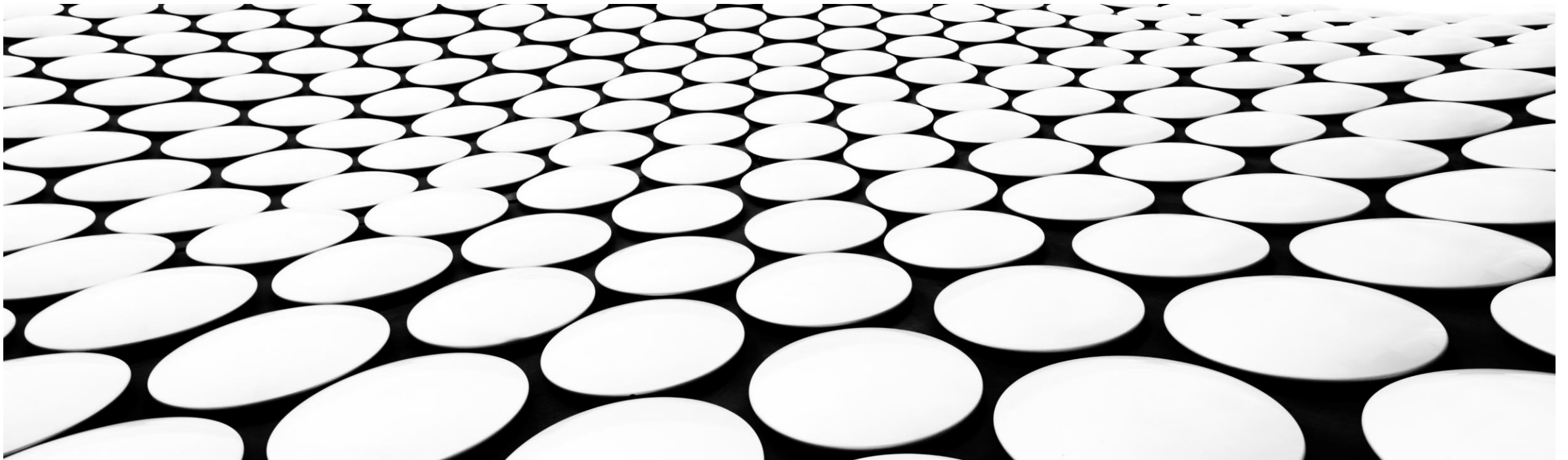


---

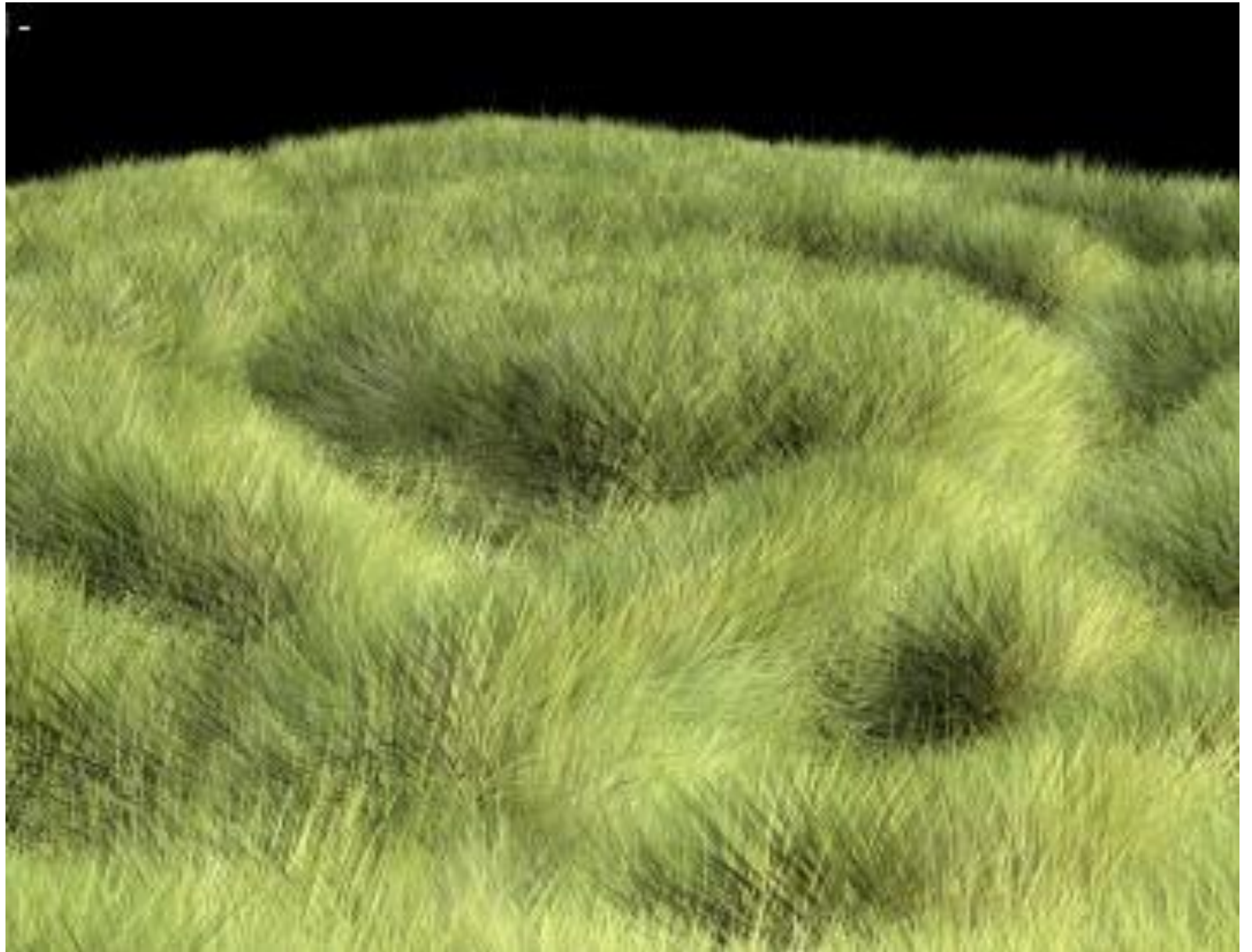
# TC2008B MODELACIÓN DE SISTEMAS MULTIAGENTES CON GRÁFICAS COMPUTACIONALES

TEXTURAS



# LÍMITES DEL MODELADO

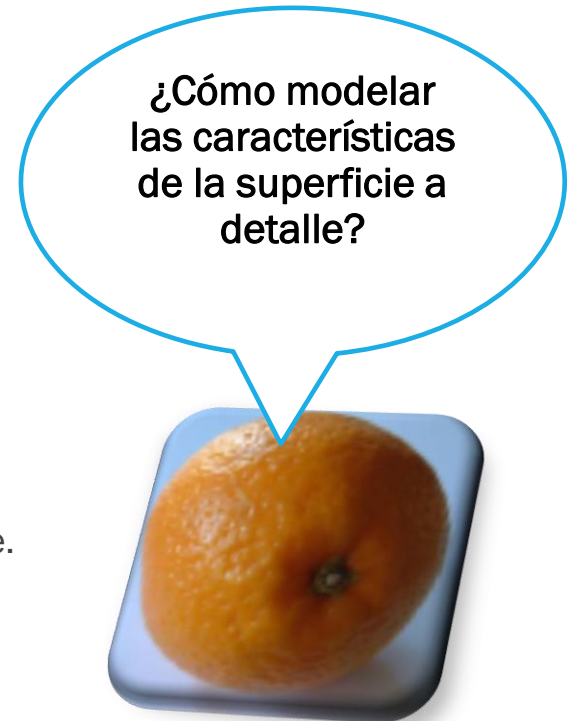
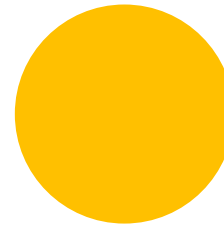
- Aún cuando las tarjetas gráficas modernas pueden renderizar millones de polígonos por segundo, ese número es insuficiente cuando se trata de objetos con superficies complejas:
  - Pasto
  - Terreno
  - Piel
  - Naranja



# EJEMPLO: NARANJA

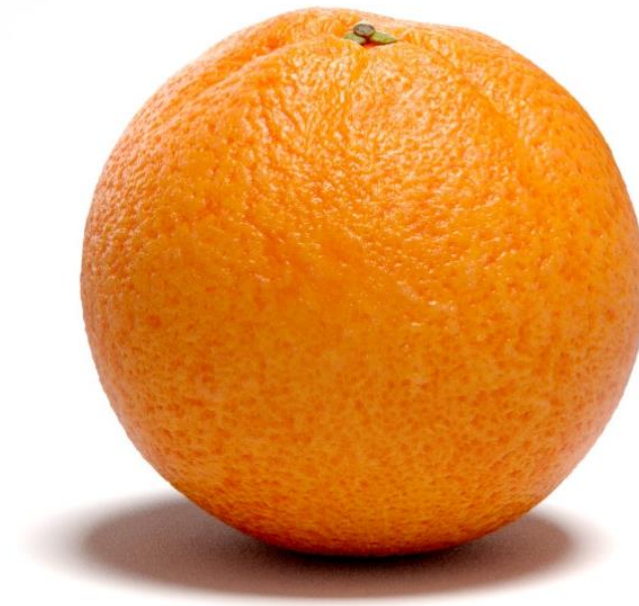
- Considera el problema de modelar una naranja:

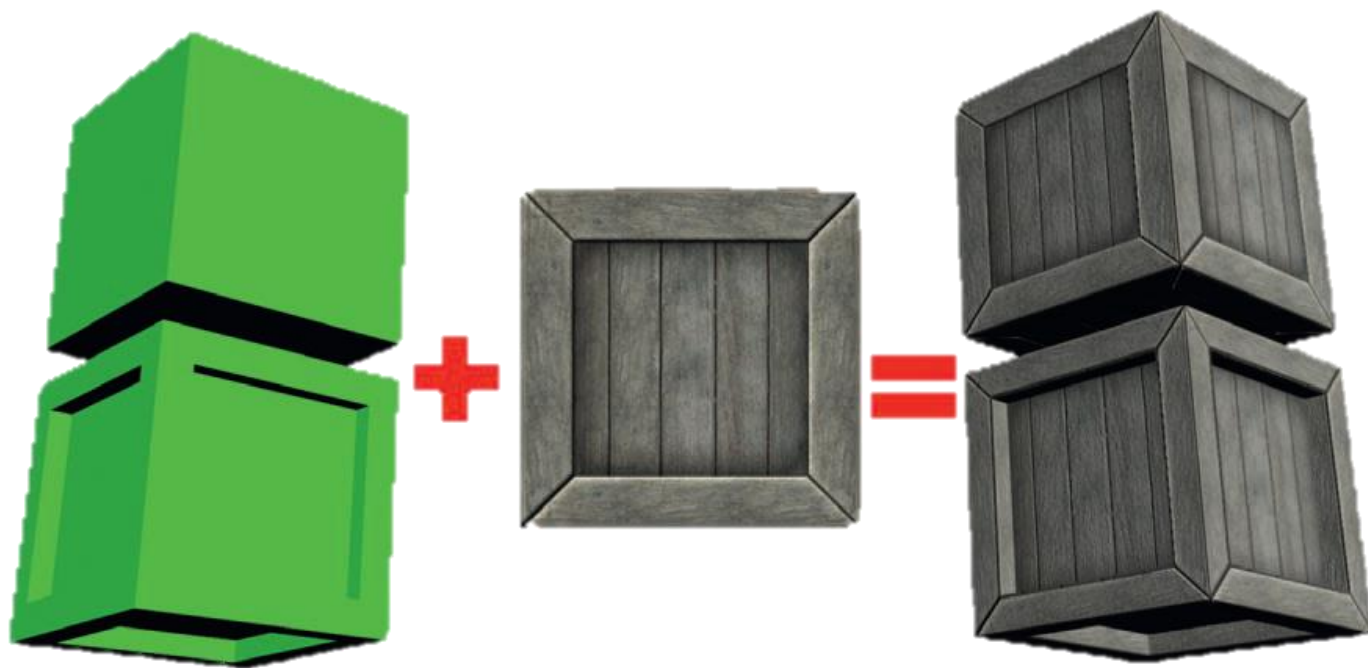
1. Una esfera naranja sería demasiado sencilla.
2. Reemplazar la esfera por una malla de miles de polígonos sería costoso computacionalmente.



