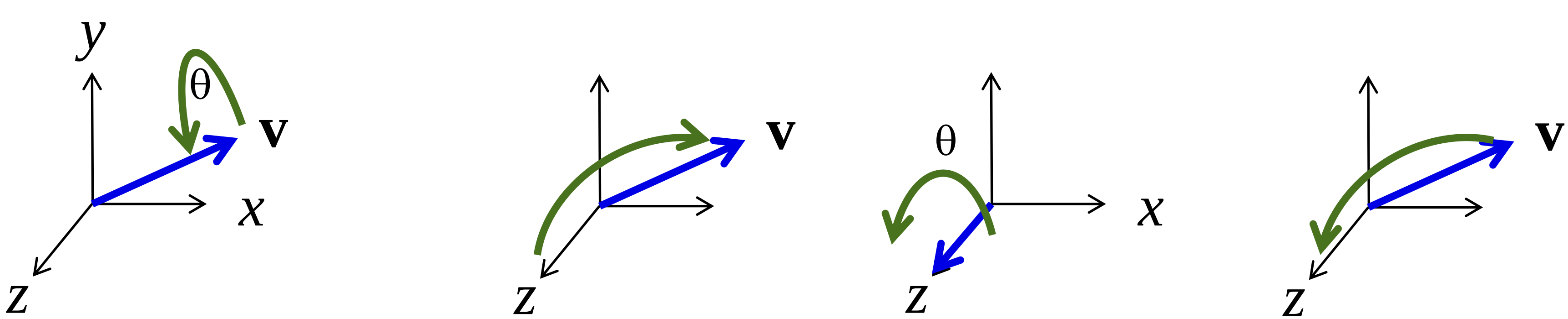




Arbitrary Rotation

- Find a rotation matrix that rotates by an angle θ about an arbitrary unit direction vector **v**

$$\mathbf{v} = (x_v, y_v, z_v),\ x_v^2 + y_v^2 + z_v^2 = 1$$

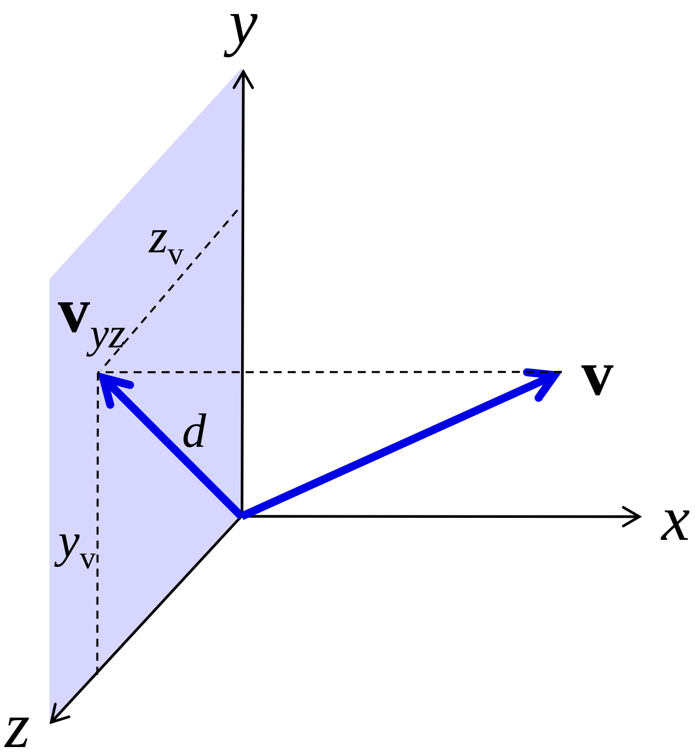


[Rotate by θ about v] = [Rotate z to **v**] [Rotate by θ about z] [Rotate **v** to z]

Rotate v to z

$\mathbf{v} = (x_v, y_v, z_v)$, $x_v^2 + y_v^2 + z_v^2 = 1$

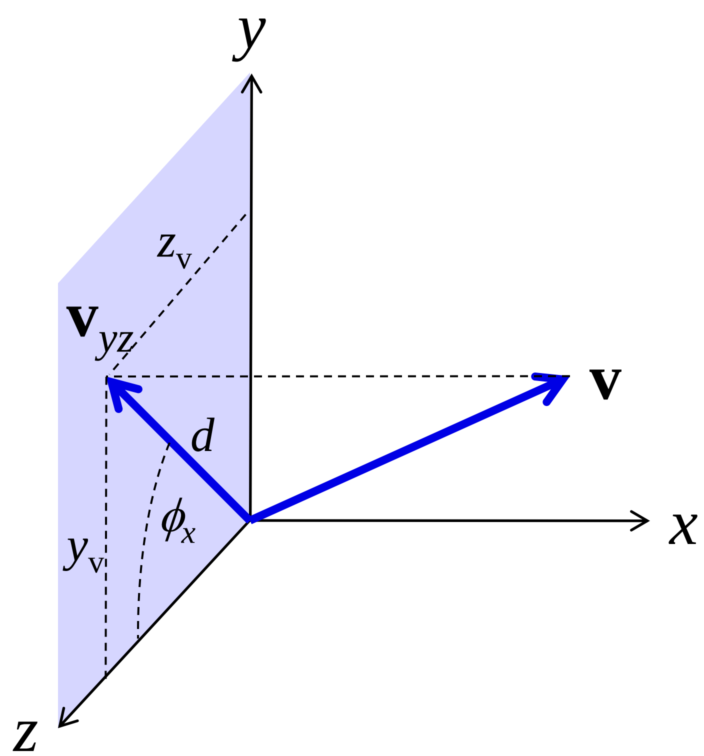
1. Project **v** onto the *yz* plane and let $d = \text{sqrt}(y_v^2 + z_v^2)$



