TRANSFORMAÇÕES EM 3D - ÊNFASE ROTAÇÕES

- Translação e Escala
- Rotação por Ângulos de Euler
- Rotação por Definição de um Referencial

TRANSLAÇÃO E ESCALA

- São extensões naturais de seus equivalentes em 2D

$$T = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{e} \quad S = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

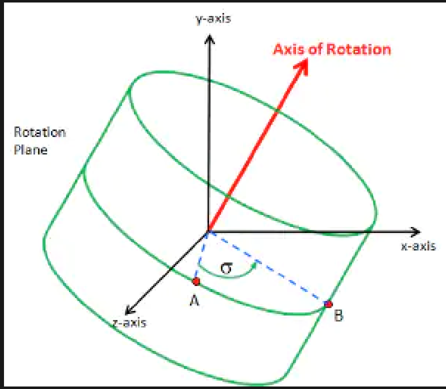
$$\text{Inversas :}$$

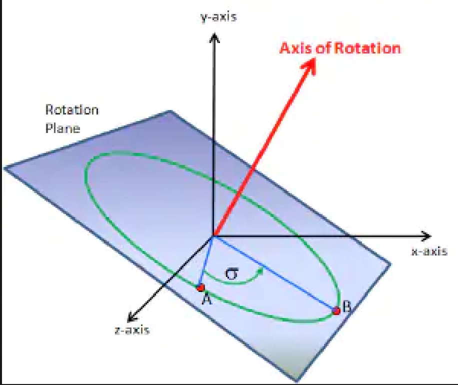
$$T^{-1}(t_x, t_y, t_z) = T(-t_x, -t_y, -t_z)$$

$$S^{-1}(s_x, s_y, s_z) = S(\frac{1}{s_x}, \frac{1}{s_y}, \frac{1}{s_z})$$

ROTAÇÃO

- Podemos visualizar a rotação por:
  - Eixo e ângulo de rotação no plano perpendicular

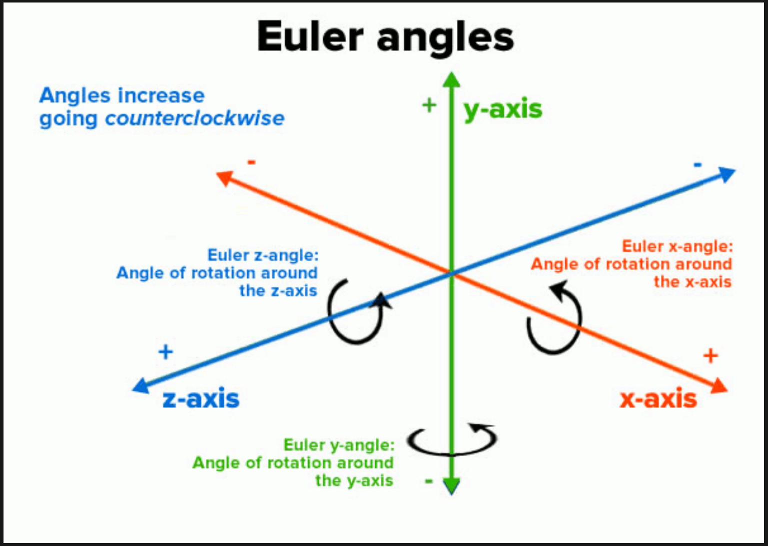




- Note que se o eixo está em Z → plano de rotação é XY

ROTAÇÃO POR ÂNGULOS DE EULER

- Cada eixo coordenado é um eixo de rotação
  - 3 planos de rotação
    - XY, ZX e YZ
    - Assume-se rotação positiva → a regra da mão direita



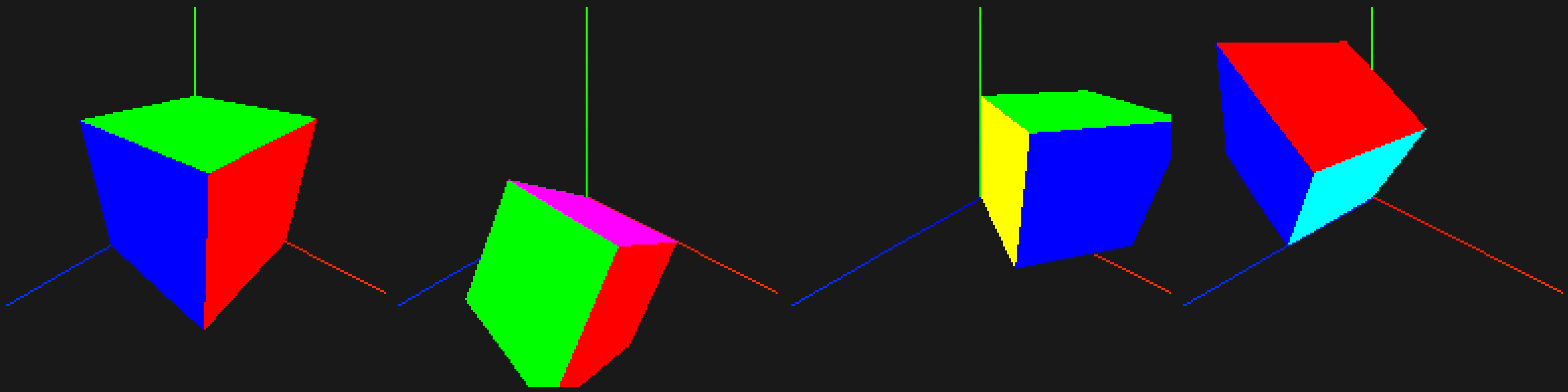
MATRIZES

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) & 0 \\ 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; R_x^{-1}(\alpha) = R_x^T(\alpha) = R_x(-\alpha)$$

$$R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; R_y^{-1}(\beta) = R_y^T(\beta) = R_y(-\beta)$$

