

Transformaciones y visualización 3D

Transformaciones 3D

El proceso de visualización

Proyecciones

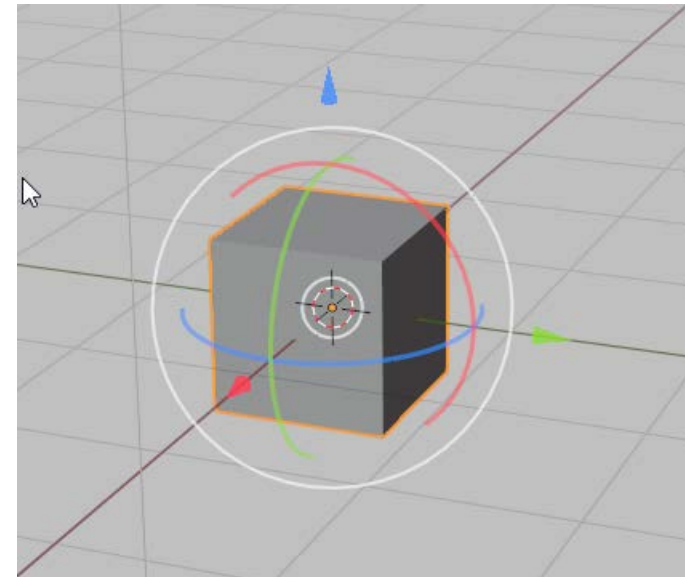
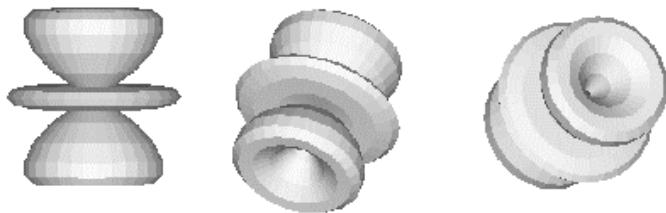
Modelo de cámara

Matrices de proyección

Recortado 3D

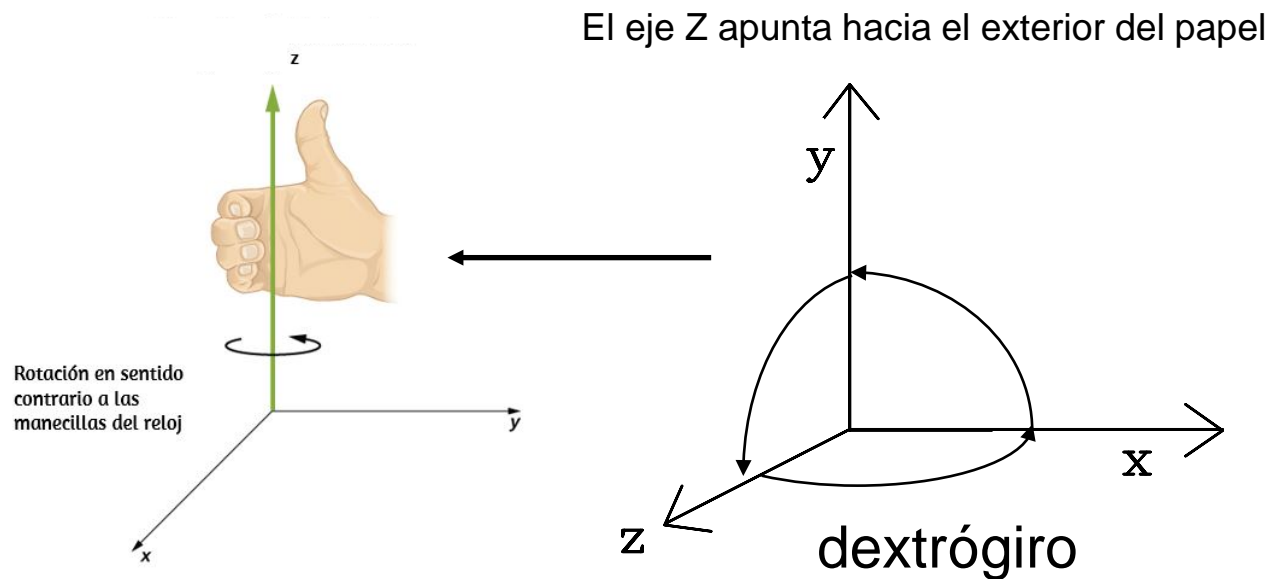
Transformaciones 3D

- ▶ Las transformaciones en 3D se utilizan para:
 - ▶ Manipular objetos en el espacio 3D
 - ▶ *Traslación, giro y escalado*
 - ▶ Ayuda para visualizar y examinar objetos

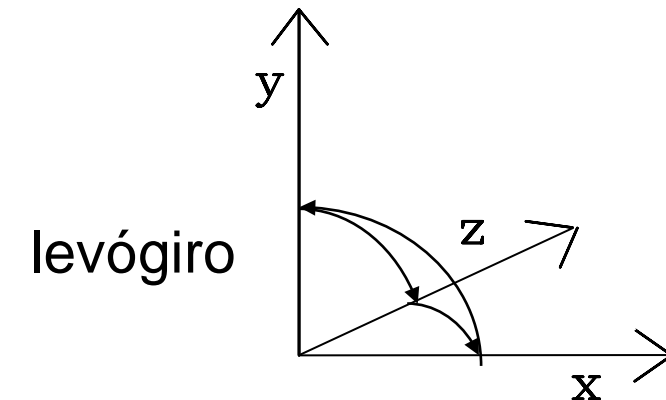


Transformaciones 3D

- ▶ Los sistemas de coordenadas pueden ser
 - ▶ *Dextrógiro* (el que utilizaremos para realizar las transformaciones)
 - ▶ *Levógiro* (se utiliza tras realizar la proyección)



El eje Z apunta hacia el interior del papel



Transformaciones 3D

- ▶ Al igual que en 2D se utilizan las coordenadas homogéneas
 - ▶ Un punto 3D (x, y, z) se representa por (x, y, z, w) y como vector columna
 - ▶ Las matrices de transformación serán, por tanto, 4x4
 - ▶ Las matrices premultiplicarán a los puntos

$$\begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix} \bullet \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}$$

Transformaciones 3D

► Traslación

$$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

► Escalado

$$\begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

○ Rotación

● Eje X

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

● Eje Y

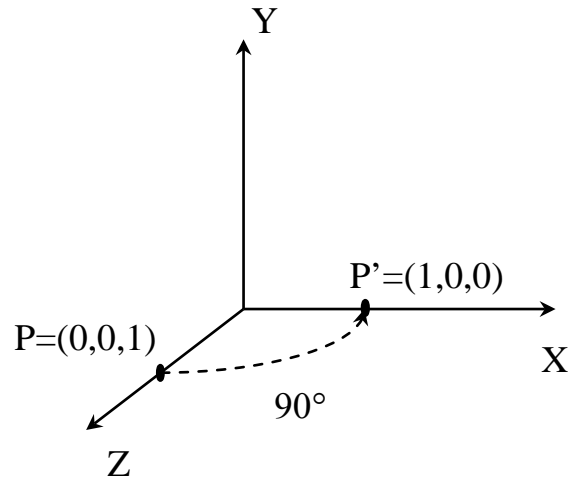
$$\begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

● Eje Z

$$\begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Transformaciones 3D

► Ejemplo

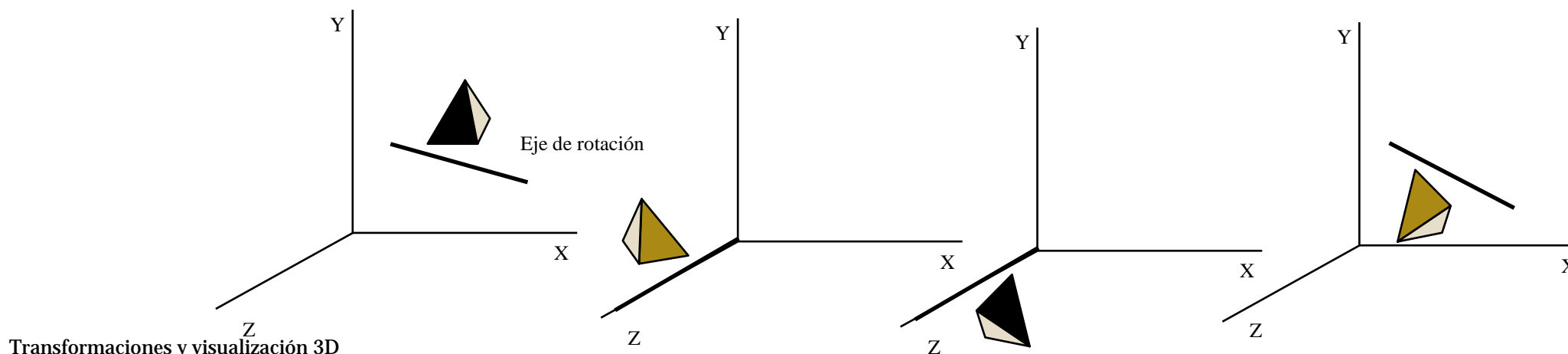


$$\mathbf{R}_y(90) = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P' = \mathbf{R}_y(90) \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

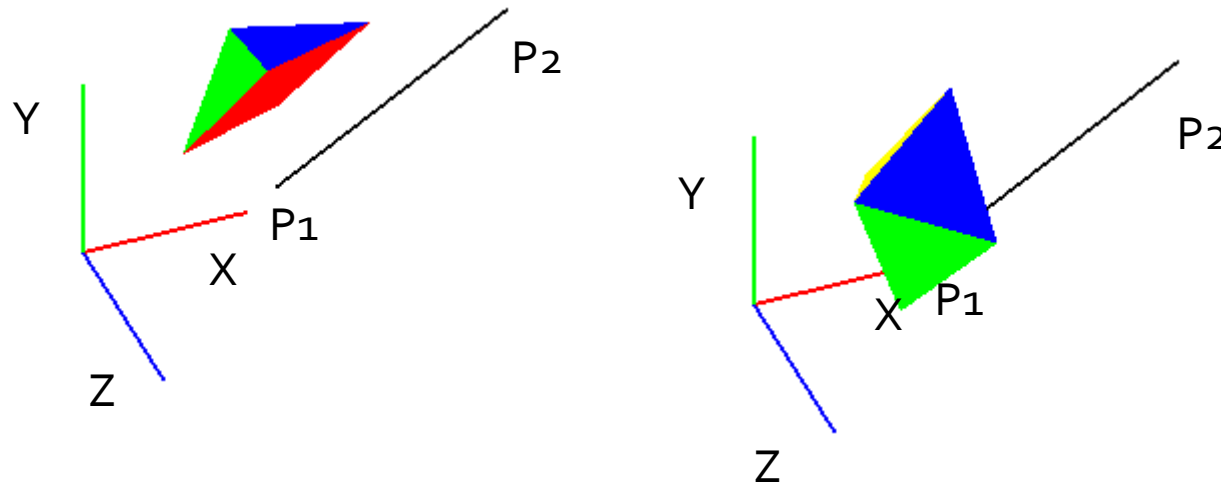
Transformaciones 3D

- ▶ Para realizar una rotación de α grados respecto a un eje cualquiera, se deben realizar los siguientes pasos:
 - ▶ Traslación para que el eje pase por el origen
 - ▶ Rotar el eje para que coincida con uno de los ejes de coordenadas
 - ▶ Realizar la rotación de α grados alrededor del eje anterior
 - ▶ Aplicar las rotaciones inversas para que el eje vuelva a su orientación original
 - ▶ Aplicar la traslación inversa para que el eje vuelva a su posición original



Transformaciones 3D

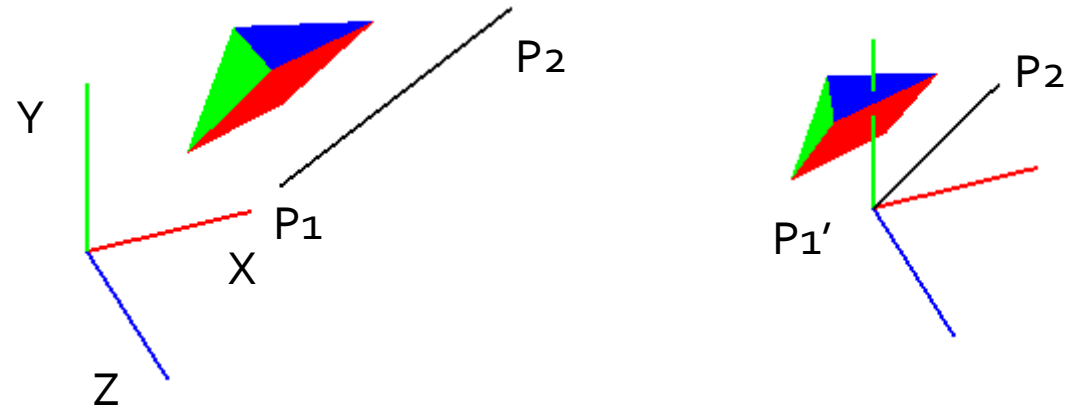
- **Ejemplo:** Girar α grados alrededor de un eje definido por los puntos P_1 y P_2 :



Transformaciones 3D

Ejemplo: Girar α grados alrededor de un eje definido por los puntos P1 y P2:

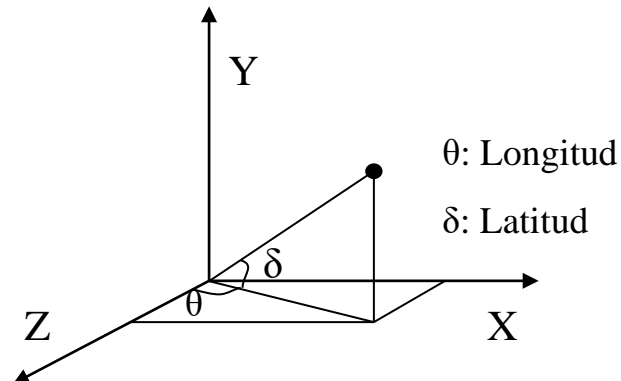
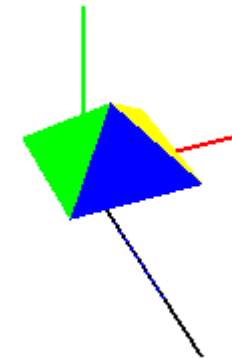
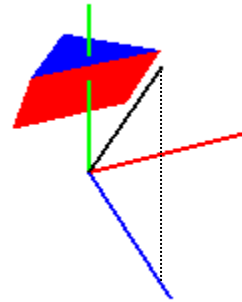
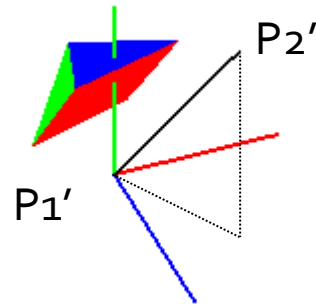
1. Trasladar el eje hasta el origen de coordenadas



Transformaciones 3D

2. Girar el eje para que coincida con uno de los ejes principales (p.ej. En la figura el eje Z)

1. Girar alrededor del eje Y de manera que P1P2 coincida con el plano YZ
2. Girar alrededor del eje X de manera que P1P2 coincida con el eje Z



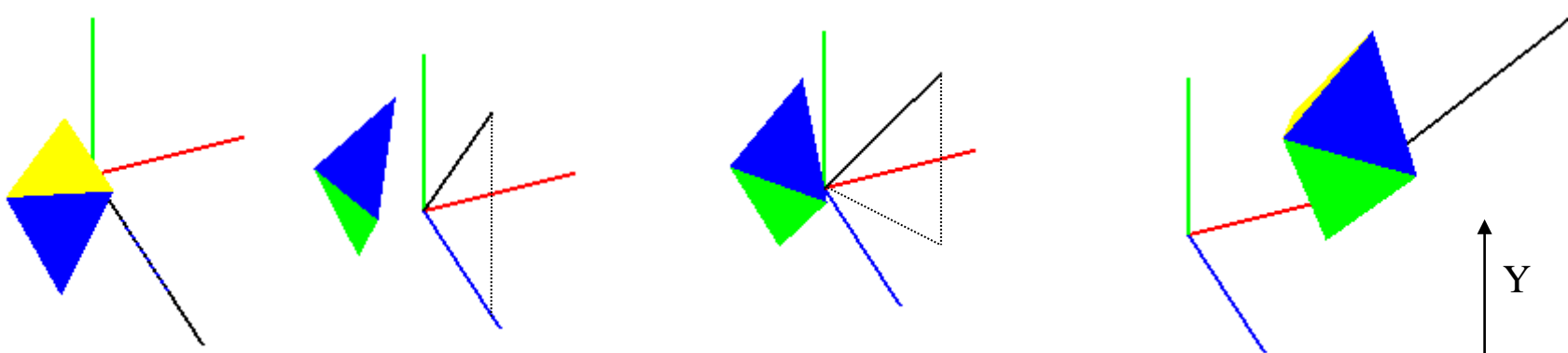
Transformaciones 3D

3. Girar α grados alrededor del eje de coordenadas (eje Z)

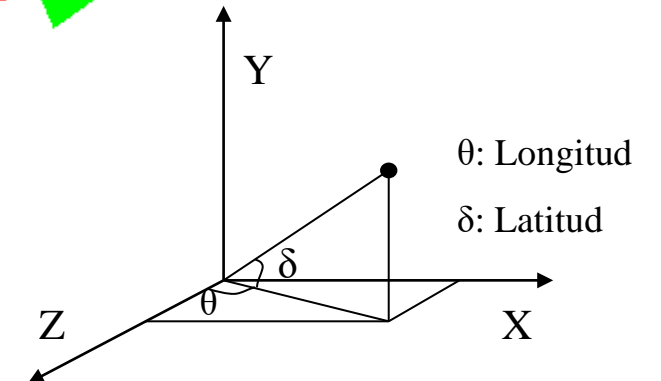


Transformaciones 3D

4. Girar el eje hasta llevarlo a su orientación original (transformaciones inversas a las del paso 2)
5. Trasladar el eje para devolverlo a su posición original (traslación inversa a la del paso 1)



$$P' = T(P1) * R_Y(\text{long}) * R_X(-\text{lat}) * R_Z(\alpha) * R_X(\text{lat}) * R_Y(-\text{long}) * T(-P1) * P$$



