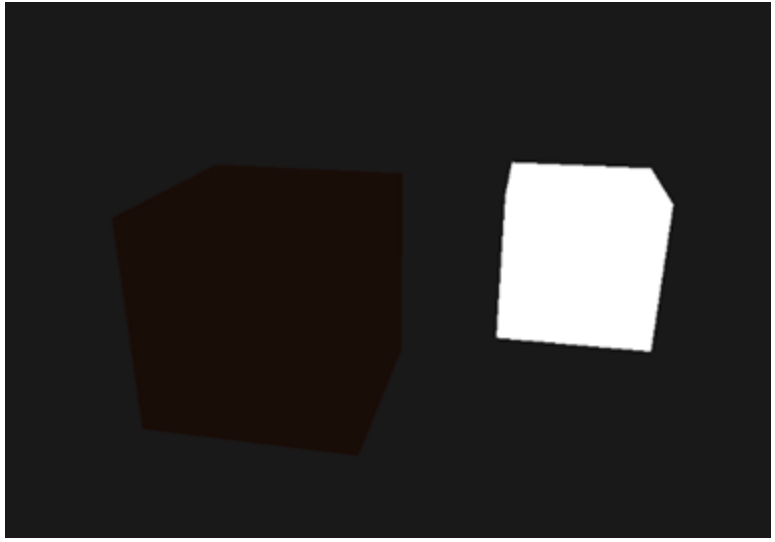
Insper

Computação Gráfica

Revisão: Iluminação e Animação

Iluminação/Reflexão Ambiente

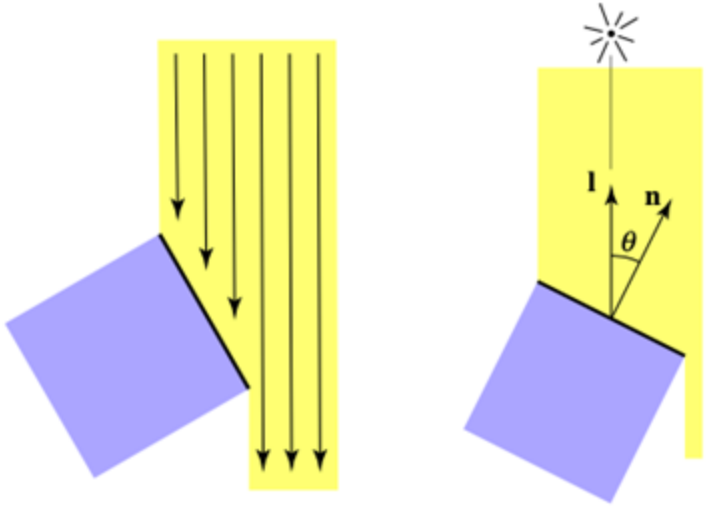
A iluminação ambiente (AmbientLight) resulta da dispersão e reflexão da luz originalmente emitida diretamente por fontes de luz. A quantidade de luz ambiente está associada às luzes individuais na cena. Esta é uma aproximação grosseira de como a reflexão ambiental realmente ocorre na natureza.

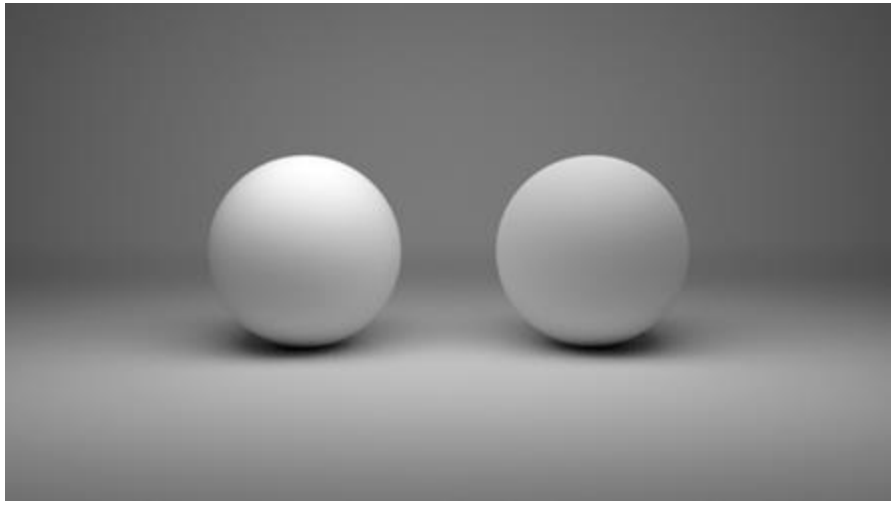


fonte: https://flylib.com/books/en/2.451.1.18/1/ e https://learnopengl.com/Lighting/Basic-Lighting

Reflexão Difusa

A reflexão difusa (Diffuse) espalha a luz de forma uniforme, assim não depende do ponto de vista, porém depende da sua relação com a normal da superfície.