# MAPEO DE TEXTURAS

- La solución podría ser tomar una imagen de la superficie de la naranja y “pegarla” sobre un modelo geométrico simple.
- La imagen es utilizada para alterar el color en cada punto de la superficie.
- Esto es mucho más eficiente porque no aumenta la complejidad de la geometría del modelo.





**MAPEO DE  
TEXTURAS**



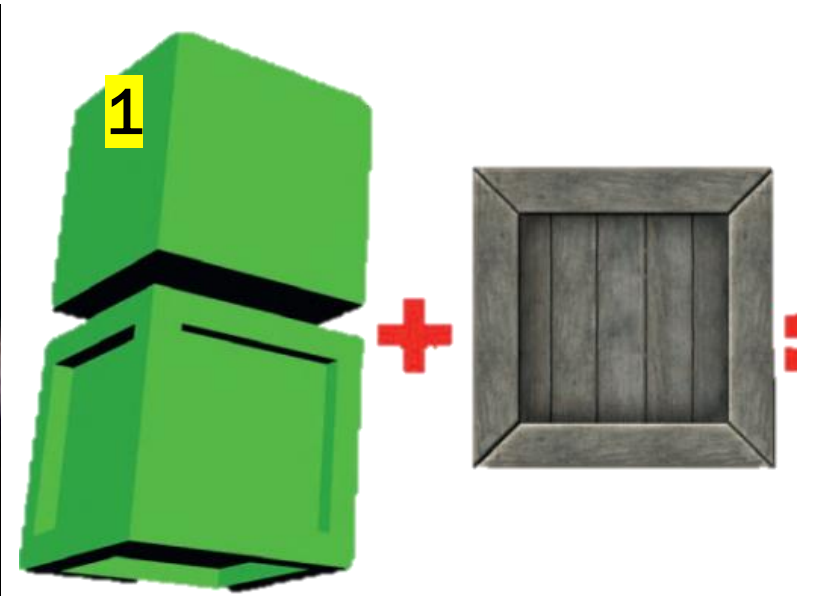
## DETALLE VISUAL

- Usar imágenes y no más polígonos para representar variaciones de pequeña escala del color, añade detalle.



# TIPOS DE MAPEO

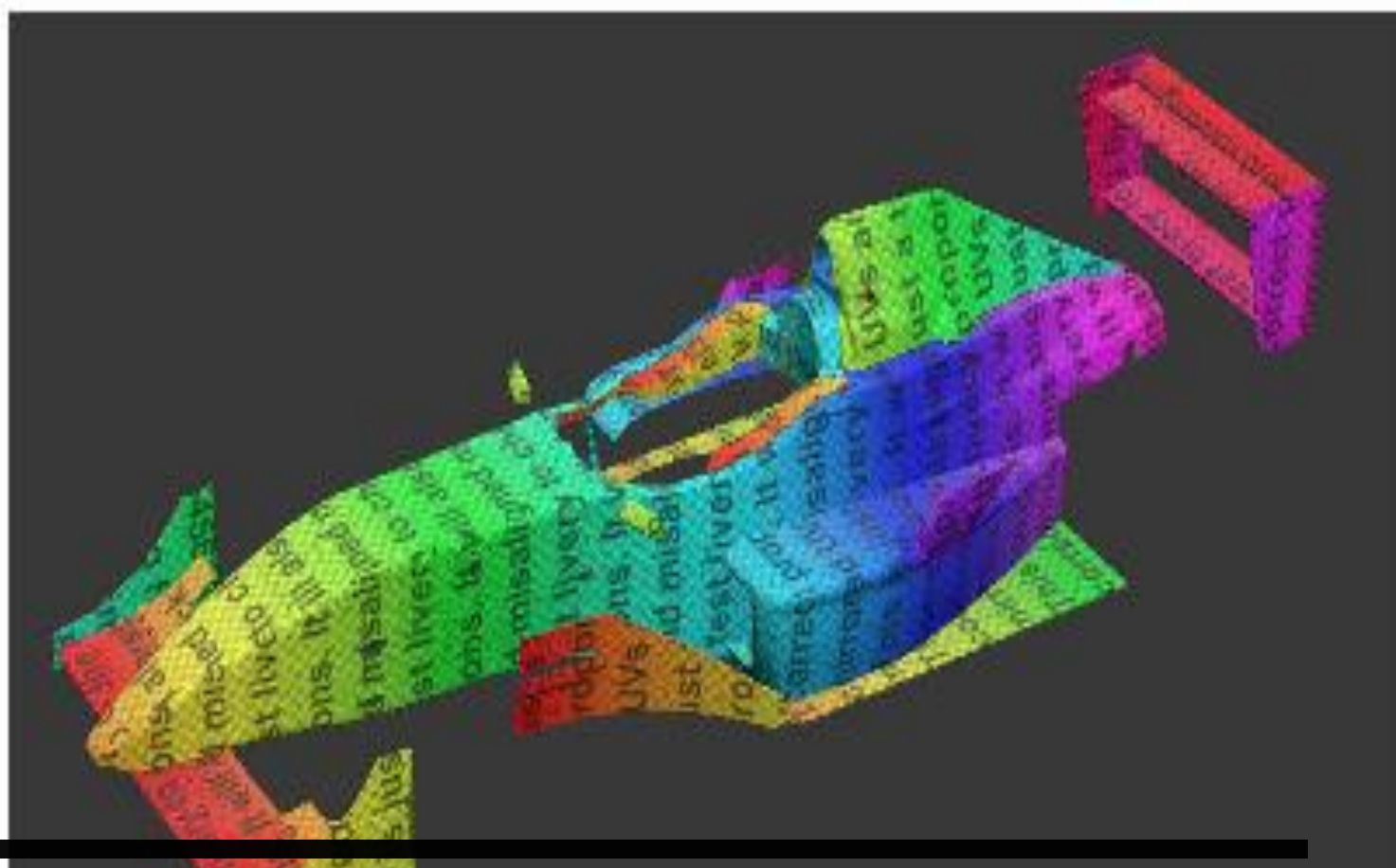
1. Mapeo de texturas:
  - Uso de imágenes.
2. Mapeo ambiental:
  - Reflexiones.
3. Mapeo de normales:
  - *Bump mapping*.





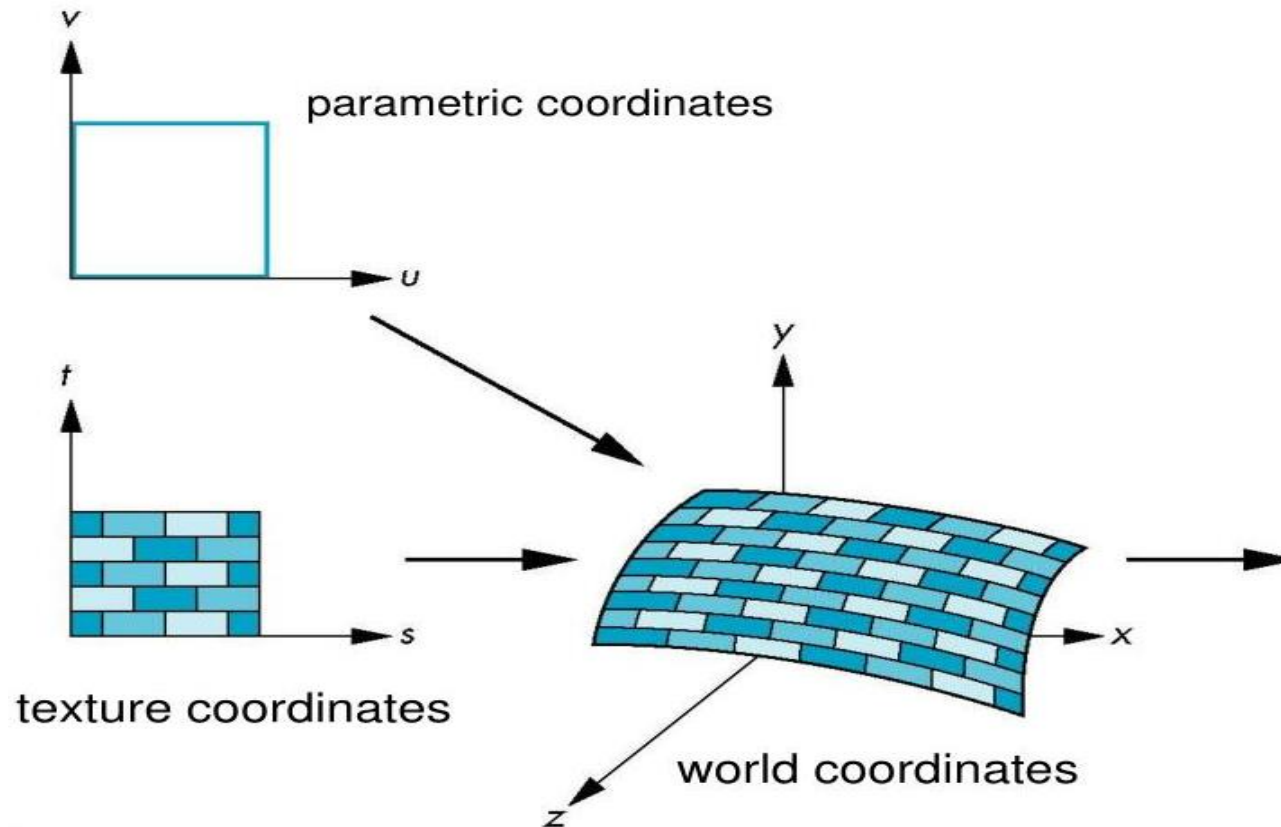


## MAPEO DE TEXTURAS BÁSICO



- Una textura es una imagen en un sistema de coordenadas 2D ( $s, t$ ).
- Cada parte de la superficie corresponde o mapea (coordenadas  $u, v$ ) a alguna parte de la textura ( $s, t$ ).
- Cada objeto 3D debe estar parametrizado mediante una función que va de 2D a 3D.



