

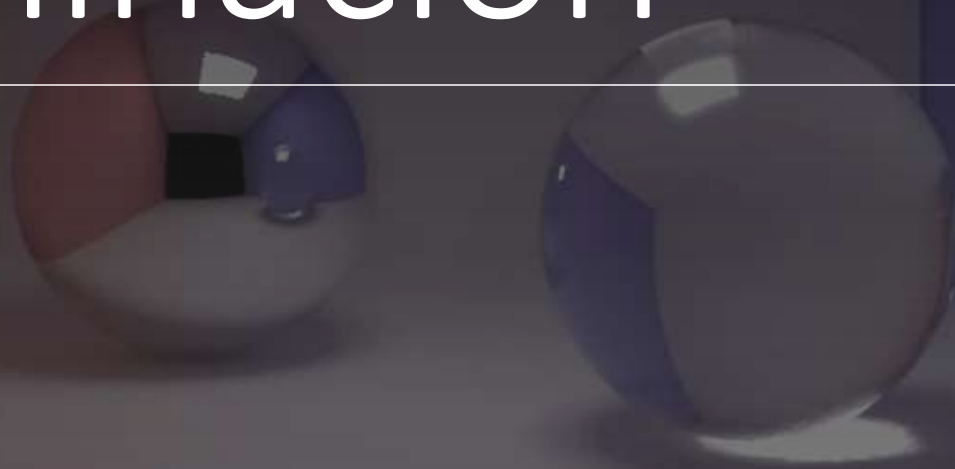
# Computación Gráfica

---

Ing. Gabriel Ávila, MSc.

# Sombreado e iluminación

---



# Sombreado

---

PROCESO DE DETERMINAR EL COLOR DE UN PIXEL, USUALMENTE BASADO EN LA ILUMINACIÓN DE LA ESCENA.

# Sombreado básico

---

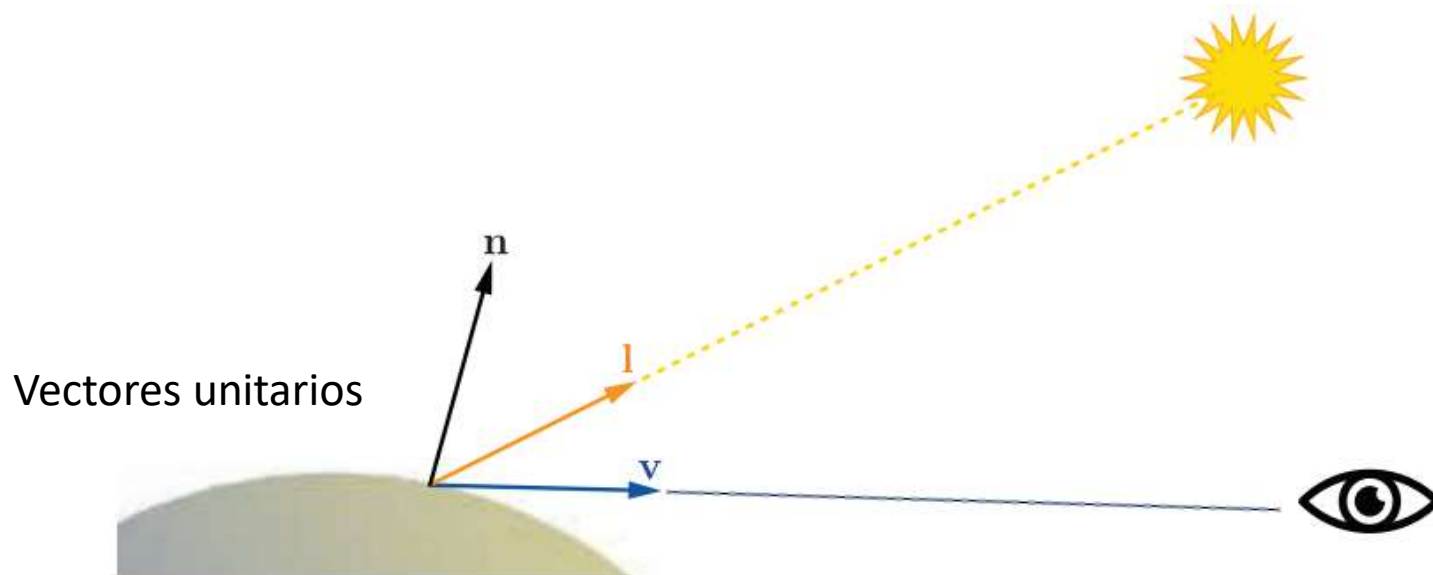
Es necesario definir modelos de sombreado, según los requerimientos de la aplicación. Estos pueden variar, desde efectos fotorrealistas hasta apariencias estilizadas.



