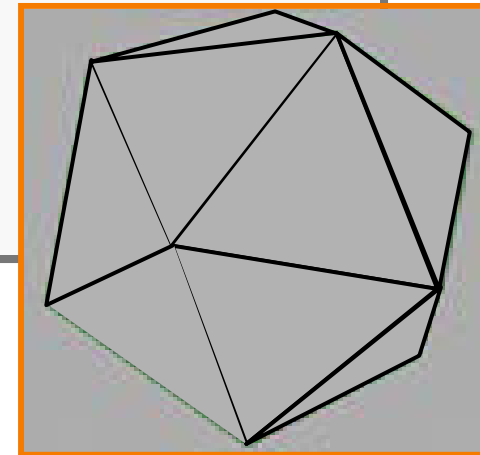


# Ray Casting

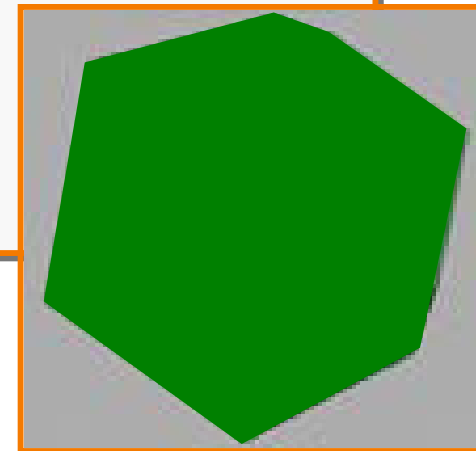
```
Image RayCast(Scene scene, int width, int height)
{
    Image image = new Image(width, height);
    for (int i = 0; i < width; i++) {
        for (int j = 0; j < height; j++) {
            Ray ray = ConstructRayThroughPixel(scene.camera, i, j);
            Intersection hit = FindIntersection(ray, scene);
            image[i][j] = GetColor(scene, ray, hit);
        }
    }
    return image;
}
```



Wireframe

# Ray Casting

```
Image RayCast(Scene scene, int width, int height)
{
    Image image = new Image(width, height);
    for (int i = 0; i < width; i++) {
        for (int j = 0; j < height; j++) {
            Ray ray = ConstructRayThroughPixel(scene.camera, i, j);
            Intersection hit = FindIntersection(ray, scene);
            image[i][j] = GetColor(scene, ray, hit);
        }
    }
    return image;
}
```



Sem Iluminação

# Ray Casting

```
Image RayCast(Scene scene, int width, int height)
{
    Image image = new Image(width, height);
    for (int i = 0; i < width; i++) {
        for (int j = 0; j < height; j++) {
            Ray ray = ConstructRayThroughPixel(scene.camera, i, j);
            Intersection hit = FindIntersection(ray, scene);
            image[i][j] = GetColor(scene, ray, hit);
        }
    }
    return image;
}
```

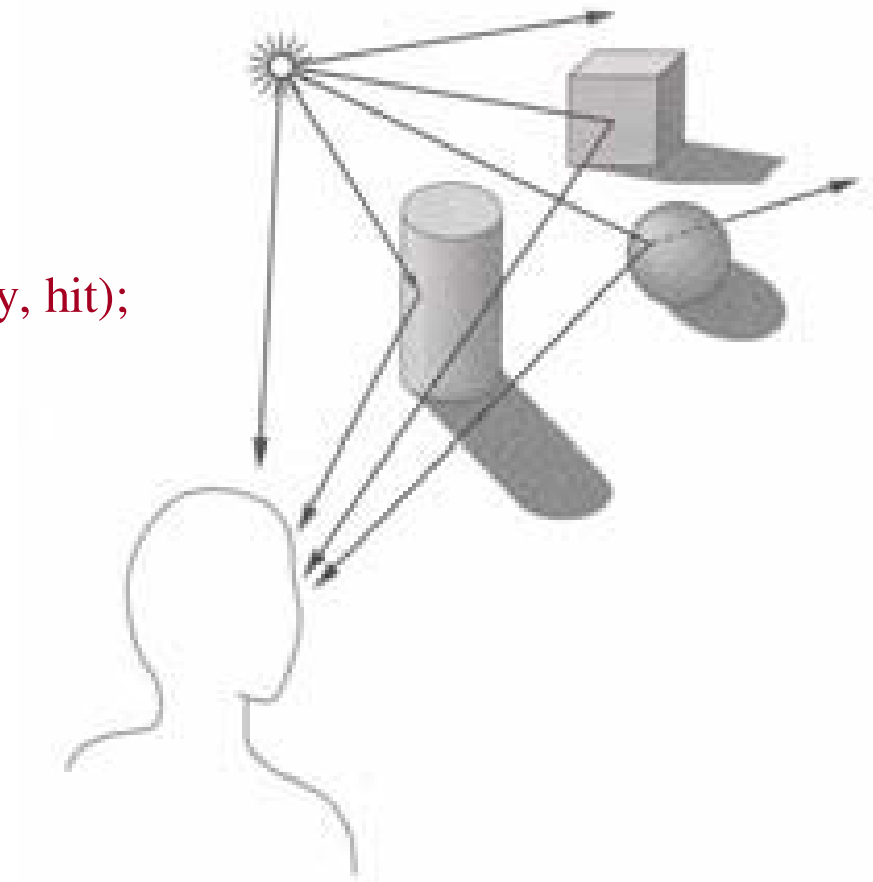


Com Iluminação

# Iluminação

- Como calcular a radiância para um raio ?