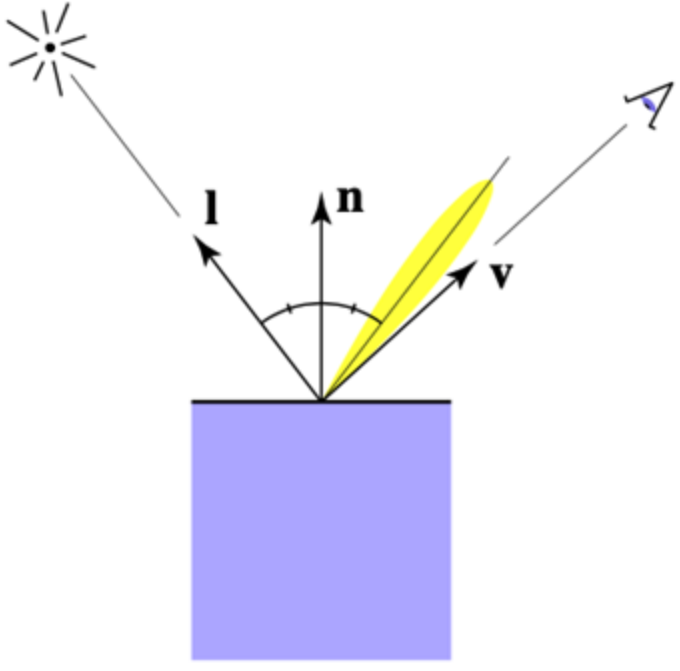


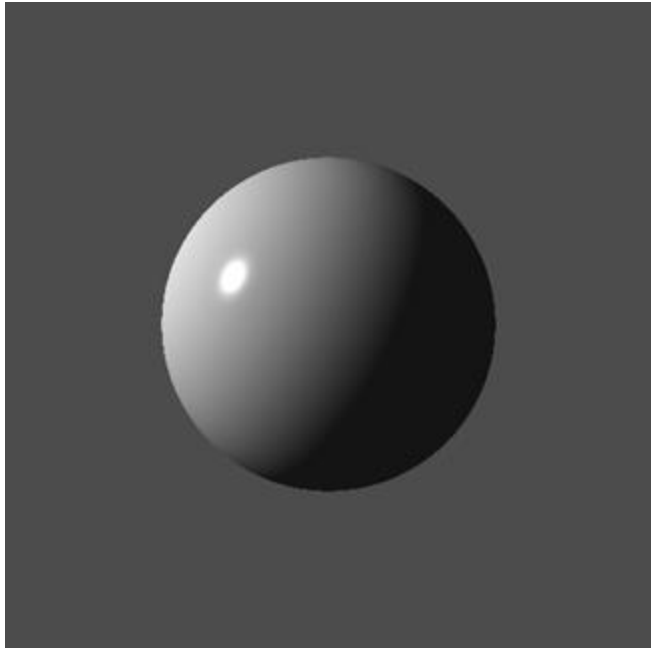


Reflexão Especular

A reflexão especular (Specular) possui uma reflexividade dependendo da origem da fonte de luz e do ponto de vista.

Nessa reflexão é possível ver pontos mais iluminados.

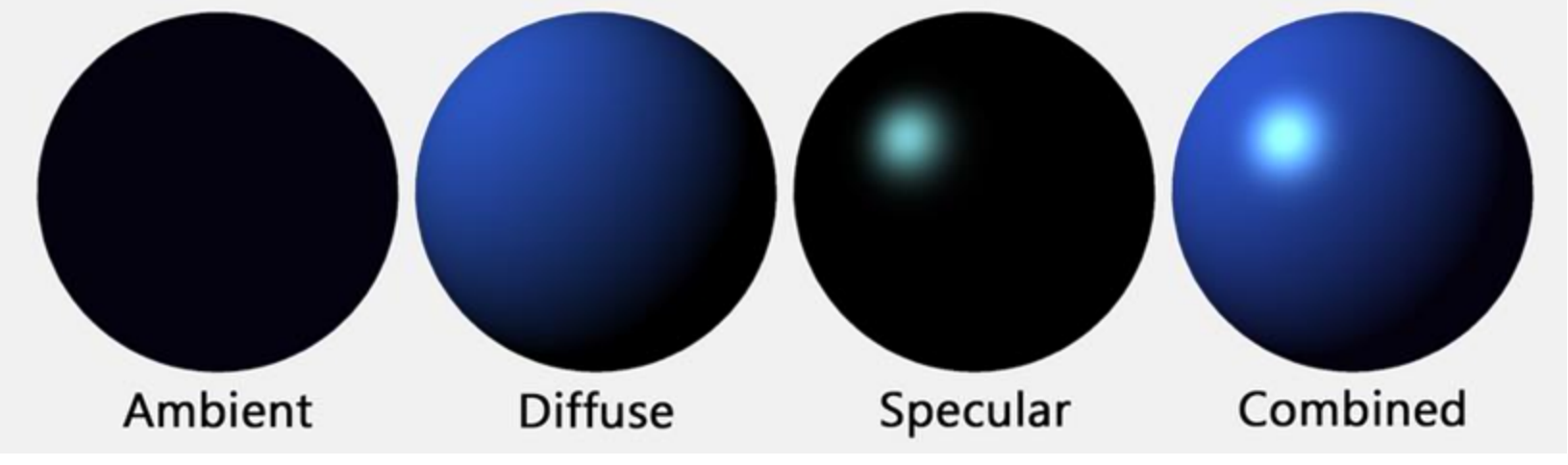




Insper

Resultado Final

$$\mathbf{I_{rgb}} = O_{Emissive\ rgb} + SUM(\ I_{Lrgb} \times (ambient_i + diffuse_i + specular_i))$$