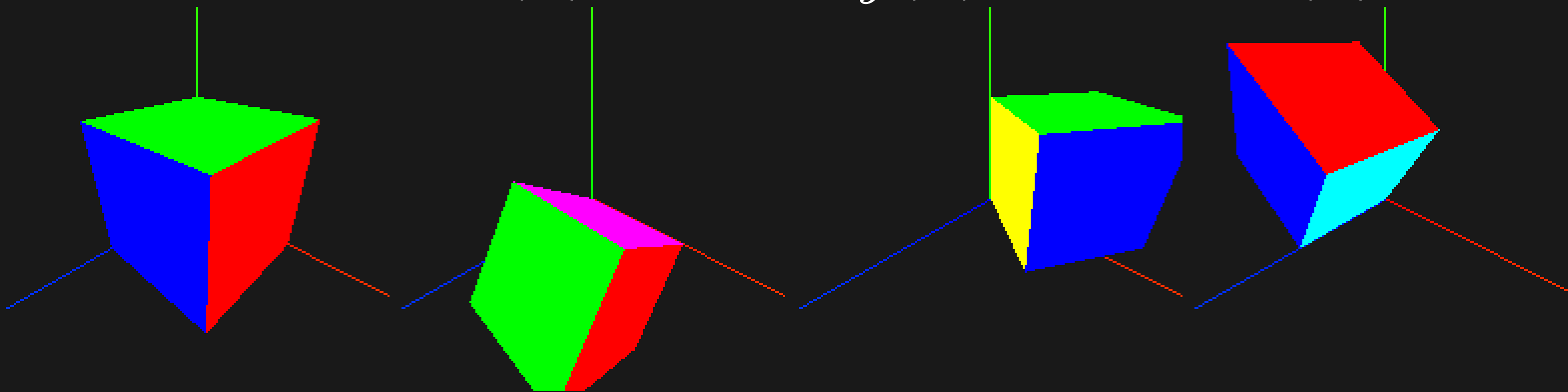
$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; R_z^{-1}(\gamma) = R_z^T(\gamma) = R_z(-\gamma)$$

EXEMPLO - ROTAÇÃO EULER



EXEMPLO - ROTAÇÃO EULER

$R_x(\theta)$ $R_y(\theta)$ $R_z(\theta)$

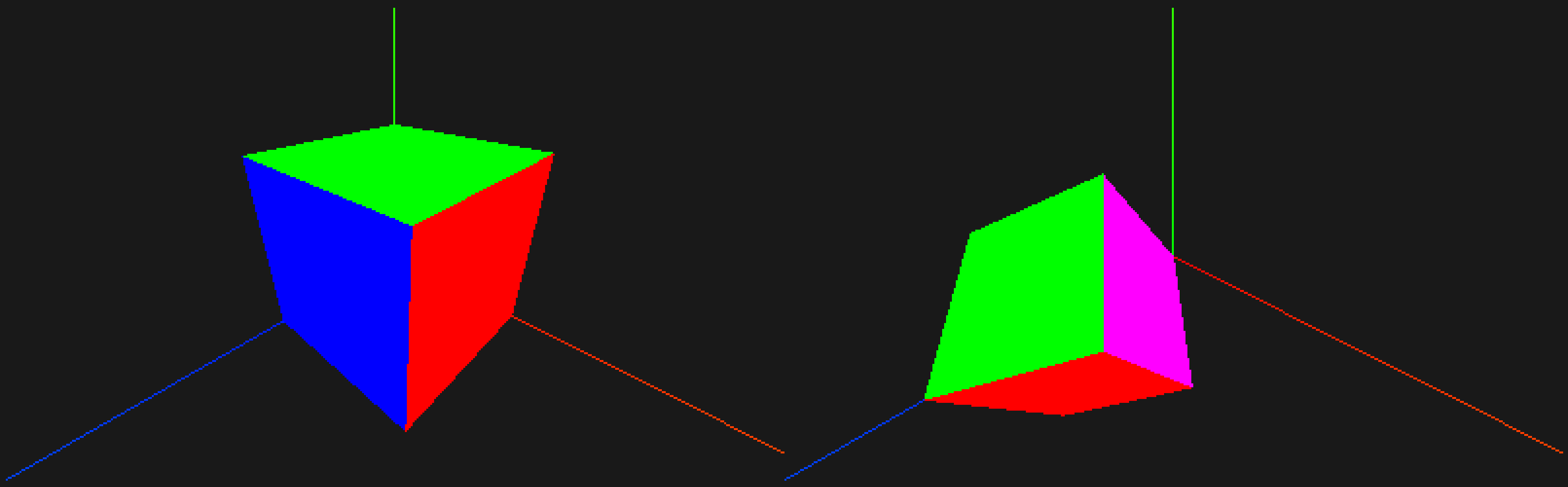


ROTAÇÃO ARBITRÁRIA

- Rotação arbitrária 3D é obtida pela combinação das 3 rotações pelos ângulos de Euler
  - Não há uma ordem específica
    - Lembre-se: a ordem importa
    - No Three.js, a ordem pode ser especificada pelo usuário