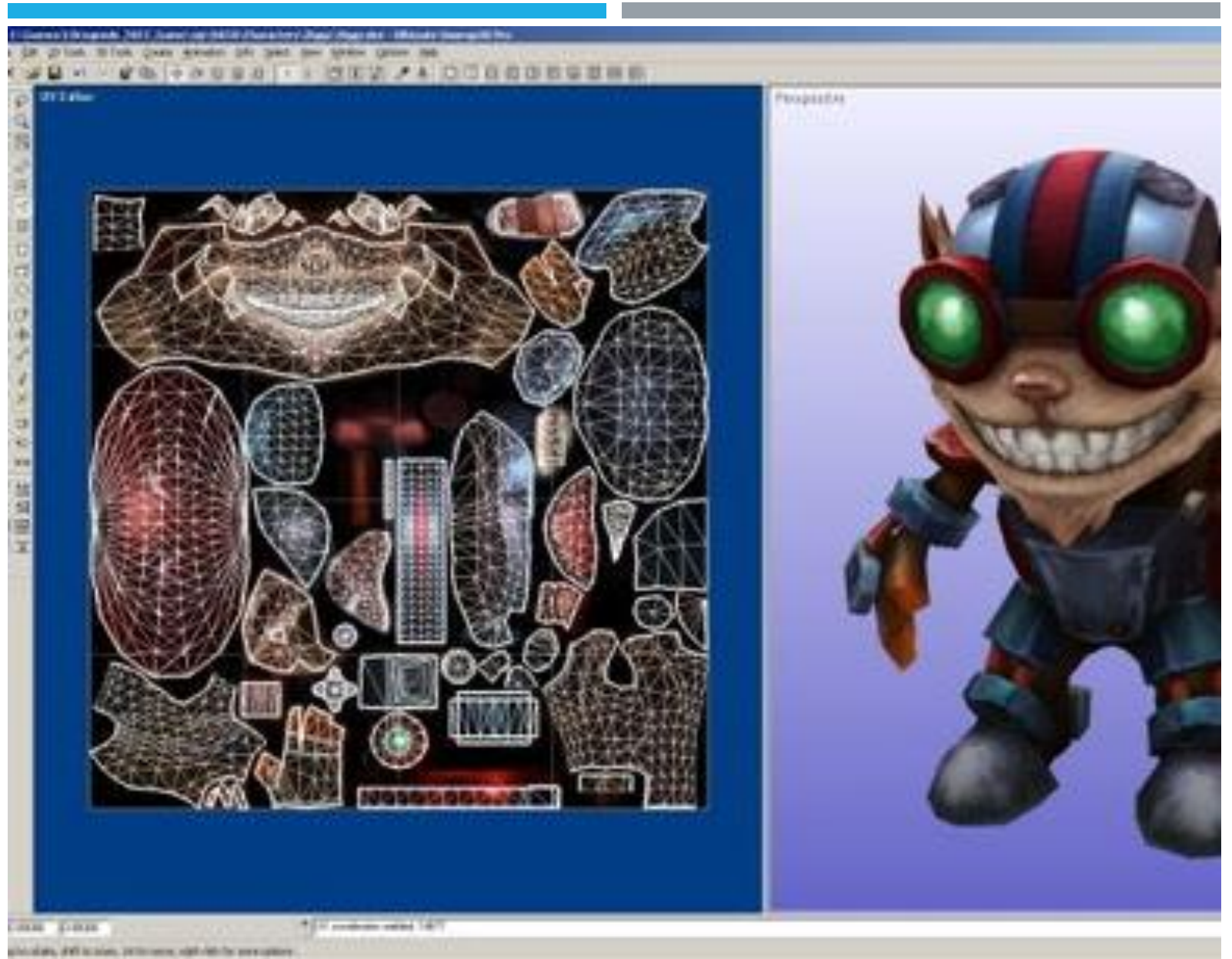


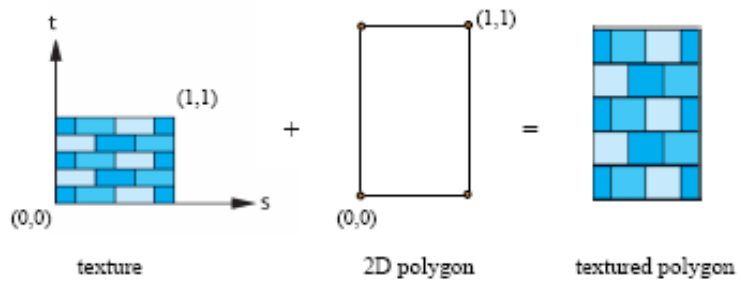
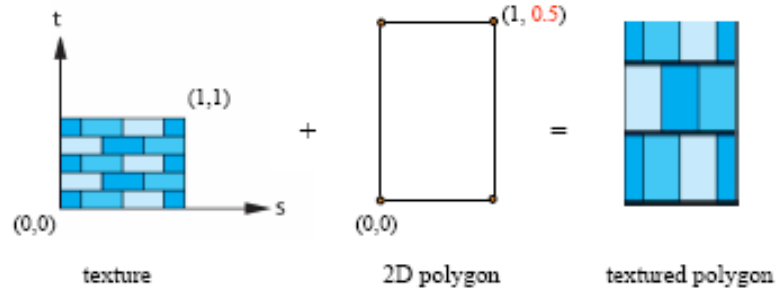
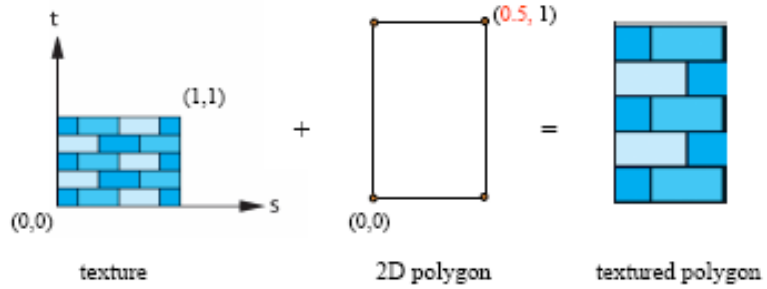
# SISTEMAS COORDENADOS

- Coordenadas de las texturas ( $s, t$ ):
  - Usadas para identificar puntos en la imagen de la textura (usualmente en el rango 0, 1).
- Coordenadas paramétricas ( $u, v$ ):
  - Usadas para mapear la superficie 3D con parámetros 2D (usualmente en el rango 0, 1).

# ¿CÓMO ASIGNAR VALORES U, V?

1. Asignar valores manualmente:
  - Dar coordenadas de la textura para cada vértice manualmente.
2. Calcular las coordenadas automáticamente:
  - Utilizar un algoritmo que calcule dichas coordenadas.