Insper

Equação de Cores (padrão X3D simplificado)

$$\mathbf{I_{rgb}} = O_{E\,rgb} + SUM(\ I_{Lrgb} \times (ambient_i + diffuse_i + specular_i))$$

$$\mathbf{ambient_i} = I_{ia} \times O_{Drgb} \times O_a$$

$$\mathbf{diffuse_i} = I_i \times O_{Drgb} \times (\ \mathbf{N} \cdot \mathbf{L}\ )$$

$$\mathbf{specular_i} = I_i \times O_{Srgb} \times (\ \mathbf{N} \cdot ((\mathbf{L} + \mathbf{v}) / |\mathbf{L} + \mathbf{v}|))^{shininess \times 128}$$

$\mathbf{I_{Lrgb}}$ = light *color* $\mathbf{I_i}$ = light *intensity* $\mathbf{I_{ia}}$ = light *ambientIntensity*

$\mathbf{O_{Ergb}}$ = material *emissiveColor* $\mathbf{O_{Drgb}}$ = material diffuse colour $\mathbf{O_{Srgb}}$ = material *specularColor*

$\mathbf{O_a}$ = material *ambientIntensity*

$\mathbf{L}$ = direction of light source

$\mathbf{N}$ = normalized normal vector at this point on geometry

$\mathbf{v}$ = normalized vector from point on geometry to viewer's position

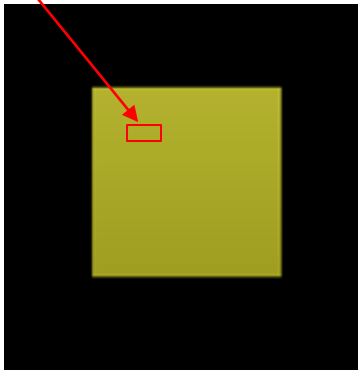
https://www.web3d.org/documents/specifications/19775-1/V3.3/Part01/components/lighting.html

Insper

Exemplo X3D

```
<Viewpoint position="0 0 10"/>
<NavigationInfo headlight='false'/>
<DirectionalLight direction="0 -0.8 -0.6" color="1 1 1" intensity="1" ambientIntensity="0"/>
<Transform>
  <Shape>
    <Box/>
    <Appearance>
      <Material specularColor="1.0 1.0 1.0" diffuseColor="1.0 1.0 0.0" shininess="0.2"
                ambientIntensity="0.2" emissiveColor="0 0 0"/>
    </Appearance>
  </Shape>
</Transform>
```

Qual a cor do pixel?



I_{Lrgb} = (1.0, 1.0, 1.0) I_i = 1.0 I_{ia} = 0.0

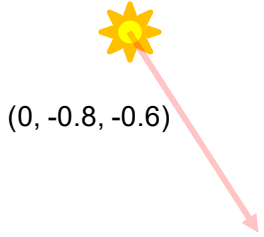
O_{Srgb} = (1.0, 1.0, 1.0) O_{Drgb} = (1.0, 1.0, 0.0) shininess = 0.2 O_a = 0.2 O_{Ergb} = (0.0, 0.0, 0.0)

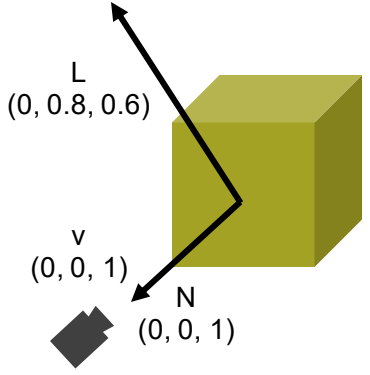
L = (0.0, 0.8, 0.6)

N = (0.0, 0.0, 1.0)

v = (0.0, 0.0, 1.0)* [Supondo no meio da tela]

*(essa é uma aproximação, mas podem usar no projeto se desejarem)



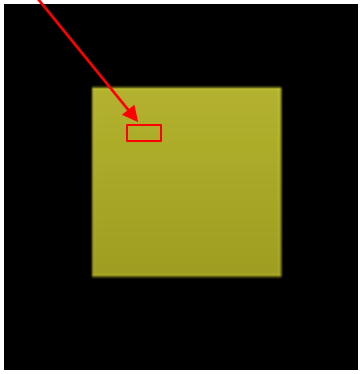


Insper

Exemplo X3D

```
<Viewpoint position="0 0 10"/>
<NavigationInfo headlight='false'/>
<DirectionalLight direction="0 -0.8 -0.6" color="1 1 1" intensity="1" ambientIntensity="0"/>
<Transform>
  <Shape>
    <Box/>
    <Appearance>
      <Material specularColor="1.0 1.0 1.0" diffuseColor="1.0 1.0 0.0" shininess="0.2"
                ambientIntensity="0.2" emissiveColor="0 0 0"/>
    </Appearance>
  </Shape>
</Transform>
```

Qual a cor do pixel?