Tomado de [1]

# Modelos de sombreado

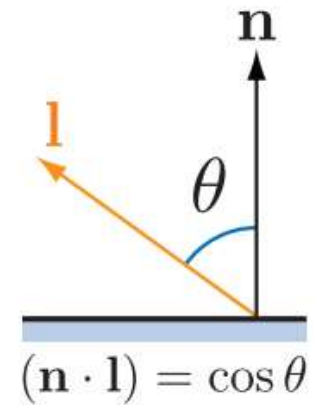
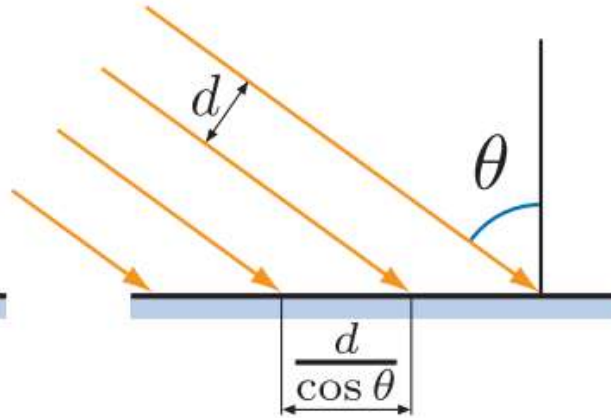
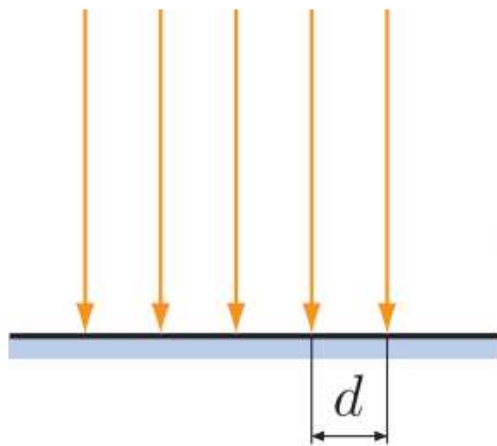
Permiten definir cómo cambia el color de un objeto, según factores como orientación o normal de la superficie ( $\mathbf{n}$ ), dirección de visualización ( $\mathbf{v}$ ) y la dirección de la iluminación ( $\mathbf{l}$ ).



Tomado de [1]

# Modelos de sombreado

---



Tomado de [1]

# Modelo general de sombreado


---

Para una única fuente luminosa:

$$\mathbf{c}_{\text{shaded}} = f_{\text{unlit}}(\mathbf{n}, \mathbf{v}) + \mathbf{c}_{\text{light}} f_{\text{lit}}(\mathbf{l}, \mathbf{n}, \mathbf{v})$$

Modelo general:

$$\mathbf{c}_{\text{shaded}} = f_{\text{unlit}}(\mathbf{n}, \mathbf{v}) + \sum_{i=1}^n \mathbf{c}_{\text{light}_i} f_{\text{lit}}(\mathbf{l}_i, \mathbf{n}, \mathbf{v})$$


$$f_{\text{unlit}}() = (0, 0, 0)$$

# Modelo general de sombreado

---

Modelo general, teniendo en cuenta la dirección de la luz y la normal de la superficie:

$$\mathbf{c}_{\text{shaded}} = f_{\text{unlit}}(\mathbf{n}, \mathbf{v}) + \sum_{i=1}^n (\mathbf{l}_i \cdot \mathbf{n})^+ \mathbf{c}_{\text{light}_i} f_{\text{lit}}(\mathbf{l}_i, \mathbf{n}, \mathbf{v})$$



# Un ejemplo:

## *Modelo Lambertiano*

---

Definido por Johann Heinrich Lambert en 1760. Simplifica el modelo de sombreado, haciendo que el color de la superficie iluminada sea el color de la superficie:

$$f_{\text{lit}}() = \mathbf{c}_{\text{surface}}$$

Simplificando el modelo a:

$$\mathbf{c}_{\text{shaded}} = f_{\text{unlit}}(\mathbf{n}, \mathbf{v}) + \sum_{i=1}^n (\mathbf{l}_i \cdot \mathbf{n})^+ \mathbf{c}_{\text{light}_i} \mathbf{c}_{\text{surface}}$$

# Un ejemplo: *Sombreado Gooch*

---

$$\mathbf{c}_{\text{shaded}} = s \mathbf{c}_{\text{highlight}} + (1 - s) (t \mathbf{c}_{\text{warm}} + (1 - t) \mathbf{c}_{\text{cool}})$$



$$\begin{aligned}\mathbf{c}_{\text{cool}} &= (0, 0, 0.55) + 0.25 \mathbf{c}_{\text{surface}}, \\ \mathbf{c}_{\text{warm}} &= (0.3, 0.3, 0) + 0.25 \mathbf{c}_{\text{surface}}, \\ \mathbf{c}_{\text{highlight}} &= (1, 1, 1), \\ t &= \frac{(\mathbf{n} \cdot \mathbf{l}) + 1}{2}, \\ \mathbf{r} &= 2 (\mathbf{n} \cdot \mathbf{l}) \mathbf{n} - \mathbf{l}, \\ s &= (100 (\mathbf{r} \cdot \mathbf{v}) - 97)^{\overline{+}}.\end{aligned}$$

Tomado de [1]

# Sombreado - *Shading*

---

En modelado poligonal, existen 3 tipos:

- Sombreado plano.
- Sombreado de Gouraud.
- Sombreado de Phong.