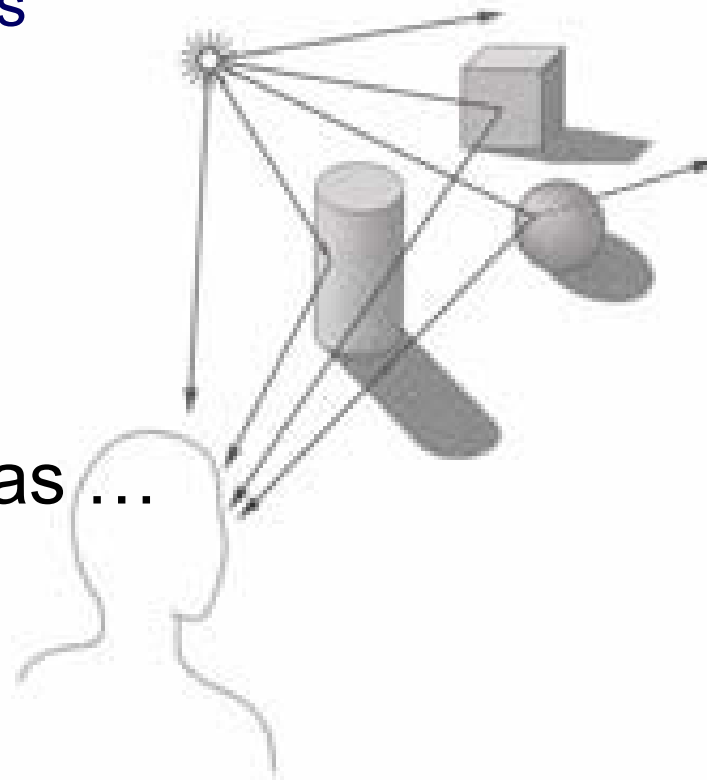
`image[i][j] = GetColor(scene, ray, hit);`



Angel Figure 6.2

# Objetivos

- Deve-se construir modelos computacionais para ...
  - Emissão nas fontes luminosas
  - Dispersão nas superfícies
  - Recepção na câmera
- Características desejadas ...
  - Concisão
  - Eficiência nos cálculos
  - “Precisão”



# Overview

---

- Iluminação Direta
  - Emissão nas fontes luminosas
  - Dispersão nas superfícies
- Iluminação Global
  - Sombras
  - Refrações
  - Reflexão entre objetos



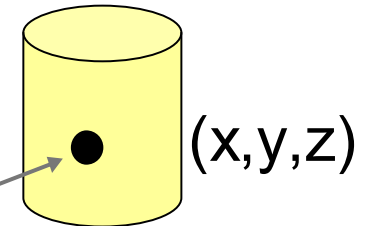
Iluminação Direta

# Emissão nas Fontes Luminosas

- $I_L(x, y, z, \theta, \phi, \lambda) \dots$ 
  - descreve a intensidade da energia,
  - saindo da fonte luminosa, ...
  - chegando na posição  $(x, y, z)$ , ...
  - da direção  $(\theta, \phi)$ , ...
  - com comprimento de onda  $\lambda$

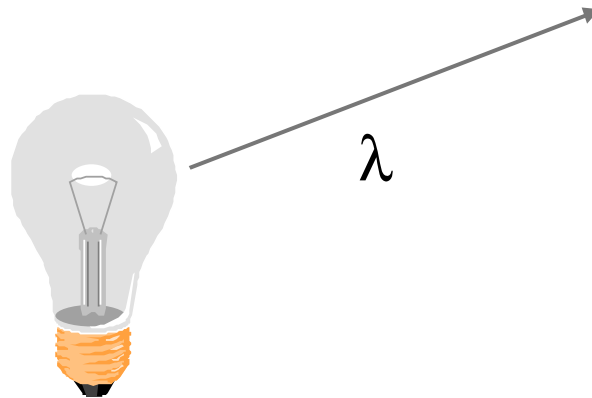


Light



# Modelos Empíricos

- Idealmente, mensurar energia em “todas” as situações
  - Ocupa muito espaço de armazenamento
  - Difícil de colocar em prática

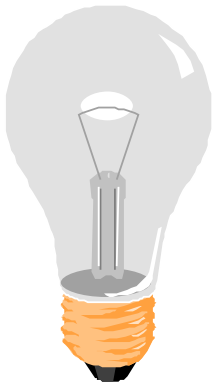




# Modelos de Fontes Luminosas - OpenGL

---

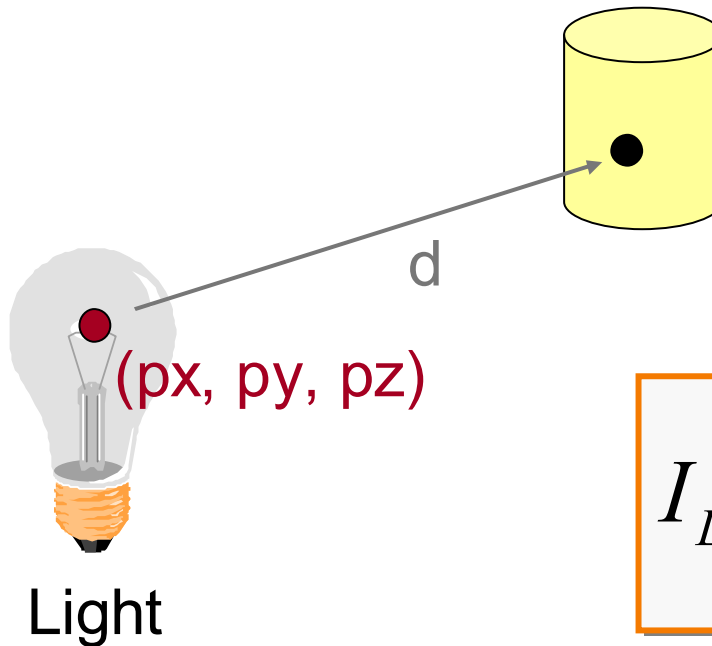
- Modelos matematicamente simples:
  - Luz Pontual (Point light)
  - Luz Spot (Spot light)
  - Luz Direcional (Directional light)



# Luz Pontual



- Modela fontes pontuais omni-direcionais
  - intensidade ( $I_0$ ),
  - posição ( $p_x, p_y, p_z$ ),
  - coeficientes ( $c_a, l_a, q_a$ ) para atenuação com distância ( $d$ )