I_{Lrgb} = (1.0, 1.0, 1.0) I_i = 1.0 I_{ia} = 0.0

O_{Srgb} = (1.0, 1.0, 1.0) O_{Drgb} = (1.0, 1.0, 0.0) shininess = 0.2 O_a = 0.2 O_{Ergb} = (0.0, 0.0, 0.0)

L = (0.0, 0.8, 0.6)

N = (0.0, 0.0, 1.0)

v = (0.0, 0.0, 1.0)* [Supondo no meio da tela]

*(essa é uma aproximação, mas podem usar no projeto se desejarem)

N · **L** = 0.6

(**L** + **v**) / |**L** + **v**| = (0.0, 0.8, 1.6)/1.79

(**L** + **v**) / |**L** + **v**| = (0.0, 0.45, 0.90)

N · ((**L** + **v**) / |**L** + **v**|) = 0.9

Insper

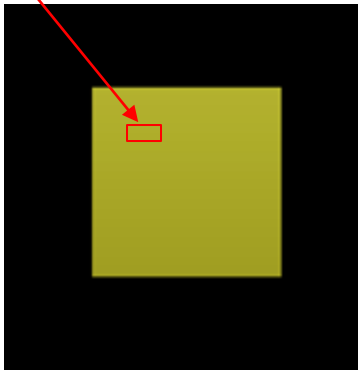
Exemplo X3D

```
<Viewpoint position="0 0 10"/>
<NavigationInfo headlight='false'/>

<DirectionalLight direction="0 -0.8 -0.6" color="1 1 1" intensity="1" ambientIntensity="0"/>

<Transform>
  <Shape>
    <Box/>
    <Appearance>
      <Material specularColor='1.0 1.0 1.0' diffuseColor='1.0 1.0 0.0' shininess='0.2'
                ambientIntensity="0.2" emissiveColor="0 0 0"/>
    </Appearance>
  </Shape>
</Transform>
```

Qual a cor do pixel?



$$\mathbf{ambient_i} = I_{ia} \times O_{Drgb} \times O_a = 0.0 \times (1.0, 1.0, 0.0) \times 0.2 = (0.0, 0.0, 0.0)$$

$$\mathbf{diffuse_i} = I_i \times O_{Drgb} \times (\mathbf{N} \cdot \mathbf{L}) = 1.0 \times (1.0, 1.0, 0.0) \times 0.6 = (0.6, 0.6, 0.0)$$

$$\mathbf{specular_i} = I_i \times O_{Srgb} \times (\mathbf{N} \cdot ((\mathbf{L} + \mathbf{v}) / |\mathbf{L} + \mathbf{v}|))^{\text{shininess} \times 128}$$

$$= 1.0 \times (1.0, 1.0, 1.0) \times 0.9^{25.6} = (0.07, 0.07, 0.07)$$

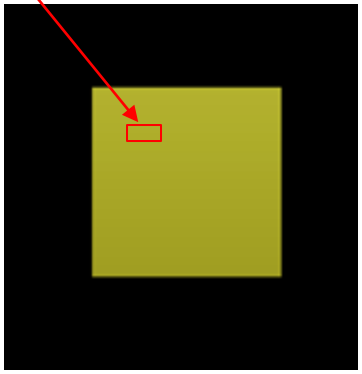
Insper

Exemplo X3D

```
<Viewpoint position="0 0 10"/>
<NavigationInfo headlight='false'/>

<DirectionalLight direction="0 -0.8 -0.6" color="1 1 1" intensity="1" ambientIntensity="0"/>

<Transform>
  <Shape>
    <Box/>
    <Appearance>
      <Material specularColor='1.0 1.0 1.0' diffuseColor='1.0 1.0 0.0' shininess='0.2'
                ambientIntensity="0.2" emissiveColor="0 0 0"/>
    </Appearance>
  </Shape>
</Transform>
```



$$\mathbf{I_{rgb}} = O_{E\,rgb} + SUM(I_{Lrgb} \times (ambient_i + diffuse_i + specular_i))$$

$$\mathbf{I_{rgb}} = (0.0, 0.0, 0.0) + SUM((1.0, 1.0, 1.0) \times ((0.0, 0.0, 0.0) + (0.6, 0.6, 0.0) + (0.07, 0.07, 0.07)))$$