Rotate v to z

$\mathbf{v} = (x_v, y_v, z_v)$, $x_v^2 + y_v^2 + z_v^2 = 1$

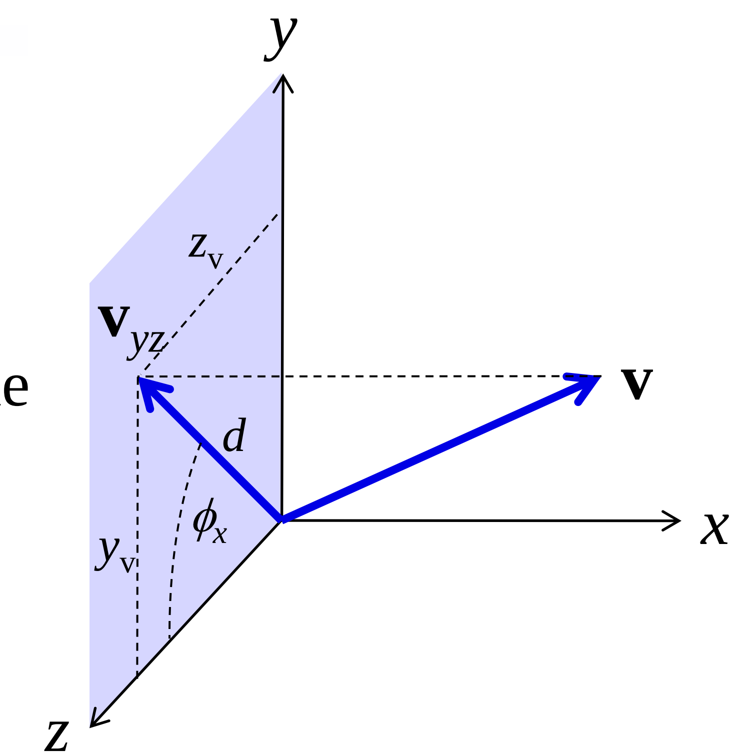
1. Project **v** onto the *yz* plane and let $d = \text{sqrt}(y_v^2 + z_v^2)$
2. Then $\cos\phi_x = z_v/d$ and $\sin\phi_x = y_v/d$



Rotate v to z

$\mathbf{v} = (x_v, y_v, z_v),\ x_v^2 + y_v^2 + z_v^2 = 1$

1. Project $\mathbf{v}$ onto the yz plane and let $d = \mathrm{sqrt}(y_v^2 + z_v^2)$
2. Then $\cos\phi_x = z_v/d$ and $\sin\phi_x = y_v/d$
3. Rotate $\mathbf{v}$ by ϕ_x about x into the xz plane

