



Básicos de 3D

El explorador como plataforma

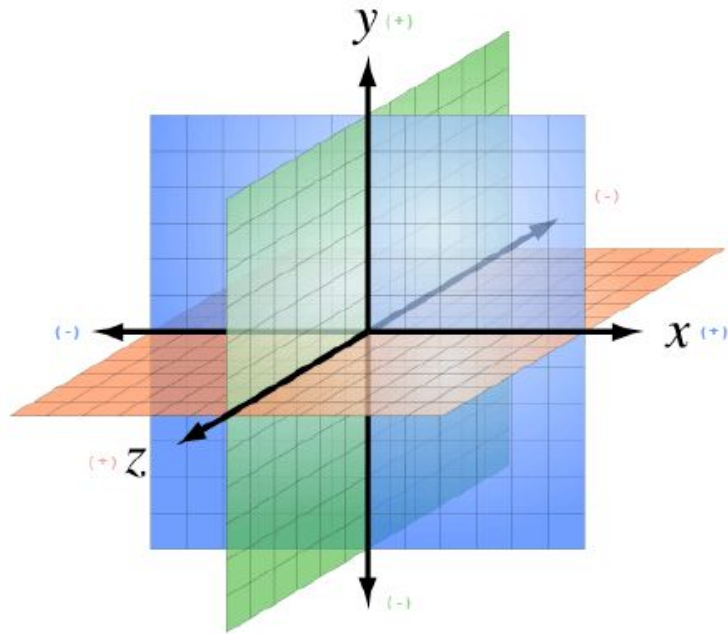
- Máquinas virtuales de Javascript
 - Las animaciones e interacción van tan rápido como las MV lo soporten.
- Composición acelerada
 - Combinar (componer) varios elementos de una página rápidamente.
 - Uso de GPUs.
- Soporte de animaciones

Gráficas en 3D

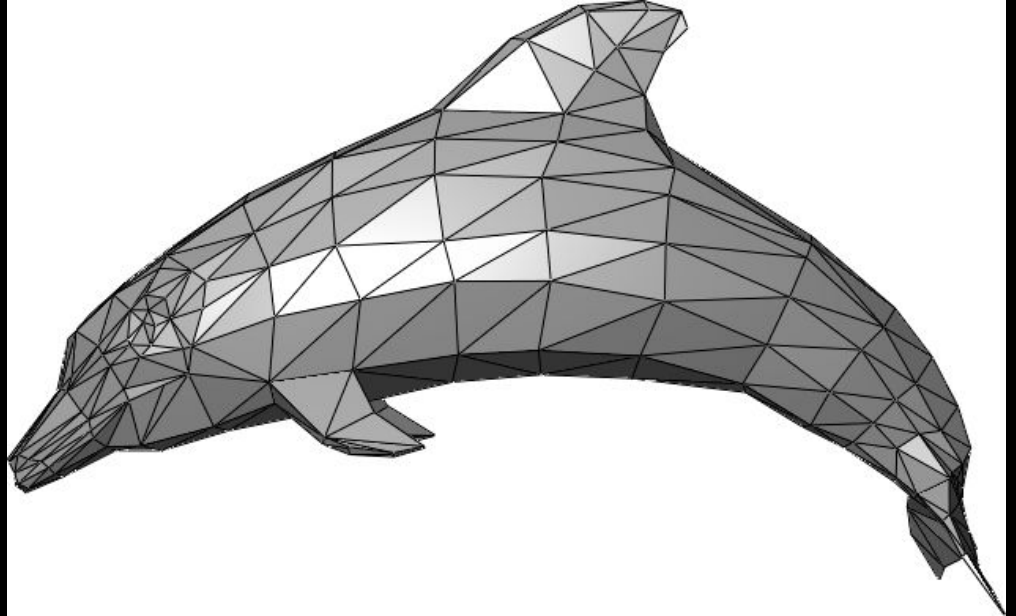
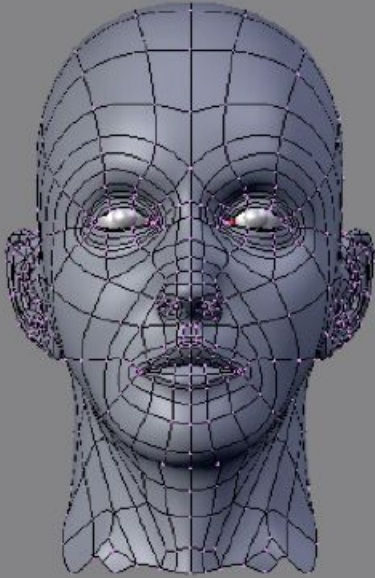
Los gráficos 3D (a diferencia de los 2D) usan una **representación tridimensional** de datos geométricos con el propósito de realizar cálculos y **renderizar** imágenes en 2D. Dichas imágenes pueden almacenarse para verlas más tarde o mostrarse en **tiempo real**.