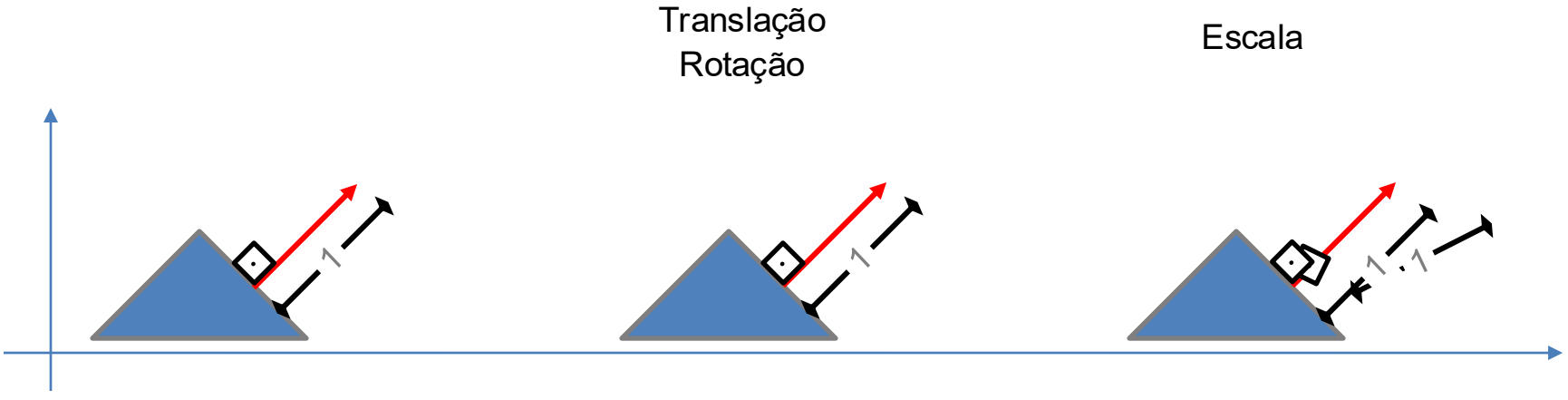
$$\mathbf{I_{rgb}} = (0.0, 0.0, 0.0) + SUM((1.0, 1.0, 1.0) \times (0.67, 0.67, 0.07))$$

$$\mathbf{I_{rgb}} = (0.0, 0.0, 0.0) + (0.67, 0.67, 0.07)$$

$$\mathbf{I_{rgb}} = (0.67, 0.67, 0.07)$$

Transformações nas Normais

Podemos usar a mesma transformação da geometria sobre suas normais?

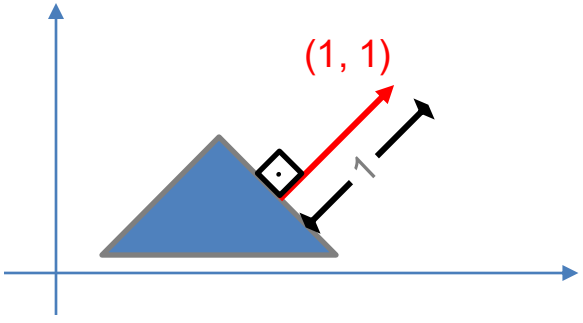


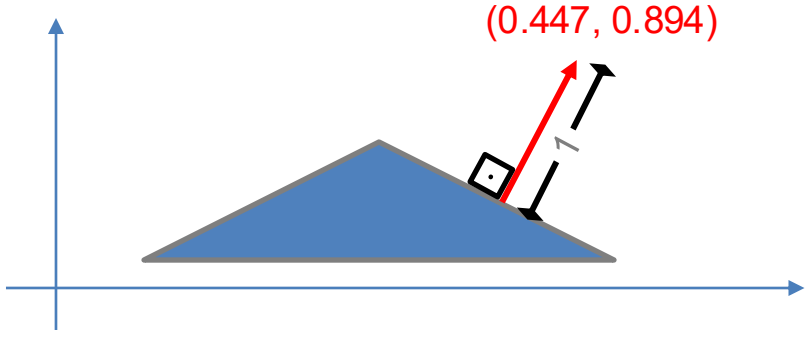
Solução: A transposta da inversa da matriz de transformação

$$M^{-1T}$$

Mais detalhes: https://www.scratchapixel.com/lessons/mathematics-physics-for-computer-graphics/geometry/transforming-normals.html

Escala nas Normais (Exemplo)





$$S = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$S_N = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^{-1T} = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$N_s = \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1 \\ 0 \end{bmatrix}$$