Estos normalmente corresponden a:

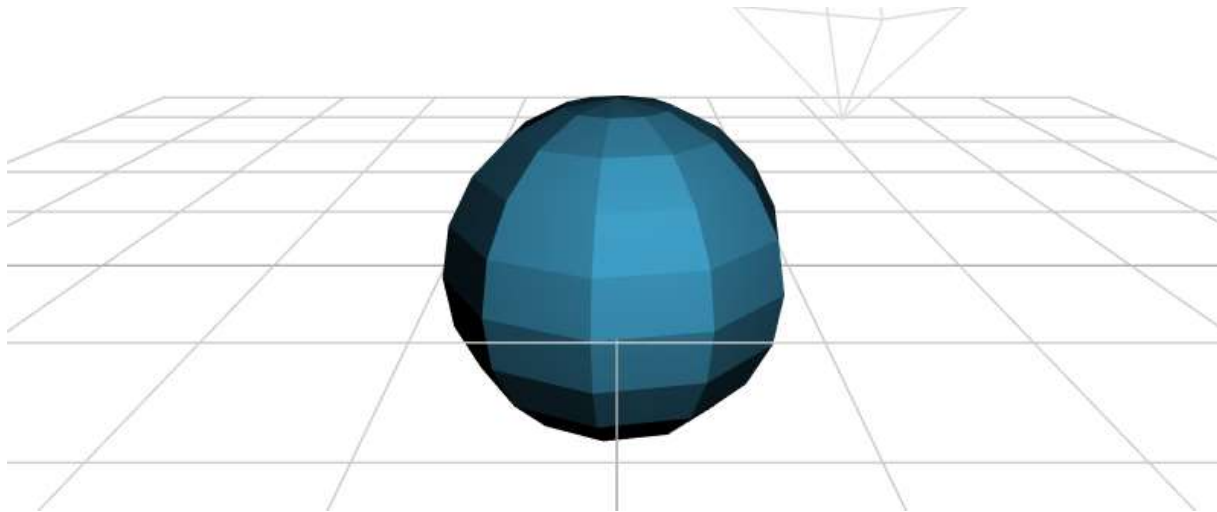
- Sombreado por polígonos.
- Sombreado por vértices.
- Sombrado por píxeles.

# Sombreado plano: *Flat shading*

---

Es el modelo mas simple. Consiste en:

- Calcular el color del polígono (normalmente en el centroide).
- Aplicar ese color para cada pixel de ese polígono.



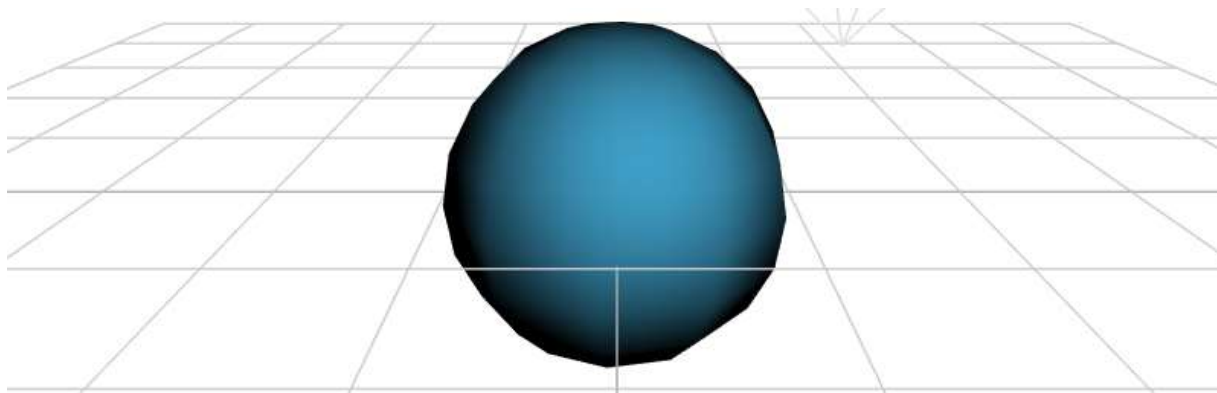
# Sombreado de Gouraud

---

También conocido como sombreado por vértices (vertex shading) o sombreado interpolado.

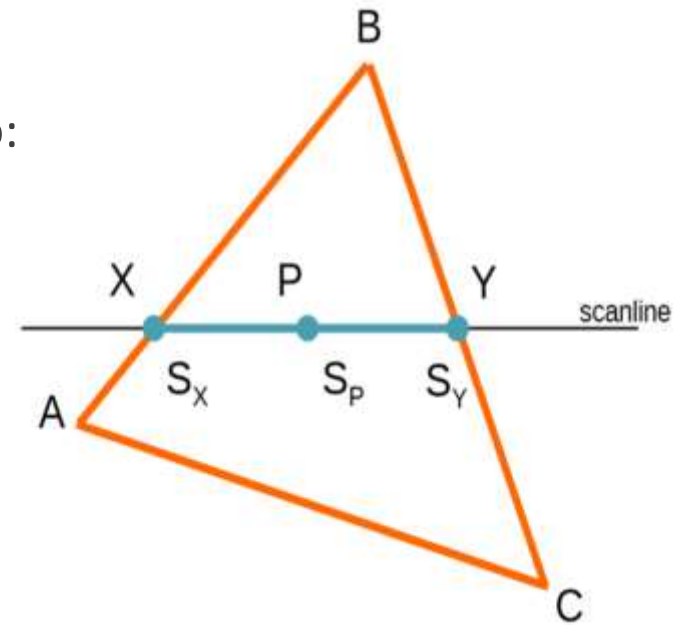
Es el ideal para representar superficies curvas, aproximadas por mallas poligonales. Consiste en:

- Calcular el color correspondiente a cada vértice.
- Interpolarse linealmente el color para cada pixel interior, dados los valores de los otros vértices.