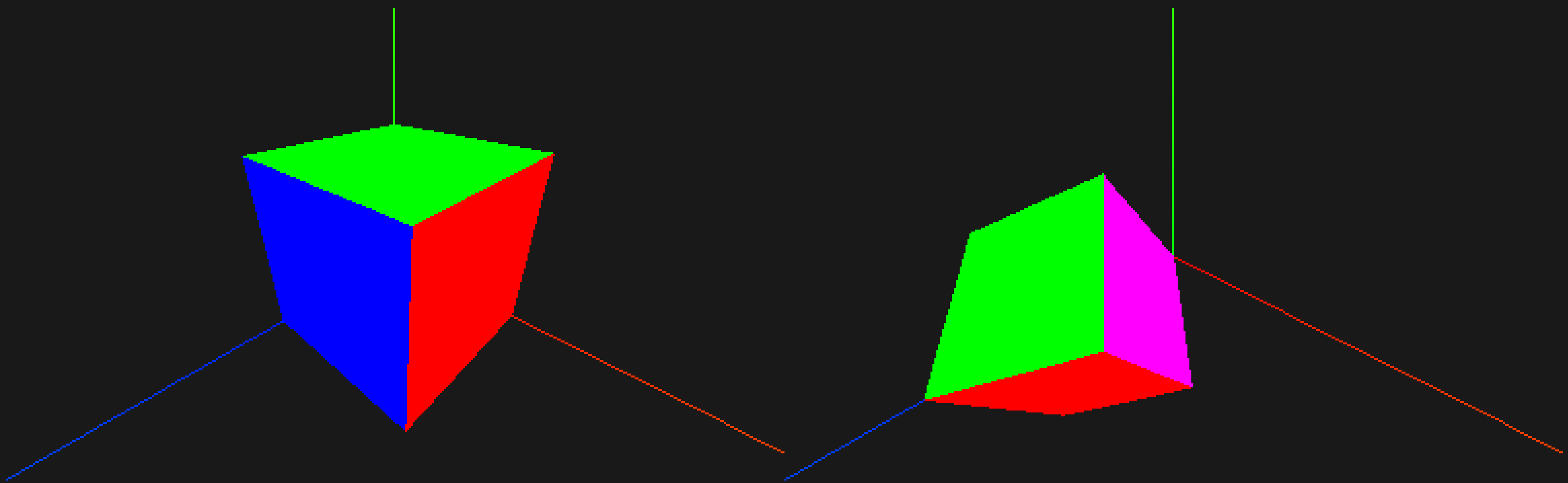
EXEMPLO

Qual a matriz de Rotação, usando ângulos de Euler?



EXEMPLO

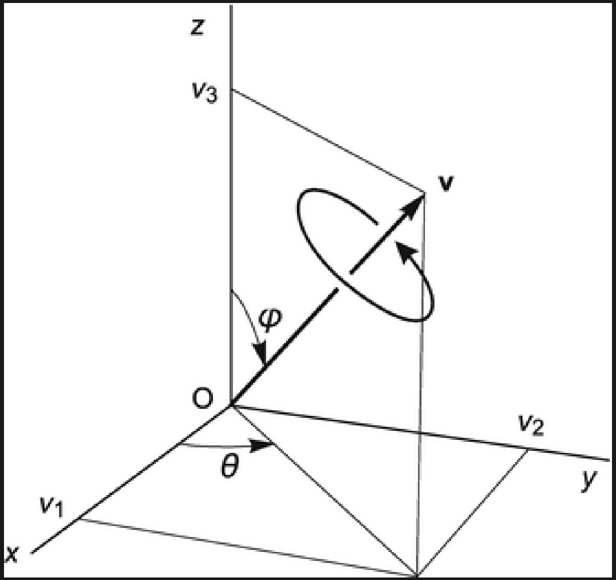
Qual a matriz de Rotação, usando ângulos de Euler?



$$\begin{aligned} M &= R_x(\arcsin(\tfrac{1}{\sqrt{3}}))R_y(-\arccos(\tfrac{1}{\sqrt{2}})) \\ &= R_x(35.26^{\text{o}})R_y(-45^{\text{o}}) \end{aligned}$$