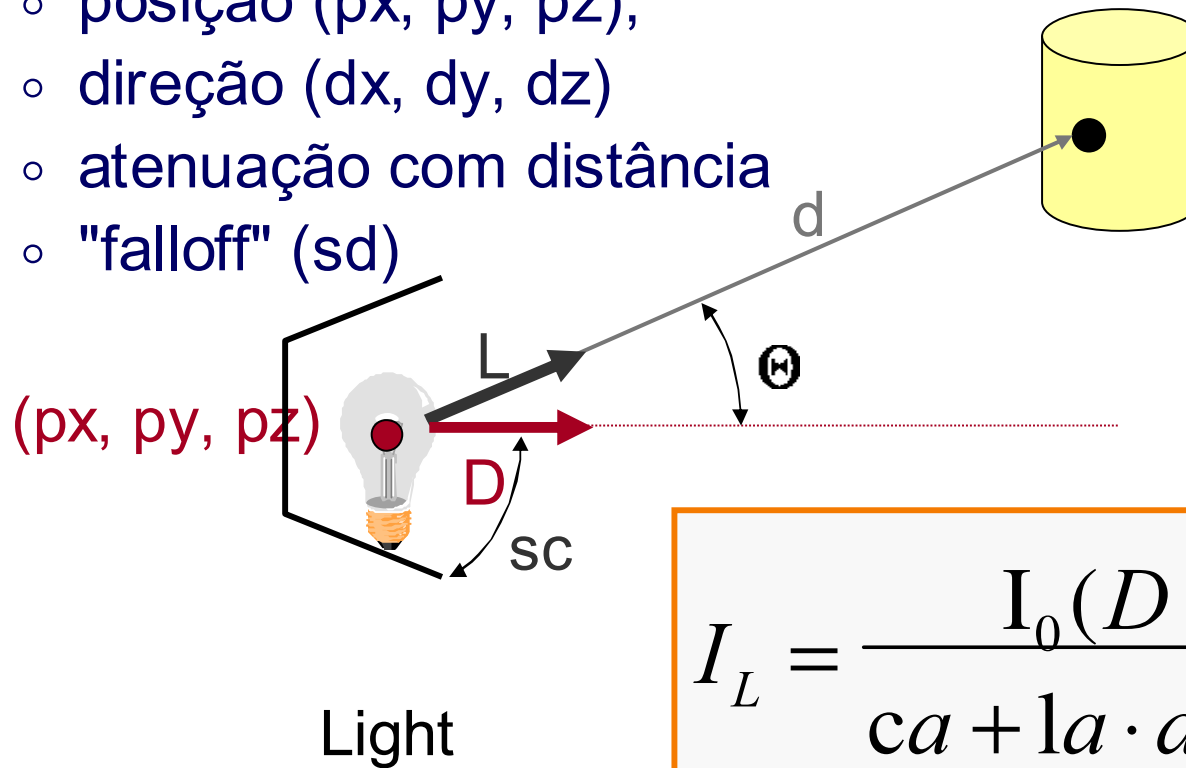


$$I_L = \frac{I_0}{c_a + l_a \cdot d + q_a \cdot d^2}$$

# Luz Spot



- Modela fontes pontuais com direção
  - intensidade ( $I_0$ ),
  - posição ( $p_x, p_y, p_z$ ),
  - direção ( $d_x, d_y, d_z$ )
  - atenuação com distância
  - "falloff" ( $sd$ )



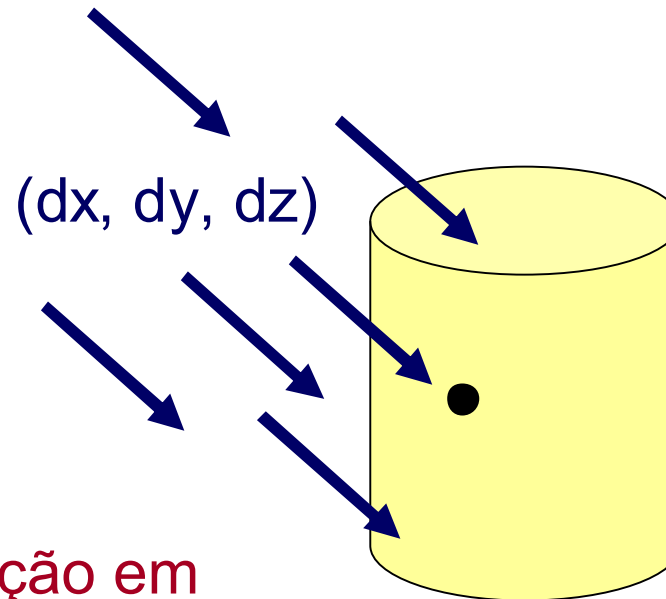
$$I_L = \frac{I_0 (D \cdot L)^{sd}}{ca + la \cdot d + qa \cdot d^2}$$

if  $(\Theta > sc) I_L = 0$

# Luz Direcional



- Modela luz pontual no infinito
  - intensidade ( $I_0$ ),
  - direção (dx,dy,dz)