Explicado en este screencast:

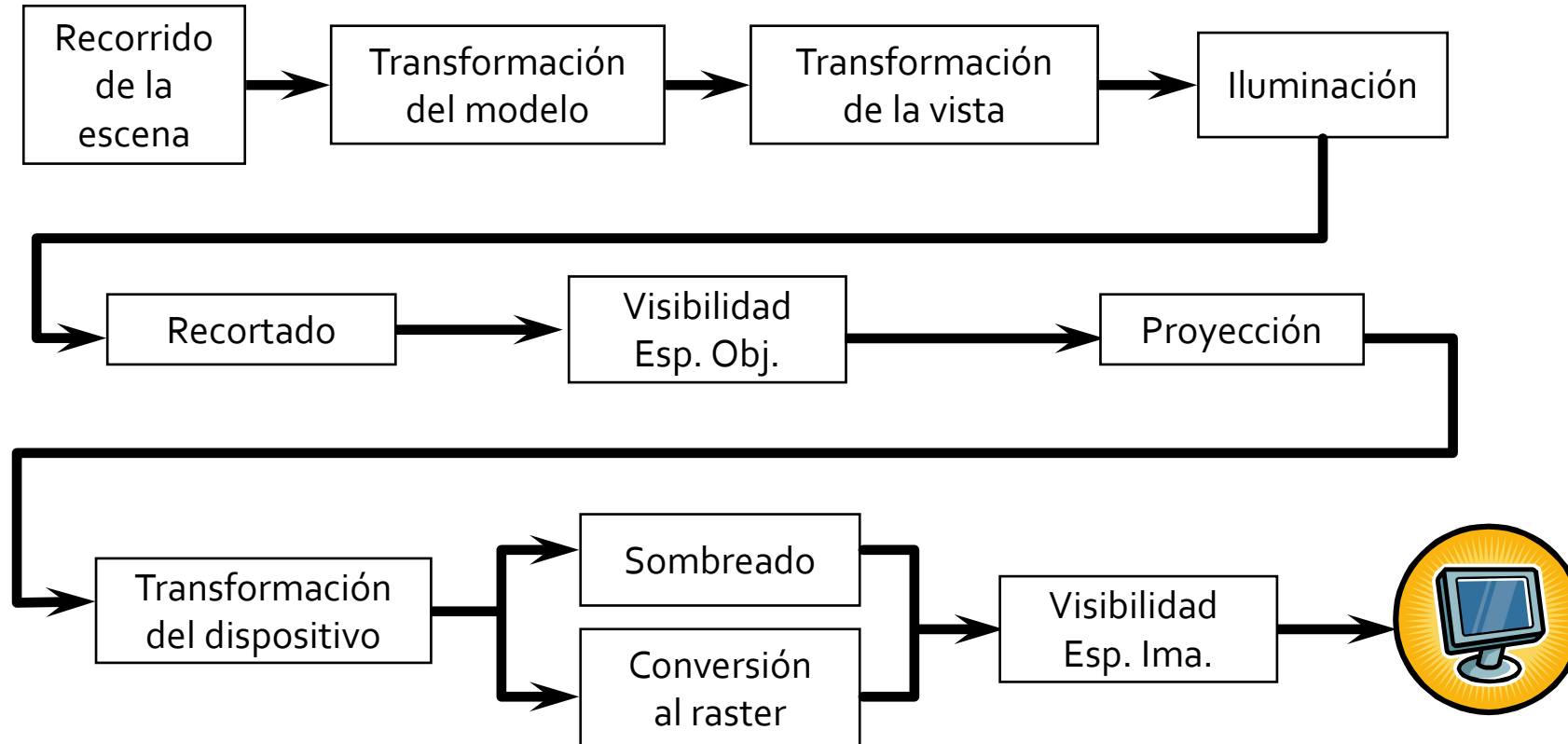
<https://media.upv.es/player/?id=b57be490-1219-11eb-ab27-e1603a595a55>

El proceso de visualización 3D

- El proceso de síntesis de una imagen (proceso de visualización) es el conjunto de operaciones (en 3D y en 2D) sobre un modelo informático de datos que resultan en una representación gráfica del mismo en un dispositivo físico de representación

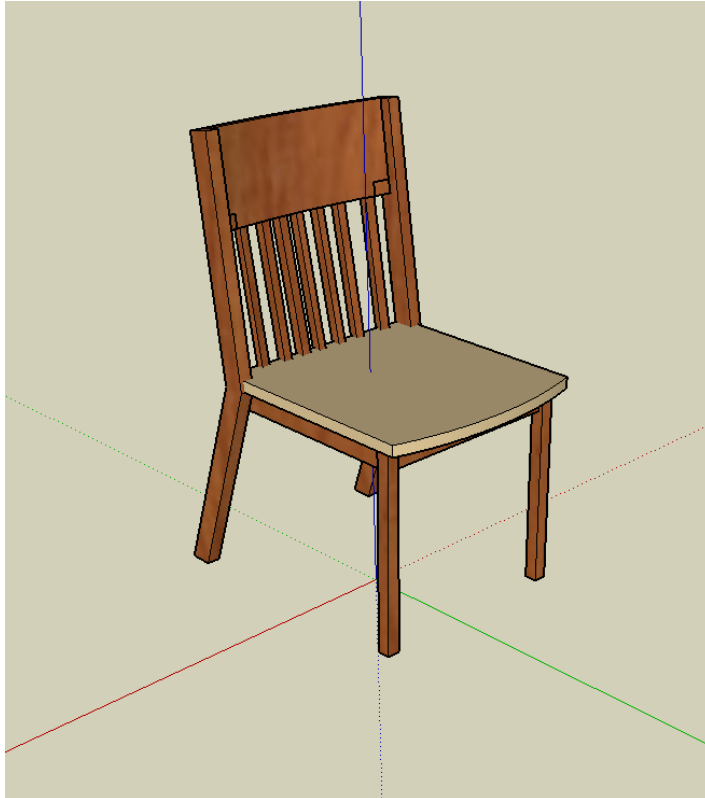
El proceso de visualización

Pipeline 3D



El proceso de visualización

Pipeline 3D. Recorrido de la escena

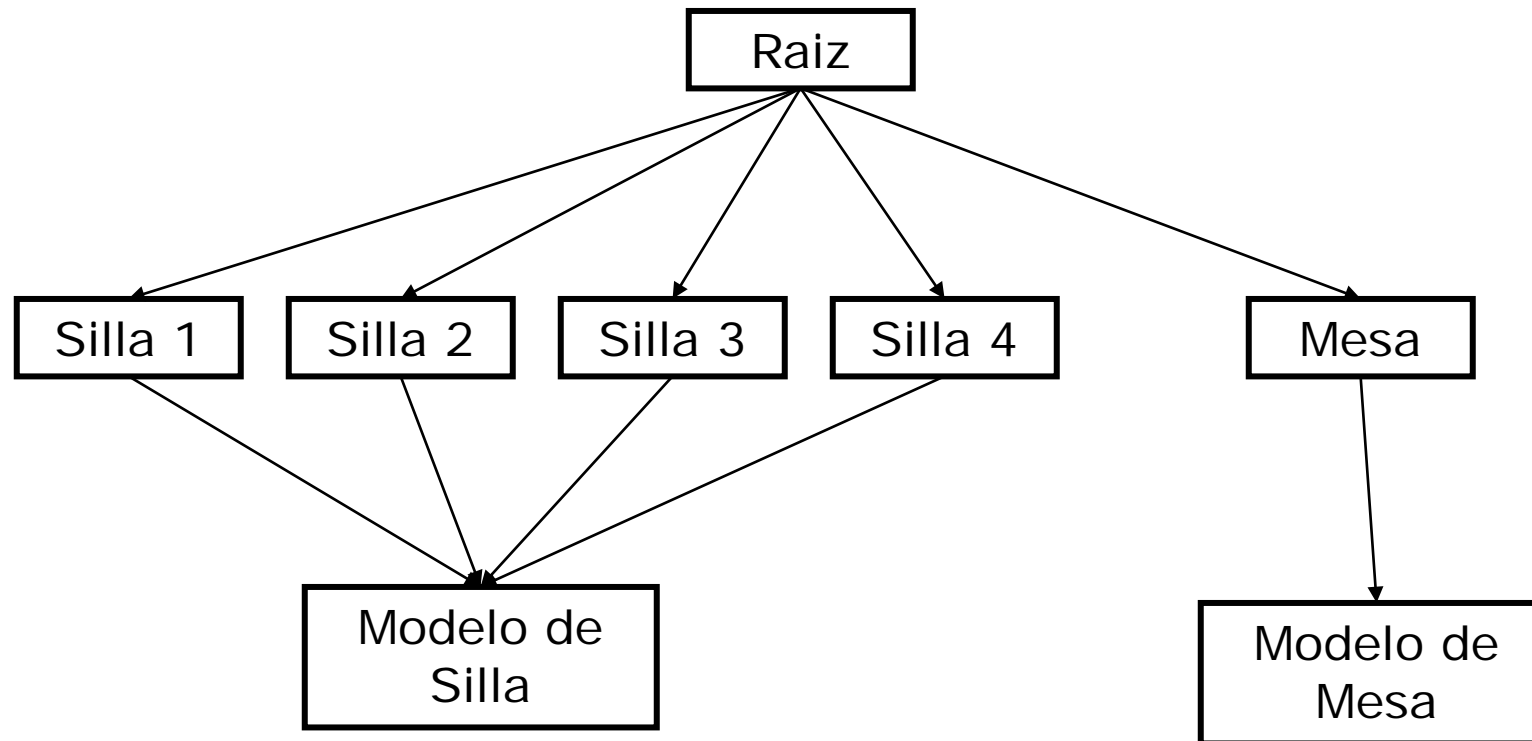


Cada objeto se modela en su propio sistema de coordenadas

El proceso de visualización

Pipeline 3D. Recorrido de la escena

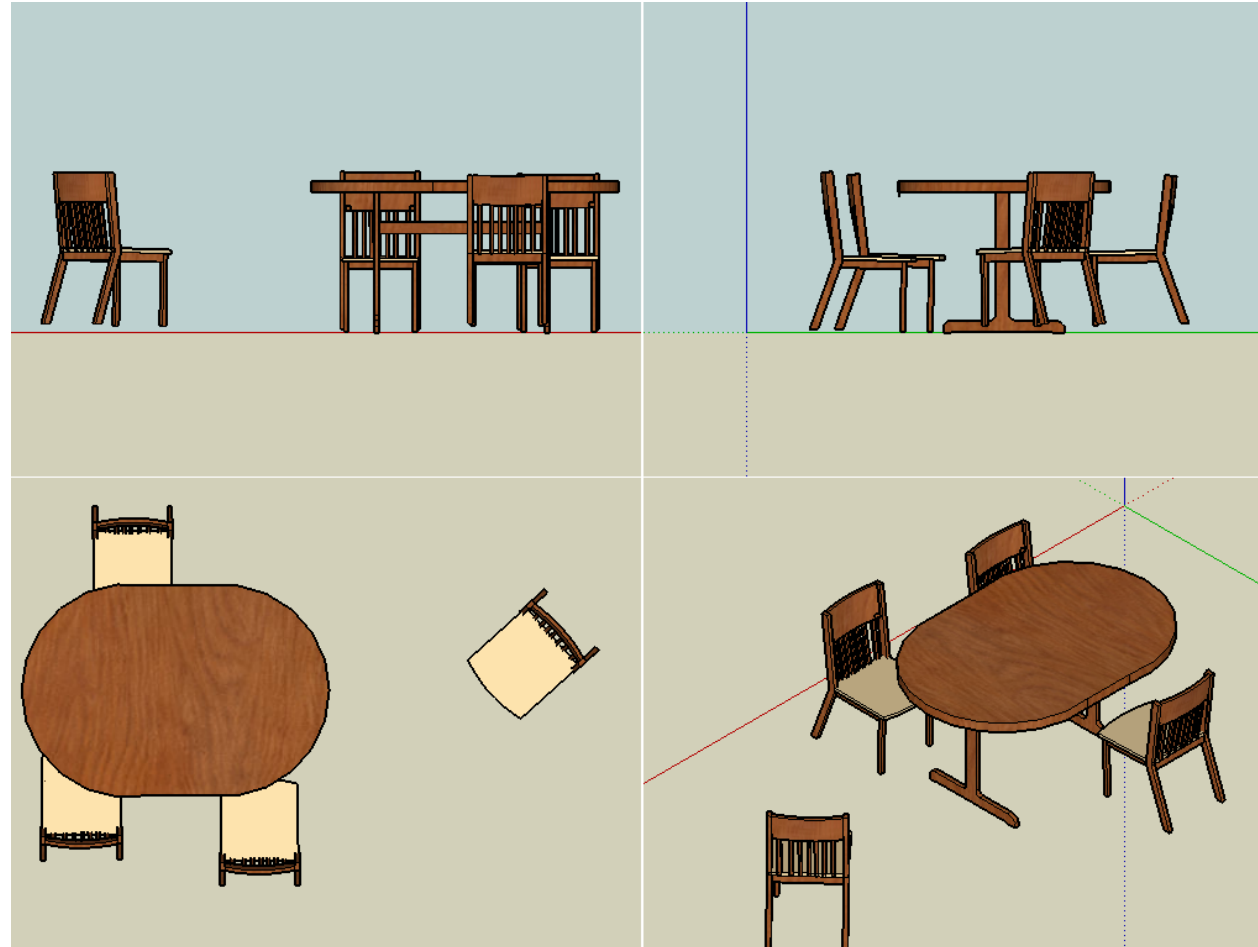
Normalmente la escena se representa mediante un grafo:



El proceso de visualización

Pipeline 3D. Transformación del modelo

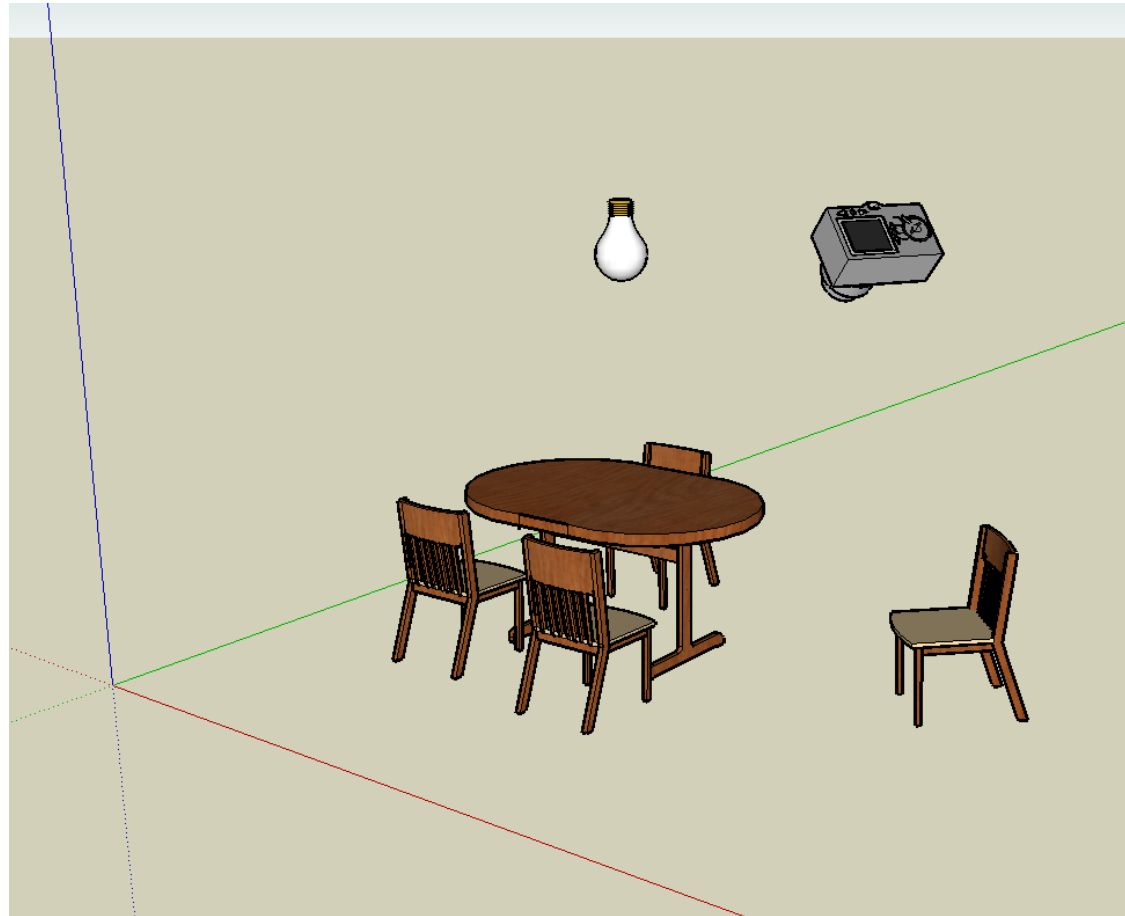
Los objetos se distribuyen en la escena:



El proceso de visualización

Pipeline 3D. Transformación del modelo

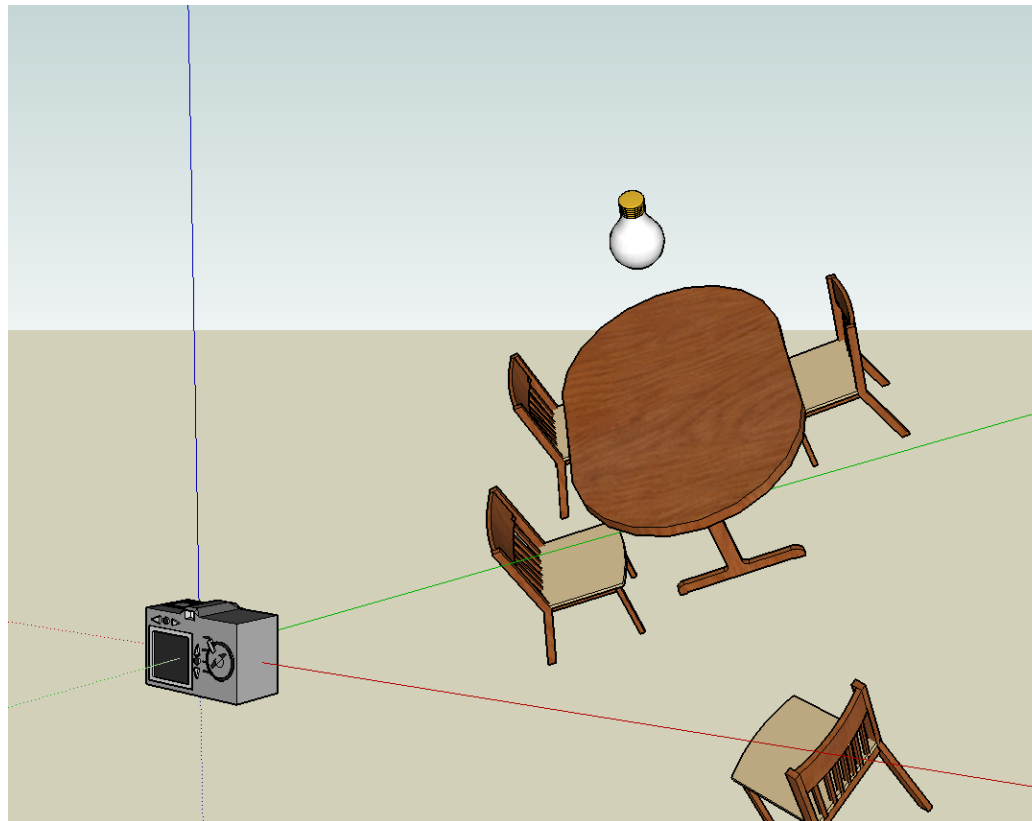
La cámara y luces también forman parte de la escena:



El proceso de visualización

Pipeline 3D. Transformación de la vista

- ▶ La escena se transforma a una posición estándar:
 - ▶ La cámara se mueve al origen
 - ▶ El vector de la vista se sitúa sobre el eje Z, la vertical es Y



El proceso de visualización

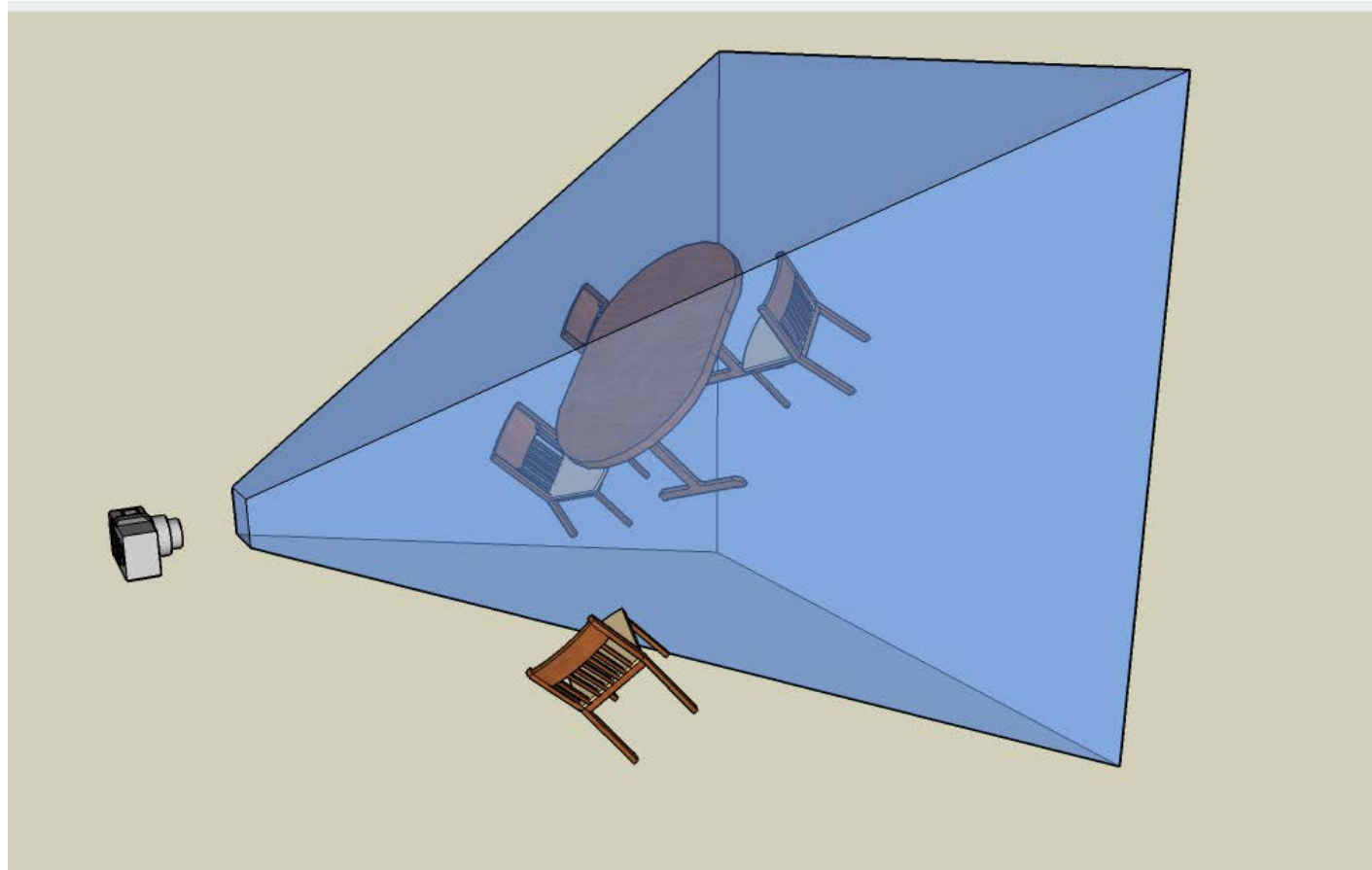
Pipeline 3D. Iluminación

- ▶ Cada objeto está definido por:
 - ▶ Geometría (una serie de vértices)
 - ▶ Propiedades de color
- ▶ Las fuentes de luz están definidas por:
 - ▶ El color e intensidad de la luz que emiten
 - ▶ Su posición
- ▶ En esta etapa se calcula el color que debe tener cada vértice, teniendo en cuenta todos los parámetros anteriores

El proceso de visualización

Pipeline 3D. Recortado

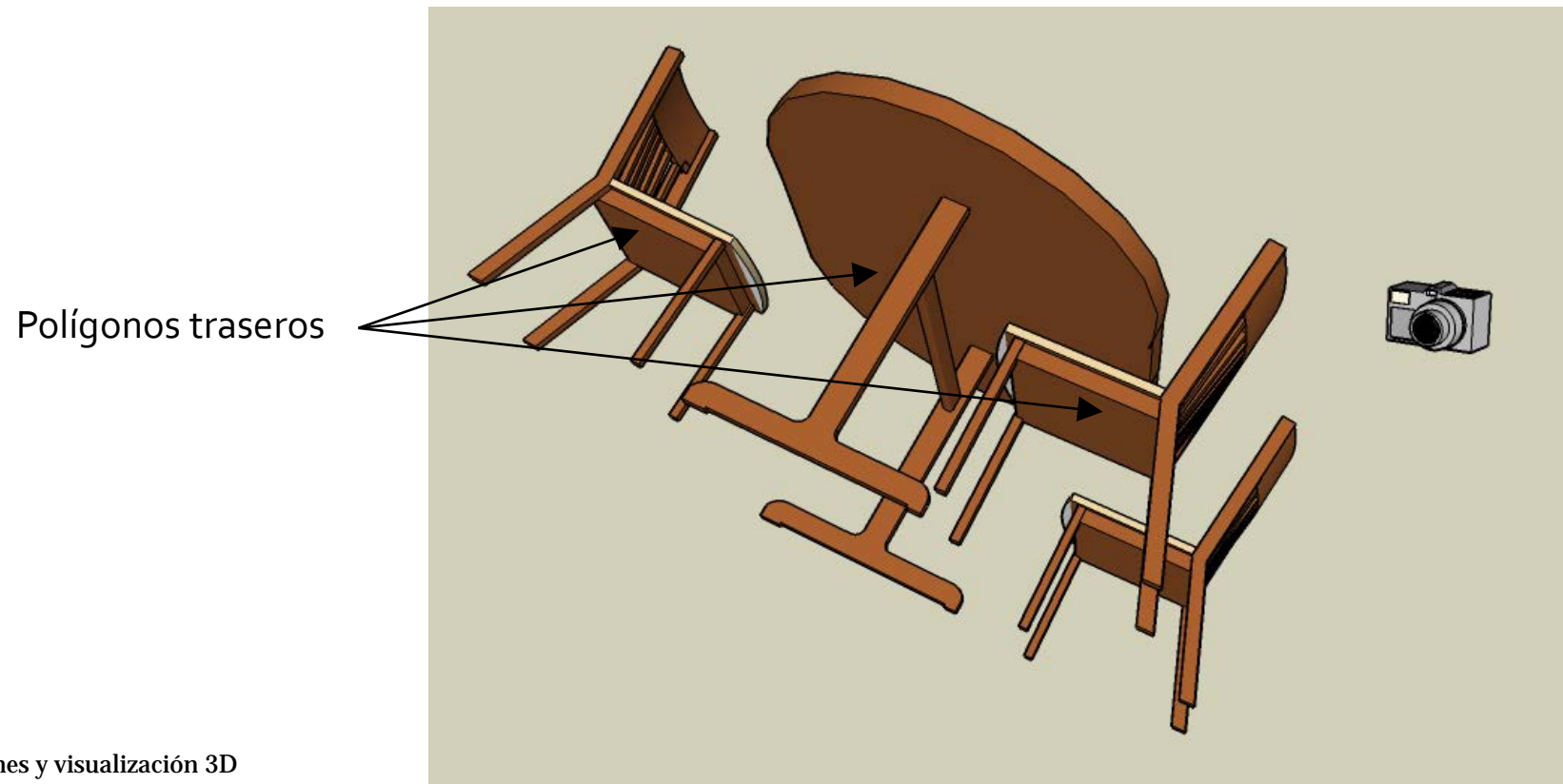
Se define el volumen de la vista y se recortan los objetos que quedan en el exterior de dicho volumen.



El proceso de visualización

Pipeline 3D. Visibilidad

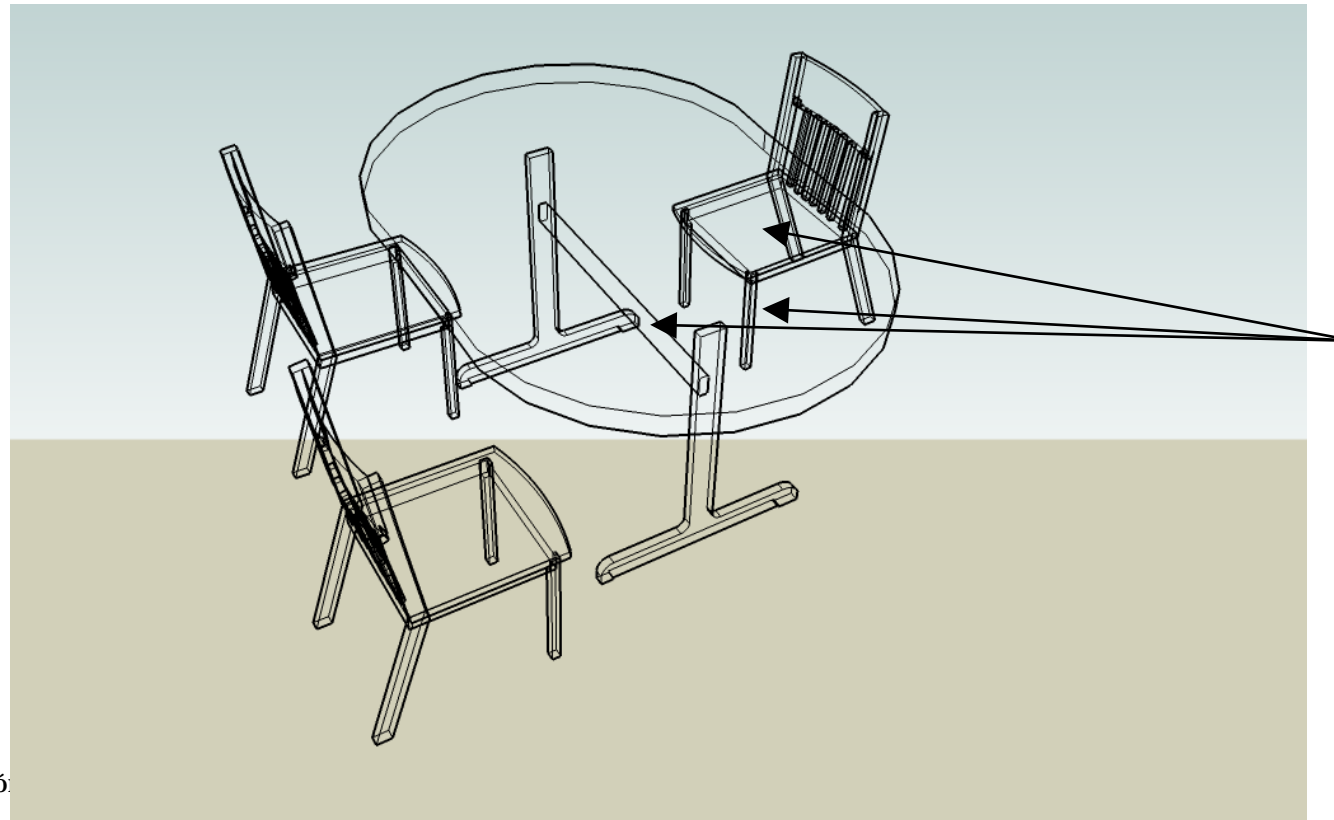
- ▶ Se eliminan los objetos de la escena que no se pueden ver:
 - ▶ Tapados por partes del propio objeto



El proceso de visualización

Pipeline 3D. Visibilidad

- ▶ Se eliminan los objetos de la escena que no se pueden ver:
 - ▶ Tapados por otros objetos