# Sombreado de Gouraud: Algoritmo

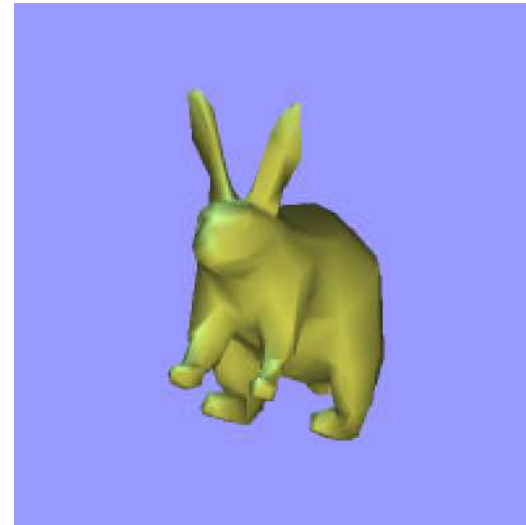
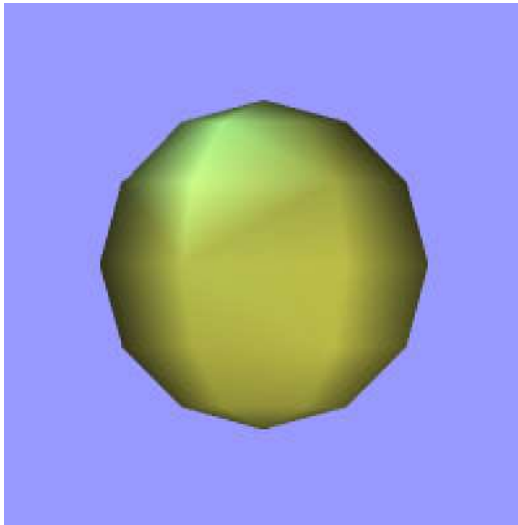
- Determinar el vector unitario normal promedio en cada vértice.
- Aplicar iluminación, para obtener la intensidad luminosa en cada vértice.
- Interpolarse linealmente las intensidades de los vértices, para cada pixel en el área del polígono:
  - En el borde, interpolar entre vértices.
  - Al interior, interpolar con respecto al borde.



# Sombreado de Gouraud: Desventajas

---

Se presentan las bandas de Mach debido a la aproximación en la interpolación lineal.



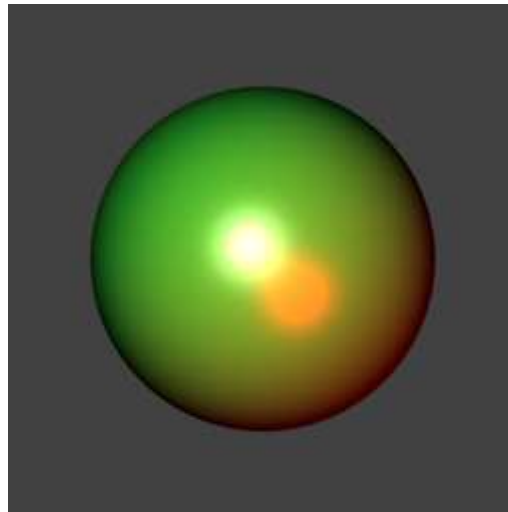
La solución: Subdividir polígonos o hacer interpolación de mayor orden.

# Sombreado de Phong

---

El cálculo de iluminación de cada pixel se basa en interpolación de los vectores normales, no del color.

- En los bordes, a partir de las normales en los vértices.
- Al interior, a partir de las normales en los bordes.