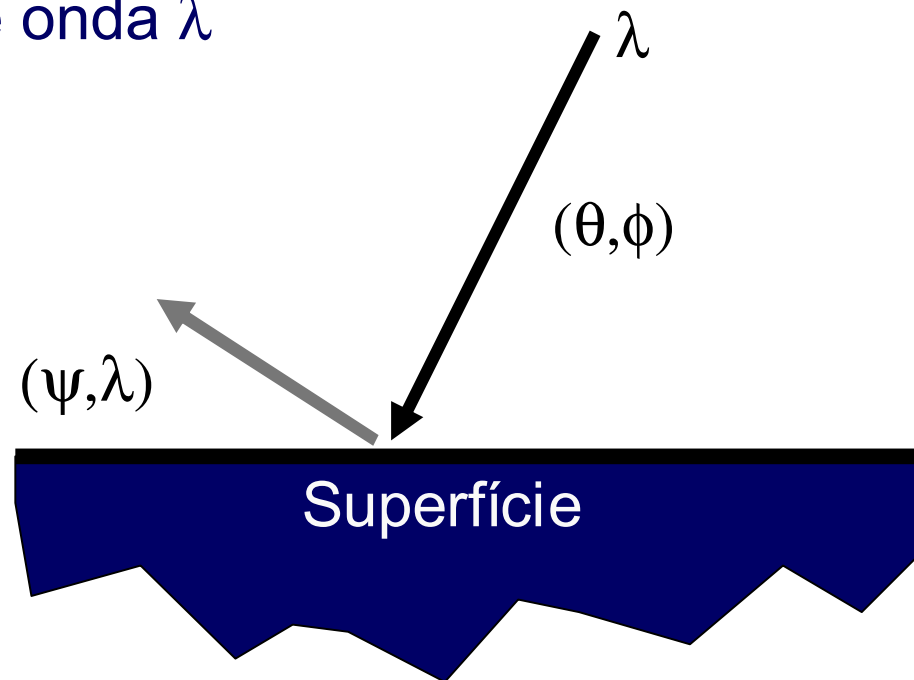


Sem atenuação em  
função da distância

$$I_L = I_0$$

# Dispersão nas Superfícies

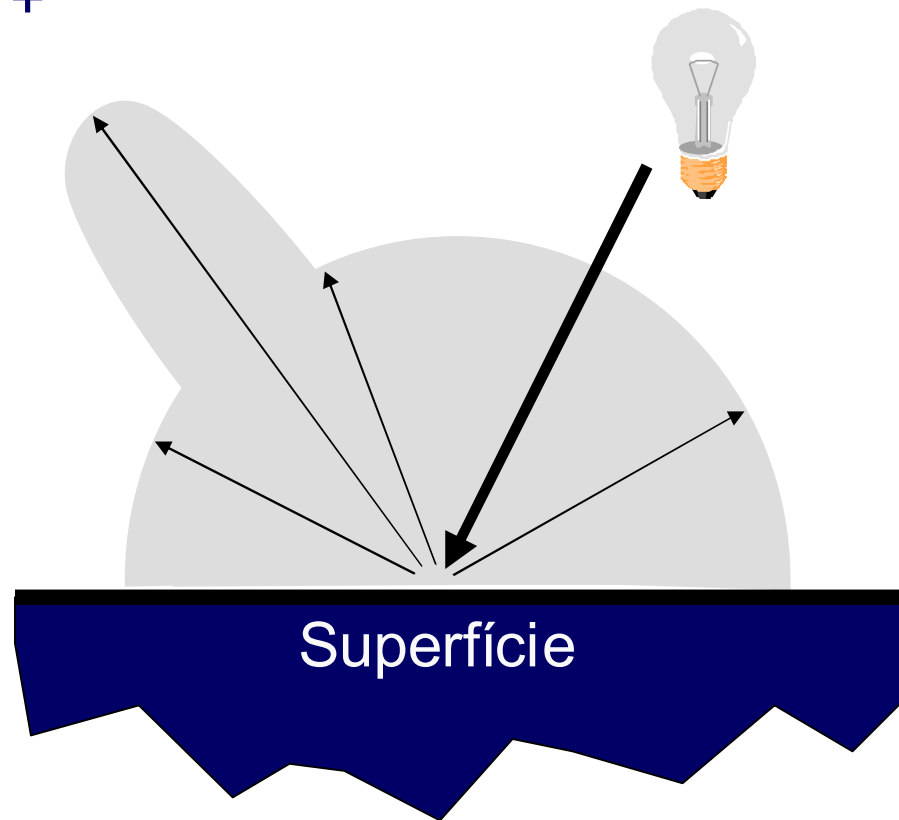
- $R_s(\theta, \phi, \gamma, \psi, \lambda)$  ...
  - descreve a quantidade de energia incidente,
  - chegando de uma direção  $(\theta, \phi)$ , ...
  - partindo na direção  $(\gamma, \psi)$ , ...
  - com comprimento de onda  $\lambda$



# Modelo de Iluminação - OpenGL

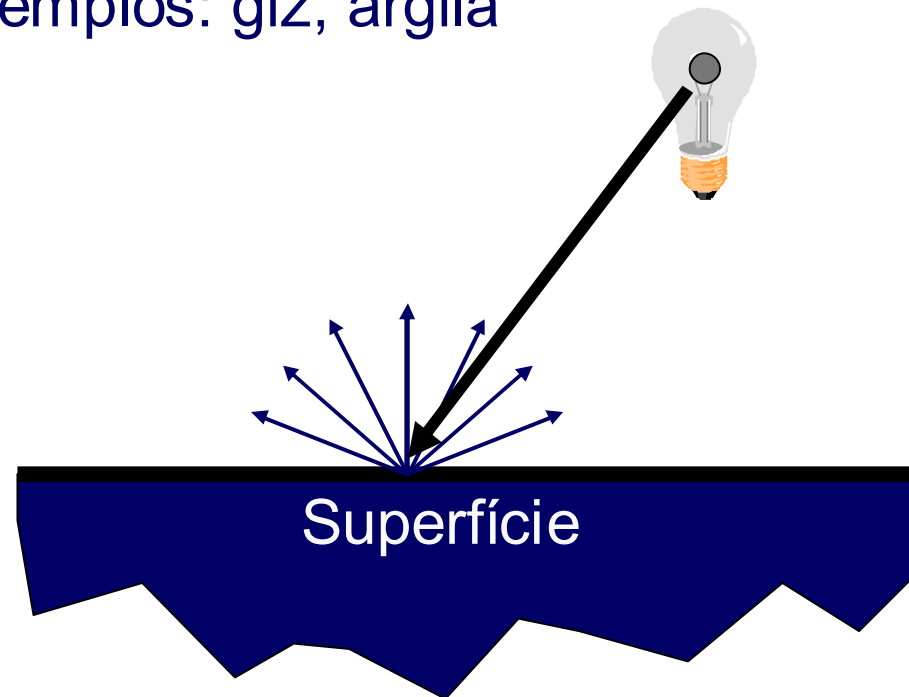
- Modelo analítico simples:
  - reflexão difusa +
  - reflexão especular +
  - emissão +
  - “ambiente”