Sistema de coordenadas

- WebGL: y up
 - y es positiva de abajo hacia arriba de la ventana.
- Canvas: y down
 - Basado en el estándar de HTML



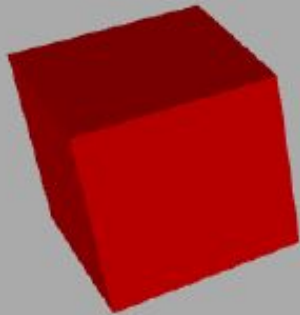
Meshes, Polígonos, Vertices



Materiales, Texturas, Luces

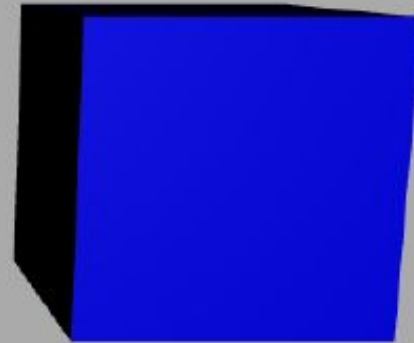


Transformaciones



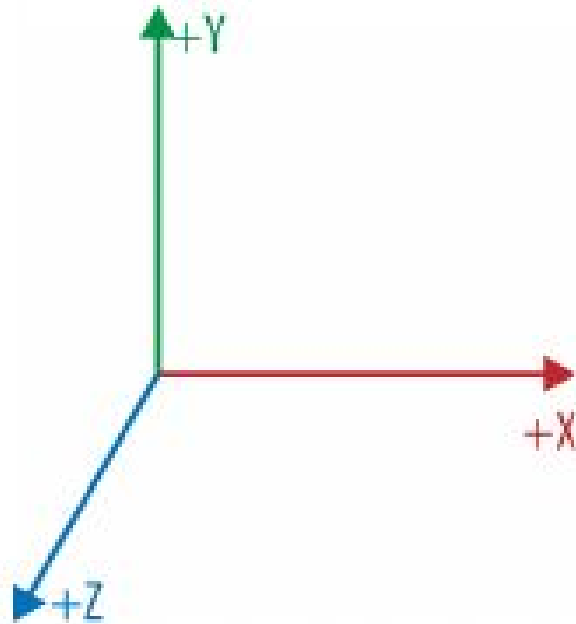
Translate $[-4, 0, 0]$

Rotate $[\pi / 8, \pi / 5, 0]$



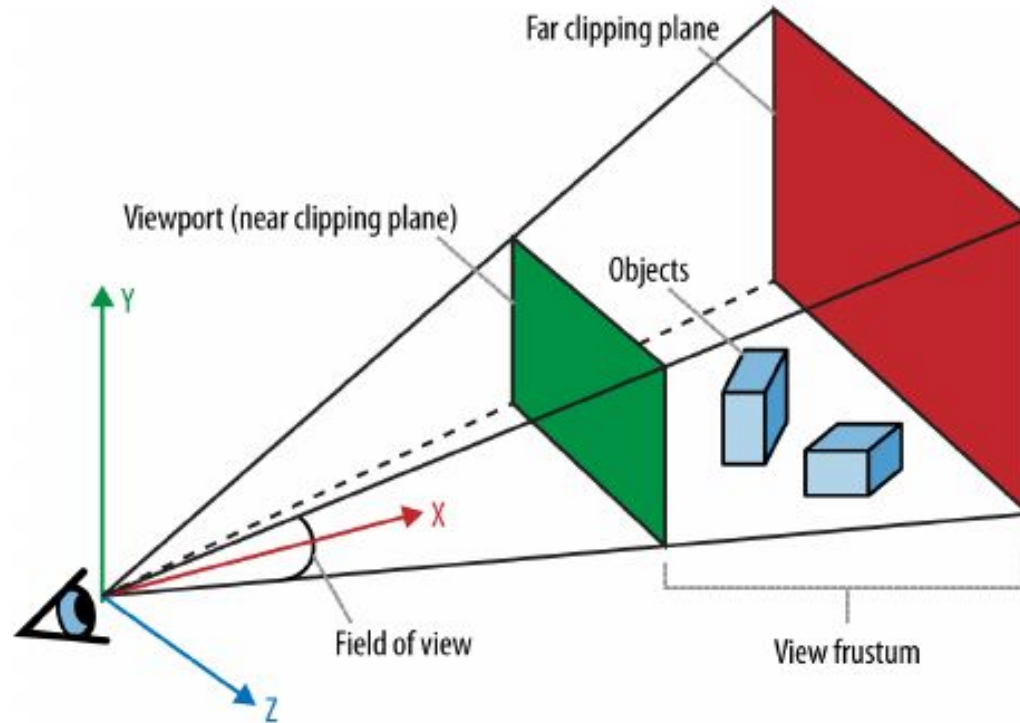
Scale $[1.5, 1.5, 1.5]$

Matrices



Left (X) Axis	Up (Y) Axis	Forward(Z) Axis	Translation
$\left\{ \begin{array}{l} m_0 \\ m_1 \\ m_2 \\ m_3 \end{array} \right\}$	$\left\{ \begin{array}{l} m_4 \\ m_5 \\ m_6 \\ m_7 \end{array} \right\}$	$\left\{ \begin{array}{l} m_8 \\ m_9 \\ m_{10} \\ m_{11} \end{array} \right\}$	$\left\{ \begin{array}{l} m_{12} \\ m_{13} \\ m_{14} \\ m_{15} \end{array} \right\}$