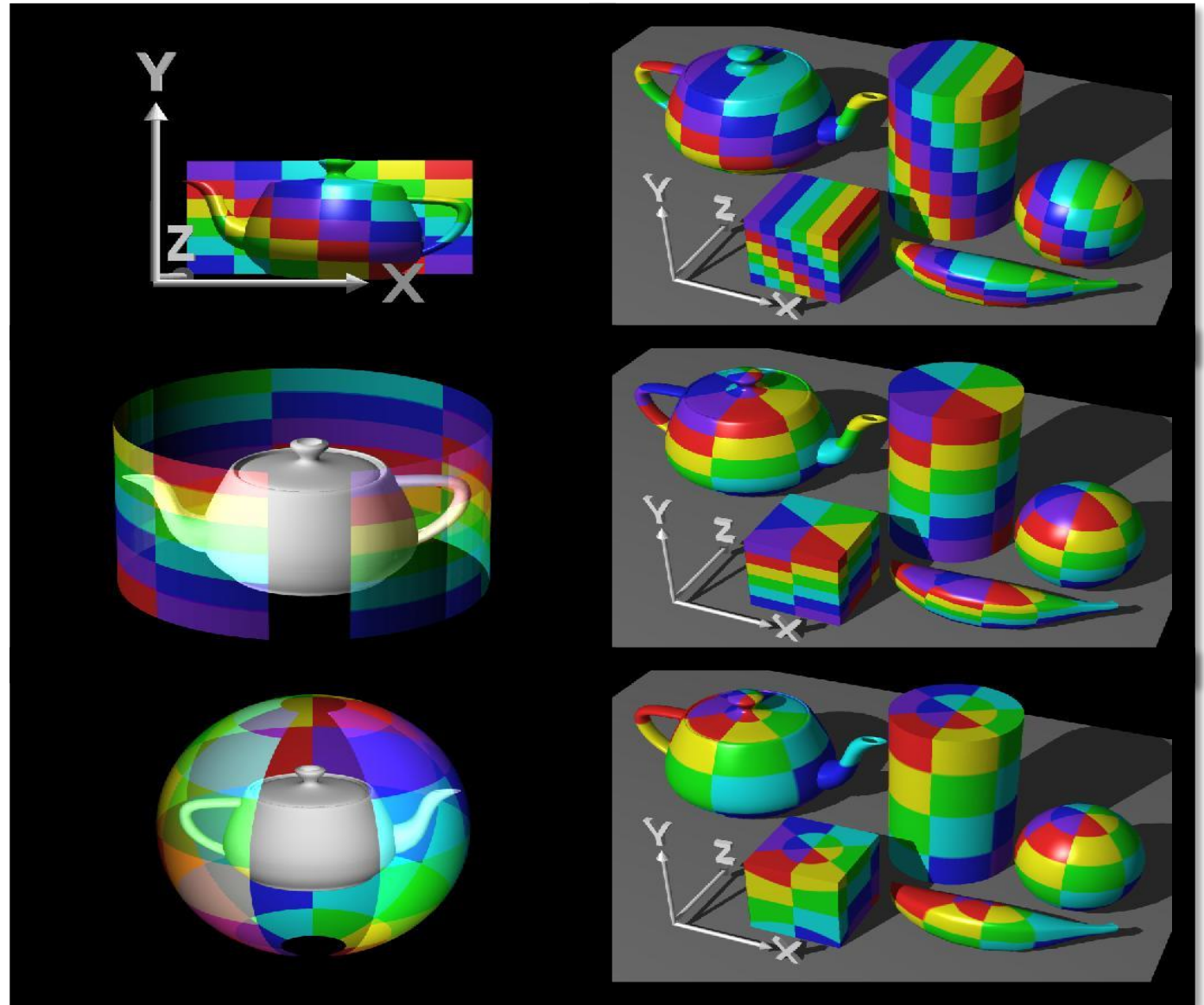


# ASIGNACIÓN MANUAL DE COORDENADAS U, V

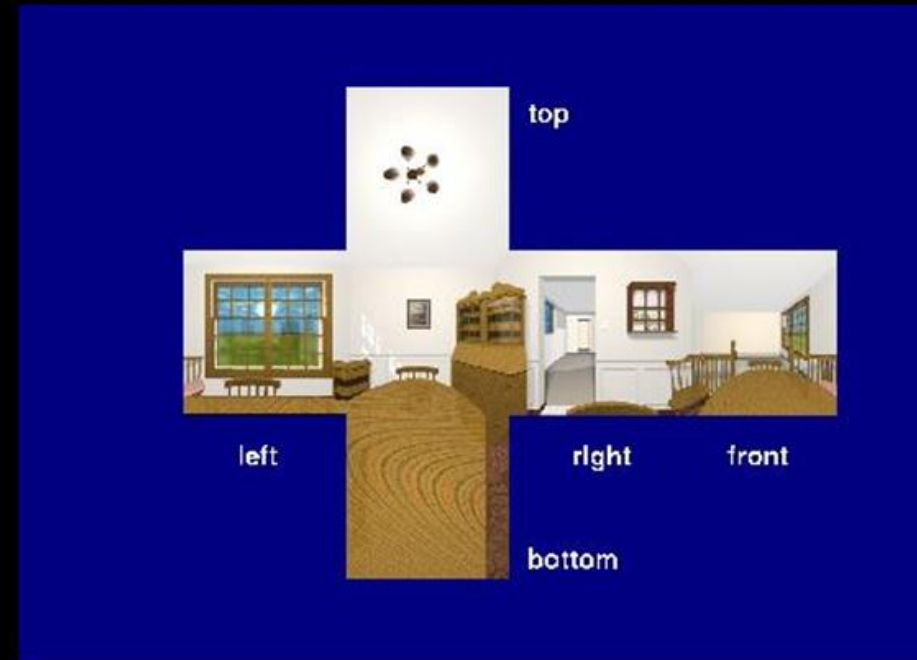
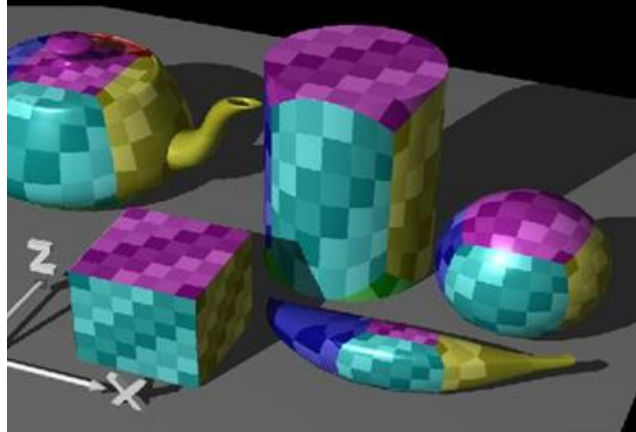
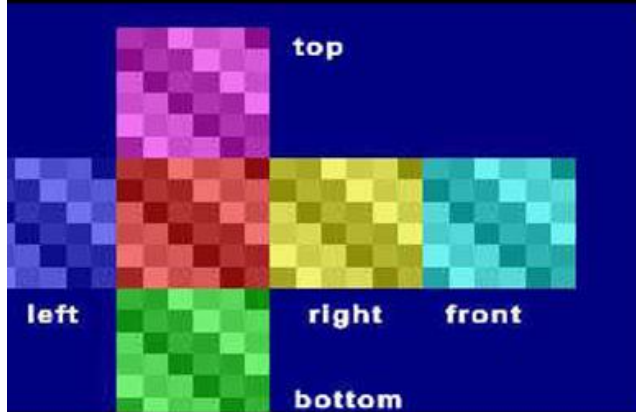
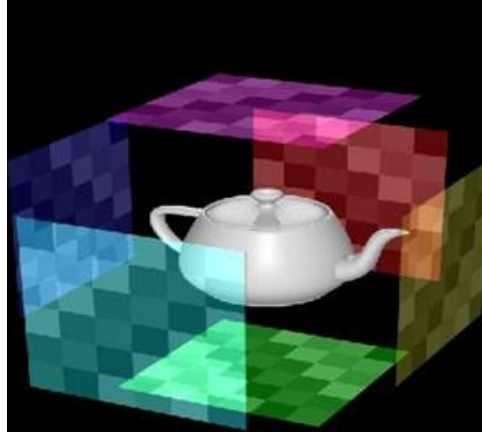
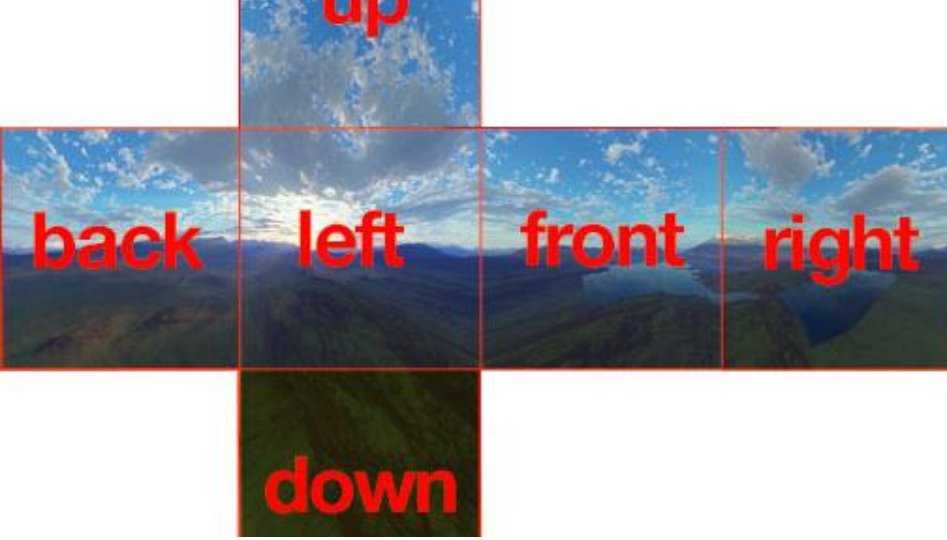
- Arriba: se usa la mitad izquierda de la textura (desde 0 hasta 0.5 en la dimensión U) para cubrir todo el polígono.
- En medio: se usa la mitad inferior de la textura (desde 0 hasta 0.5 en la dimensión V) para cubrir todo el polígono.
- Abajo: se usa toda la textura para cubrir todo el polígono.
- Nota que en todos los casos hay una deformación de la apariencia original de la textura.

# MAPEO U, V MEDIANTE ALGORITMOS

- Es fácil asignar coordenadas de texturas para objetos sencillos en 2D, pero puede ser muy difícil asignar valores cuando se trata de objetos complicados en 3D.
- Mapeo en dos etapas o pasos
  - La solución consiste en primero mapear la textura a una superficie intermedia, luego mapear de la intermedia a la superficie final.
- Existen mapeos estándar: plano, cilíndrico, esférico y cúbico.



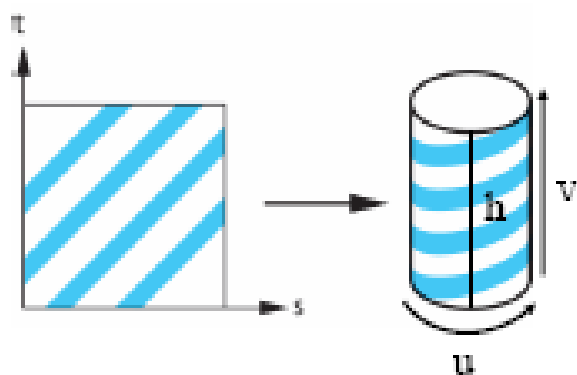




## MAPEO U, V CÚBICO

- Se utiliza para hacer un "skybox" es decir, rodear el punto de vista del usuario en una escena virtual, para simular que está dentro de un ambiente extenso.





## MAPEO CILÍNDRICO

- Cilindro paramétrico:

$$x = r \cos (2 \Pi u)$$

$$y = v/h$$

$$z = r \sin (2 \Pi u)$$

- Mapear un rectángulo en el espacio  $u, v$  a un cilindro de radio  $r$  y altura  $h$  en las coordenadas del mundo:

$$s = u$$

$$t = v$$

# MAPEO ESFÉRICO

- Podemos mapear mediante una esfera paramétrica (los polos se verán distorsionados a menos que la textura se corrija):