Flat

Gouraud

Phong

# Comparación

---

# Comparación

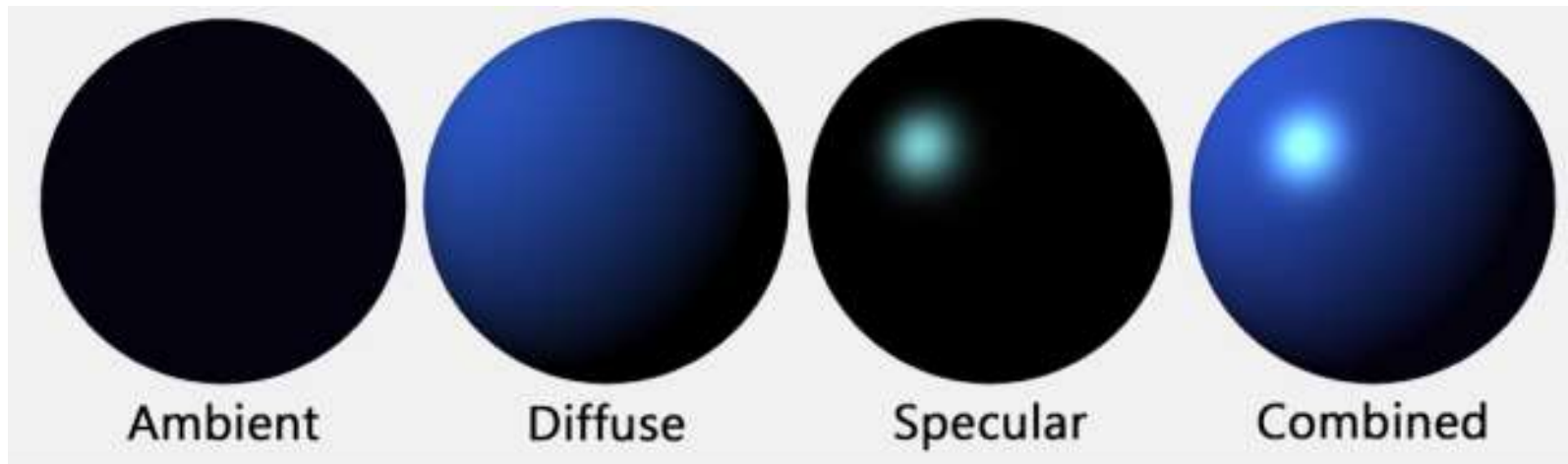
---



# Sombreado: Ambiente + Difuso + Especular

---

El sombreado de un objeto puede basarse en el modelo de iluminación utilizado. Recordando, la intensidad de la iluminación corresponde a la suma de la luz ambiente, el componente difuso y el componente especular.





# Iluminación

---

DESIGNA LA INTERACCIÓN ENTRE LOS MATERIALES  
Y LAS FUENTES DE LUZ (FÍSICA).

# Fuentes luminosas

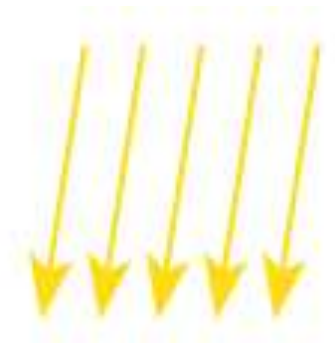
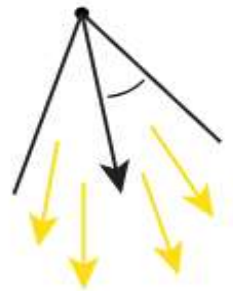
## *Light sources*