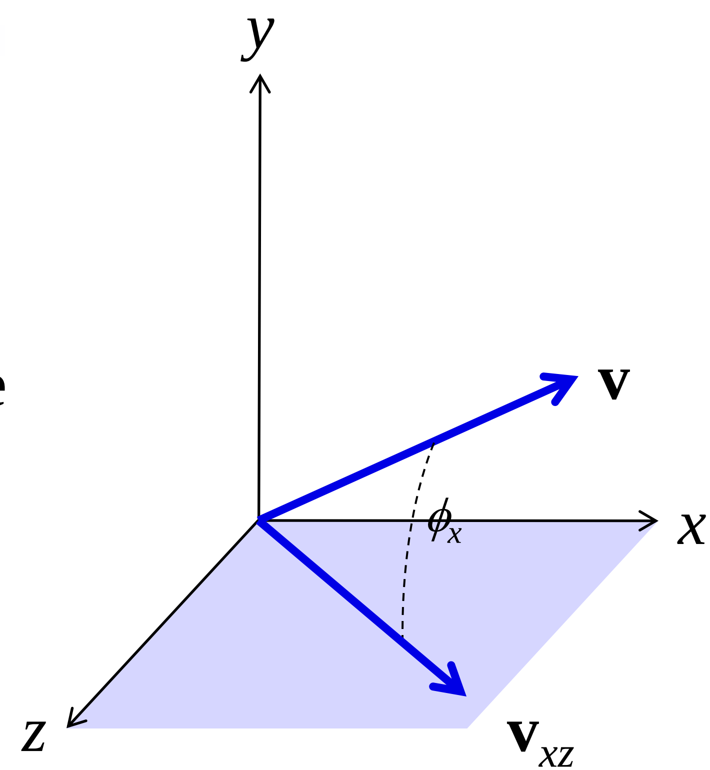


$$\begin{bmatrix} 1 & & & \\ & \frac{z_{\mathbf{v}}}{d} & -\frac{y_{\mathbf{v}}}{d} & \\ & \frac{y_{\mathbf{v}}}{d} & \frac{z_{\mathbf{v}}}{d} & \\ & & & 1 \end{bmatrix}$$

Rotate v to z

$\mathbf{v} = (x_v, y_v, z_v),\ x_v{}^2 + y_v{}^2 + z_v{}^2 = 1$

1. Project **v** onto the *yz* plane and let $d = \text{sqrt}(y_v{}^2 + z_v{}^2)$
2. Then $\cos\phi_x = z_v/d$ and $\sin\phi_x = y_v/d$
3. Rotate **v** by ϕ_x about *x* into the *xz* plane

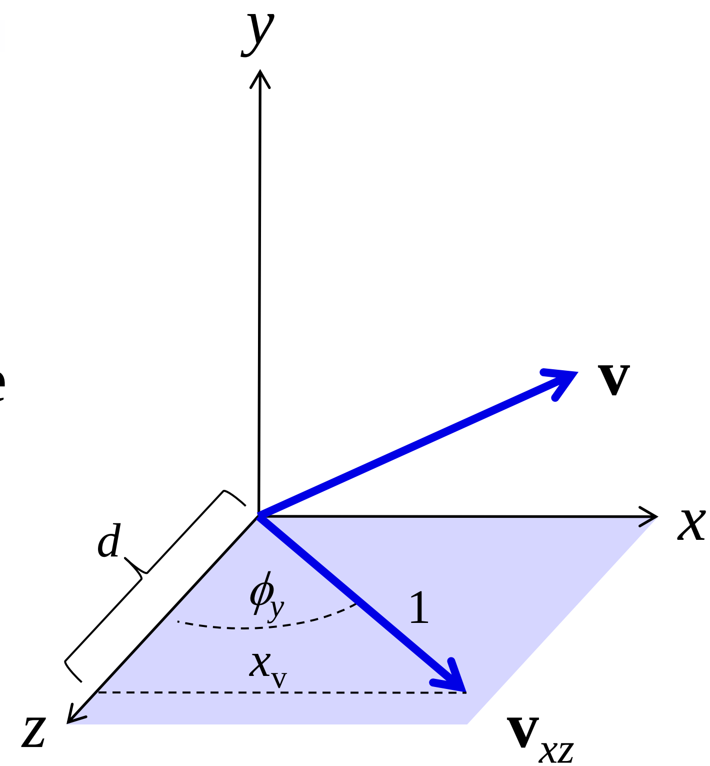


$$\begin{bmatrix} 1 & & & \\ & \frac{z_{\mathbf{v}}}{d} & -\frac{y_{\mathbf{v}}}{d} & \\ & \frac{y_{\mathbf{v}}}{d} & \frac{z_{\mathbf{v}}}{d} & \\ & & & 1 \end{bmatrix}$$

Rotate v to z

$\mathbf{v} = (x_v, y_v, z_v)$, $x_v^2 + y_v^2 + z_v^2 = 1$

1. Project **v** onto the *yz* plane and let $d = \text{sqrt}(y_v^2 + z_v^2)$
2. Then $\cos\phi_x = z_v/d$ and $\sin\phi_x = y_v/d$
3. Rotate **v** by ϕ_x about x into the xz plane
4. Then $\cos\phi_y = d$ and $\sin\phi_y = x_v$