Baseado na  
proposta de Phong



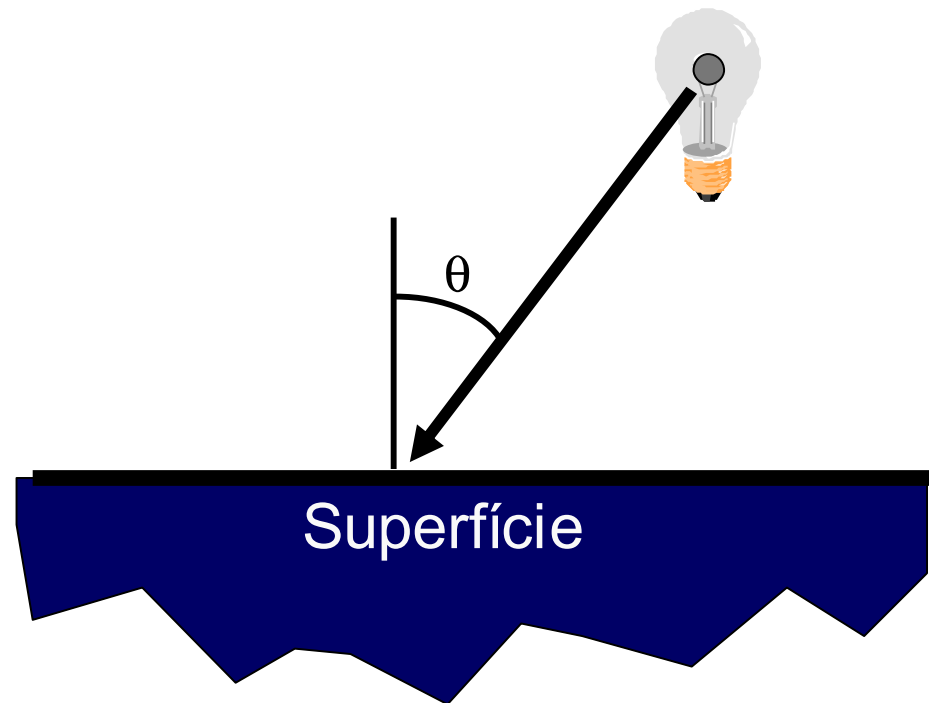
# Reflexão Difusa

- Suposição: a superfície reflete igualmente em todas as direções
  - Exemplos: giz, argila



# Reflexão Difusa

- Qual a quantidade de luz que é refletida ?
  - Depende do ângulo de incidência

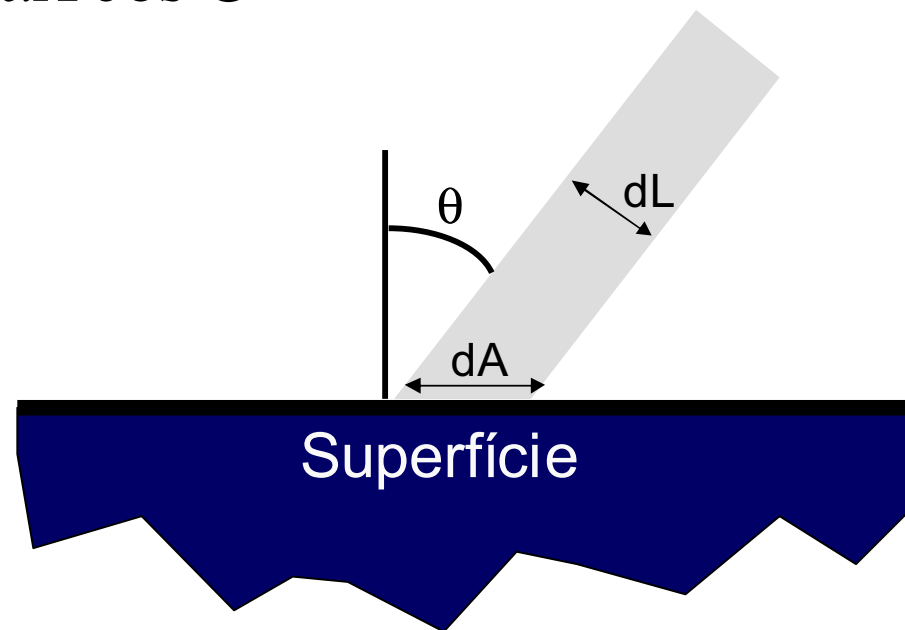




# Reflexão Difusa

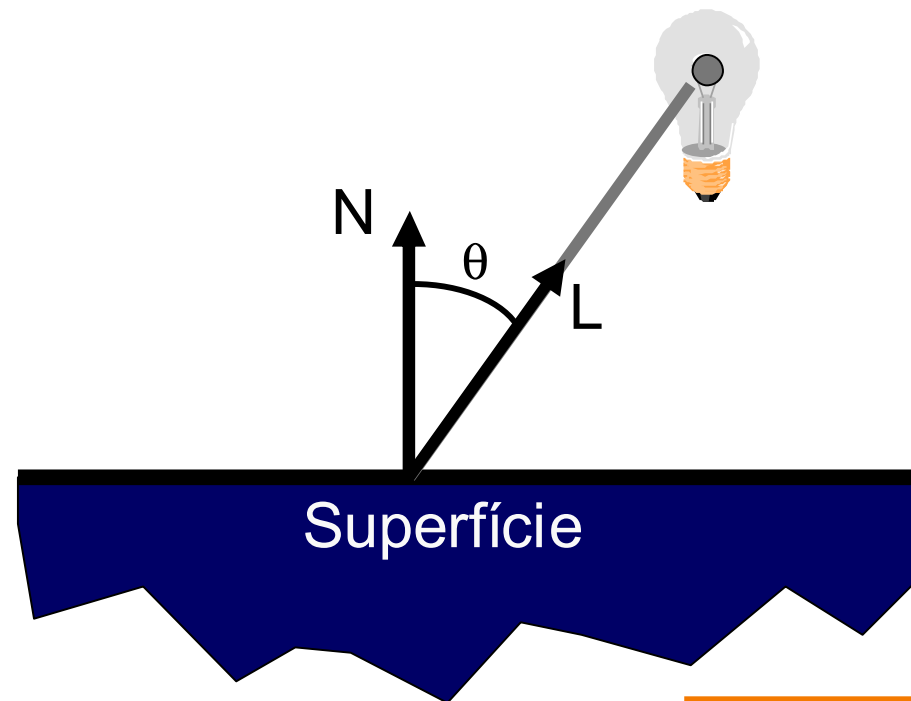
- Qual a quantidade de luz que é refletida ?
  - Depende do ângulo de incidência