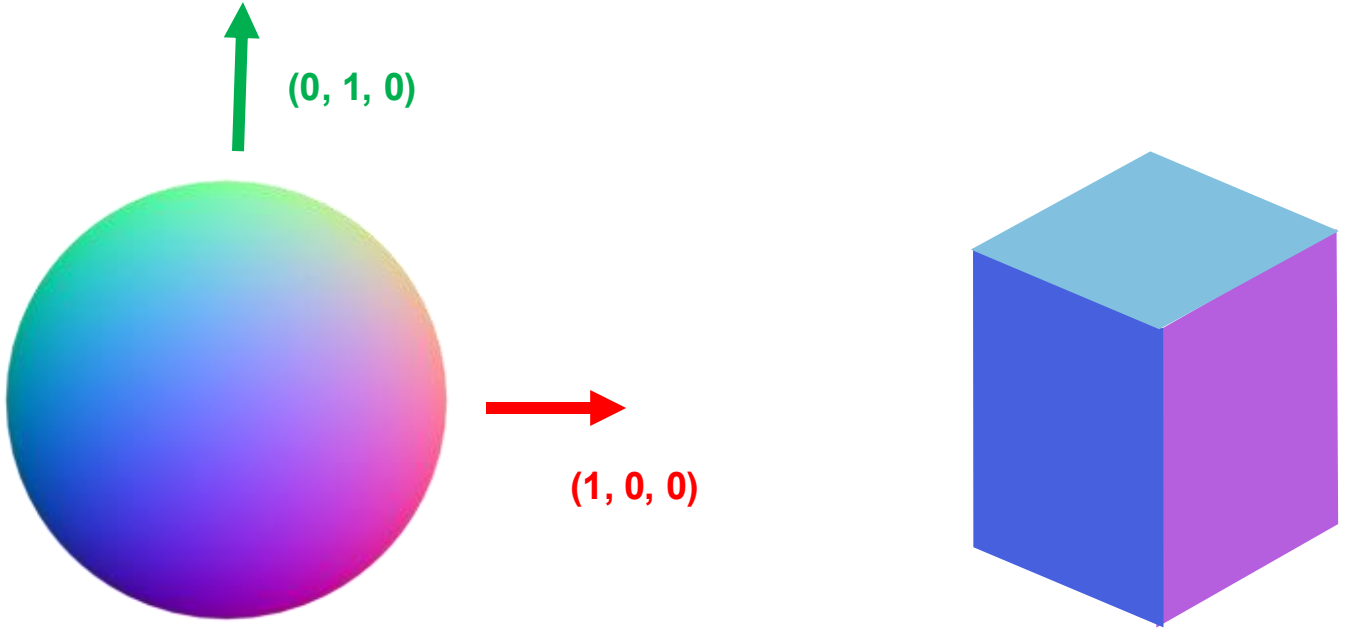
$$\widehat{N_S} \cong \begin{bmatrix} 0.5/1.118 \\ 1/1.118 \\ 0 \end{bmatrix} \cong \begin{bmatrix} 0.447 \\ 0.894 \\ 0 \end{bmatrix}$$

$$\|N_s\| = \sqrt{0.5^2 + 1^2} = \sqrt{1.25} \cong 1.118$$

Insper

Truque para visualizar as normais

Use o valor das cores pegando das normais



NavigationInfo – headlight (Web3D)

The headlight field specifies whether a browser shall turn on a headlight. A headlight is a **directional light** that always points in the direction the user is looking.

The headlight shall have:

- *intensity* = 1,
- *color* = (1 1 1),
- *ambientIntensity* = 0.0,
- *direction* = (0 0 −1).

https://www.web3d.org/documents/specifications/19775-1/V3.3/Part01/components/navigation.html#NavigationInfo

Insper

NavigationInfo – headlight (Web3D)

O campo **headlight** especifica se um navegador deve ou não ativar uma luz no observador. O headlight é uma luz direcional que sempre aponta na direção em que o usuário está olhando.

O headlight deve ter:

- *intensity* = 1,
- *color* = (1 1 1),
- *ambientIntensity* = 0.0,
- *direction* = (0 0 −1).

https://www.web3d.org/documents/specifications/19775-1/V3.3/Part01/components/navigation.html#NavigationInfo

Hermite spline interpolation (X3D simplificado)

($t_i \leq$ fraction $< t_{i+1}$), where t_i is the key at (i), and t_{i+1} is the key at (i+1)

$s = (t - t_i) / (t_{i+1} - t_i)$

The keyValue at key (i) is denoted as $\mathbf{v}_i$ and the keyValue at key (i+1) is denoted as $\mathbf{v}_{i+1}$.

$$\mathbf{v}_s = \mathbf{S}^T \mathbf{H} \mathbf{C}$$

$$\mathbf{S} = \begin{vmatrix} s^3 \\ s^2 \\ s \\ 1 \end{vmatrix} \quad \mathbf{H} = \begin{vmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix} \quad \mathbf{C} = \begin{vmatrix} \mathbf{v}_i \\ \mathbf{v}_{i+1} \\ \mathbf{T}^0_i \\ \mathbf{T}^1_{i+1} \end{vmatrix}$$

If the velocity vector is not specified, it is calculated as follows:

$$\mathbf{T}_i = (\mathbf{v}_{i+1} - \mathbf{v}_{i-1}) / 2$$

If the interpolator is not closed, and the first and last velocity vectors are not specified by the author:

$$\mathbf{T}^0_0 = \mathbf{T}^1_0 = \mathbf{T}^0_{N-1} = \mathbf{T}^1_{N-1} = 0$$

https://www.web3d.org/documents/specifications/19775-1/V3.3/Part01/components/interp.html

Insper

Hermite spline interpolation (X3D simplificado)

$(t_i \leq fraction < t_{i+1})$, onde t_i é a key em (i), e t_{i+1} é a key em (i+1)

$s = (t - t_i) / (t_{i+1} - t_i)$

O keyValue na key (i) i é denotado como $\mathbf{v}_i$ e o keyValue na key (i+1) é denotado como $\mathbf{v}_{i+1}$.

$$\mathbf{v}_s = \mathbf{S}^T \mathbf{H} \mathbf{C}$$

$$\mathbf{S} = \begin{vmatrix} s^3 \\ s^2 \\ s \\ 1 \end{vmatrix} \qquad \mathbf{H} = \begin{vmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{vmatrix} \qquad \mathbf{C} = \begin{vmatrix} \mathbf{v}_i \\ \mathbf{v}_{i+1} \\ \mathbf{T}^0_i \\ \mathbf{T}^1_{i+1} \end{vmatrix}$$