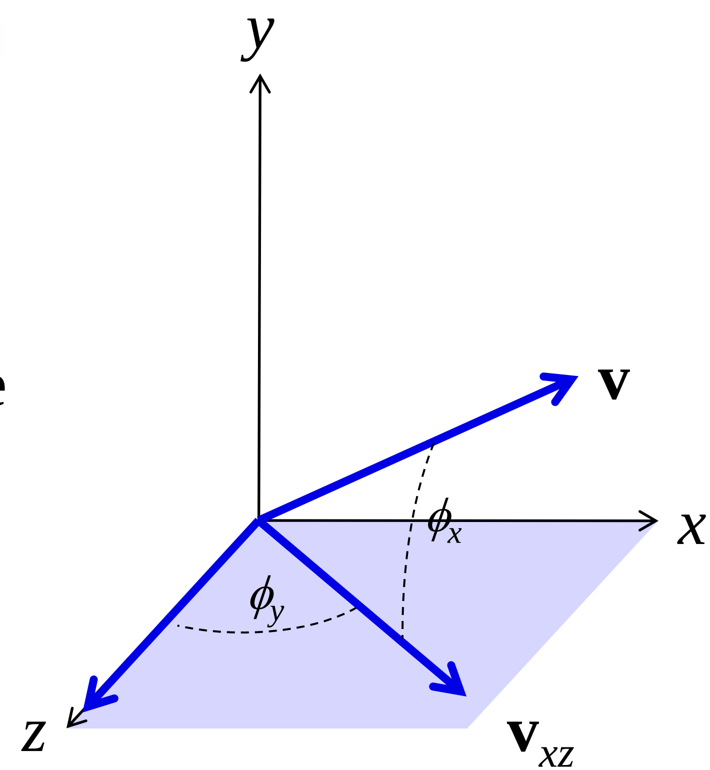


$$\begin{bmatrix} 1 & & & \\ & \frac{z_{\mathbf{v}}}{d} & -\frac{y_{\mathbf{v}}}{d} & \\ & \frac{y_{\mathbf{v}}}{d} & \frac{z_{\mathbf{v}}}{d} & \\ & & & 1 \end{bmatrix}$$

Rotate v to z

$\mathbf{v} = (x_v, y_v, z_v)$, $x_v^2 + y_v^2 + z_v^2 = 1$

1. Project $\mathbf{v}$ onto the yz plane and let $d = \text{sqrt}(y_v^2 + z_v^2)$
2. Then $\cos\phi_x = z_v/d$ and $\sin\phi_x = y_v/d$
3. Rotate $\mathbf{v}$ by ϕ_x about x into the xz plane
4. Then $\cos\phi_y = d$ and $\sin\phi_y = x_v$
5. Rotate $\mathbf{v}_{xz}$ by ϕ_y about y into the z axis



$$\begin{bmatrix} d & & -x_{\mathbf{v}} & \\ & 1 & & \\ x_{\mathbf{v}} & & d & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & \frac{z_{\mathbf{v}}}{d} & -\frac{y_{\mathbf{v}}}{d} & \\ & \frac{y_{\mathbf{v}}}{d} & \frac{z_{\mathbf{v}}}{d} & \\ & & & 1 \end{bmatrix}$$

Rotate θ about **v**

- Let $R_{\mathbf{v}}(\theta)$ be the rotation matrix for rotation by θ about arbitrary axis direction **v**
- Recall $(R_x\ R_y)$ is the matrix (product) that rotates direction **v** to z axis
- Then

$$R_{\mathbf{v}}(\theta) = (R_y\ R_x)^{-1}\ R_z(\theta)\ (R_y\ R_x)$$
$$= R_x^{-1}\ R_y^{-1}\ R_z(\theta)\ R_y\ R_x$$
$$= R_x^{T}\ R_y^{T}\ R_z(\theta)\ R_y\ R_x$$

(since the inverse of a rotation matrix is the transpose of the rotation matrix)

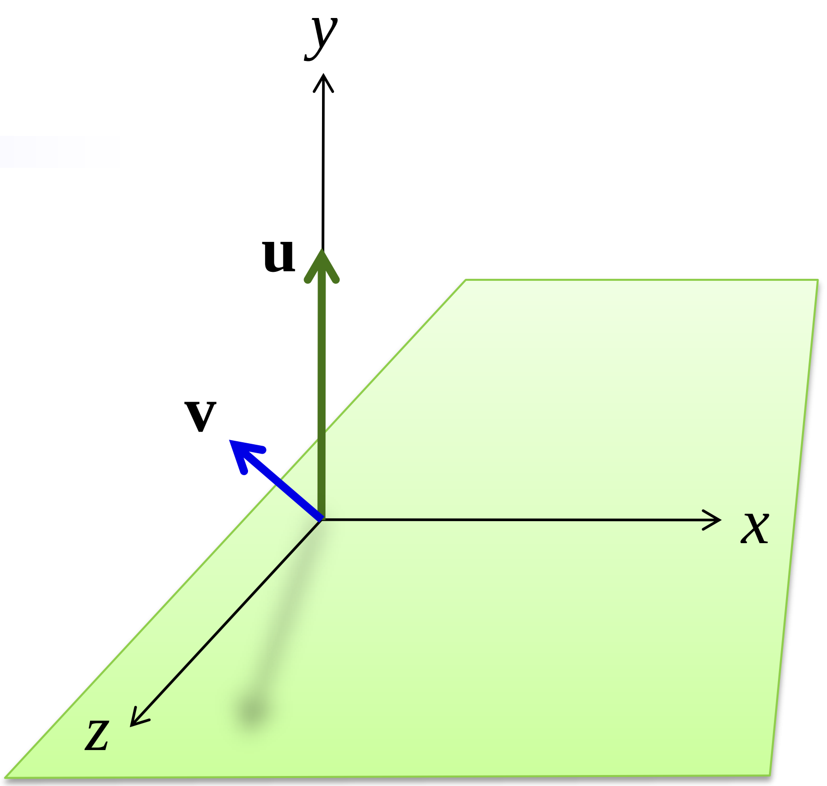
$$\mathrm{R}_x = \begin{bmatrix} 1 & & & \\ & \frac{z_v}{d} & -\frac{y_v}{d} & \\ & \frac{y_v}{d} & \frac{z_v}{d} & \\ & & & 1 \end{bmatrix}$$