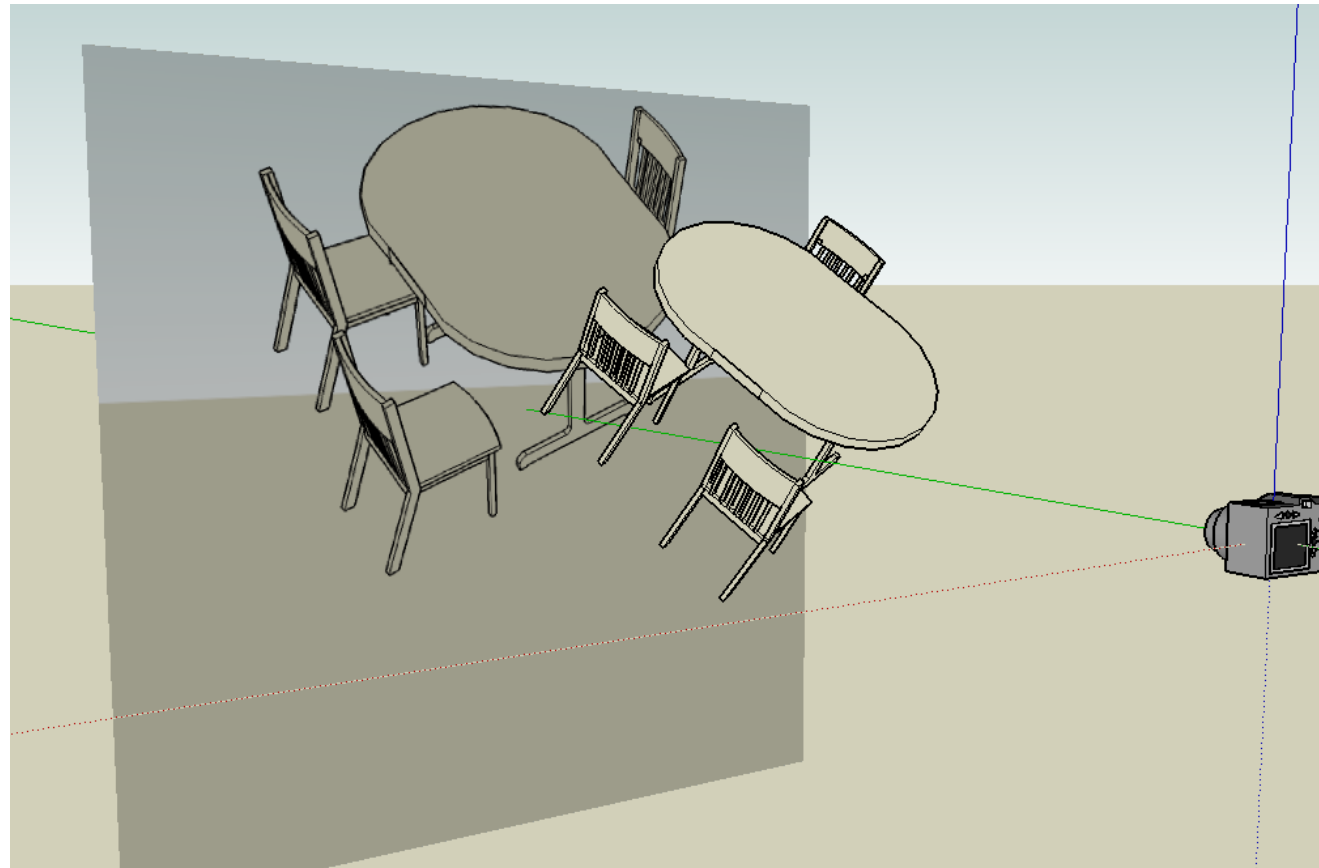


Estos polígonos no se pueden ver desde la cámara

El proceso de visualización

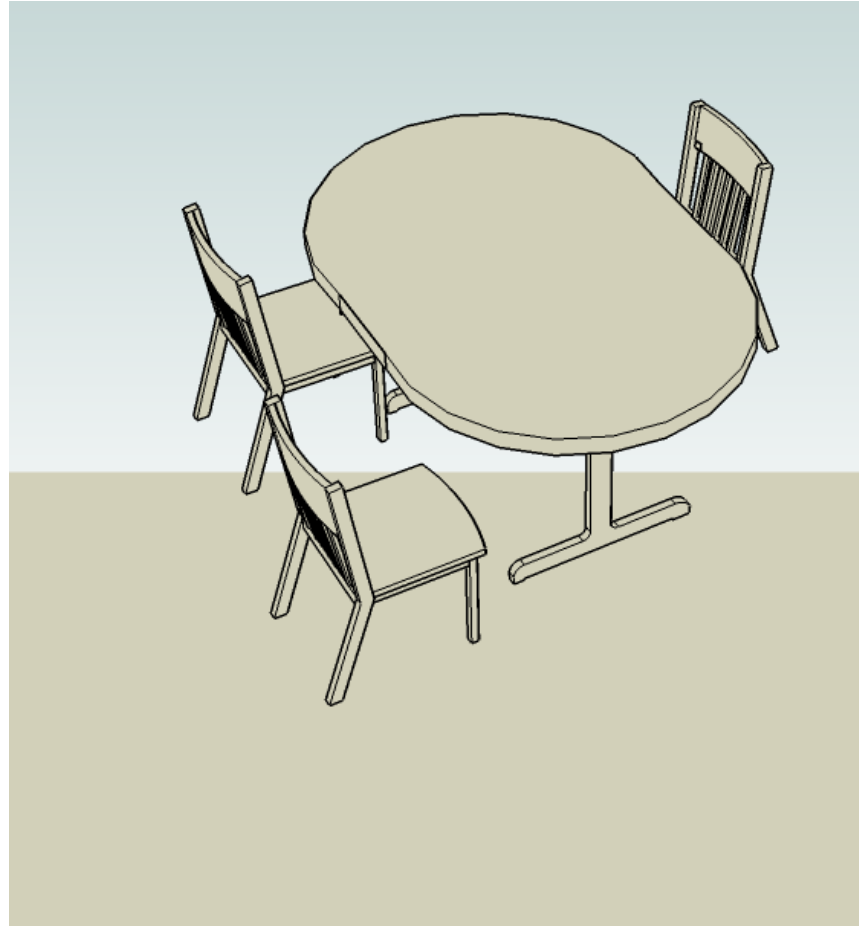
Pipeline 3D. Proyección

Proyección: Convierte un espacio 3D en uno 2D

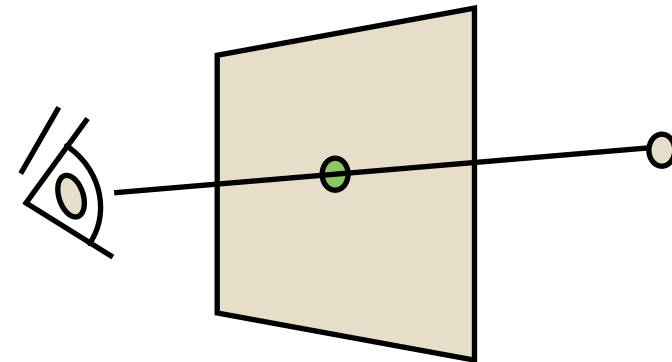


El proceso de visualización

Pipeline 3D. Proyección



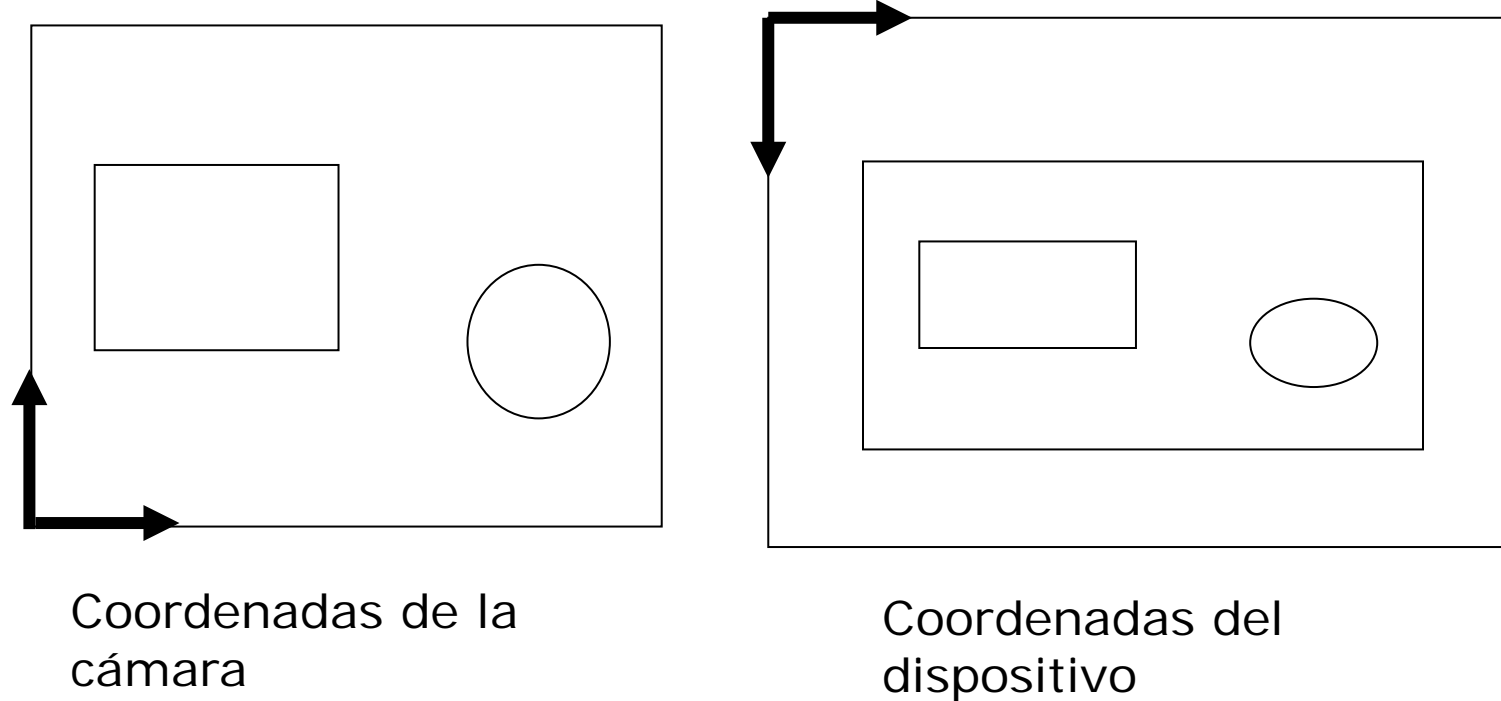
(esto todavía es un conjunto de objetos matemáticos, pero ahora son 2D)



El proceso de visualización

Pipeline 3D. Transformación del dispositivo

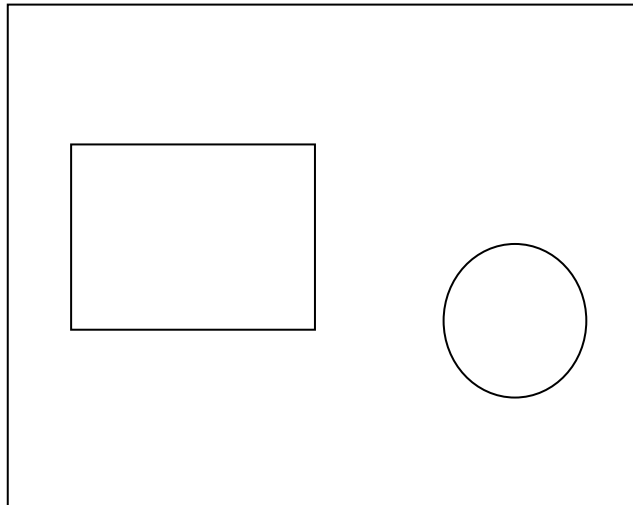
Adaptar el sistema de coordenadas de la vista al sistema de coordenadas del dispositivo



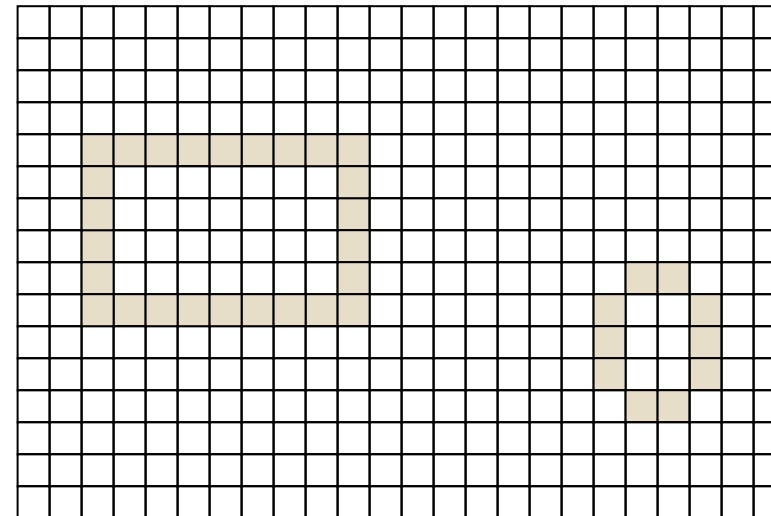
El proceso de visualización

Pipeline 3D. Conversión al raster

Convertir un conjunto de primitivas matemáticas 2D en píxeles



Primitivas proyectadas



Raster

El proceso de visualización

Pipeline 3D. Sombreado

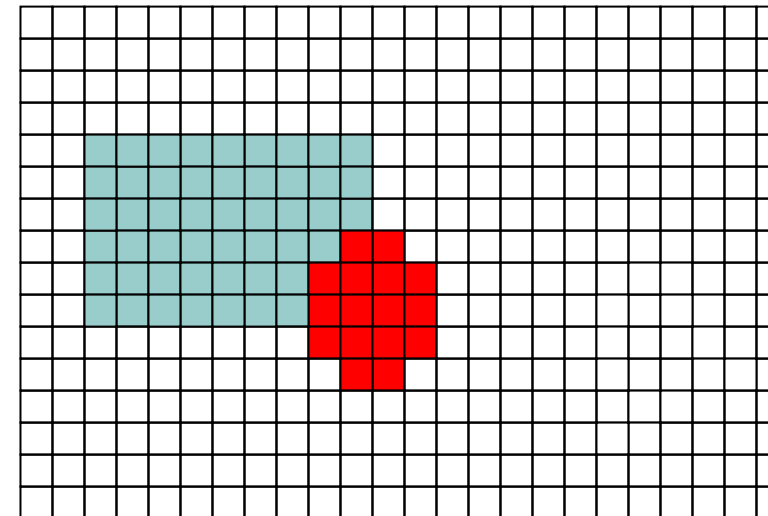
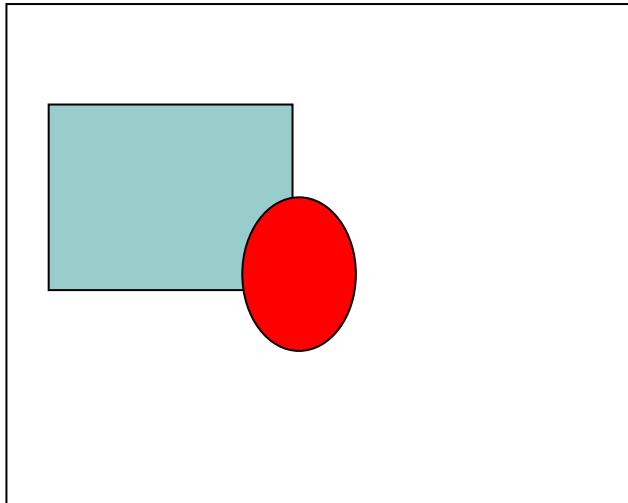
Decidir el color de cada píxel de un polígono en función del color de sus vértices.