$$x = r \sin(2\pi v) \sin(2\pi u)$$

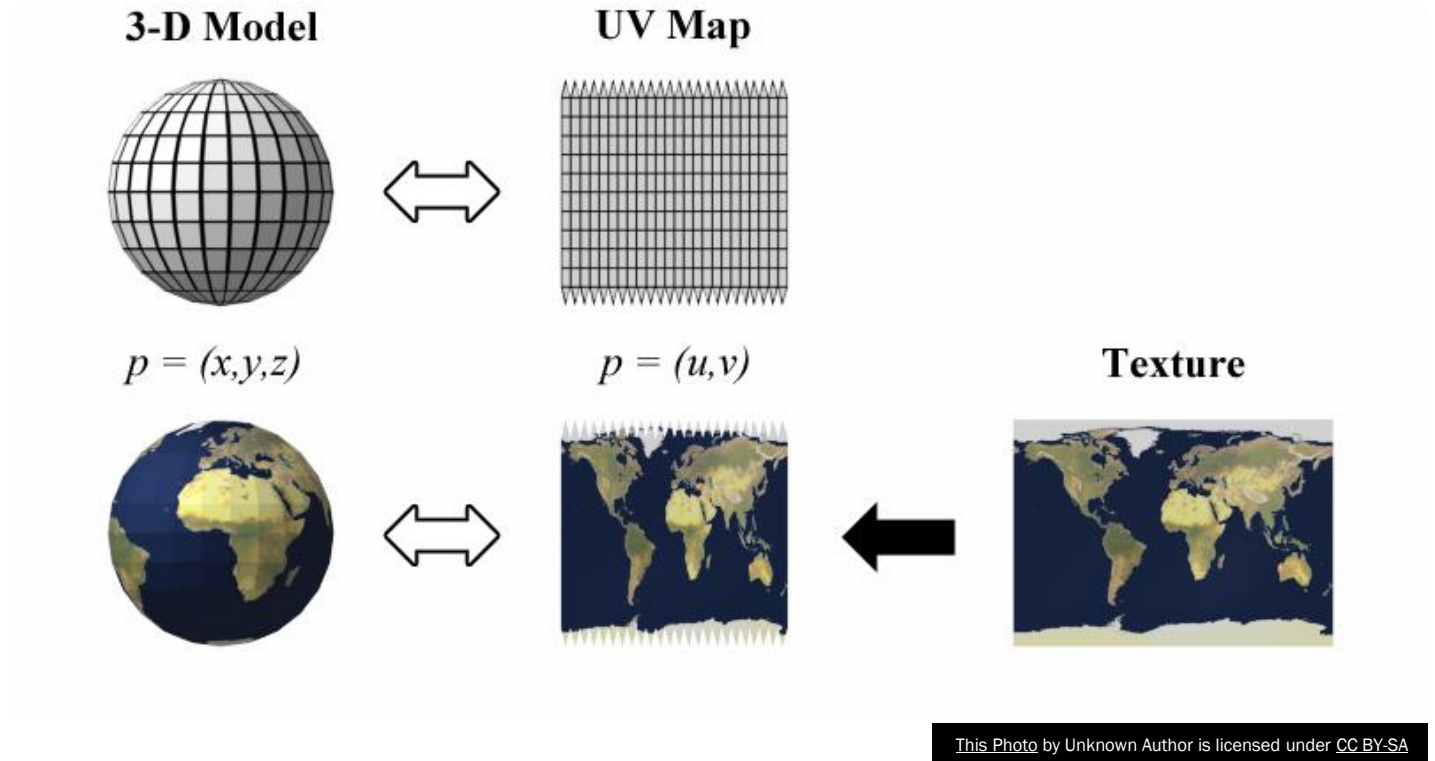
$$y = r \cos(2\pi v)$$

$$z = r \sin(2\pi v) \cos(2\pi u)$$

$$u = 0.5 + \frac{\arcsin(\hat{n}_x)}{\pi}$$

$$v = 0.5 + \frac{\arcsin(\hat{n}_y)}{\pi}$$

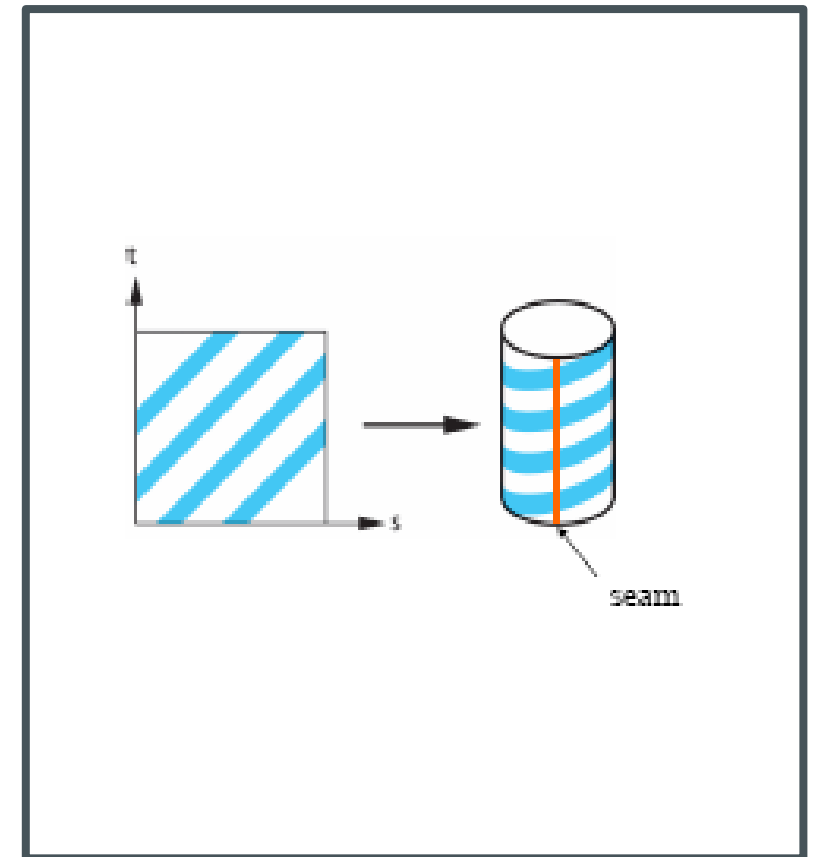
$$\hat{n} = \|Pol - SC\|$$



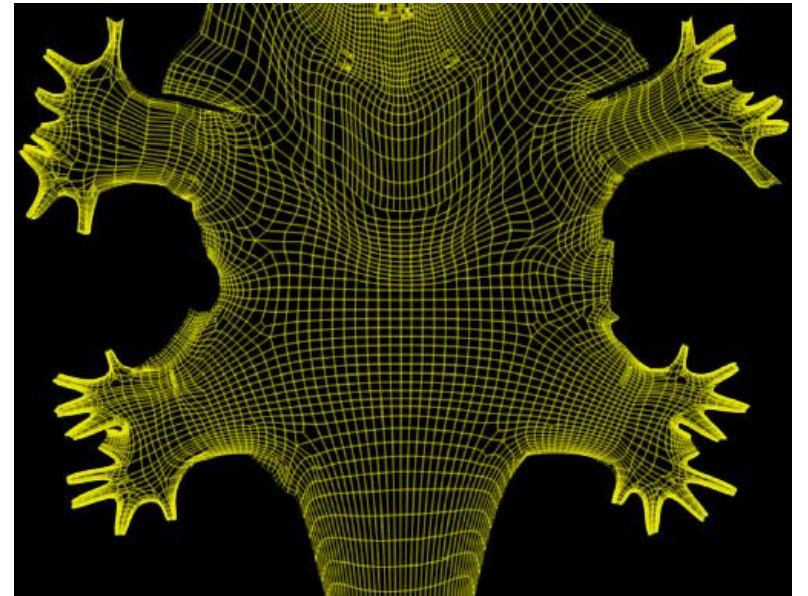
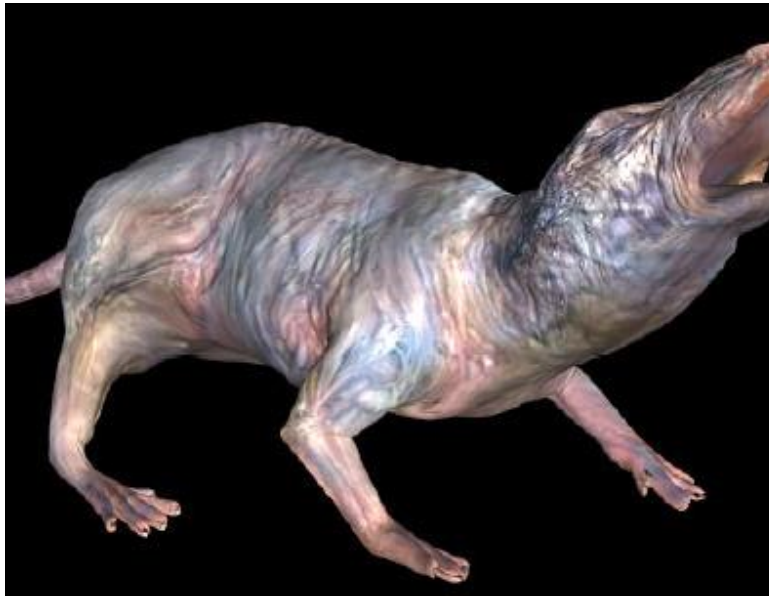
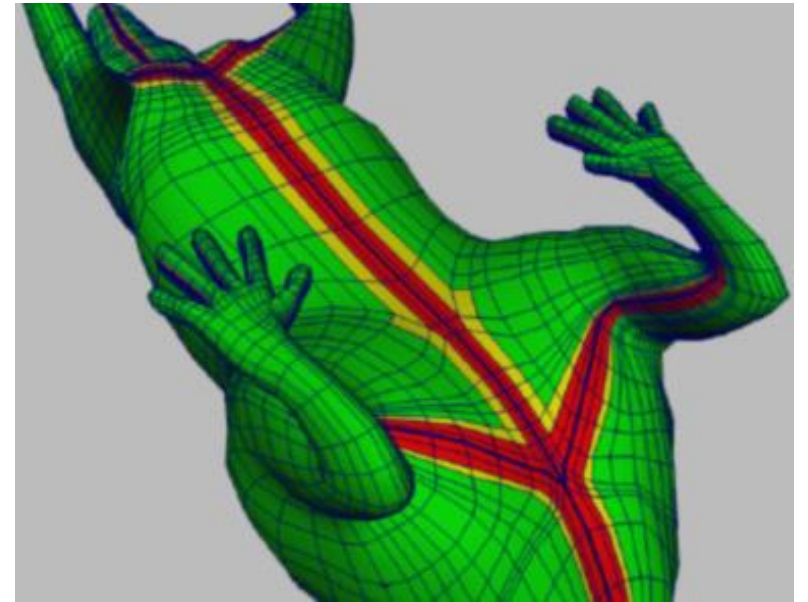
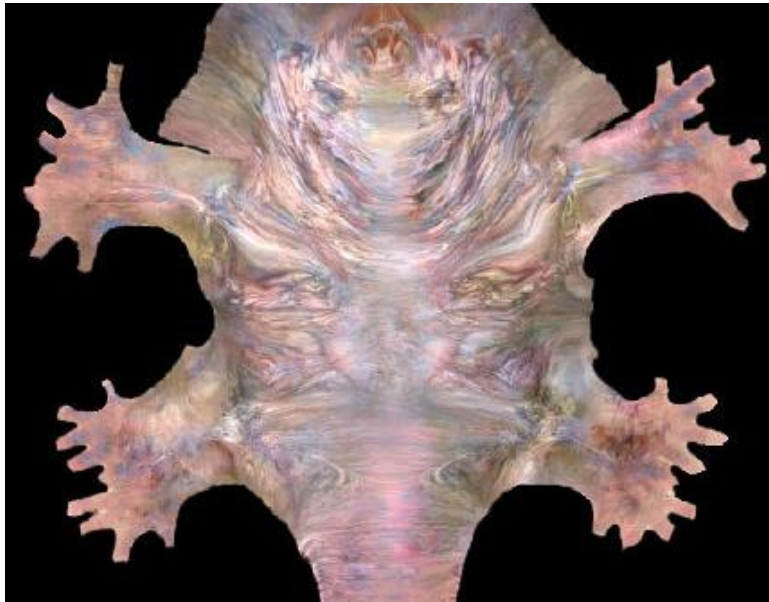


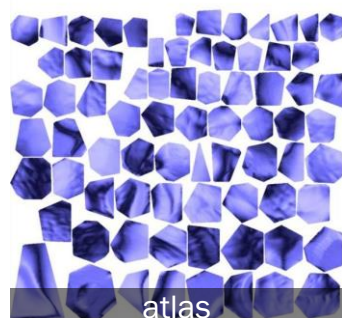
## COSTURAS (SEAMS)

- Al repetir las texturas sobre los objetos (porque son muy pequeñas para cubrirlos totalmente) se pueden generar costuras (seams) que son visibles y rompen el efecto de la simulación de superficie. Para corregir el problema existen tres técnicas.