$$dL = dA \cos \Theta$$



# Reflexão Difusa

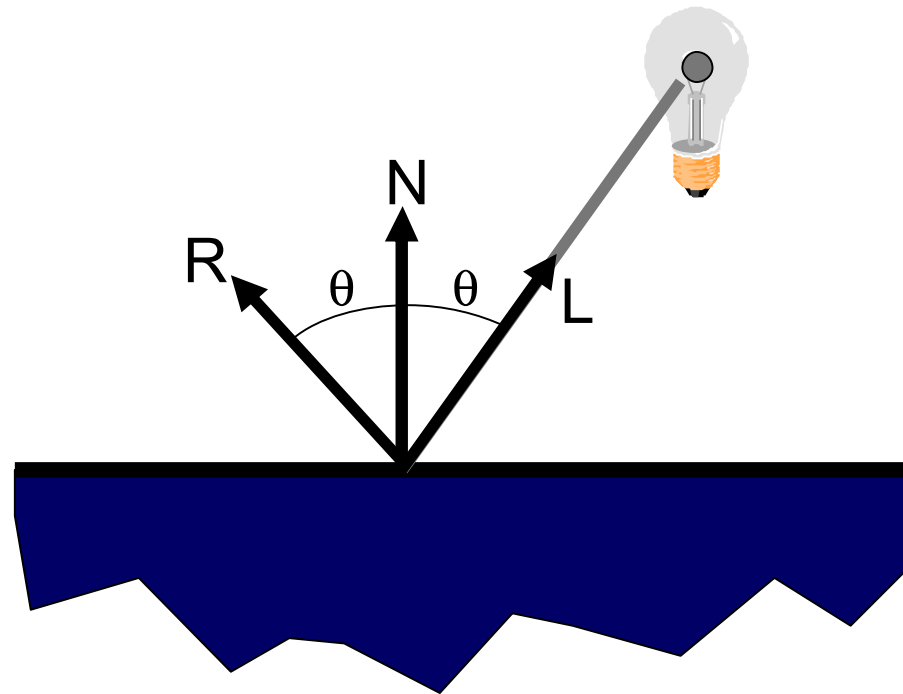
- Modelo Lambertiano
  - Lei do cosseno (produto interno)



$$I_D = K_D (N \bullet L) I_L$$

# Reflexão Especular

- Reflexão é mais intensa próxima do ângulo "espelho"
  - Exemplos: espelhos, metais

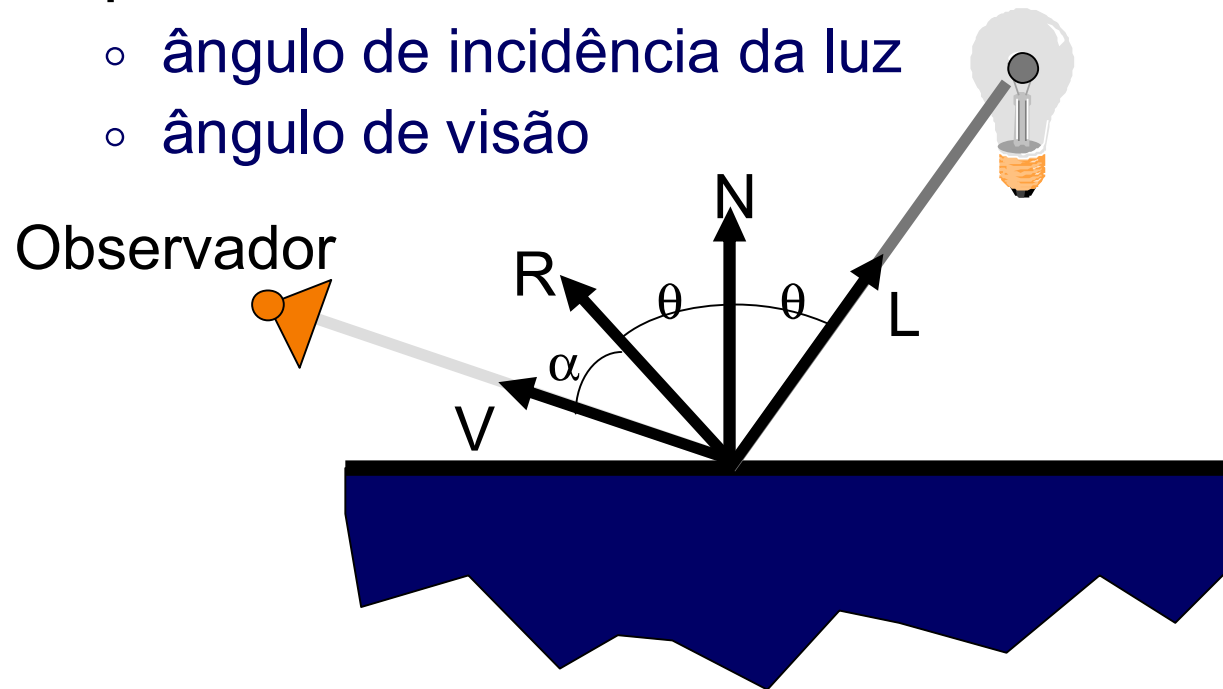


# Reflexão Especular

Qual a quantidade de luz que é refletida ?

Depende do:

- ângulo de incidência da luz
- ângulo de visão

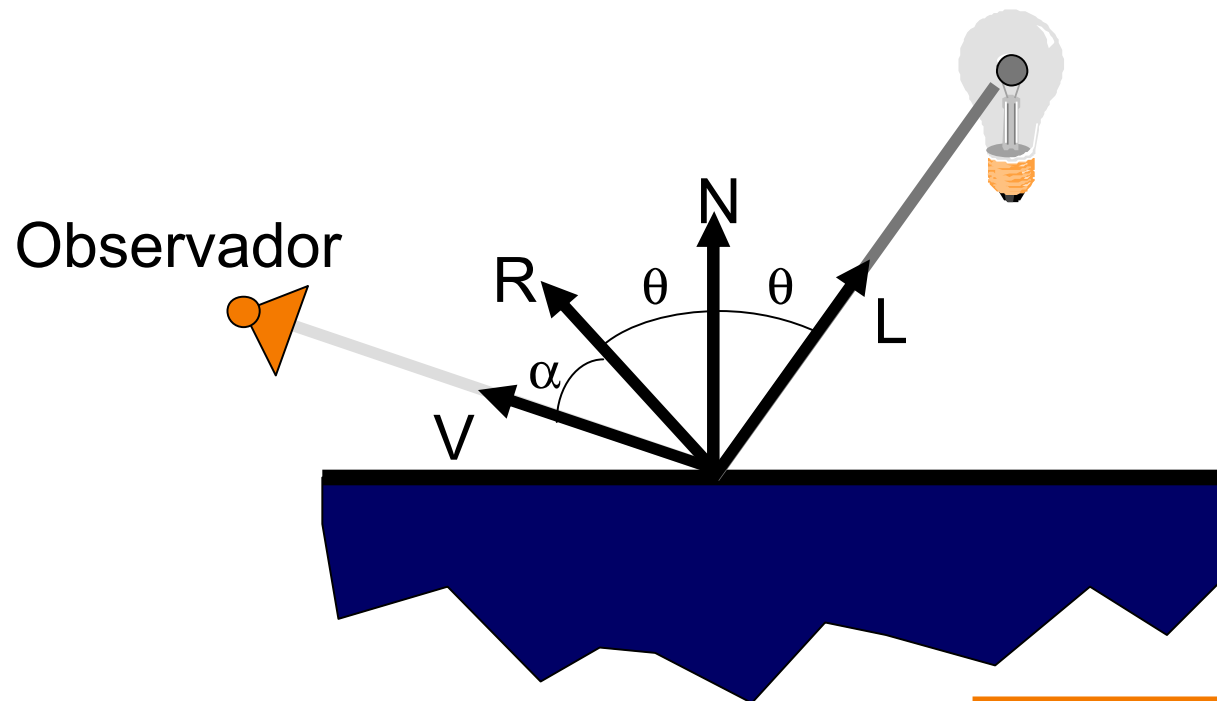


# Reflexão Especular

- Modelo de Phong

- $\cos(\alpha)^n$

Este é um "artifício" em bases físicas



$$I_S = K_S (V \bullet R)^n I_L$$

# Emissão

- Representa a luz emanada diretamente de um polígono

Emissão  $\neq 0$



# Ambiente

- Representa reflexão de toda iluminação indireta

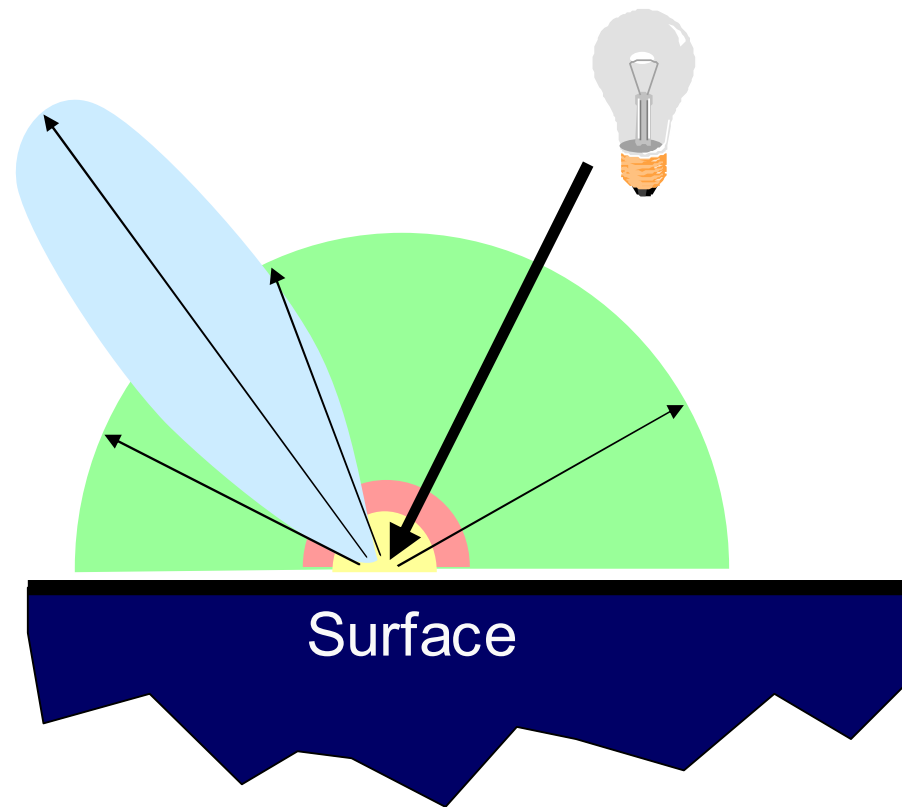


Este é um "macete" (evita a complexidade da iluminação global)!

# Modelo de Iluminação - OpenGL

- Modelo analítico simples:

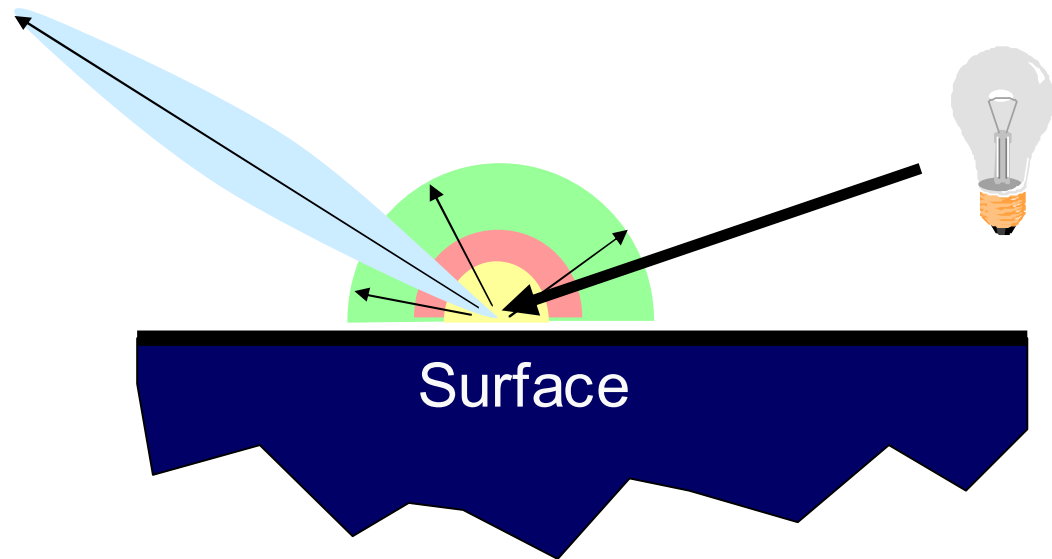
- reflexão difusa +
- reflexão especular +
- emissão +
- “ambiente”












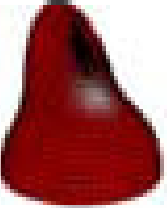



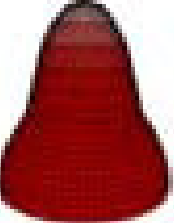
# Modelo de Iluminação - OpenGL

- Modelo analítico simples:
  - reflexão difusa +
  - reflexão especular +
  - emissão +
  - “ambiente”