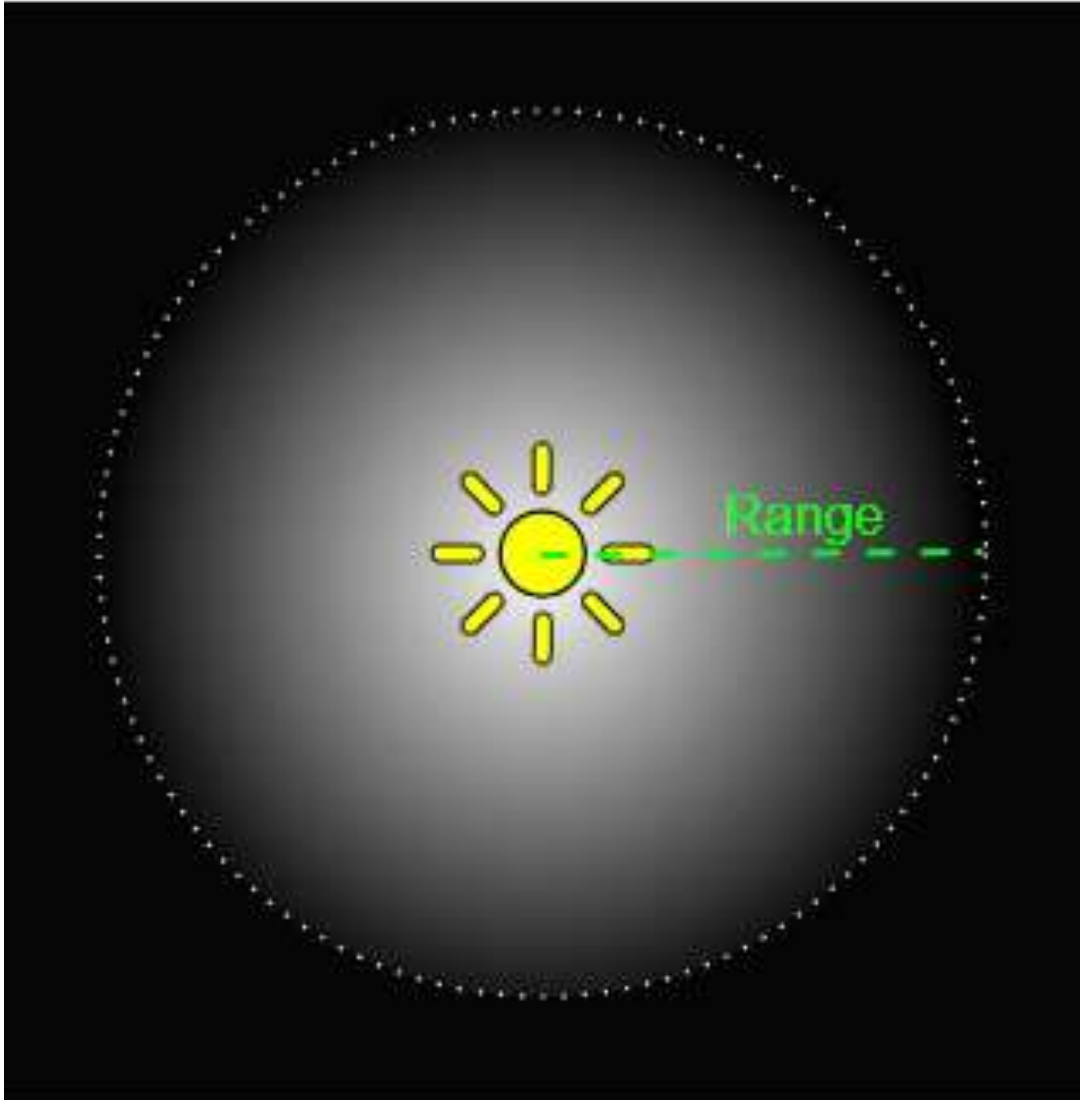
---

Con el fin de renderizar una escena, es necesario describir la iluminación de la escena al motor de render.

Esta descripción consiste básicamente en una lista de fuentes luminosas, con diferentes características (color, intensidad, forma, tamaño, etc). Las más usadas son:

- Puntuales – ***Point light***.
- Direccionales – ***Directional light***.
- Luz concentrada – ***Spot Light***.
- Ambiental – ***Ambient light***.





## Light sources: Point lights

---

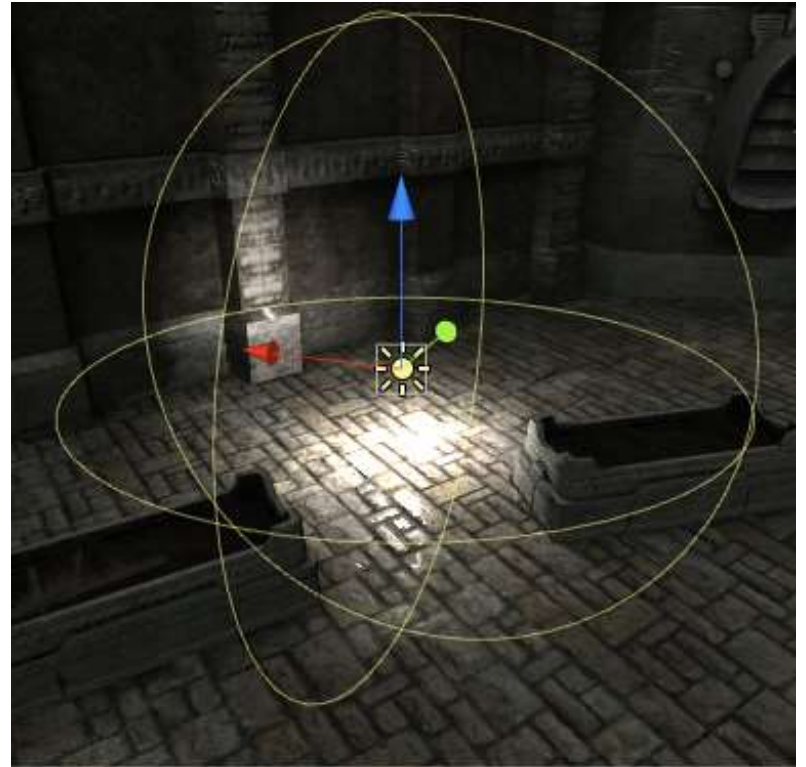
*Luz que emana desde un punto en el espacio, hacia afuera, en todas las direcciones.*

La intensidad de la luz normalmente decrece a medida que se aleja del centro de luz y es cero al finalizar el rango.

Útiles para simular lámparas, fuego u otras fuentes de luz en una escena.

# Light sources: Point lights

---



# Light sources:

## Directional lights

---

Representa luz que emana de un punto distante en el espacio, tan lejano, que los rayos de luz apuntan en la misma dirección, paralelos unos a otros.

Una luz direccional no tiene una posición, ni se atenúa.

El ejemplo más claro es el sol.

En este modelo el vector  $\mathbf{l}$  y el color de la luz  $C_{\text{light}}$  son constantes para toda la escena.  $C_{\text{light}}$  podría verse atenuado por las sombras en la escena.



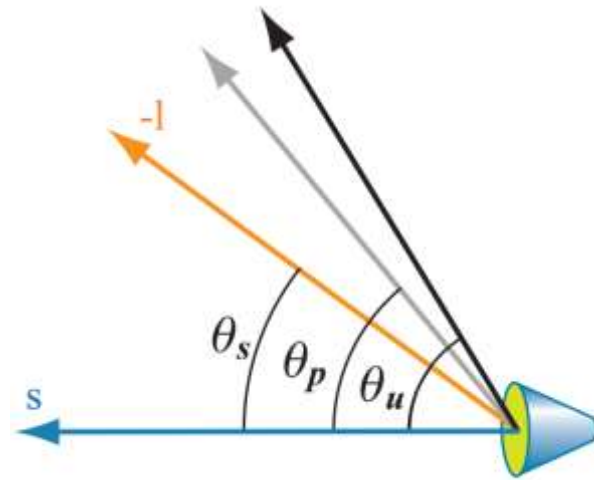
# Light sources: Spot lights

---

Luz que varía de intensidad, tanto en distancia como en dirección.