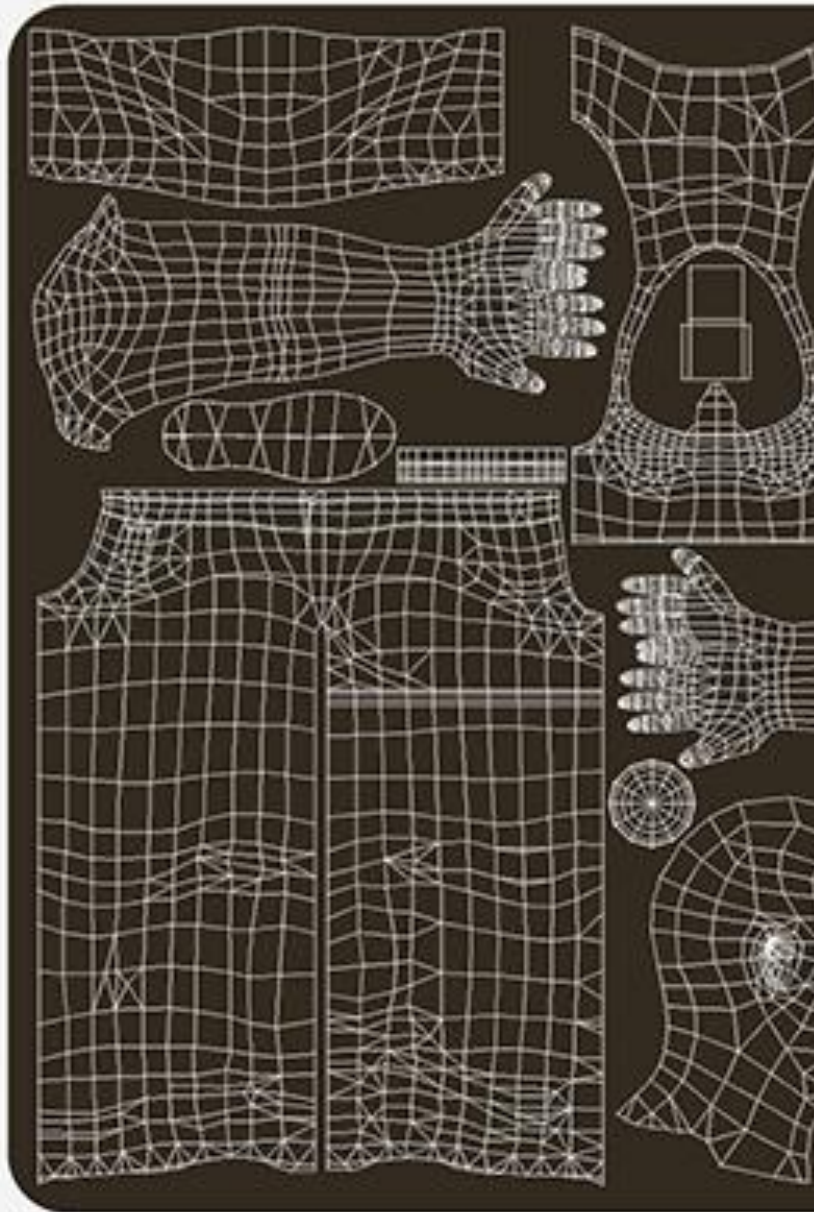
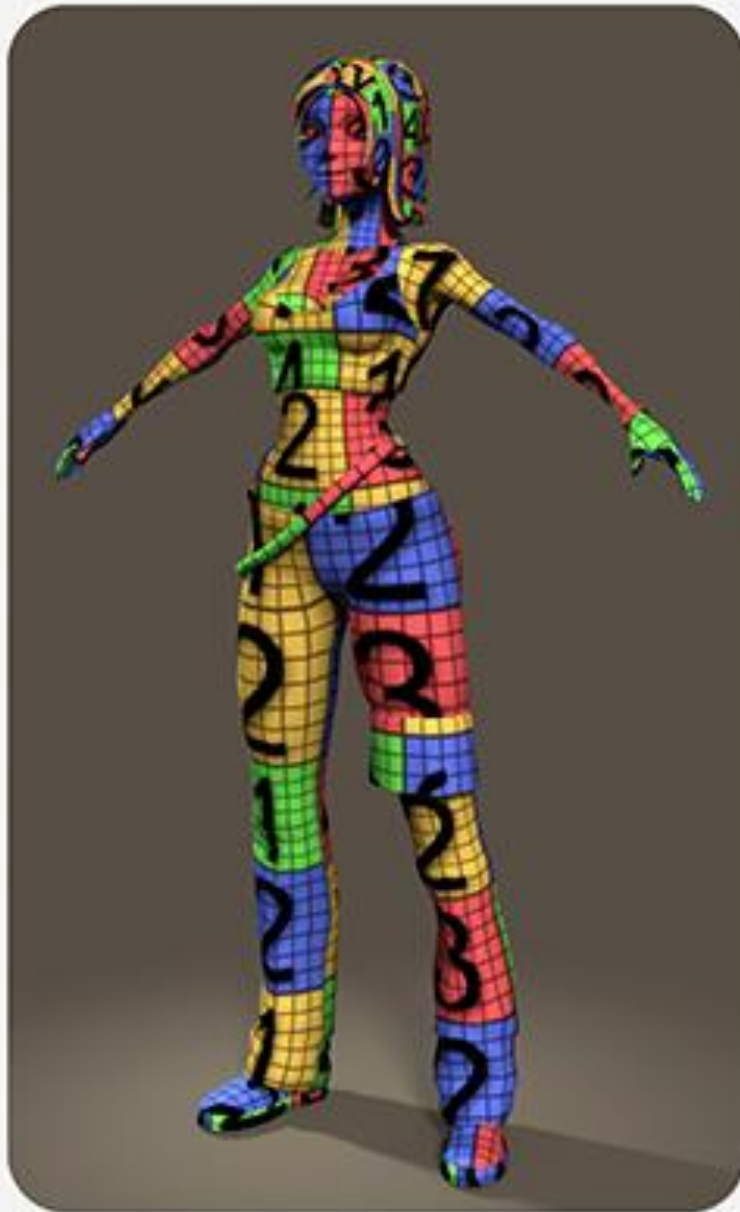
# SOLUCIÓN 1: DESDOBLAR LA SUPERFICIE





## SOLUCIÓN 2: ATLAS DE TEXTURA





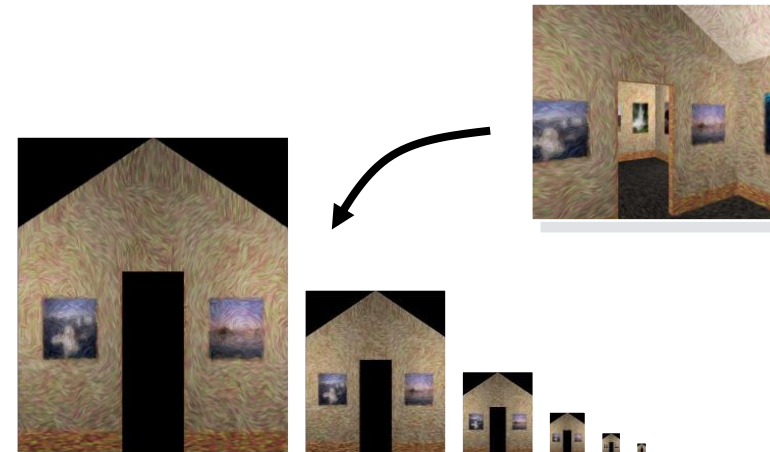
## **SOLUCIÓN 3: PROCESO ARTÍSTICO**

LO REALIZA UN EXPERTO EN  
MODELADO 3D Y TEXTURIZADO



# MIP MAPS

- Técnica que almacena la textura pre filtrada en múltiples resoluciones.
- En tiempo de ejecución, un algoritmo seleccionará la textura apropiada para ahorrar recursos computacionales, de acuerdo al número de píxeles en pantalla que correspondan a la geometría, considerando la posición y orientación de la cámara.
- Mip Map significa:
  - **MIP:** *Multum In Parvo* (Mucho En Poco)



## EJEMPLO DE USO DE MIP MAPS

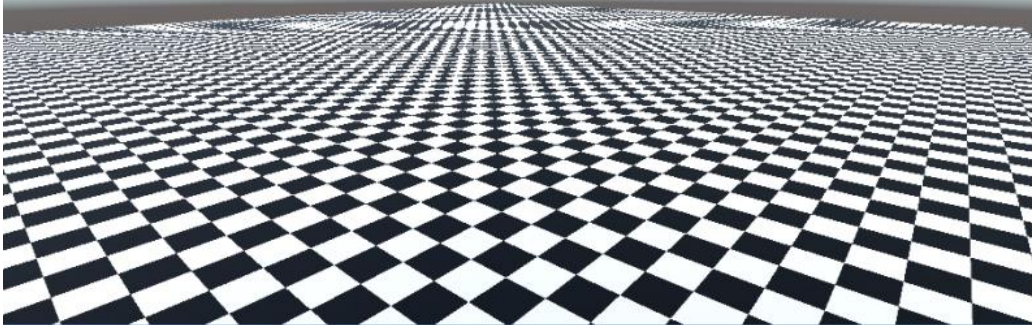
- Derecha: sin filtrado
- Izquierda: al usar **MipMaps**, desaparecen artefactos de distorsión no deseados (esto se descubrió por accidente al implementar el uso de MipMaps. De aquí procede la técnica de Anisotropic Filtering y anti aliasing).



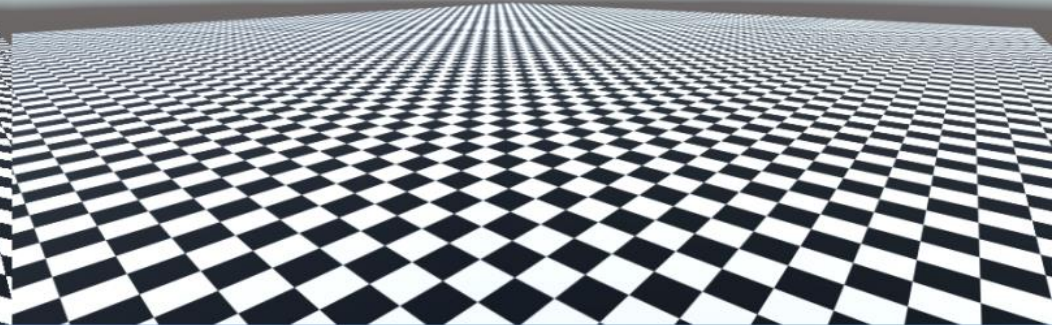




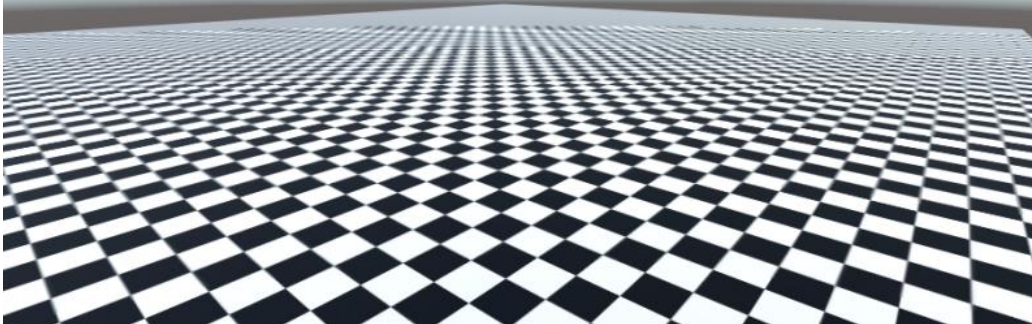
no MipMaps (=essentially Point Filtering)



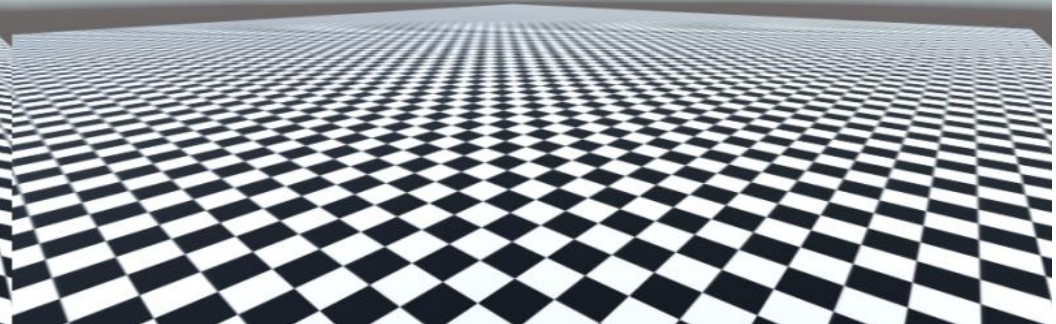
Anisotropic Filtering >0



MipMaps, bilinear



MipMaps, trilinear





# OTROS USOS DE TEXTURAS EN GRÁFICAS COMPUTACIONALES

- Al mapear una textura se puede alterar el color difuso, ambiental y especular.
- También es posible simular vectores normales, donde cada vector se almacena en un texel (se almacena x, y, z en lugar de r, g, b).