$$\mathrm{R}_y = \begin{bmatrix} d & & x_v & \\ & 1 & & \\ x_v & & d & \\ & & & 1 \end{bmatrix}$$

$$\mathrm{R}_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & & \\ \sin\theta & \cos\theta & & \\ & & 1 & \\ & & & 1 \end{bmatrix}$$

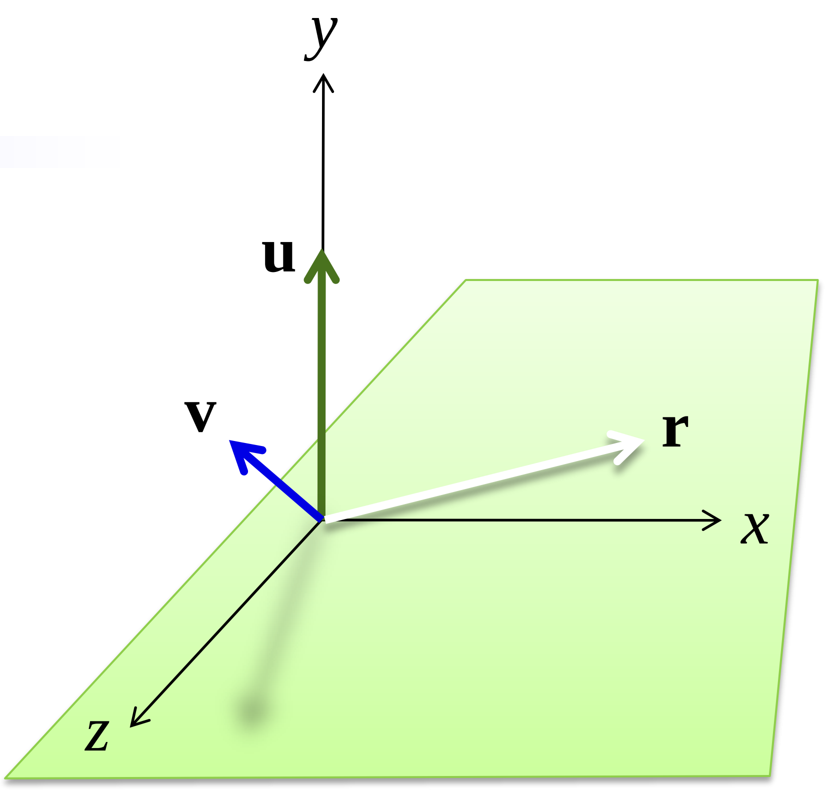
Easier Way

- Find an orthonormal vector system



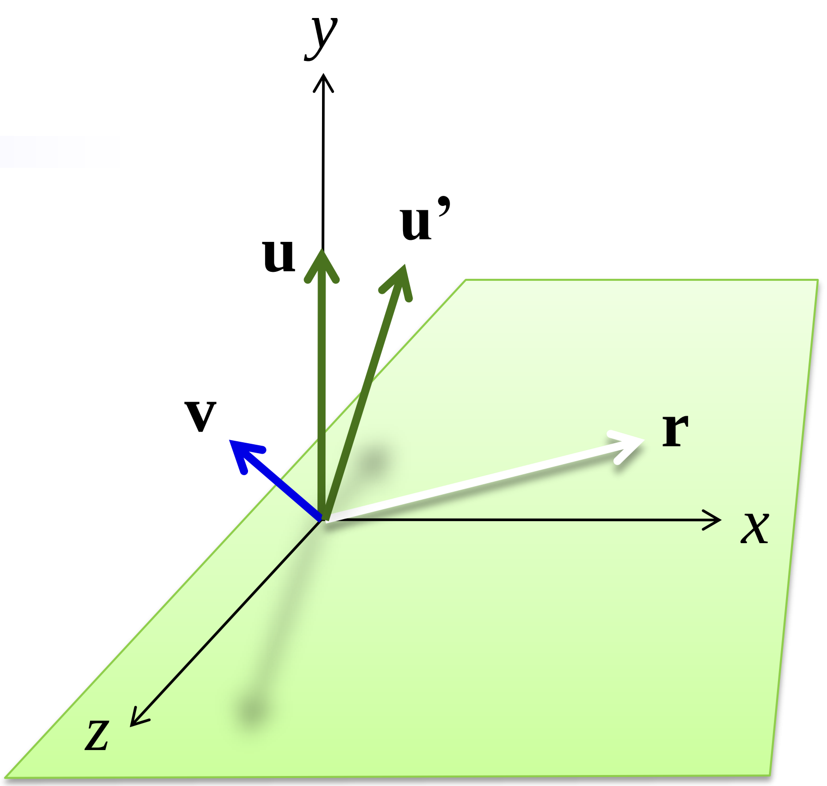