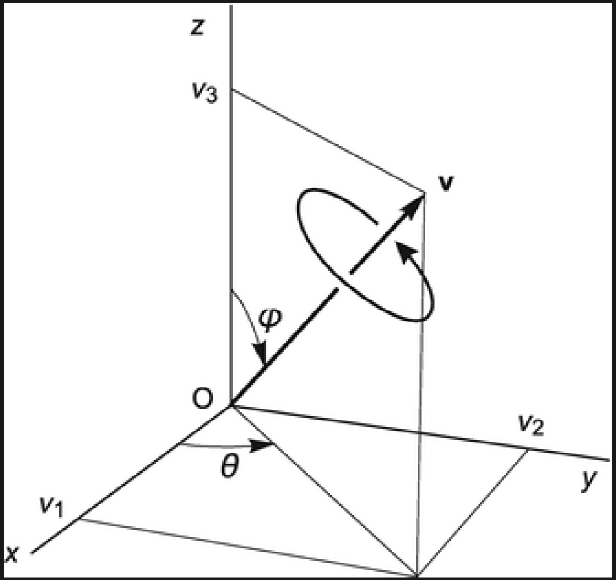
EIXO ARBITRÁRIO

- Rotação → plano XY
- Rotação → plano ZX
- Rotação → eixo cartesiano
- Rotação inversa → plano ZX
- Rotação inversa → plano XY



EIXO ARBITRÁRIO

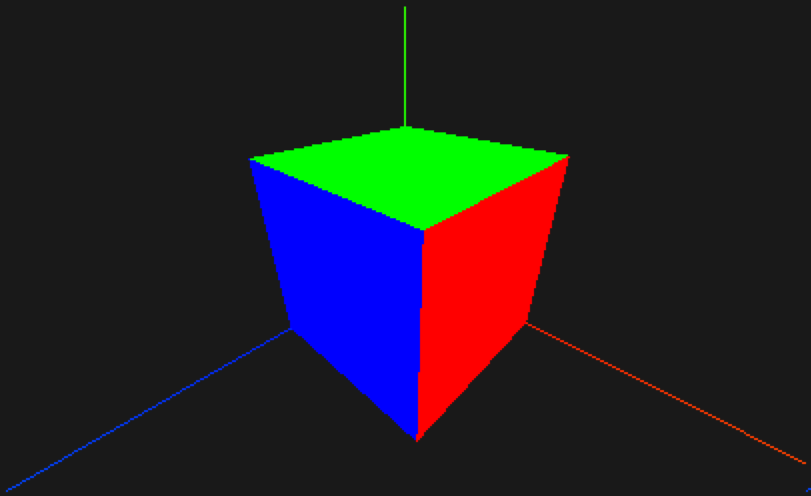
- Rotação → plano XY
- Rotação → plano ZX
- Rotação → eixo cartesiano
- Rotação inversa → plano ZX
- Rotação inversa → plano XY