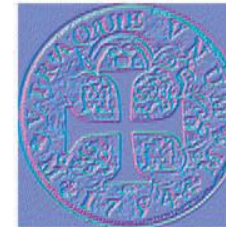
**TEXTURE MAPS**



**DIFFUSE (color) MAP**



**SPECULAR MAP**

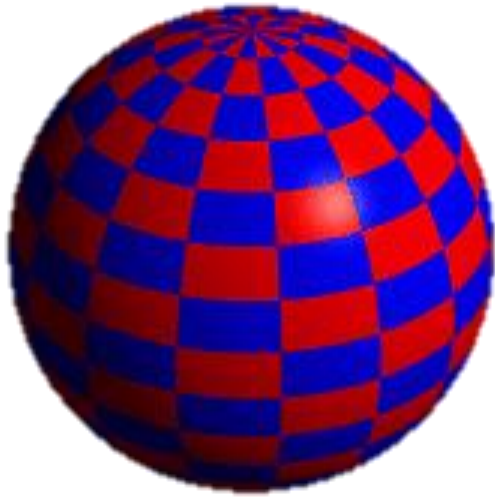


**NORMAL MAP**

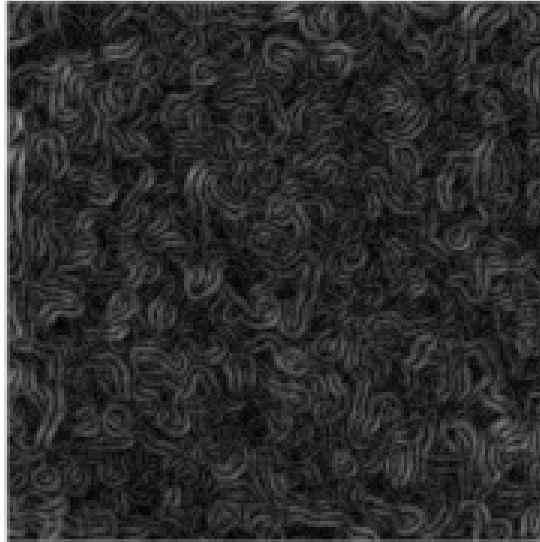
**FINAL**



## BUMP MAPPING (MAPA DE NORMALES)



*Esfera con textura difusa*

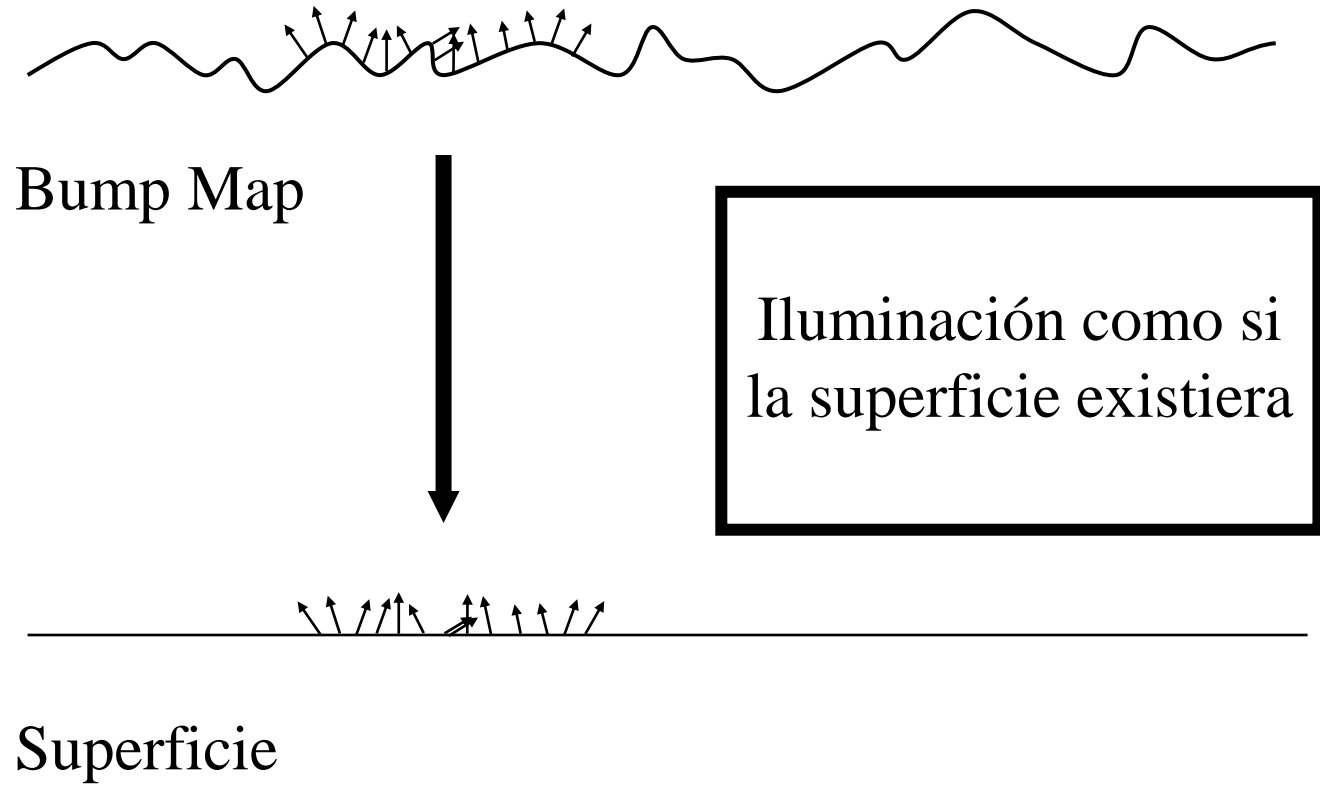


*Swirly bump map*



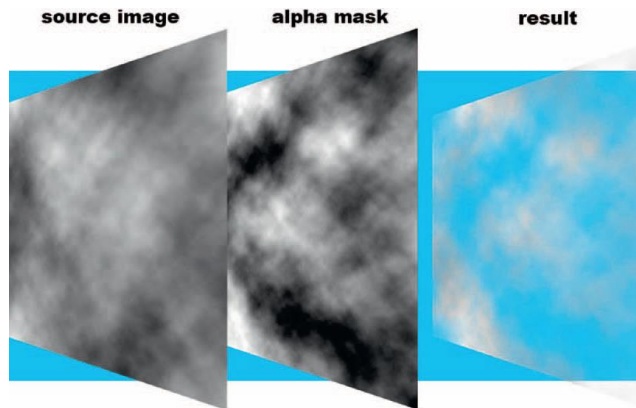
*Esfera con textura difusa  
y swirly bump map*

# BUMP MAPPING



# OTROS USOS DE TEXTURAS EN GRÁFICAS COMPUTACIONALES

- También se pueden determinar niveles de transparencia o mapear transparencias usando el canal Alpha.