Se o vetor de velocidade não for especificado, ele é calculado da seguinte forma:

$$\mathbf{T}_i = (\mathbf{v}_{i+1} - \mathbf{v}_{i-1}) / 2$$

Tangentes de Hermite / Interpolação Catmull-Rom

Se o interpolador não for fechado, e os primeiros e últimos vetores de velocidade não forem especificados pelo autor:

$$\mathbf{T}^0_0 = \mathbf{T}^1_0 = \mathbf{T}^0_{N-1} = \mathbf{T}^1_{N-1} = 0$$

https://www.web3d.org/documents/specifications/19775-1/V3.3/Part01/components/interp.html

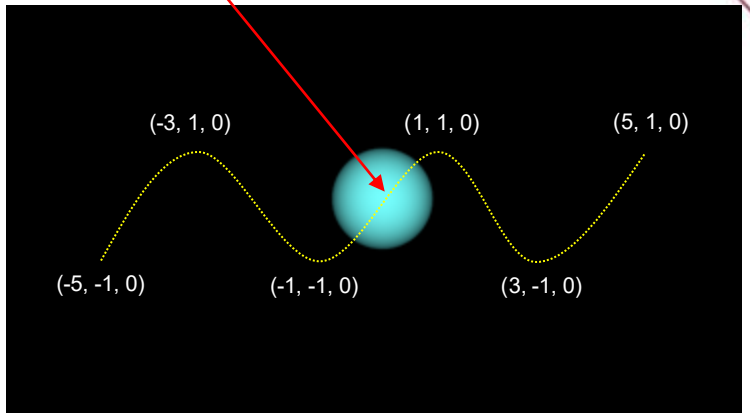
Exemplo X3D

Qual a posição da interpolação depois de 4 segundos?

```
<TimeSensor DEF='relogio' cycleInterval='8' loop='true'/>

<SplinePositionInterpolator DEF="move" closed="false"
      key="      0.00      0.2       0.4       0.6       0.8       1.00"
      keyValue="-5 -1 0  -3 1 0  -1 -1 0  1 1 0  3 -1 0  5 1 0"/>
                 t0        t1        t2        t3        t4        t5
<Transform DEF='esfera'>
  <Shape>
    <Sphere/>
    <Appearance>
      <Material diffuseColor='0.0 1.0 1.0'/>
    </Appearance>
  </Shape>
</Transform>

<ROUTE fromNode='relogio' fromField='fraction_changed' toNode='move' toField='set_fraction'/>
<ROUTE fromNode='move' fromField='value_changed' toNode='esfera' toField='translation'/>
```



t = *frac*(tempo / cycleInterval) = (4 *mod* 8) / 8 = 0.5

$$frac(x) = x - \lfloor x \rfloor$$

floor

(key anterior) $\mathbf{t}_i = \mathbf{t}_2$

(key posterior) $\mathbf{t}_{i+1} = \mathbf{t}_3$

Insper

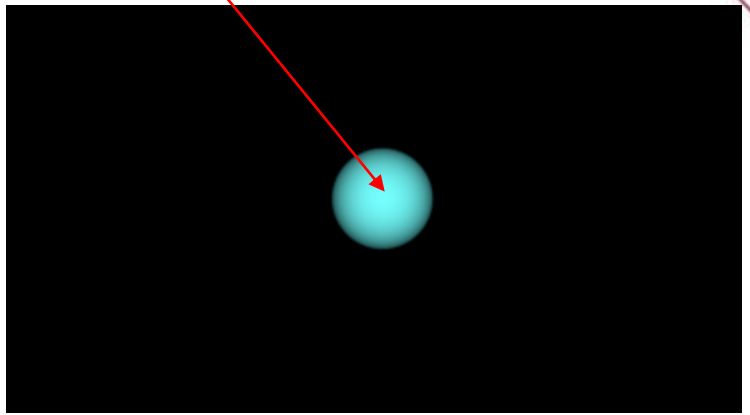
Exemplo X3D

Qual a posição da interpolação depois de 4 segundos?

```
<TimeSensor DEF='relogio' cycleInterval='8' loop='true'/>

<SplinePositionInterpolator DEF="move" closed="false"
      key="      0.00      0.2       0.4       0.6       0.8       1.00"
      keyValue="-5 -1 0  -3 1 0  -1 -1 0  1 1 0  3 -1 0  5 1 0"/>
                 t0        t1        t2        t3        t4        t5
<Transform DEF='esfera'>
  <Shape>
    <Sphere/>
    <Appearance>
      <Material diffuseColor='0.0 1.0 1.0'/>
    </Appearance>
  </Shape>
</Transform>

<ROUTE fromNode='relogio' fromField='fraction_changed' toNode='move' toField='set_fraction'/>
<ROUTE fromNode='move' fromField='value_changed' toNode='esfera' toField='translation'/>
```



$\mathbf{s} = (t - t_i) / (t_{i+1} - t_i) = (t - t_2) / (t_3 - t_2) = (0.5 - 0.4) / (0.6 - 0.4) = 0.1 / 0.2 = 0.5$