El proceso de visualización

Pipeline 3D. Visibilidad de píxeles

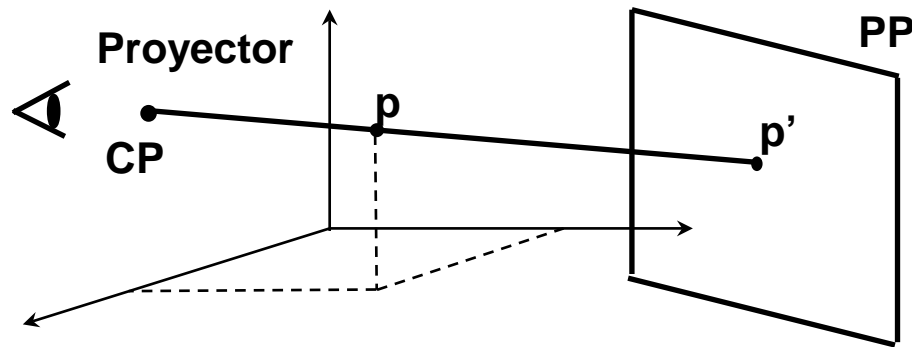
Para todos los píxeles de la imagen, se selecciona el color del objeto más cercano



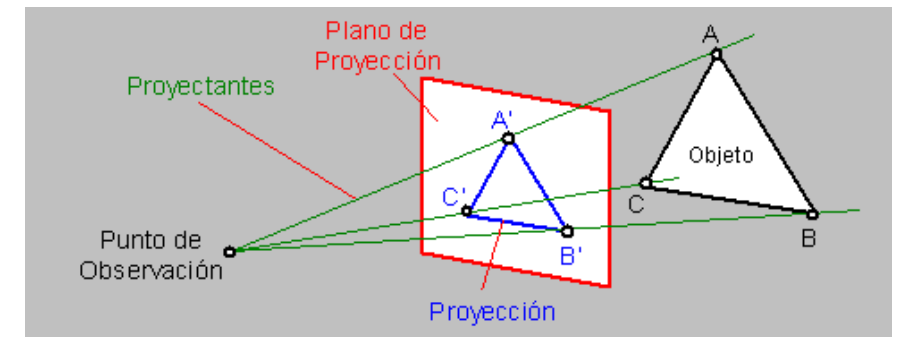
Raster

Proyecciones

- Proyección: conversión de 3D a 2D

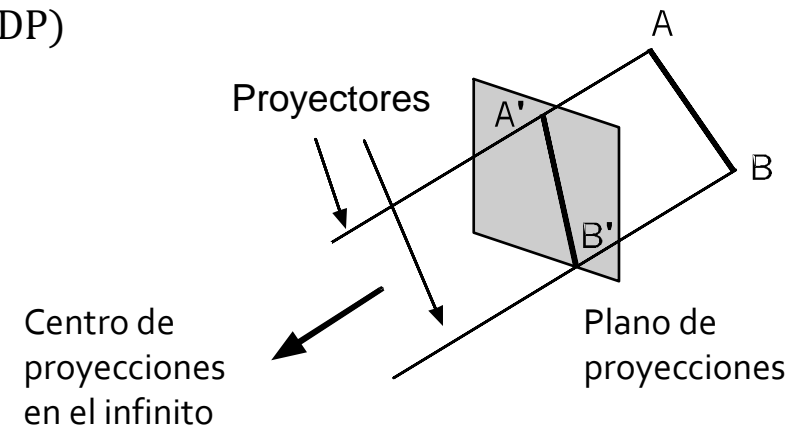
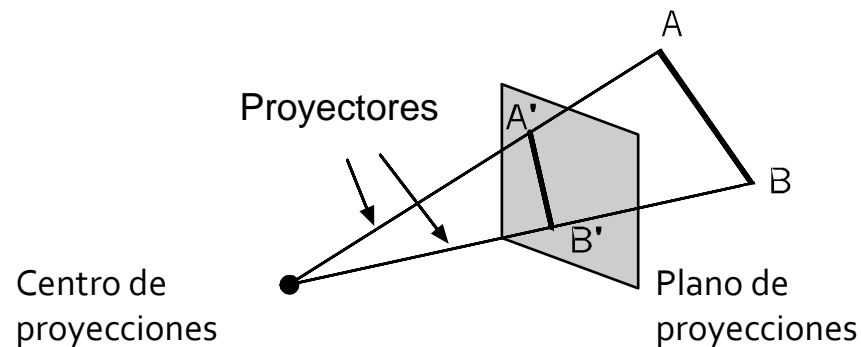


PP: Plano de proyección
CP: Centro de proyección
p: Punto en 3D
p': Proyección de p
(intersección entre la visual y PP)



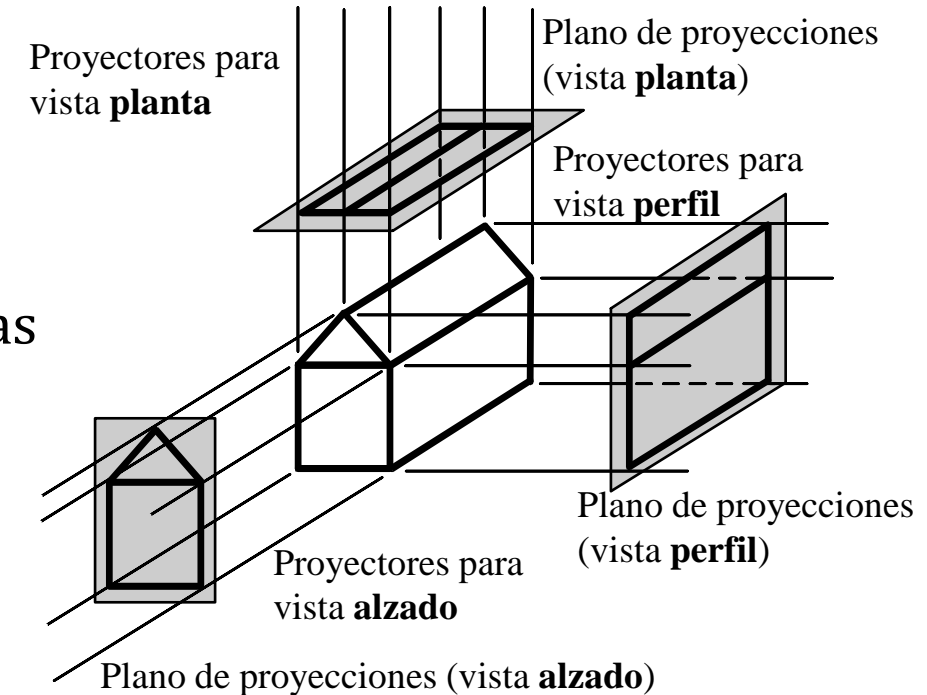
- Dos tipos de proyecciones:

- Perspectiva: definida por un Centro de Proyecciones (CP)
- Paralela: definida por una Dirección de Proyección (DP)



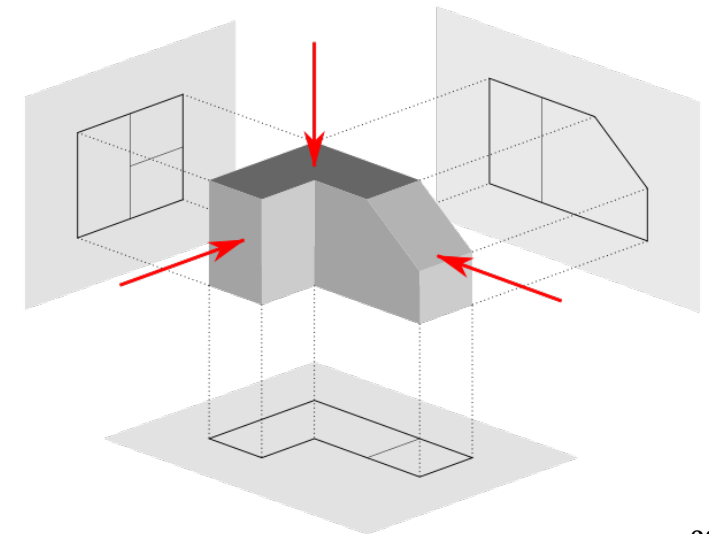
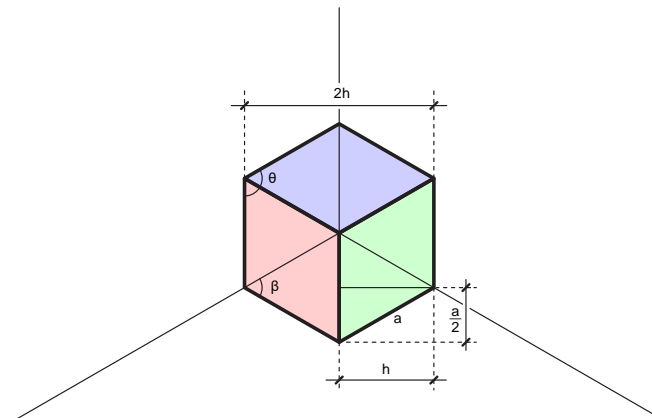
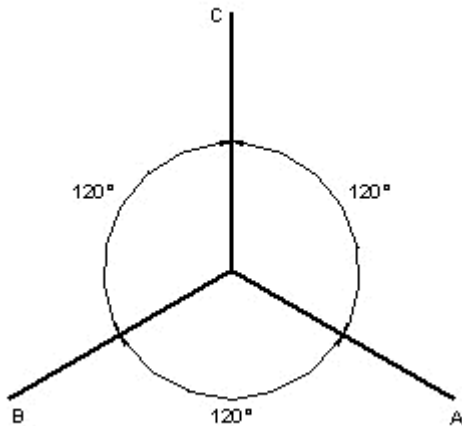
Tipos de proyecciones paralelas

- ▶ **Multivista ortográfica: $DP \perp PP$**
 - ▶ **PLANTA:** $PP \perp \text{eje Y.}$
 - ▶ **ALZADO:** $PP \perp \text{eje Z.}$
 - ▶ **PERFIL:** $PP \perp \text{eje X.}$
 - ▶ Se utilizan en:
 - ▶ diseños de ingeniería: máquinas, partes de máquinas
 - ▶ planos de arquitectura
 - ▶ **Ventajas:**
 - ▶ es posible realizar medidas precisas
 - ▶ todas las vistas tienen la misma escala
 - ▶ **Inconvenientes:**
 - ▶ no proporciona una visión realista de los objetos 3D.
Generalmente es necesario varias vistas para percibir las 3D



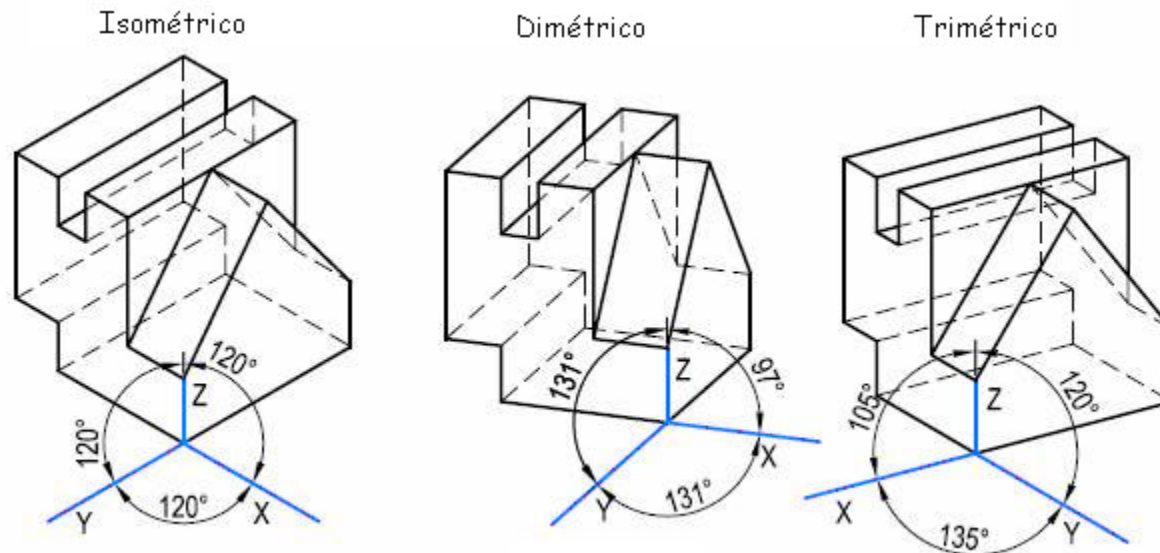
Tipos de proyecciones paralelas

- ▶ **Proyecciones Axonométricas ($DP \perp PP$)**
 - ▶ El plano de proyecciones no es perpendicular a ningún eje
 - ▶ El tamaño de las líneas paralelas se reduce en la proyección en la misma medida
 - ▶ Puedes tomar medidas, pero tienes visión 3D
 - ▶ **ISOMETRICA:** Los ángulos entre las proyecciones de los tres ejes son iguales (120°). Se aplica el mismo factor de escala a lo largo de cada eje. La dirección de proyección es el vector $[1 \ 1 \ 1]$



Tipos de proyecciones paralelas

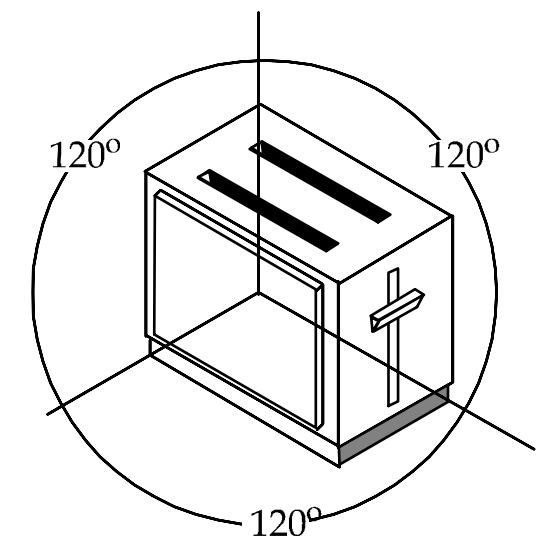
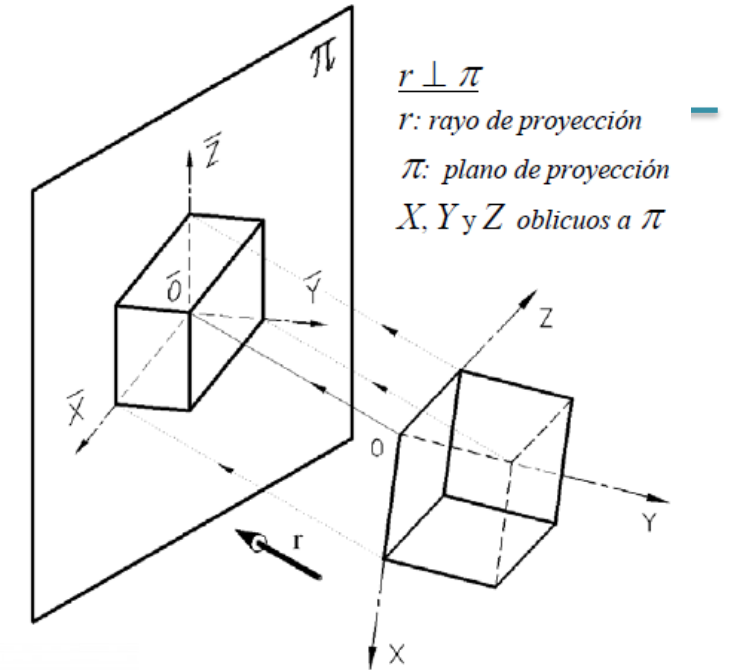
- ▶ Proyecciones Axonométricas ($DP \perp PP$)
 - ▶ DIMETRICA: Los ángulos entre dos de los ejes son iguales. Se necesitan dos factores de escala
 - ▶ TRIMETRICA: Los ángulos entre los tres ejes son diferentes. Se necesitan tres factores de escala.
 - ▶ En función de los ángulos utilizados, los factores de escala variarán



Tipos de proyecciones paralelas

► Proyección isométrica

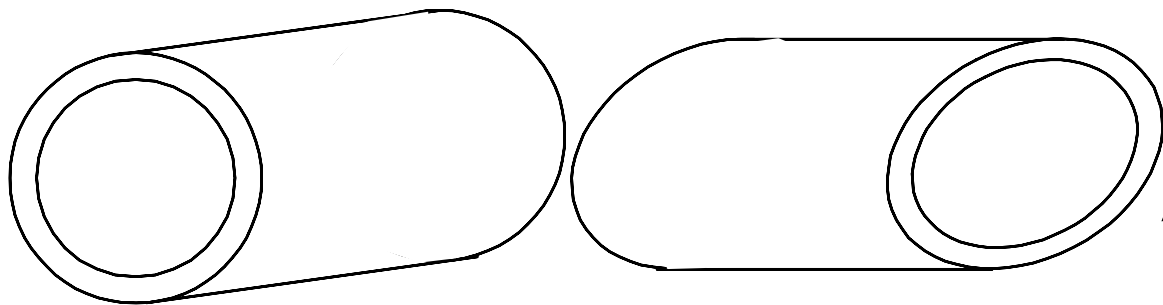
- Se utiliza en:
 - ilustraciones de catálogos, registros de oficinas de patentes, diseño de muebles
- Ventajas:
 - no es necesario utilizar múltiples vistas
 - muestra la naturaleza tridimensional de los objetos
 - se pueden realizar medidas escalando en los ejes
 - algunos videojuegos la utilizan
- Inconvenientes:
 - la falta de disminución del tamaño en la proyección produce distorsiones
 - más útil para superficies planas que para superficies curvas



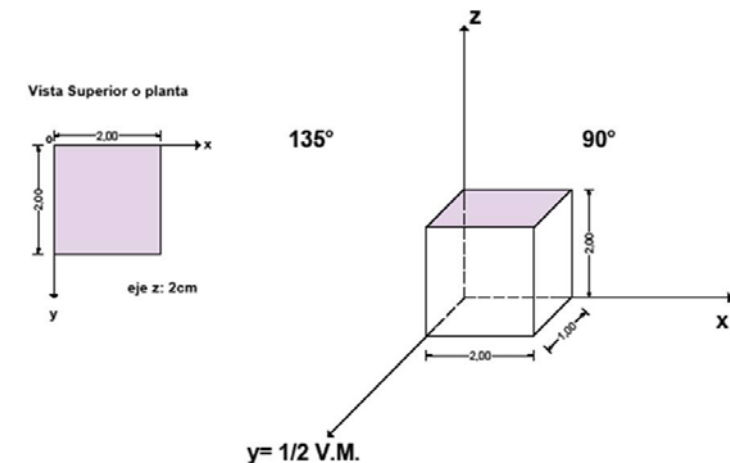
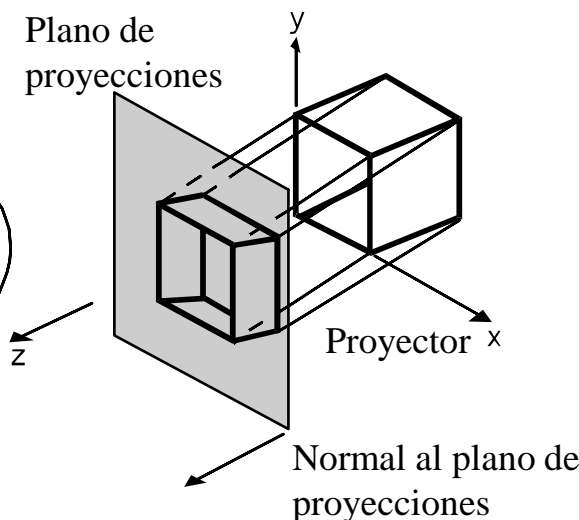
Tipos de proyecciones paralelas