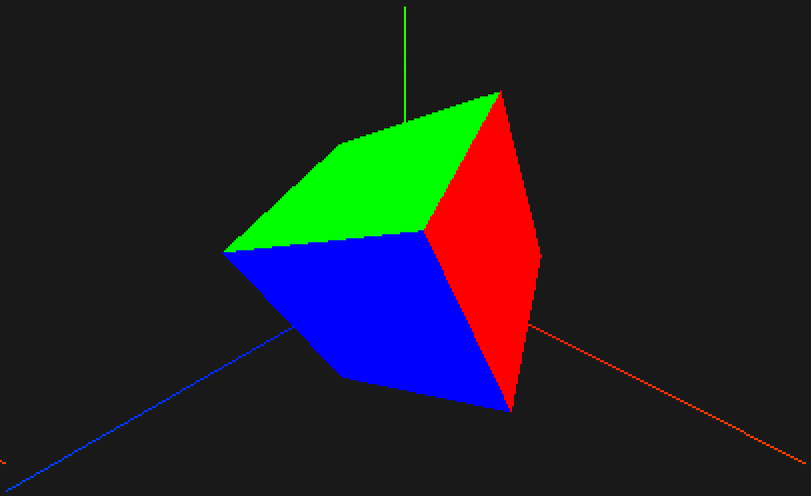


$$R(\alpha) = R_z(\theta)R_y(\phi)R_z(\alpha)R_y(-\phi)R_z(-\theta)$$

EXEMPLO

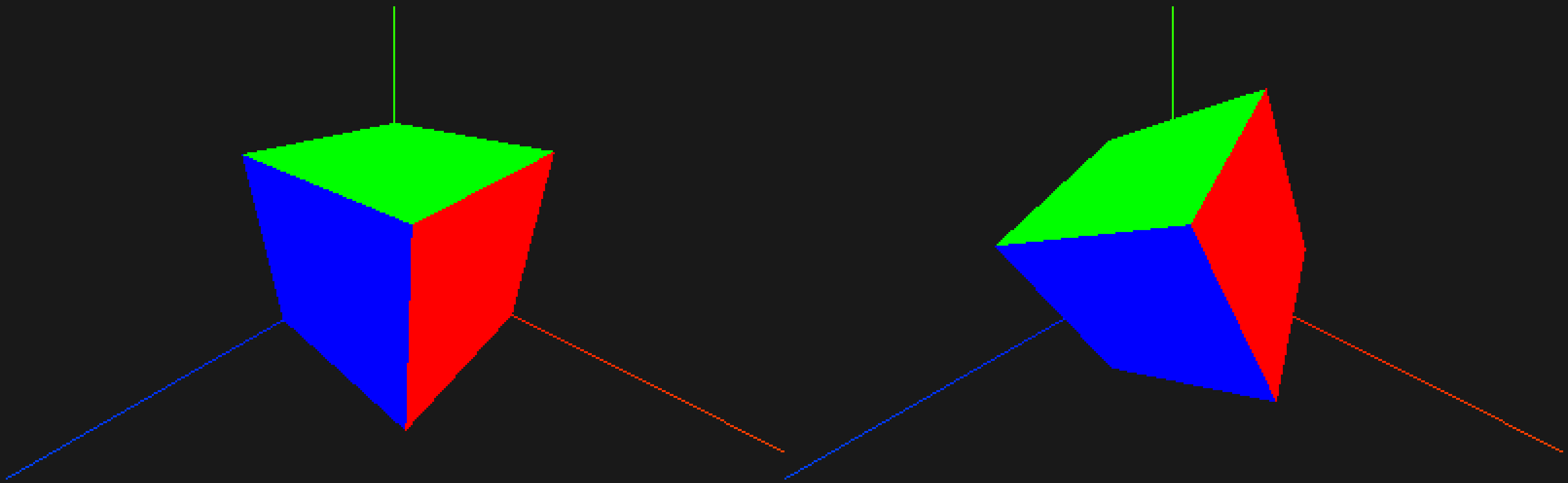
Qual a matriz de Rotação, usando ângulos de Euler?





EXEMPLO

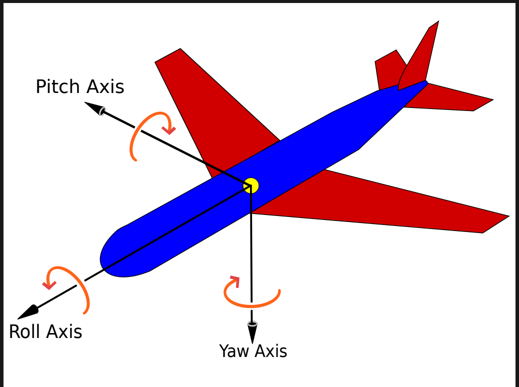
Qual a matriz de Rotação, usando ângulos de Euler?



$$M = R_y(45^o).R_x(-35.26^o).R_z(\theta).R_x(35.26^o).R_y(-45^o)$$

YAW, PITCH E ROLL

- Nomenclatura usada na aviação
  - Pode-se pensar como uma das variações dos Ângulos de Euler
    - No artigo do Wikipedia há maiores detalhes sobre as diferenças



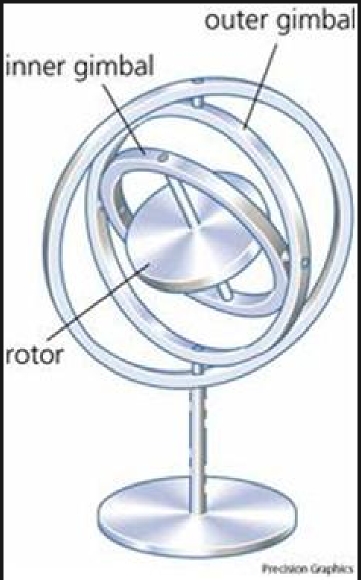
GIMBAL LOCK

- Da documentação do Three.js

Three.js provides two ways of representing 3D rotations: Euler angles and Quaternions, as well as methods for converting between the two. Euler angles are subject to a problem called “gimbal lock,” where certain configurations can lose a degree of freedom (preventing the object from being rotated about one axis). For this reason, object rotations are always stored in the object’s quaternion.