$\mathbf{T}_i = (\mathbf{v}_{i+1} - \mathbf{v}_{i-1}) / 2$:

$\mathbf{T}_2 = (\mathbf{v}_3 - \mathbf{v}_1) / 2 = ((1, 1, 0) - (-3, 1, 0)) / 2 = (4, 0, 0) / 2 = (2, 0, 0)$

$\mathbf{T}_3 = (\mathbf{v}_4 - \mathbf{v}_2) / 2 = ((3, -1, 0) - (-1, -1, 0)) / 2 = (4, 0, 0) / 2 = (2, 0, 0)$

Exemplo X3D

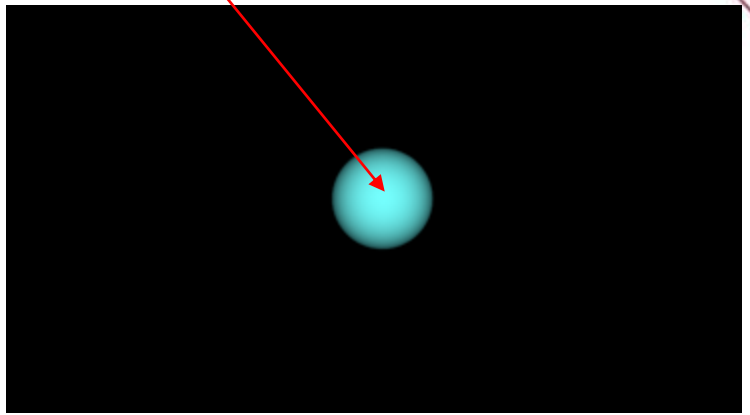
Qual a posição da interpolação depois de 4 segundos?

```
<TimeSensor DEF='relogio' cycleInterval='8' loop='true'/>

<SplinePositionInterpolator DEF="move" closed="false"
      key="     0.00     0.2     0.4      0.6    0.8     1.00"
      keyValue="-5 -1 0  -3 1 0  -1 -1 0  1 1 0  3 -1 0  5 1 0"/>

<Transform DEF='esfera'>
  <Shape>
    <Sphere/>
    <Appearance>
      <Material diffuseColor='0.0 1.0 1.0'/>
    </Appearance>
  </Shape>
</Transform>

<ROUTE fromNode='relogio' fromField='fraction_changed' toNode='move' toField='set_fraction'/>
<ROUTE fromNode='move' fromField='value_changed' toNode='esfera' toField='translation'/>
```



$$S = \begin{bmatrix} 0.125 \\ 0.25 \\ 0.5 \\ 1 \end{bmatrix}^T \quad H = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad C = \begin{bmatrix} -1 & -1 & 0 \\ 1 & 1 & 0 \\ 2 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix}$$

$$\mathbf{v}_s = \mathbf{S}^T \mathbf{H} \mathbf{C}$$

Insper

Exemplo X3D

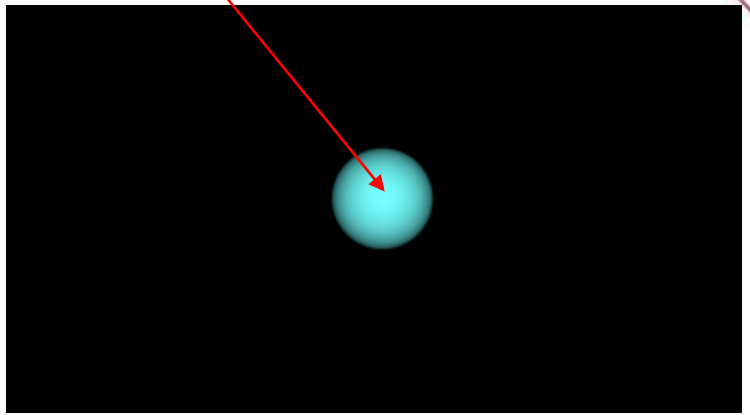
Qual a posição da interpolação depois de 4 segundos?

```
<TimeSensor DEF='relogio' cycleInterval='8' loop='true'/>

<SplinePositionInterpolator DEF="move" closed="false"
      key="     0.00     0.2     0.4      0.6     0.8     1.00"
      keyValue="-5 -1 0  -3 1 0  -1 -1 0  1 1 0  3 -1 0  5 1 0"/>

<Transform DEF='esfera'>
  <Shape>
    <Sphere/>
    <Appearance>
      <Material diffuseColor='0.0 1.0 1.0'/>
    </Appearance>
  </Shape>
</Transform>

<ROUTE fromNode='relogio' fromField='fraction_changed' toNode='move' toField='set_fraction'/>
<ROUTE fromNode='move' fromField='value_changed' toNode='esfera' toField='translation'/>
```



$$V_s = [0.125 \quad 0.25 \quad 0.5 \quad 1] \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} -1 & -1 & 0 \\ 1 & 1 & 0 \\ 2 & 0 & 0 \\ 2 & 0 & 0 \end{bmatrix} = [0 \quad 0 \quad 0]$$

Insper

Computação Gráfica

Luciano Soares
<lpsoares@insper.edu.br>

Fabio Orfali
<fabioO1@insper.edu.br>