► Proyecciones oblicuas

- Los proyectores son oblicuos al plano de proyecciones
- El plano de proyecciones es normal a uno de los ejes
- Ventajas:
 - pueden representar de forma exacta una cara del objeto (es decir, se pueden tomar medidas exactas): mejor para formas elípticas que las axonométricas.
 - la comparación de tamaños es más sencilla que con la perspectiva
 - representa la apariencia tridimensional
- Inconvenientes:
 - los objetos aparecen distorsionados si no se elige bien el plano de proyecciones (p.ej: los círculos pueden aparecer como elipses)
 - puede no tener una apariencia real

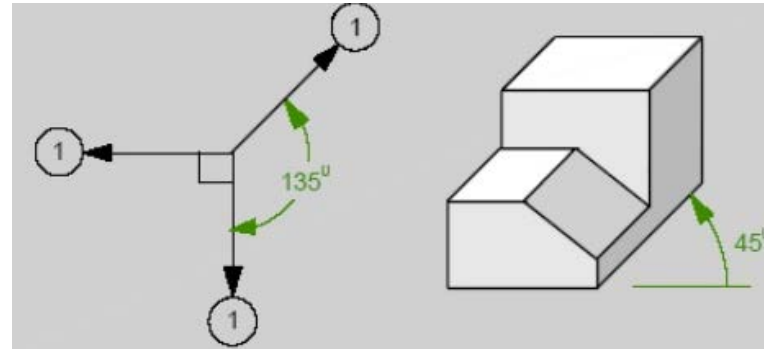


Transformaciones y visualización 3D

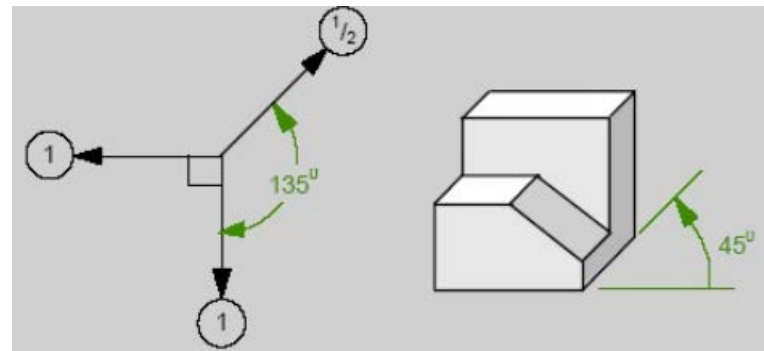


Tipos de proyecciones paralelas

- ▶ Principales tipos de proyecciones oblicuas
 - ▶ Caballera: El ángulo entre el plano de proyecciones y los proyectores es 45° . Las caras perpendiculares se proyectan a escala 1

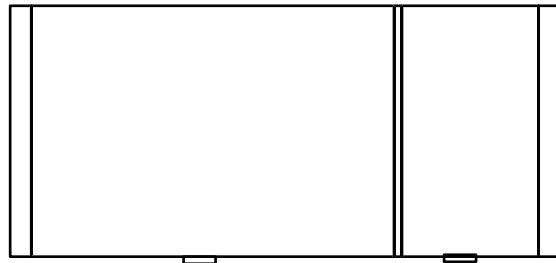


- ▶ Gabinete: Las caras perpendiculares se proyectan a escala $1/2$

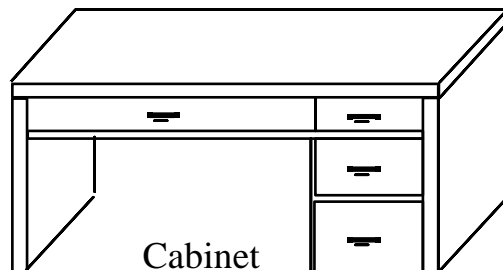
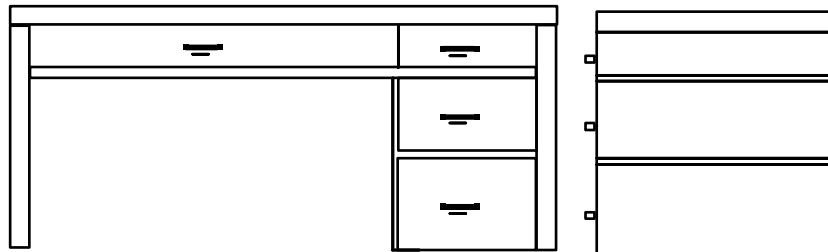


Tipos de proyecciones paralelas

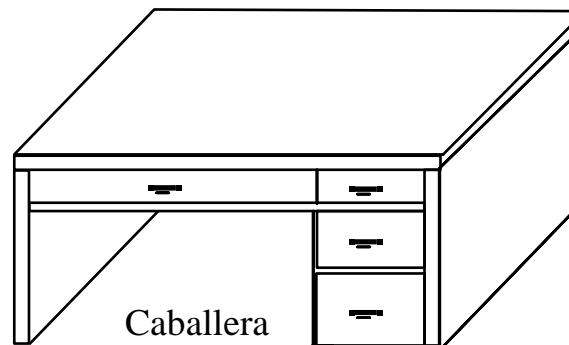
► Ejemplos de proyecciones ortográfica y oblicuas



Ortografía
multivista

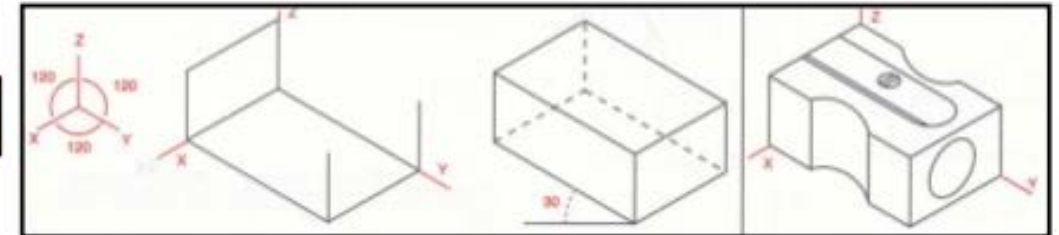


Cabinet

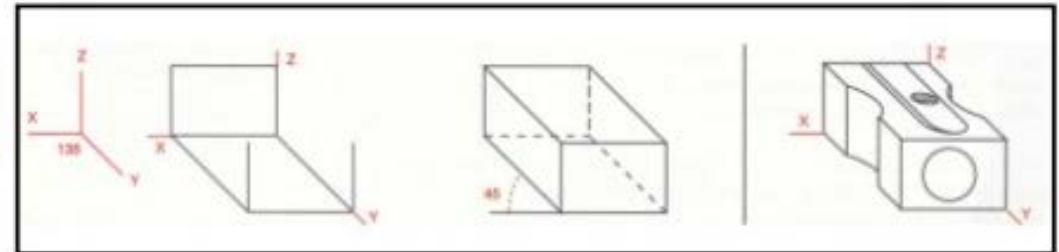


Caballera

ISOMÉTRICA



CABALLERA



Tipos de proyecciones paralelas

► Sumario de proyecciones paralelas

- Asume que la cara principal del objeto se encuentra en el plano principal, p.ej: paralela al plano XY, YZ o ZX.

1. Multivista ortográfica

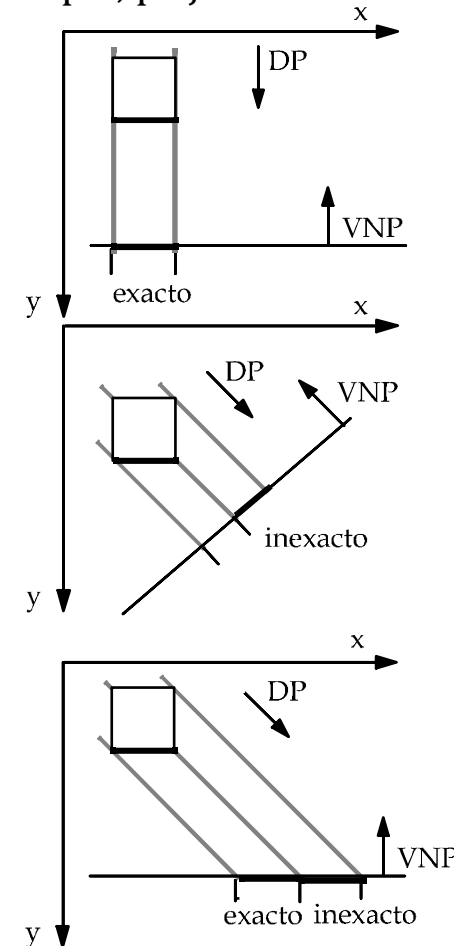
- VNP // ejes
- DP // VNP
- muestra una cara
- medidas exactas

2. Axonométricas

- VNP \perp // ejes
- DP // VNP
- muestra caras adyacentes
- medidas inexactas
- disminución del tamaño en la proyección de forma uniforme en función del ángulo que forman la normal a la cara y DP

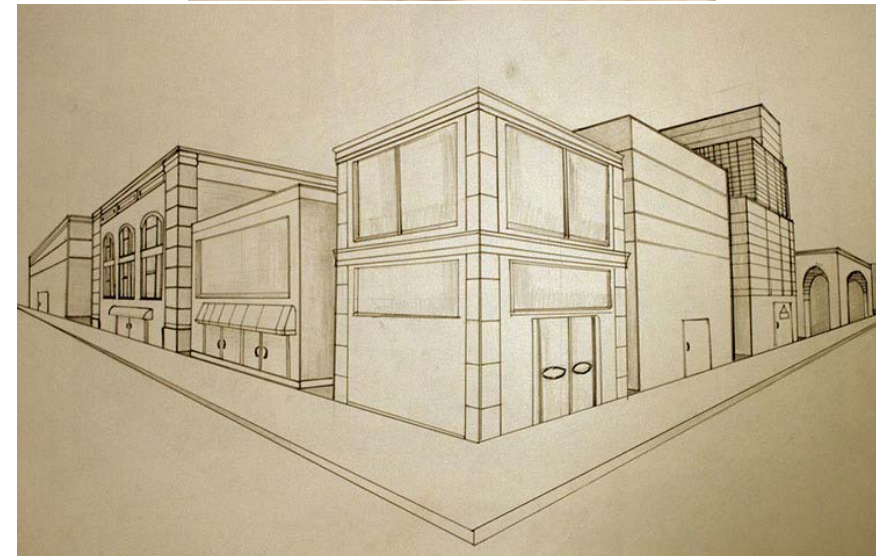
3. Oblicuas

- VNP // ejes
- DP \perp // VNP
- muestra caras adyacentes
- una cara medidas exactas
- las otras disminución del tamaño en la proyección de forma uniforme



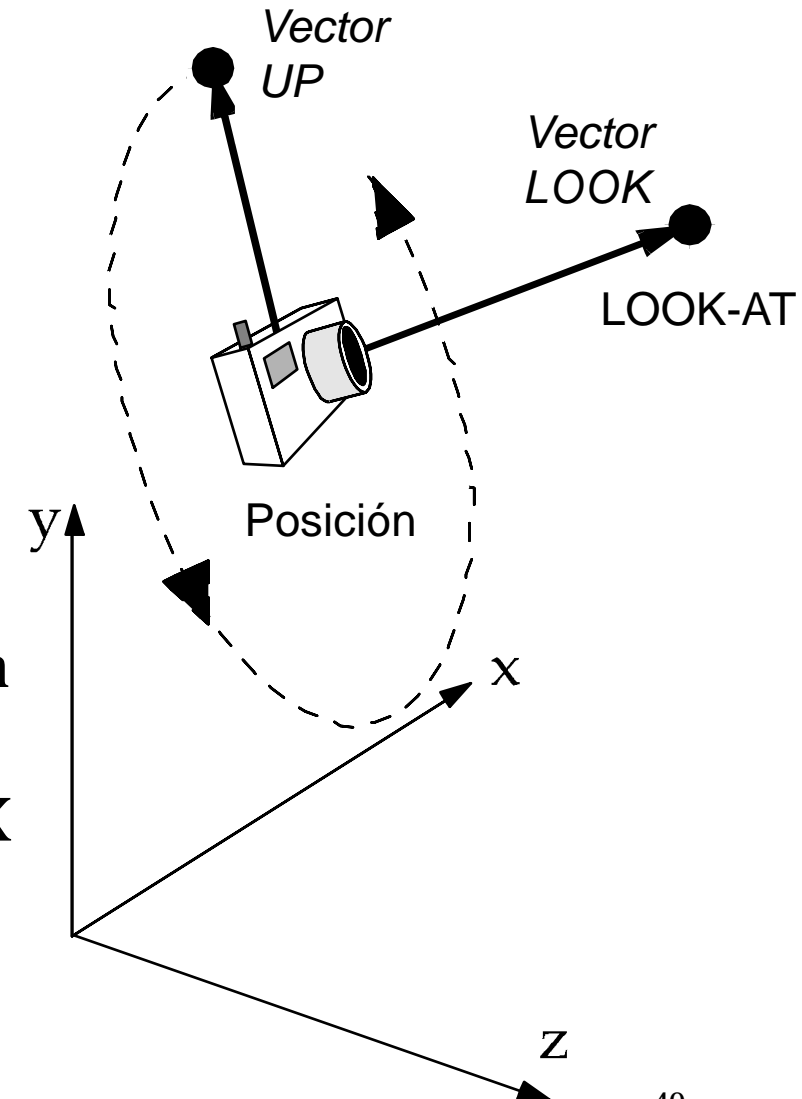
Proyecciones

- Proyección perspectiva
 - El centro de proyección está en un punto (x,y,z)
 - La proyección está definida por el centro de proyección (CP) y el plano de proyección (PP)
 - Ventajas:
 - Proporciona realismo visual y sensación tridimensional (efecto tamaño distancia)
 - Inconvenientes:
 - No mantiene la forma ni la escala del objeto (excepto en los planos paralelos al plano de proyección)
 - Es diferente de la proyección paralela porque:
 - Líneas paralelas dejan de serlo al proyectar
 - El tamaño de los objetos disminuye con la distancia
 - La disminución del tamaño no es uniforme



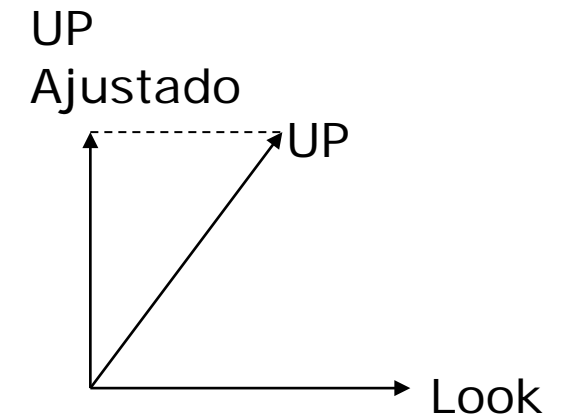
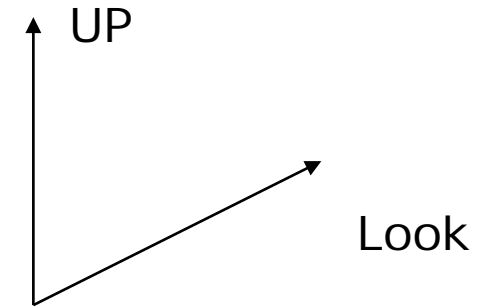
Modelo de cámara

- ▶ Parámetros del modelo de cámara
- ▶ Posición
 - ▶ Tres grados de libertad:
 - ▶ Las coordenadas x, y, z de la cámara en el espacio 3D
- ▶ Orientación
 - ▶ Viene dada por dos vectores: LOOK and UP
 - ▶ El vector LOOK indica hacia donde mira la cámara
 - ▶ Con la posición de la cámara y un punto de interés LOOK-AT también se puede obtener LOOK
 - ▶ UP define una rotación alrededor del eje definido por LOOK