Easier Way

- Find an orthonormal vector system
  - Let $\mathbf{r} = \mathbf{u} \times \mathbf{v} / ||\mathbf{u} \times \mathbf{v}||$



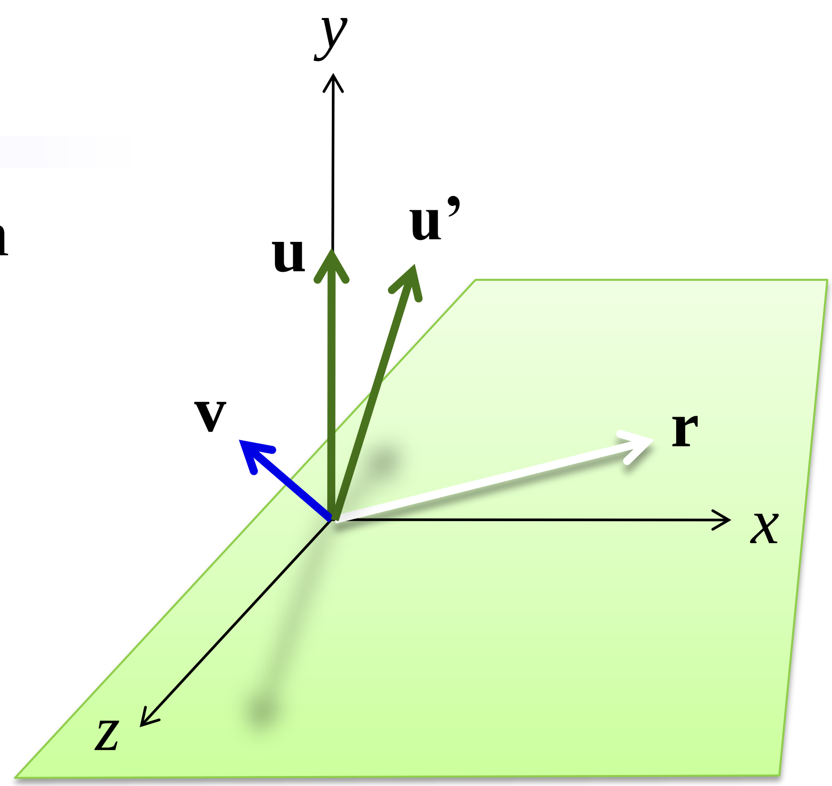
Easier Way

- Find an orthonormal vector system
  - Let $\mathbf{r} = \mathbf{u} \times \mathbf{v} / ||\mathbf{u} \times \mathbf{v}||$
  - Let $\mathbf{u}' = \mathbf{v} \times \mathbf{r}$