**IMAGE**



**ALPHA**



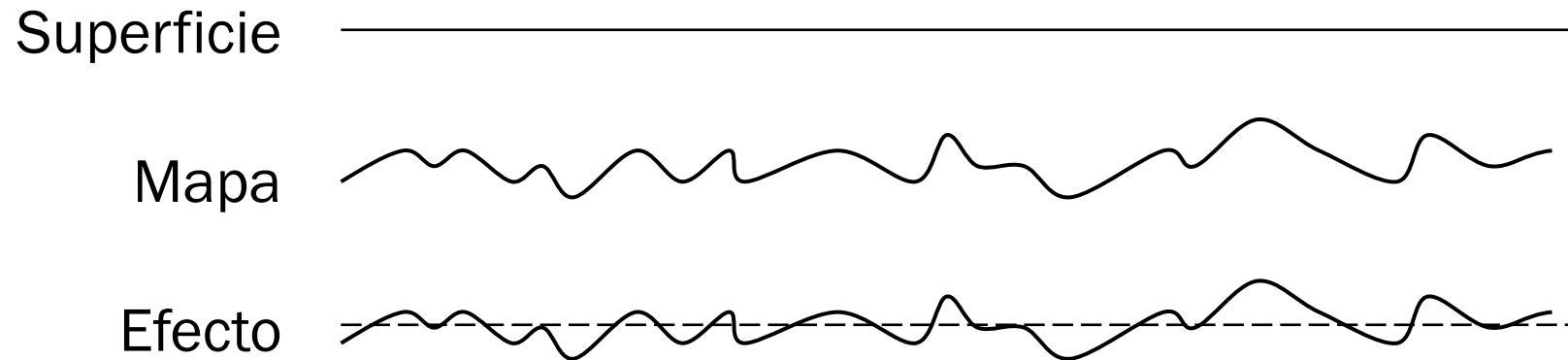
**IMAGE OVER TEXTURE**

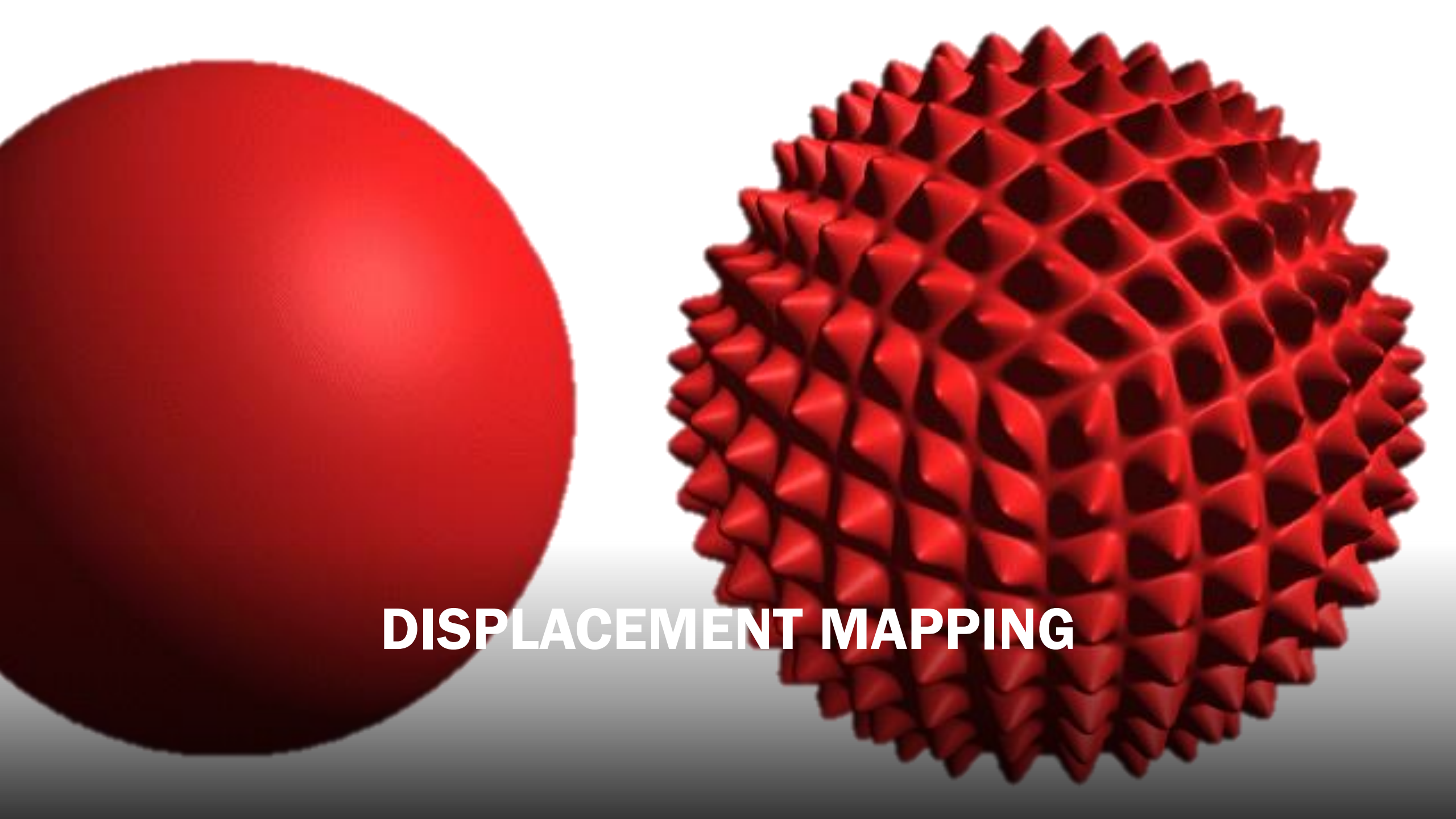




# DISPLACEMENT MAPPING (MAPA DE DESPLAZAMIENTO)

- La textura representa un offset (desplazamiento) en la dirección normal.
- “Difícil” de implementar.
- Alto realismo.
- Más caro computacionalmente que un mapa de normales.

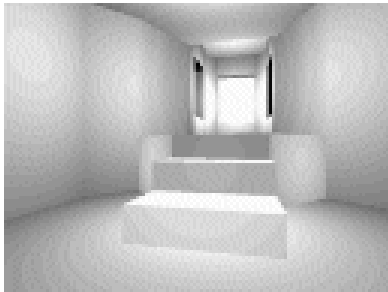




**DISPLACEMENT MAPPING**

# LIGHT MAPPING

- Quake® introduce lo que se conoce como *illumination maps* o *light maps* para capturar efectos de iluminación.



*Light map*

Textura  
+ light map





# ENVIRONMENTAL MAPPING

- Renderiza un objeto como si fuera perfectamente reflejante.
- Lo que vemos depende de la posición del ojo y de las normales a la superficie.
- James F. Blinn 1976.



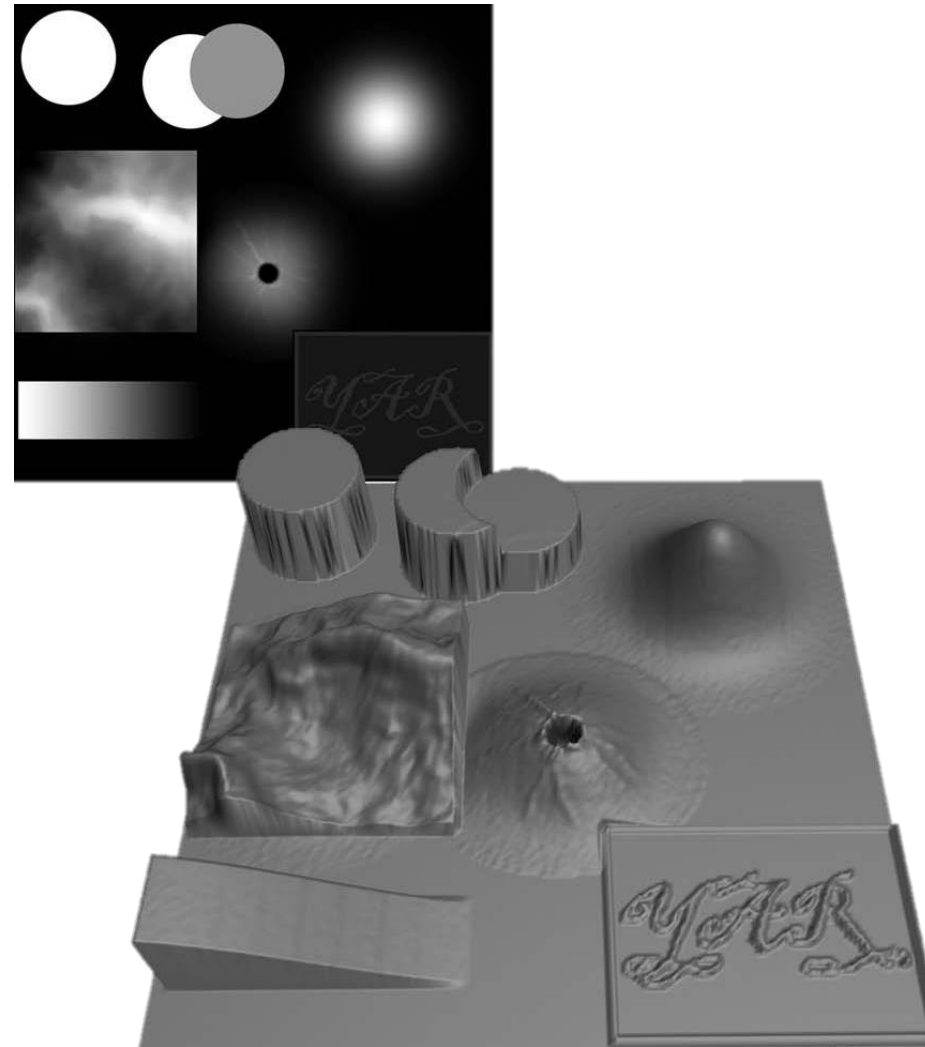


# ENVIRONMENTAL MAPPING



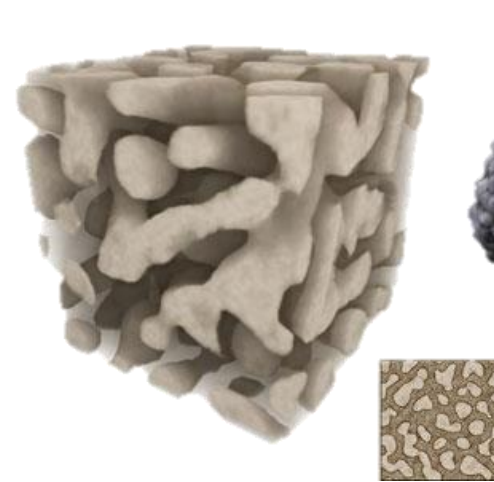
# HEIGHT MAPPING (MAPEO DE ALTURAS)

- Usado para terrenos.
- Los colores representan alturas.
- Soportan multi resolución.
- Pueden combinarse con texturas procedurales.

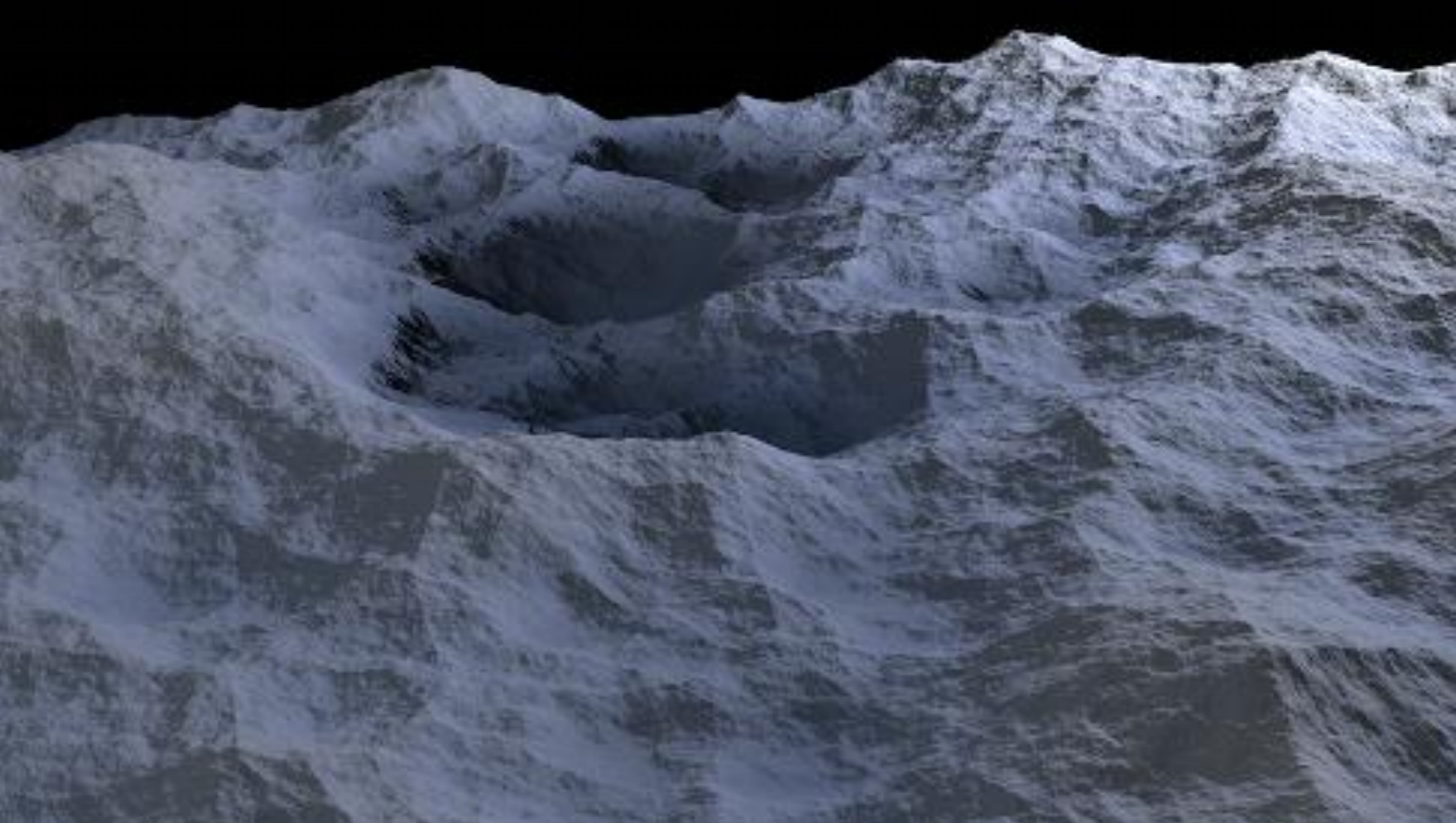


# TEXTURAS SÓLIDAS

- Valores de Textura indizados por localización en 3D (x,y,z):
  - Tamaño de archivo grande.
  - Se pueden calcular en tiempo de ejecución: texturas procedurales como Perlin noise.







**TEXTURAS PROCEDURALES**