GIMBAL

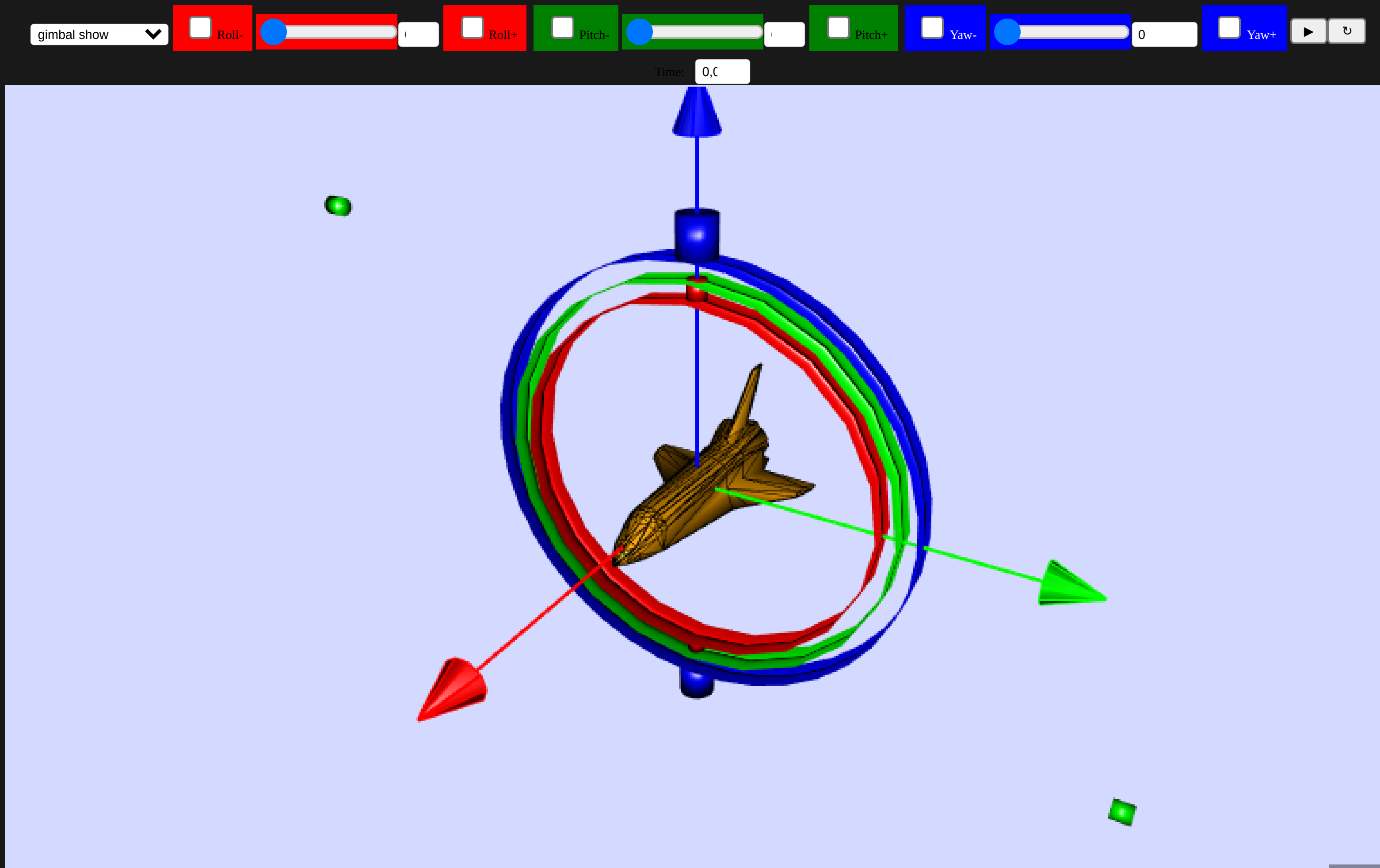
- Dispositivo de anéis concêntricos, que permite orientar um objeto no espaço 3D
  - Possui 3 graus de liberdade



GIMBAL LOCK

- Situação que ocorre, devido a certas orientações dos anéis, no qual se perde um grau de liberdade
  - Em ângulos de Euler, dada uma ordem ($R_x R_y R_z$), *gimbal lock* ocorre quando a rotação do meio for de 90º, ou seja, $R_x(\theta) R_y(\beta = 90º) R_z(\alpha)$
  - Um vídeo com uma boa explicação → *Euler (gimbal lock) Explained*
  - Uma aplicação para se testar online → *Flight Dynamics Gimbal JavaScript HTML5 Applet Simulation Model*

DEMONSTRAÇÃO



ROTAÇÃO POR DEFINIÇÃO DE UM REFERENCIAL

- Construção de um sistema referencial ortogonal → defini uma matriz de rotação

UMA METODOLOGIA

- Para construir o unicamente o referencial $\vec{u}\vec{v}\vec{n}$
  - Precisam ser dados dois vetores:
    - Um deles deve estar na direção de um dos eixos
      - Vamos chamá-lo de $\vec{U}$
    - Outro deve estar no plano $\vec{u}\vec{v}$, não colinear com $\vec{U}$
      - Vamos chamá-lo de $\vec{V}'$

O MÉTODO

$$\vec{u} = \frac{\vec{U}}{\|\vec{U}\|}$$

$$\vec{n} = \frac{\vec{u} \times \vec{V}'}{\|\vec{u} \times \vec{V}'\|}$$

$$\vec{v} = \frac{\vec{n} \times \vec{u}}{\|\vec{n} \times \vec{u}\|} = \vec{n} \times \vec{u}$$

$$R = \left[\begin{array}{ccc|c} \vec{u} & \vec{v} & \vec{n} & \vec{0} \\ \hline & \vec{0} & & 1 \end{array}\right]$$

$$\therefore \begin{bmatrix} u_x & v_x & n_x & 0 \\ u_y & v_y & n_y & 0 \\ u_z & v_z & n_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

O MÉTODO

$$\vec{u} = \frac{\vec{U}}{\|\vec{U}\|}$$

$$\vec{n} = \frac{\vec{u} \times \vec{V}'}{\|\vec{u} \times \vec{V}'\|}$$

$$\vec{v} = \frac{\vec{n} \times \vec{u}}{\|\vec{n} \times \vec{u}\|} = \vec{n} \times \vec{u}$$