Easier Way

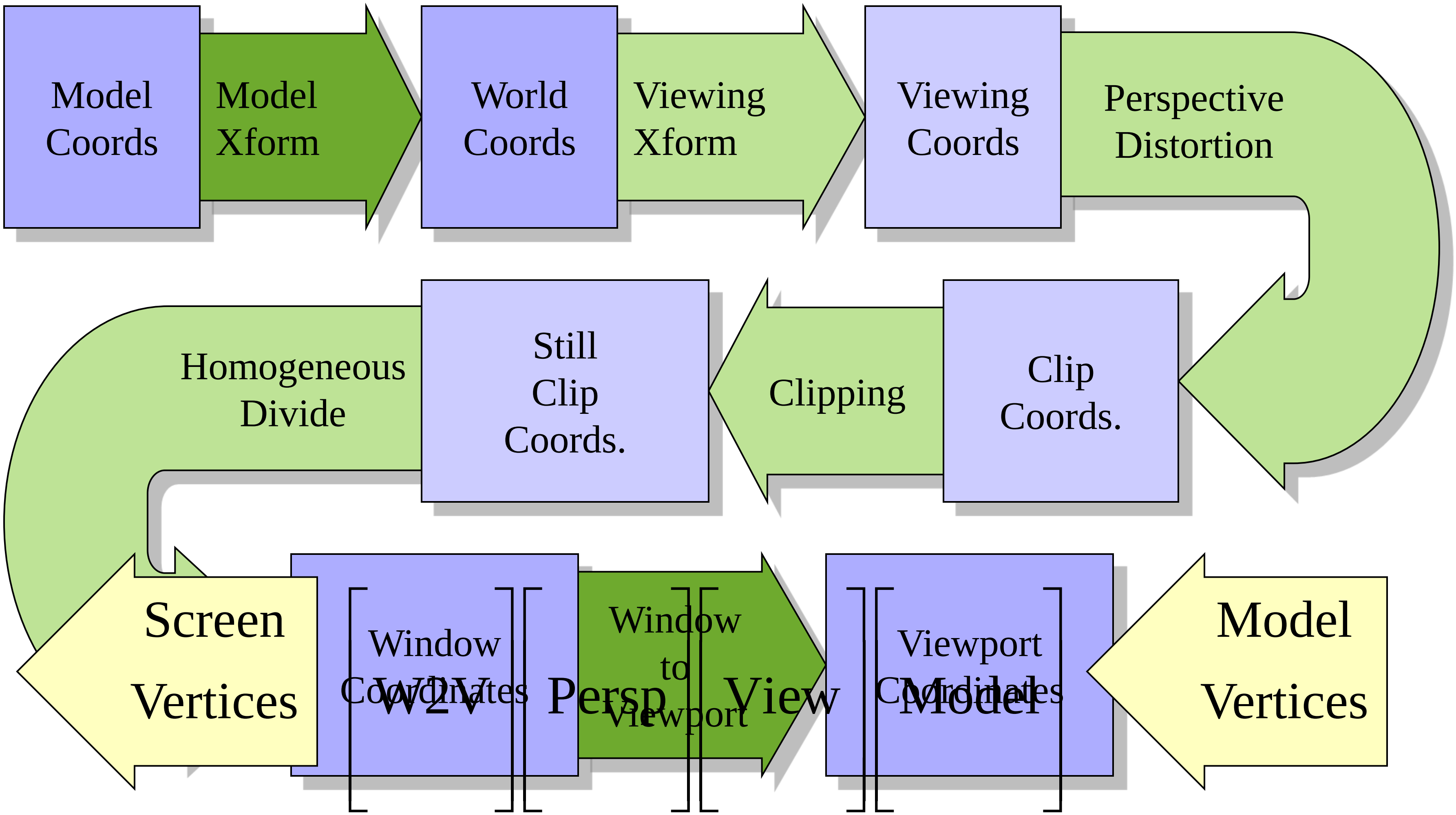
- Find an orthonormal vector system
  - Let $\mathbf{r} = \mathbf{u} \times \mathbf{v} / \|\mathbf{u} \times \mathbf{v}\|$
  - Let $\mathbf{u}' = \mathbf{v} \times \mathbf{r}$
- Find a rotation from $\langle \mathbf{r}, \mathbf{u}', \mathbf{v} \rangle \rightarrow \langle \mathbf{x}, \mathbf{y}, \mathbf{z} \rangle$



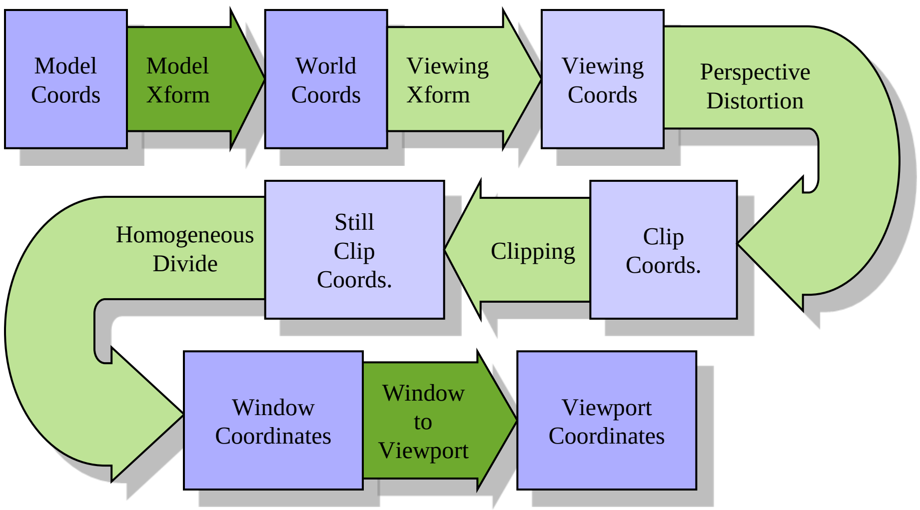
$$\begin{bmatrix} r_x & u'_x & v_x & \\ r_y & u'_y & v_y & \\ r_z & u'_z & v_z & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix}$$

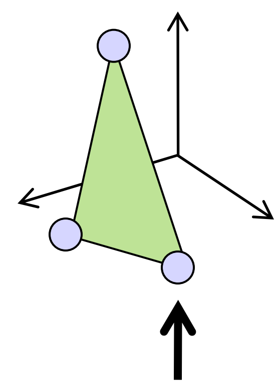
$$\begin{bmatrix} r_x & r_y & r_z & \\ u'_x & u'_y & u'_z & \\ v_x & v_y & v_z & \\ & & & 1 \end{bmatrix} \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