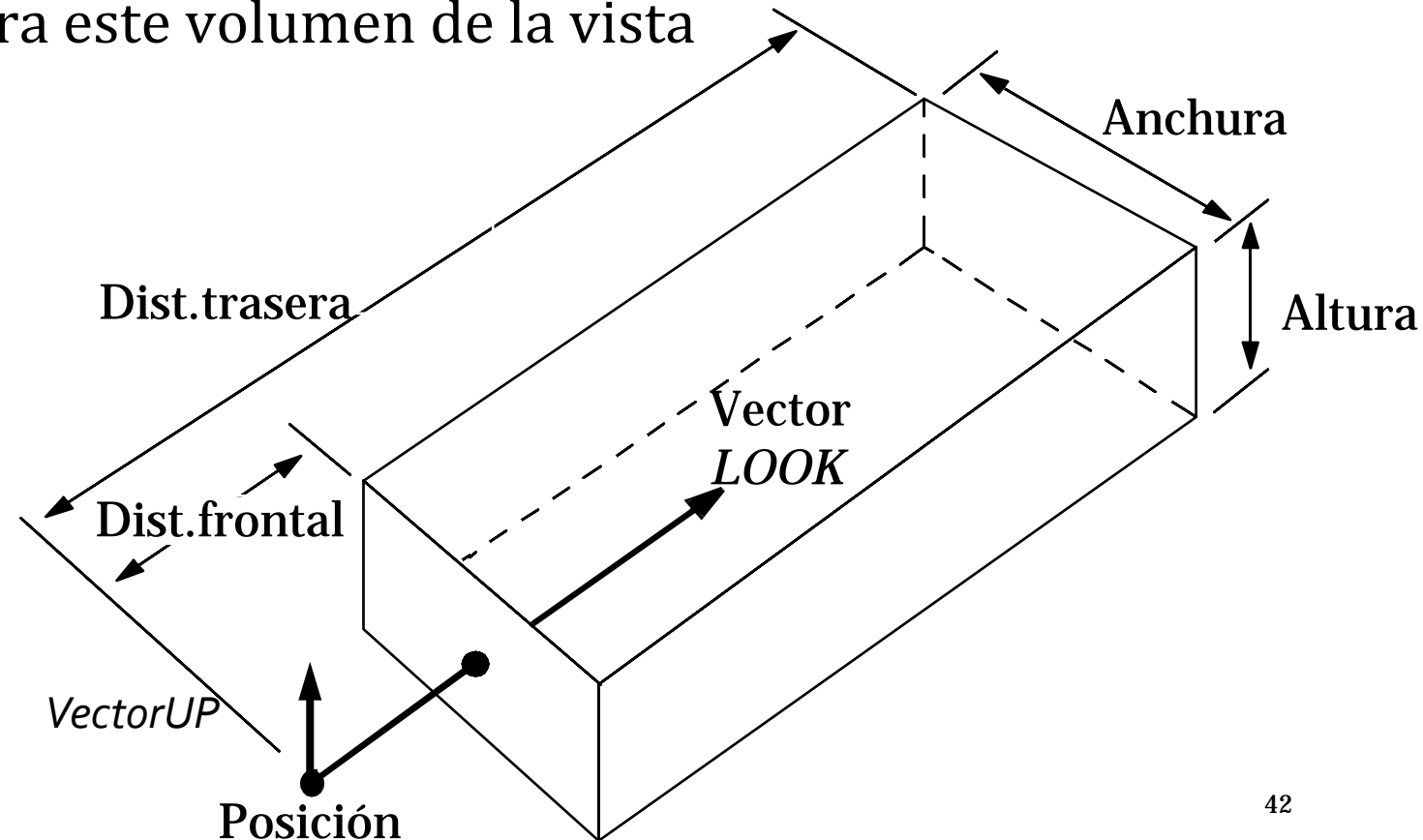
Modelo de cámara

- ▶ En principio, el vector UP debe ser perpendicular al vector LOOK
- ▶ Como puede resultar difícil determinar un vector perpendicular a LOOK, los paquetes gráficos ajustan el vector UP para que sea perpendicular a LOOK
- ▶ Por ello, UP puede ser cualquier vector mientras no sea paralelo a LOOK
- ▶ Generalmente, el vector UP se define como $(0,1,0)$



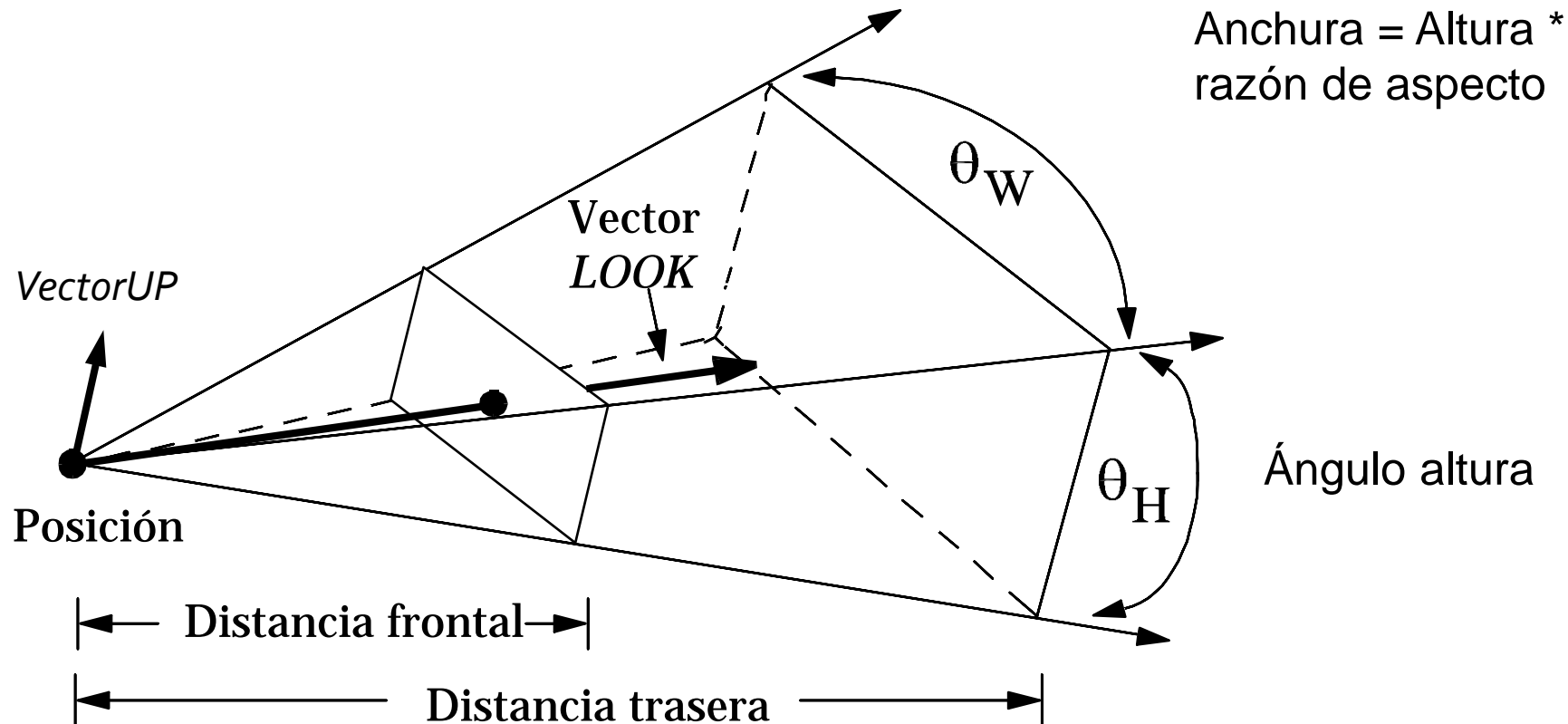
Modelo de cámara

- ▶ Volumen de la vista para una proyección paralela ortográfica
 - ▶ Define la parte de la escena visible para el usuario
 - ▶ Los objetos se recortan contra este volumen de la vista



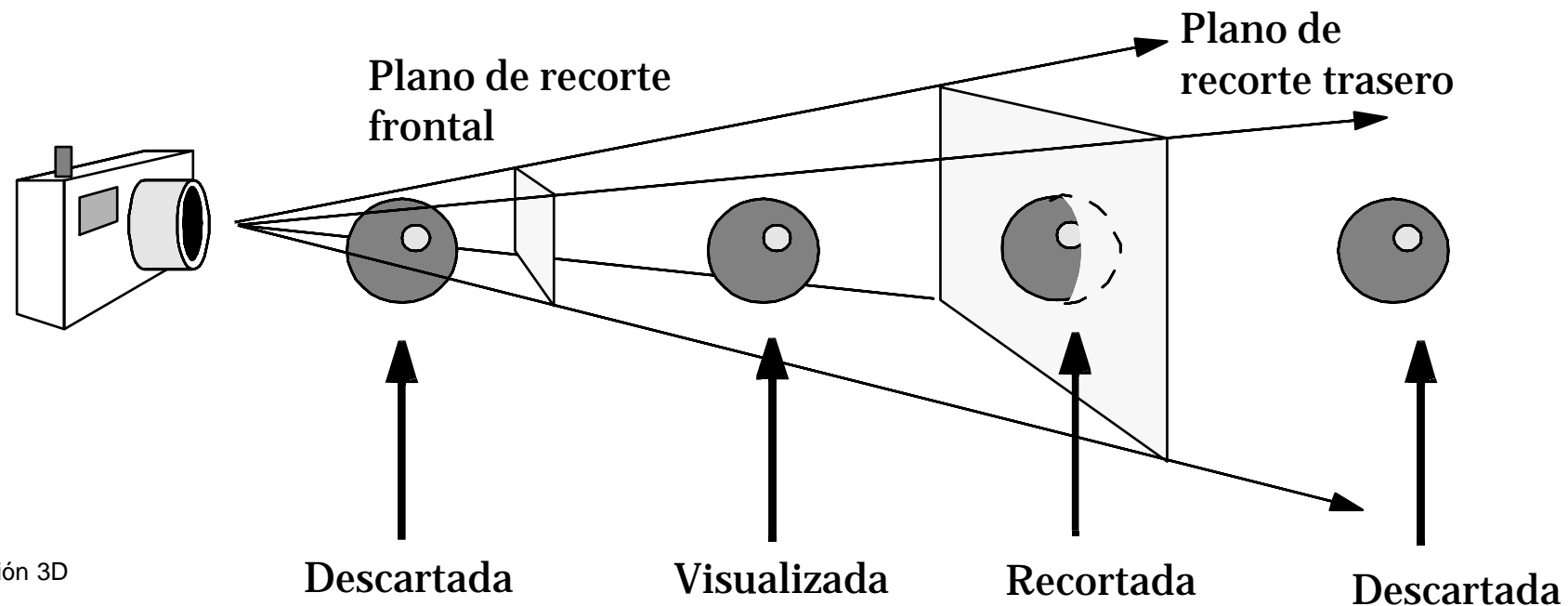
Modelo de cámara

- ▶ Volumen de la vista para una proyección perspectiva
 - ▶ Pirámide truncada



Modelo de cámara

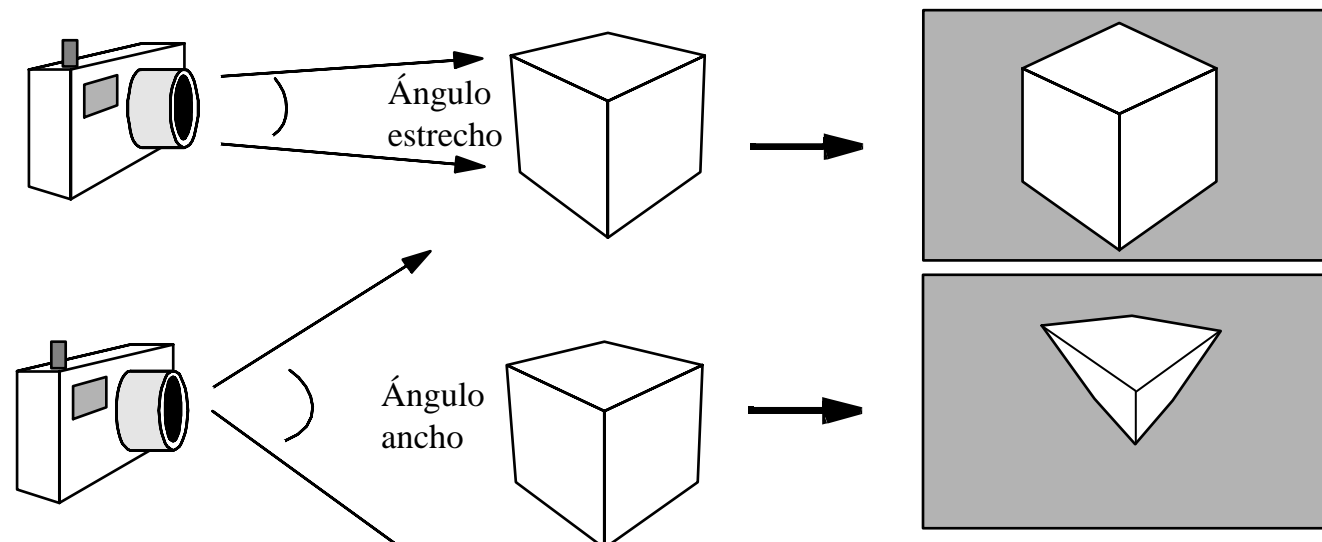
- ▶ Planos de recorte frontal y trasero
 - ▶ El volumen entre seis planos de recortado define la porción de la escena que la cámara ve
 - ▶ Las posiciones de los planos frontal y trasero vienen dadas por dos distancias a lo largo del vector LOOK
 - ▶ Los objetos que quedan fuera del volumen no se dibujan
 - ▶ Los objetos que intersectan con el volumen se recortan



Modelo de cámara

- ▶ Razón de aspecto
 - ▶ Análogo al tamaño de las fotografías, indica la proporción entre anchura y altura
- ▶ Campo de visión (anchura de campo)
 - ▶ Análogo a escoger una lente para una cámara fotográfica: ajusta el zoom y la cantidad de distorsión perspectiva

1:1	2:1
4:3	16:9



Modelo de cámara

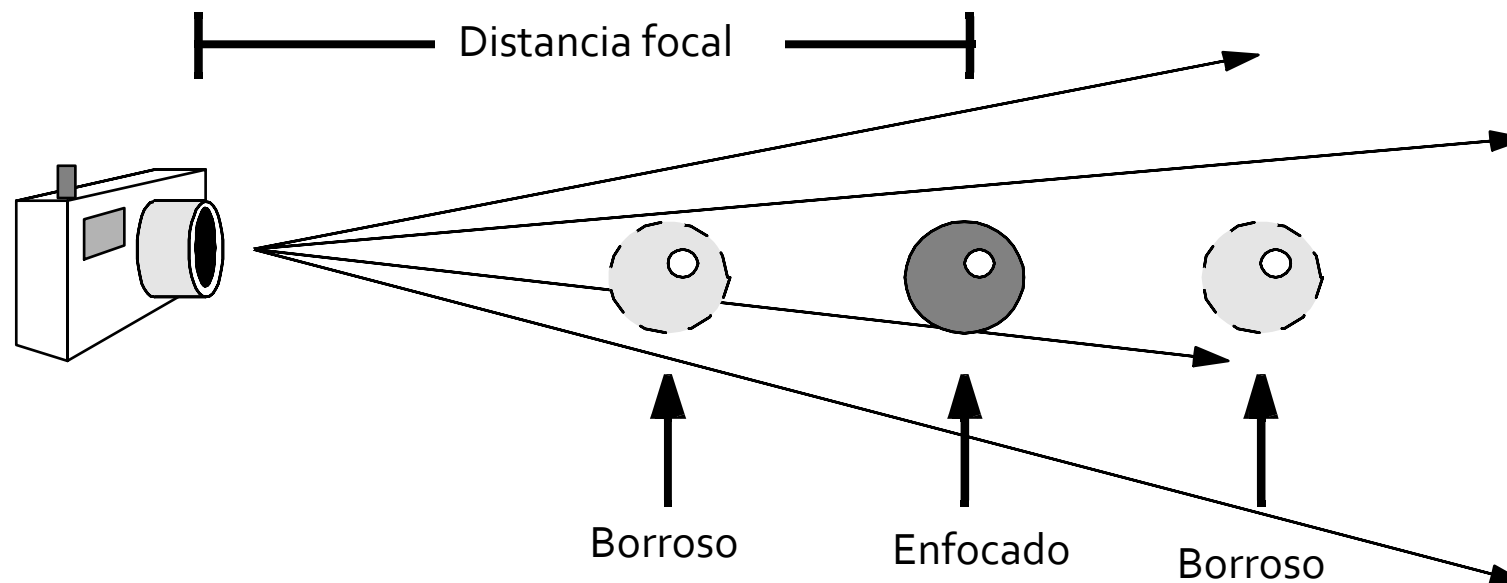
- Efecto gran angular



Modelo de cámara

► Profundidad de campo

- Algunos modelos de cámara tienen profundidad de campo para fijar el rango de enfoque ideal y así aproximar el comportamiento de una cámara real
- Objetos situados a la distancia focal se visualizarán nítidos (enfocados), los que estén más cercanos o más lejanos aparecerán borrosos (desenfocados)