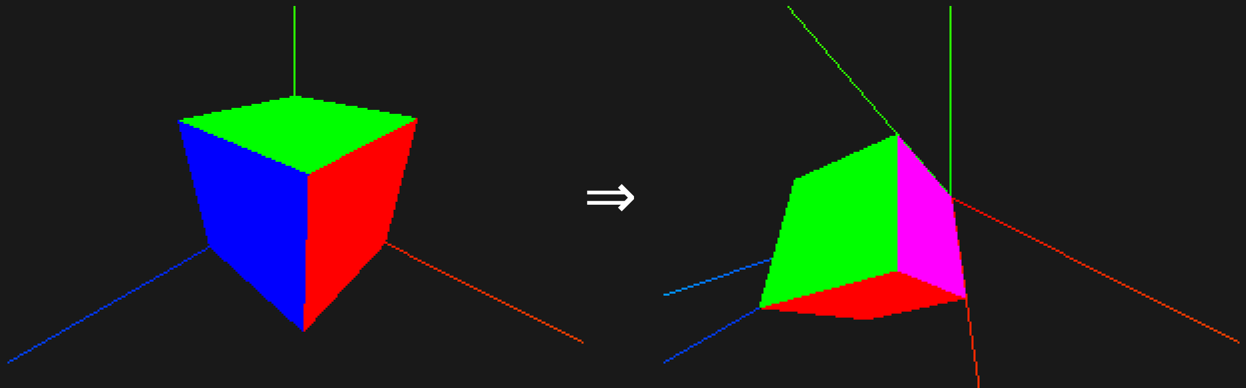
$$R = \left[\begin{array}{ccc|c} \vec{u} & \vec{v} & \vec{n} & \vec{0} \\ \hline & \vec{0} & & 1 \end{array}\right]$$

$$\therefore \begin{bmatrix} u_x & v_x & n_x & 0 \\ u_y & v_y & n_y & 0 \\ u_z & v_z & n_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Note: $\vec{n}$ e $\vec{u}$ são unitários e ortogonais $\therefore$ $\|\vec{n} \times \vec{u}\| = 1$

EXEMPLO

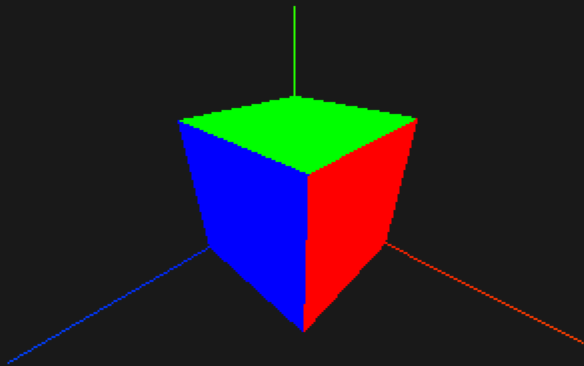
Qual é a matriz de Rotação (método: definir referencial)?



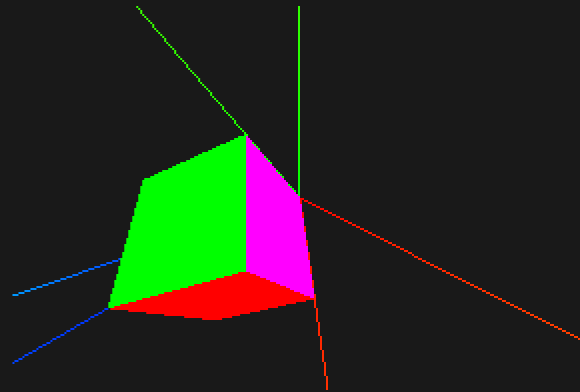
⇒

EXEMPLO

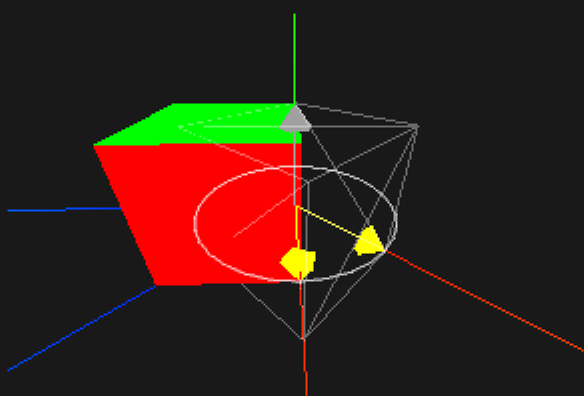
Qual é a matriz de Rotação (método: definir referencial)?