# Modelo de Iluminação - OpenGL

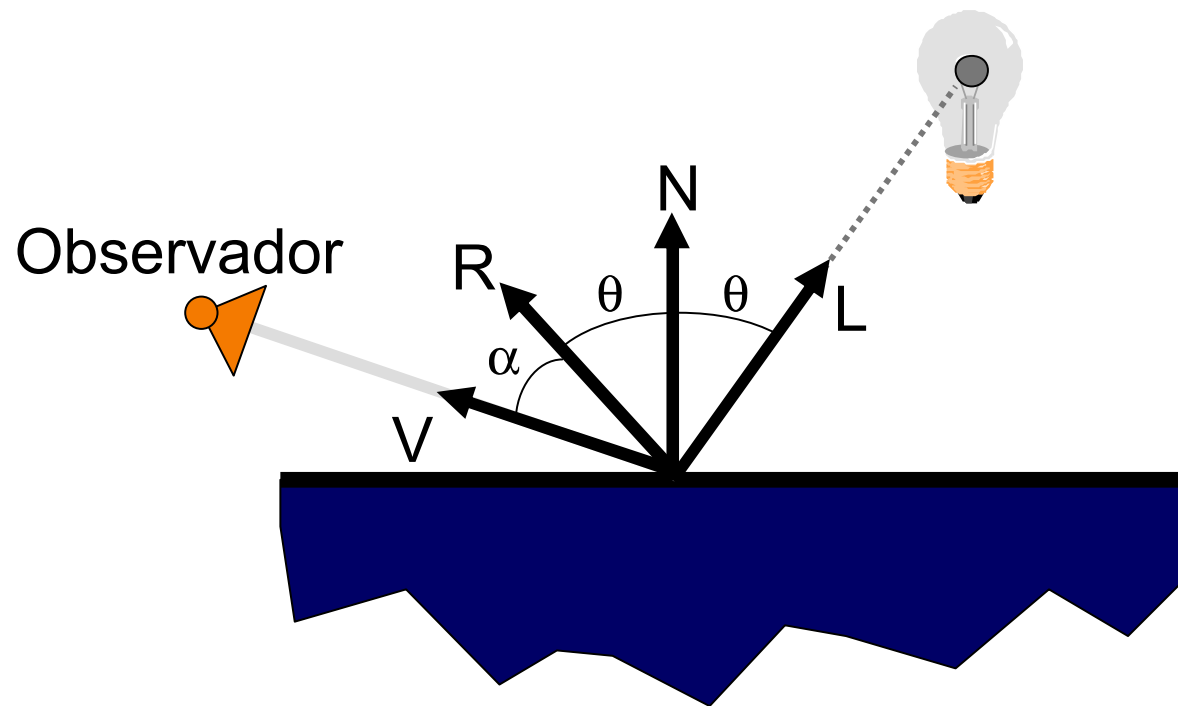
- Soma reflexão difusa, especular, emissão e ambiente

Phong	$R_{\text{ambient}}$	$R_{\text{diffuse}}$	$R_{\text{specular}}$	$R_{\text{total}}$
$\phi_i = 60^\circ$				
$\phi_i = 25^\circ$				
$\phi_i = 0^\circ$				

Leonard McMillan, MIT

# Cálculo da Iluminação Direta

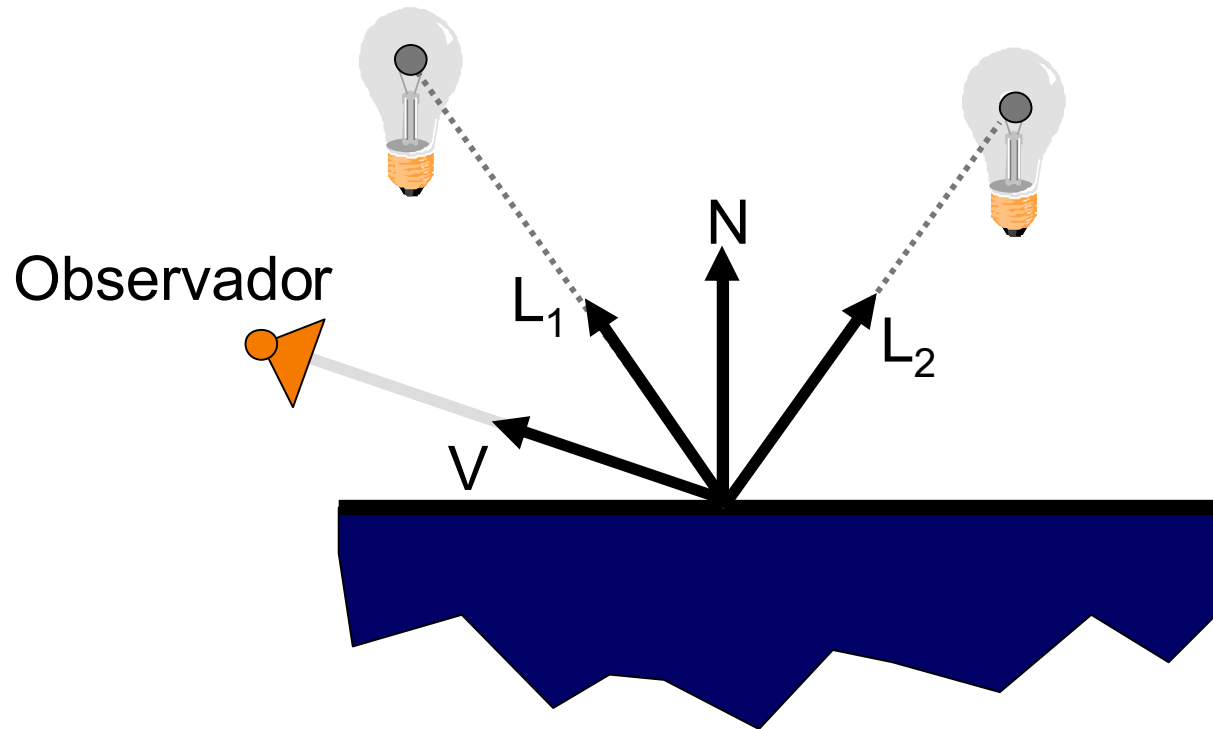
- Fonte de luz única:



$$I = I_E + K_A I_{AL} + K_D (N \cdot L) I_L + K_S (V \cdot R)^n I_L$$

# Cálculo da Iluminação Direta

- Múltiplas fontes luminosas:



$$I = I_E + K_A I_{AL} + \sum_i (K_D (N \cdot L_i) I_i + K_S (V \cdot R_i)^n I_i)$$