Similar a una linterna, la intensidad de se interpola entre 2 secciones:

- Umbra: Zona oscura.
- Penumbra: Máxima luz.



# Light sources:

## Ambient lights

---

Es la más sencilla pues se trata de un parámetro que indica el color a aplicar a los objetos cuando no tienen ninguna otra iluminación.

# Phisically Based Rendering

---



Para más información: <https://www.allegorithmic.com/pbr-guide>  
Ejemplo en Three.js: [https://threejs.org/examples/webgl\\_shading\\_physical.html](https://threejs.org/examples/webgl_shading_physical.html)

# Phisically Based Rendering

---

Modelo estándar en aplicaciones de 3D tales como Unity, Unreal Engine y 3D Studio Max.

No se usan aproximaciones a la forma como la luz interactúan. En cambio, se usa un modelo físicamente correcto. **Genera mayor realismo.**

En vez de modificar los materiales, para que funcionen correctamente bajo condiciones específicas de luz, el material es creado para que reaccione correctamente a cualquier tipo de luz.

Es un método mas costoso computacionalmente. Se basa en: rayos de luz, absorción y dispersión, reflexión difusa y especular, color, conservación de energía, entre otros fenómenos.

# Sombreado PBR

---

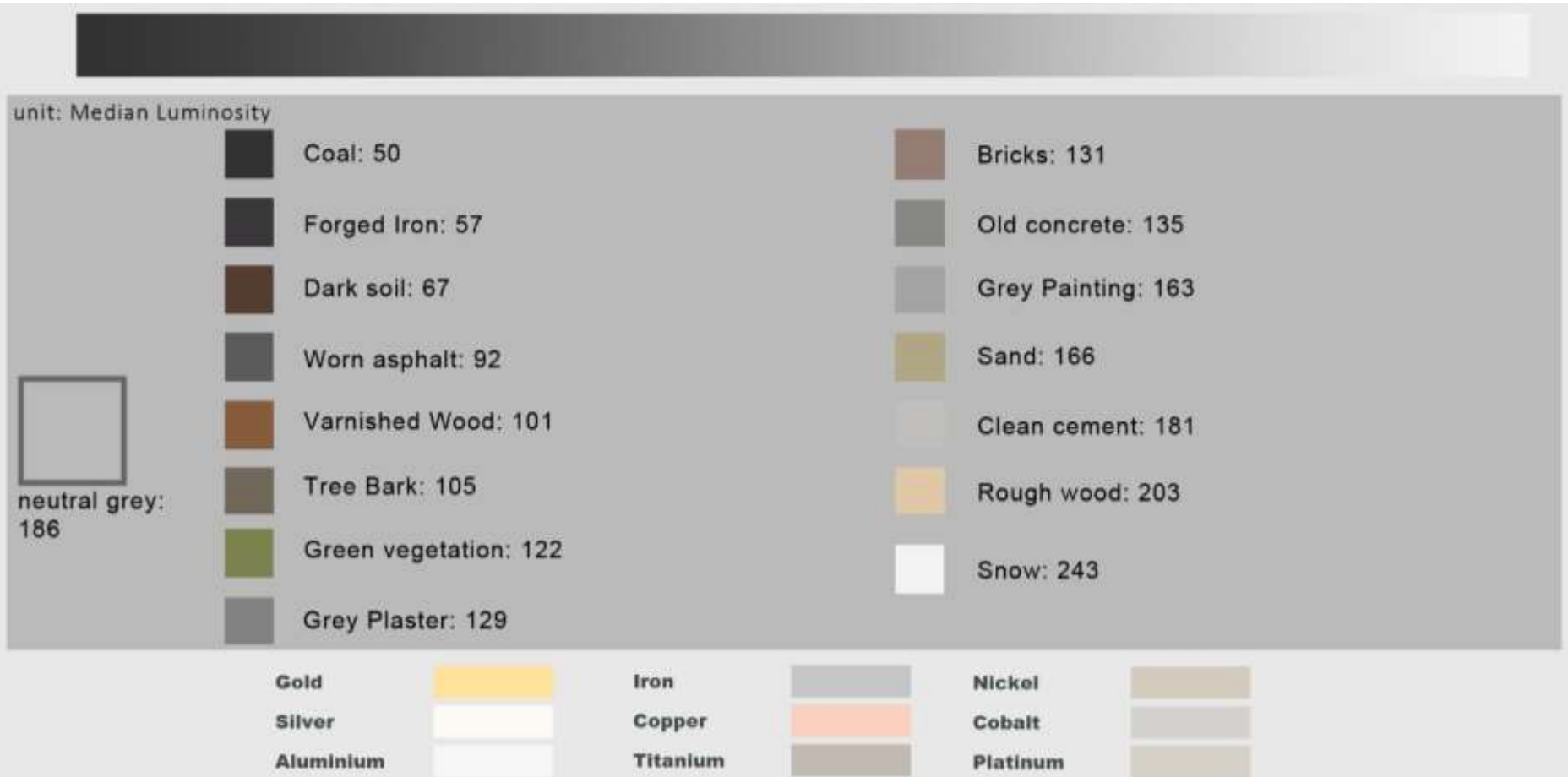
**Albedo:** porcentaje de radiación de un material.