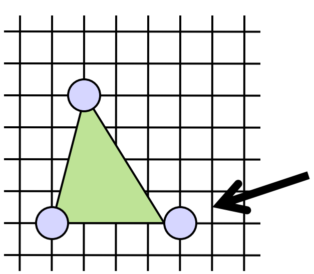
Graphics Pipeline



Graphics Pipeline

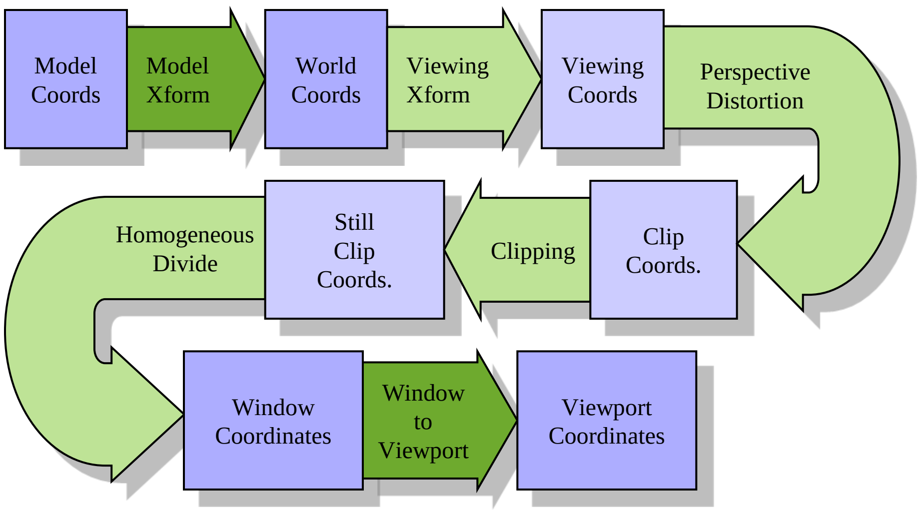


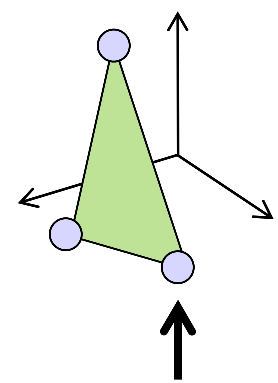


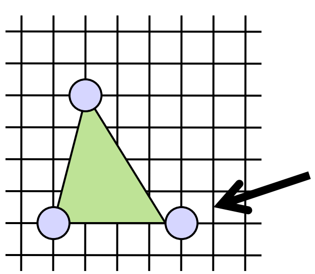


$$\begin{bmatrix} x_s \\ y_s \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} & & \\ & \text{W2V} & \\ & & \end{bmatrix} \begin{bmatrix} & & \\ & \text{Persp} & \\ & & \end{bmatrix} \begin{bmatrix} & & \\ & \text{View} & \\ & & \end{bmatrix} \begin{bmatrix} & & \\ & \text{Model} & \\ & & \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix}$$

Graphics Pipeline



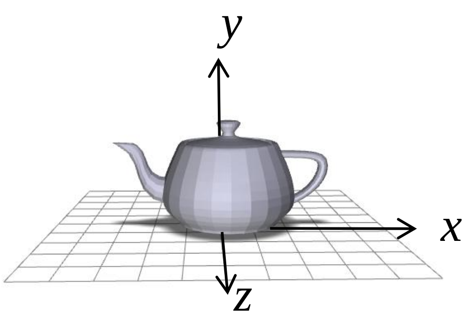




$$\begin{bmatrix} x_s \\ y_s \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} & & & \\ & \mathbf{M} & & \\ & & & \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix}$$

Transformation Order

```
glutSolidTeapot(1);
```



$$\begin{bmatrix} x_s \\ y_s \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathrm{M} \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix}$$

```
glRotate3f(-90, 0,0,1);
glTranslate3f(0,1,0);
glutSolidTeapot(1);
```

$$\begin{bmatrix} x_s \\ y_s \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathrm{M} \end{bmatrix} \begin{bmatrix} \mathrm{R} \end{bmatrix} \begin{bmatrix} \mathrm{T} \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix}$$

```
glTranslate3f(0,1,0);
glRotate3f(-90, 0,0,1);
glutSolidTeapot(1);
```

$$\begin{bmatrix} x_s \\ y_s \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathrm{M} \end{bmatrix} \begin{bmatrix} \mathrm{T} \end{bmatrix} \begin{bmatrix} \mathrm{R} \end{bmatrix} \begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix}$$