$\Rightarrow$



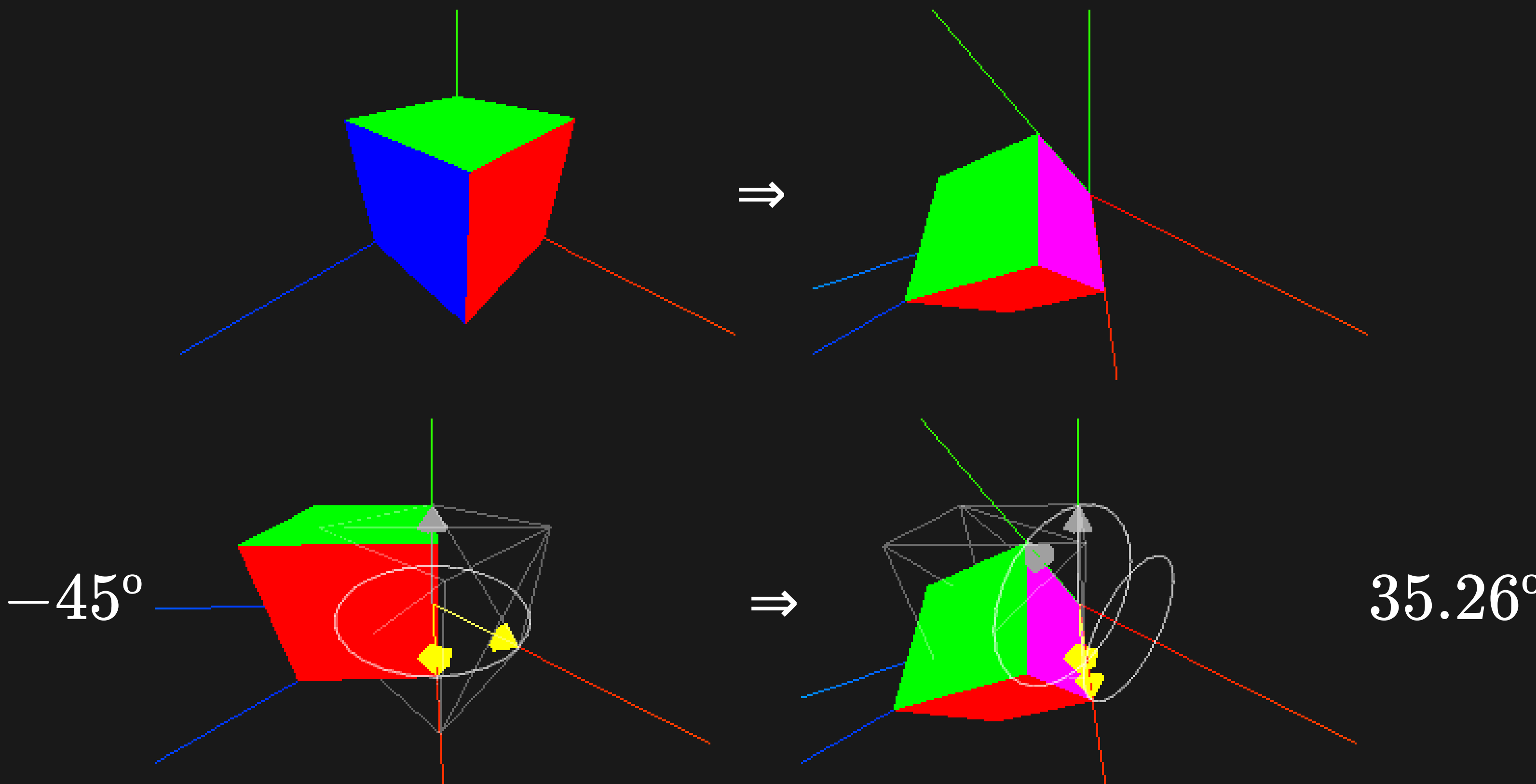
-45°



$\Rightarrow$

EXEMPLO

Qual é a matriz de Rotação (método: definir referencial)?



VETORES

$$\vec{U} : \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \xrightarrow{R_y(-45^o)} \begin{bmatrix} \frac{\sqrt{2}}{2} \\ 0 \\ \frac{\sqrt{2}}{2} \end{bmatrix} \xrightarrow{R_x(35.26^o)} \begin{bmatrix} \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{6}}{6} \\ \frac{\sqrt{3}}{3} \end{bmatrix}$$

$$\vec{V}' : \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \xrightarrow{R_x(35.26^o)} \begin{bmatrix} 0 \\ \frac{\sqrt{6}}{3} \\ \frac{\sqrt{3}}{3} \end{bmatrix}$$

CÁLCULO

$$\vec{u} = \frac{\vec{U}}{\|\vec{U}\|} = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{6}}{6} \\ \frac{\sqrt{3}}{3} \end{bmatrix}$$

$$\vec{n} = \frac{\vec{u} \times \vec{V}'}{\|\vec{u} \times \vec{V}'\|} = \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ -\frac{\sqrt{6}}{6} \\ \frac{\sqrt{3}}{3} \end{bmatrix}$$

$$\vec{v} = \frac{\vec{n} \times \vec{u}}{\|\vec{n} \times \vec{u}\|} = \vec{n} \times \vec{u} = \begin{bmatrix} 0 \\ \frac{\sqrt{6}}{3} \\ \frac{\sqrt{3}}{3} \end{bmatrix}$$

MATRIZ DE ROTAÇÃO

$$R = \left[\begin{array}{ccc|c} \vec{u} & \vec{v} & \vec{n} & \vec{0} \\ \hdashline & \vec{0} & & 1 \end{array}\right]$$

$$\therefore R = \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0 \\ -\frac{\sqrt{6}}{6} & \frac{\sqrt{6}}{3} & -\frac{\sqrt{6}}{6} & 0 \\ \frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3} & \frac{\sqrt{3}}{3} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Contra prova: Verifique se é igual à matriz:

$R_x(35.26^o)R_y(-45^o)$

// reveal.js plugins