**Metalness:** característica de un material de ser metálico.

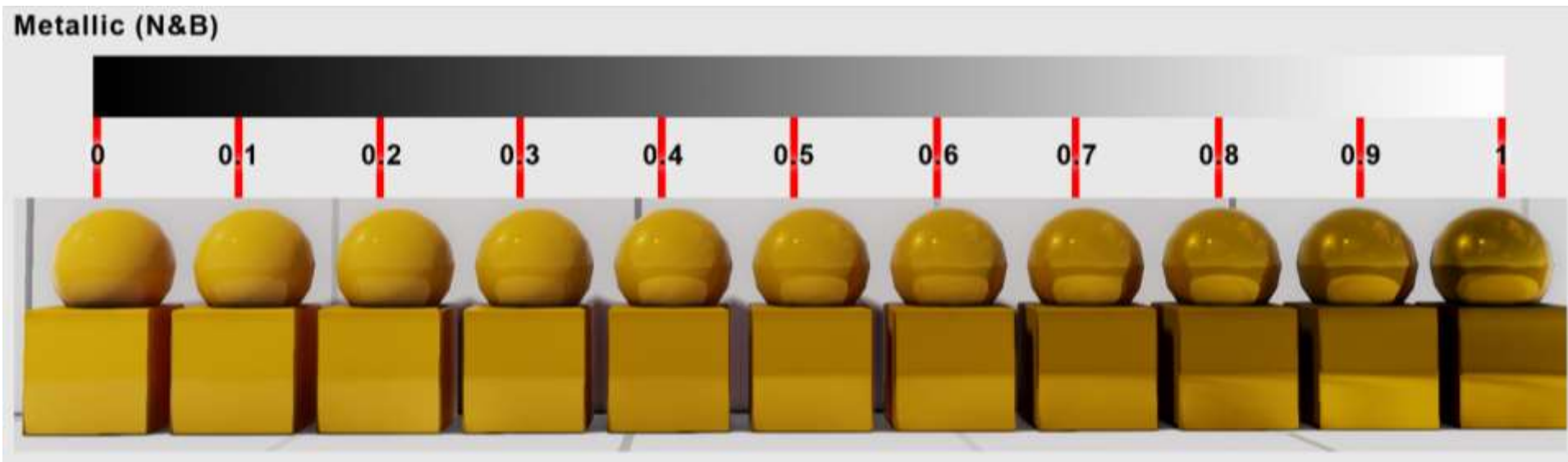
**Roughness** (Rugosidad):



# Base color: sRGB



# Metallic



# Roughness

Roughness (N&B)



0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1





# Porosity

Porosity (N&B)



# Three.js

---

Se pueden utilizar diferentes materiales, para lograr los tipos de sombreado descritos previamente.

- **MeshBasicMaterial:** Sombreado plano. No se afecta por la luz.
- **MeshLambertMaterial:** Para superficies no brillantes, sin brillos especulares. Basado en el modelo de sombreado de Gouraud.
- **MeshPhongMaterial:** Para superficies brillantes, con brillos especulares. Basado en el modelo de sombreado de Phong.
- **MeshStandardMaterial:** Material estándar, basado en las propiedades físicas de los materiales. Usa el flujo de trabajo Metálico-Rugosidad.

# Ejercicio

En grupos, investigar sobre las características de los materiales y presentar, en THREE.JS una escena similar a la siguiente imagen, incluyendo materiales como: oro, acero, aluminio, plata, cobre, madera, plástico, entre otros.





STEEL



IRON



ALUMINUM



GALVANIZED



CHROME



CARBON FIBER



PLASTIC



ROUGH PLASTIC



RUBBER



WORN RUBBER



TARP



ASPHALT



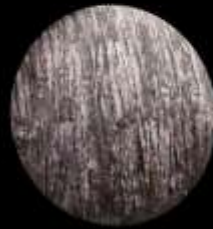
CONCRETE



COBBLESTONE



GRASS



TREE BARK - PINE

# Bibliografía

---

- [1] Akenine, T. et al. (2019). ***Real time rendering***. 4th Ed.
- [2] Dunn, F. y Parberry, I. (2002). Chapter 15 -3D Math for Graphics en: ***3D Math Primer for Graphics and Game Development***. Wordware Publishing, Inc.
- [3] Hughes, J et al. (2014). Chapter 26: Light en: ***Computer Graphics: Principles and Practice***. 3rd Ed. Addison-Wesley.
- [4] Rueda, A. (2014). ***Superficies visibles Modelos de iluminación y sombreado***. Introducción a la computación gráfica. Pontificia Universidad Javeriana.
- [5] ***[http://content.mindofmatthew.com/how\\_3d\\_works/](http://content.mindofmatthew.com/how_3d_works/)***