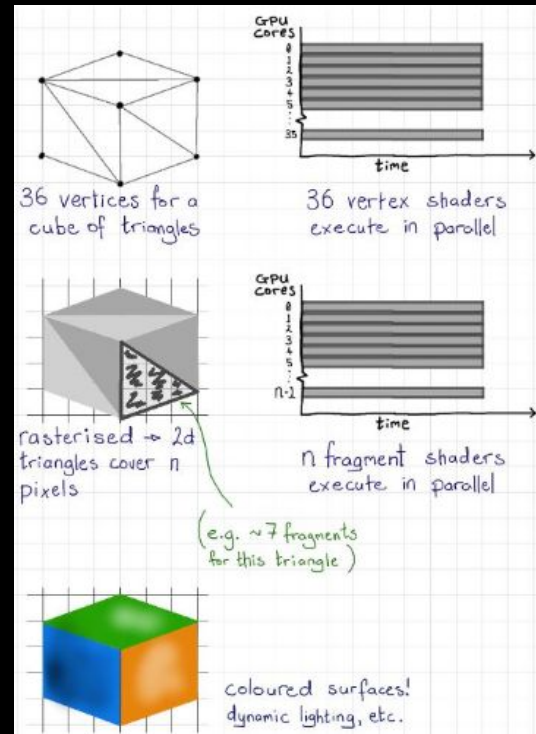
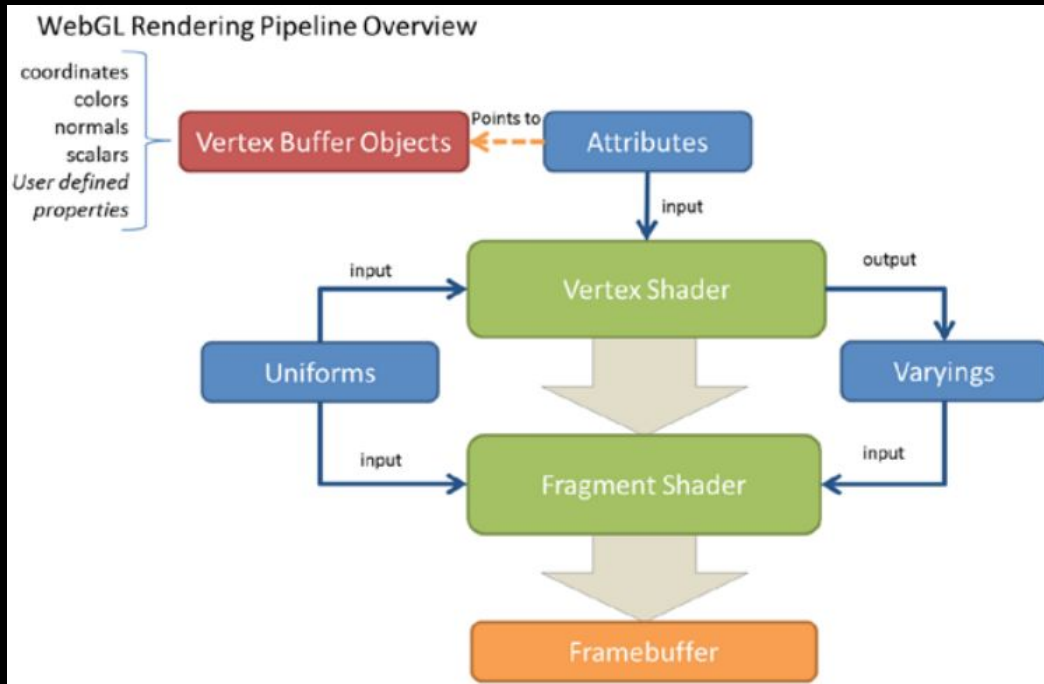
Cámaras, perspectiva, viewports, proyecciones



Shaders

- Programa para definir cómo interactúan vértices, luces, transformaciones, etc, antes de dibujarse en la pantalla (Render).
- Se ejecutan directo en la GPU.
- Necesarios para crear y desplegar los gráficos.

WebGL Rendering Pipeline



Pasos generales

1. Crear un Canvas
2. Obtener un contexto para dibujar del Canvas.
3. Iniciar un viewport.
4. Crear buffers que contengan datos para renderear.
5. Crear matrices de transformación.
6. Definir shaders para dibujar los datos.
7. Iniciar los shaders.
8. Dibujar.

Hola Mundo