Modelo de cámara

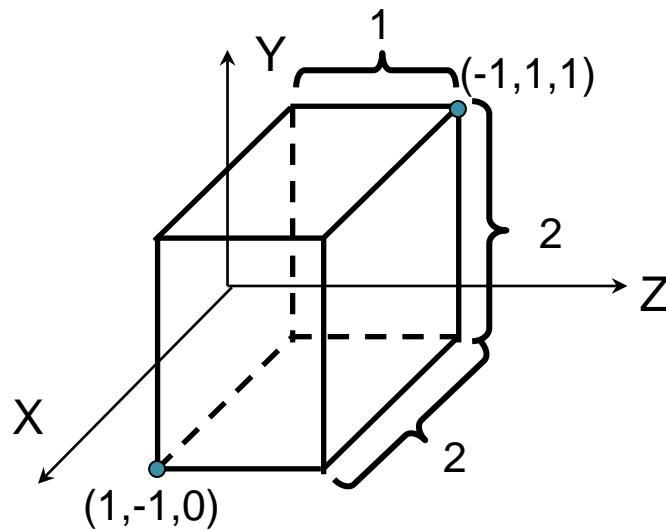
Profundidad de campo



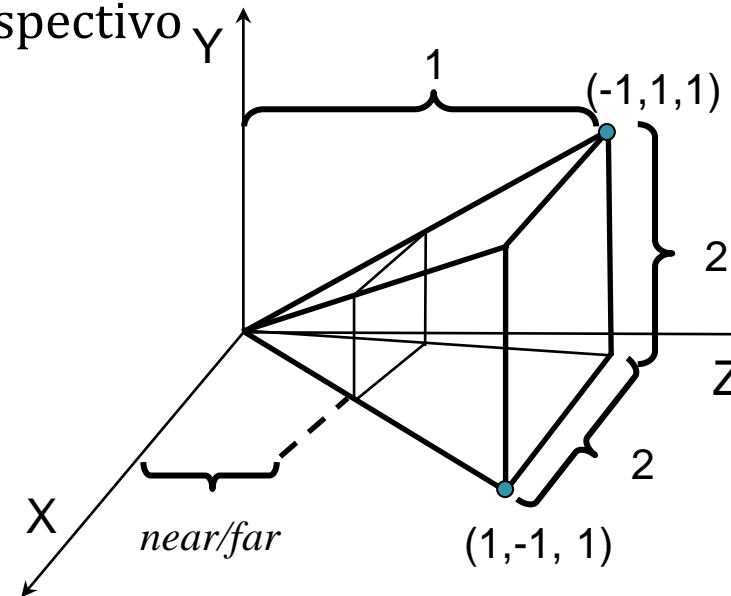
Modelo de cámara

- ▶ Volúmenes canónicos de la vista, análogo al dispositivo normalizado 2D

* Paralelo



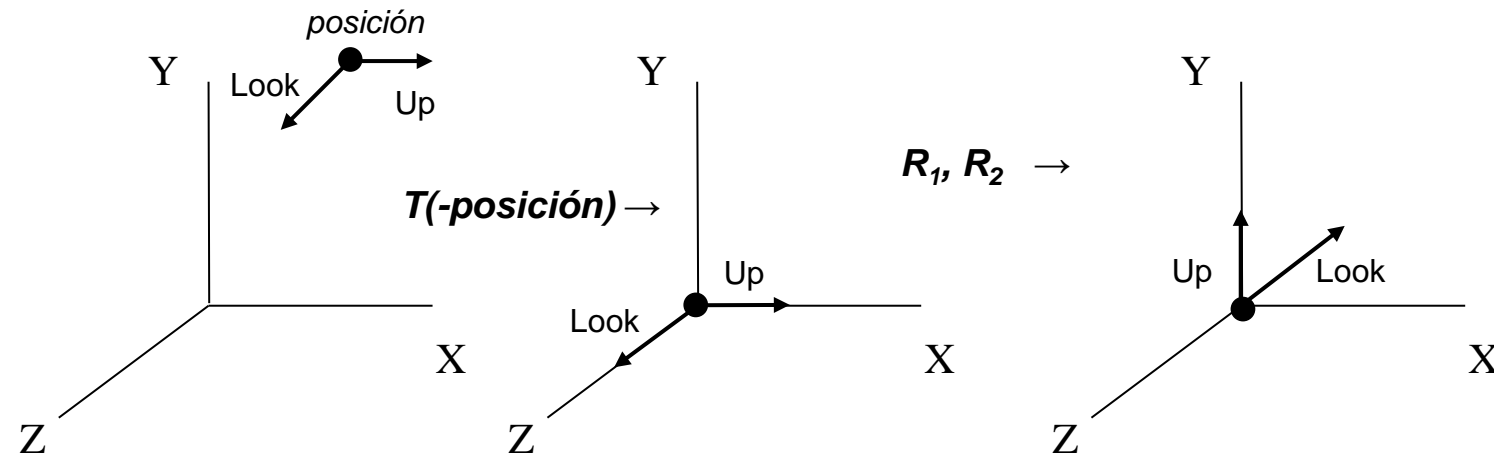
* Perspectivo



- LOOK=(0,0,1)
- UP=(0,1,0)

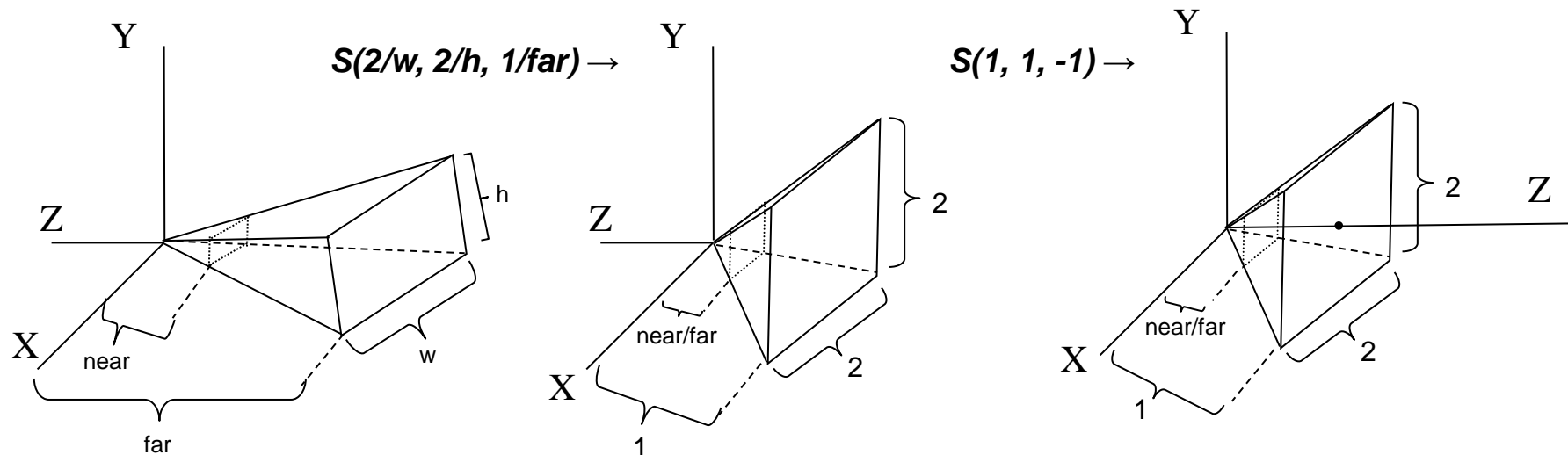
Modelo de cámara

- ▶ Transformaciones de la vista
 - ▶ Vista general → Vista simple
 - ▶ La cámara se mueve al origen
 - ▶ La dirección de proyección se lleva al eje Z
 - ▶ La dirección UP se lleva al eje Y



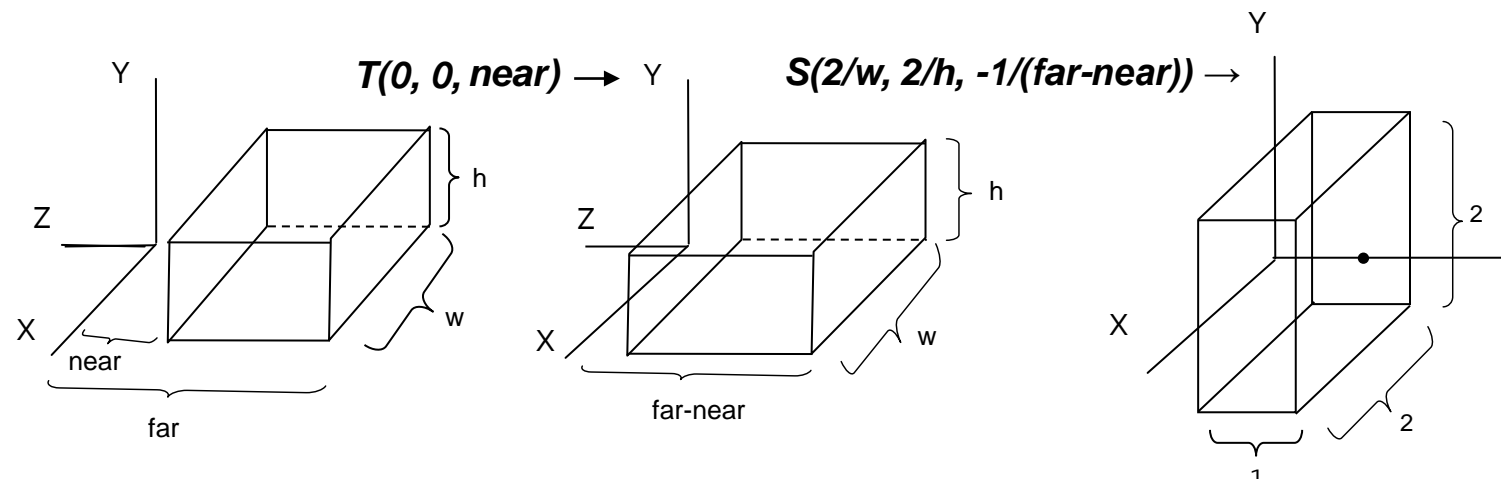
Modelo de cámara

- ▶ Transformaciones de la vista (cámara perspectiva)
 - ▶ Vista simple \rightarrow Volumen canónico
 - ▶ Ajusta el tamaño del volumen
 - ▶ Cambia a un sistema levógiro



Modelo de cámara

- ▶ Transformaciones de la vista (cámara paralela)
 - ▶ Vista simple \rightarrow Volumen canónico
 - ▶ Traslada el volumen a $z=0$
 - ▶ Ajusta el tamaño del volumen
 - ▶ Cambia a un sistema levógiro



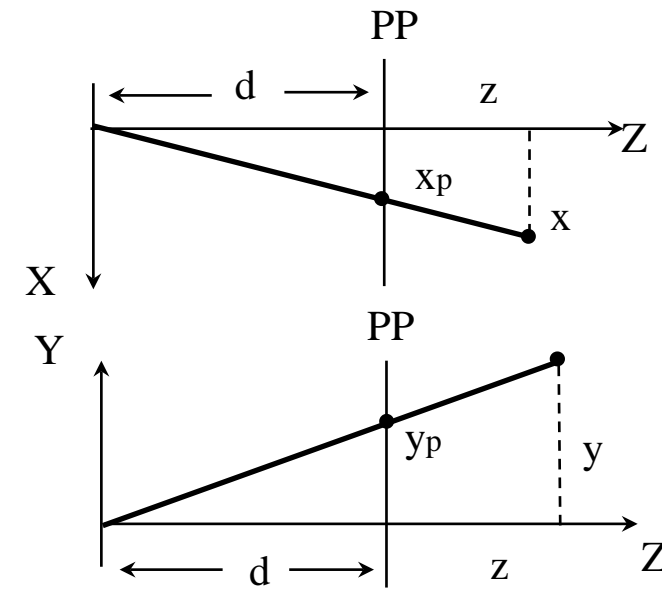
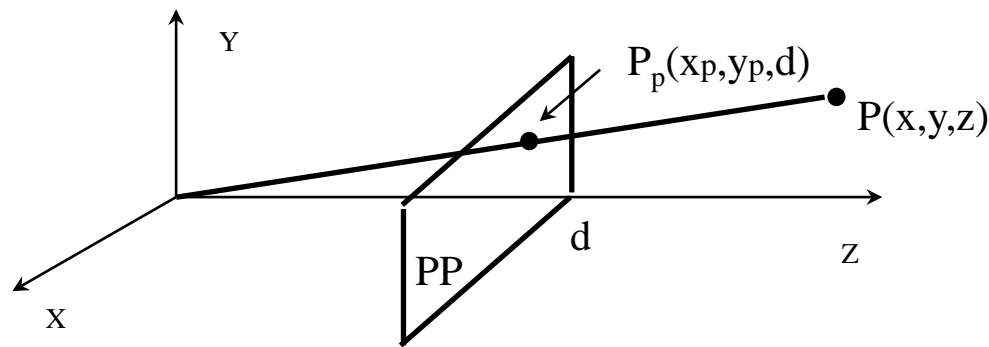
Modelo de cámara

- ▶ En este screencast se explica el modelo de cámara y cómo resolver un problema de transformación de la vista:
 - ▶ Modelo de cámara: Parámetros
 - ▶ Cámara Ortográfica
 - ▶ Cámara Perspectiva
 - ▶ Volúmenes canónicos
 - ▶ Transformación de la vista
 - ▶ Ejercicio
- ▶ <http://hdl.handle.net/10251/105167>

Matrices de proyección

Matrices de proyección simples

- ▶ Tenemos un sistema de coordenadas LEVÓGIRO
- CP en el origen y PP perpendicular a Z a una distancia d
- ▶ Proyección perspectiva simple



- ▶ Por triángulos semejantes:

$$\frac{x}{x_p} = \frac{z}{d} \rightarrow x_p = \frac{x \cdot d}{z}$$

$$\frac{y}{y_p} = \frac{z}{d} \rightarrow y_p = \frac{y \cdot d}{z}$$

$$(x_p, y_p, d) = \left(\frac{x}{z/d}, \frac{y}{z/d}, \frac{z}{z/d} \right)$$

Matrices de proyección

Matrices de proyección simples

- Proyección perspectiva:

$$(x, y, z, 1) \rightarrow (x_p, y_p, d, 1) = \left(\frac{x}{z/d}, \frac{y}{z/d}, \frac{z}{z/d}, 1 \right)$$

- Matriz resultante:

$$\begin{bmatrix} x_p \\ y_p \\ d \\ 1 \end{bmatrix} = \begin{bmatrix} 1/z/d & 0 & 0 & 0 \\ 0 & 1/z/d & 0 & 0 \\ 0 & 0 & 1/z/d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

- **Problema:** depende de la z de cada vértice
- **Solución:** dos pasos:

1. Matriz de proyección perspectiva:

$$\begin{bmatrix} x \\ y \\ z \\ z/d \end{bmatrix} = M_{per} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

2. División homogénea:

$$\frac{(x, y, z, z/d)}{z/d} = \left(\frac{x}{z/d}, \frac{y}{z/d}, \frac{z}{z/d}, 1 \right) = (x_p, y_p, d, 1)$$

Matrices de proyección

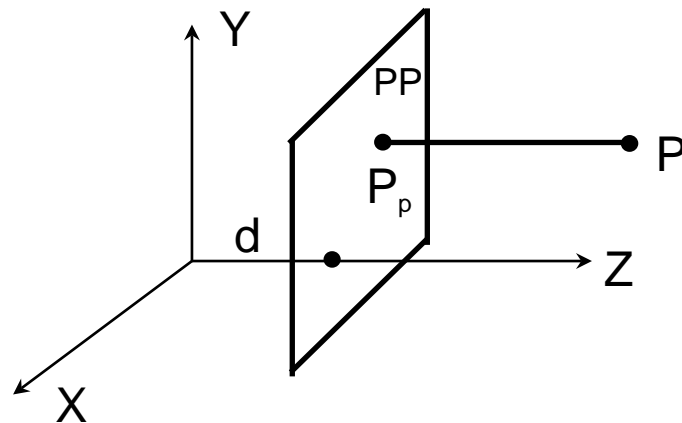
Matrices de proyección simples

- ▶ Matriz de proyección perspectiva:

$$M_{per} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/d & 0 \end{bmatrix}$$

- ▶ Proyección ortográfica simple:

$$(x, y, z, 1) \rightarrow (x, y, d, 1)$$

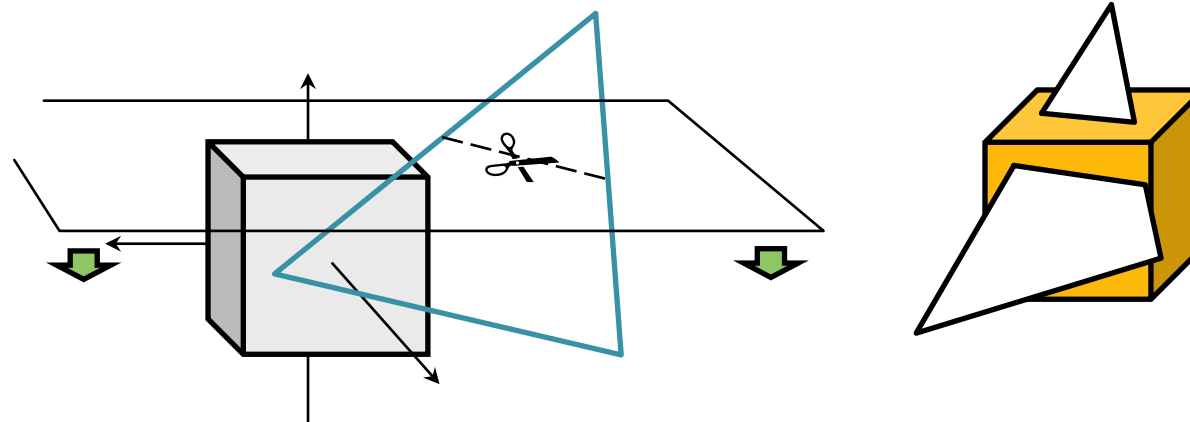


$$\begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix} = M_{ort} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$M_{ort} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Recortado 3D

- ▶ En 3D el algoritmo se aplica a cada una de las caras del poliedro a recortar contra cada uno de los planos del volumen canónico



Bibliografía

- ▶ D. Hearn, M. Baker. Computer Graphics with OpenGL. Pearson Prentice Hall, 4ª edición.
 - ▶ Capítulos 9 y 10