Colores y Atributos

<a.soroa@ehu.eus>

EHU

Atributos

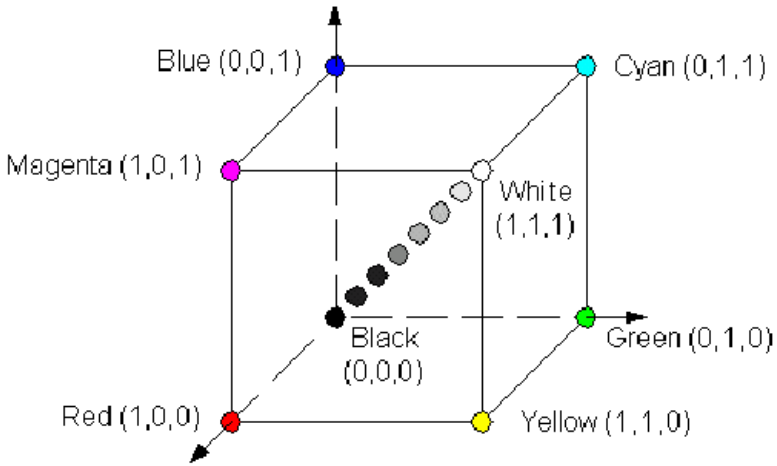
- Hasta ahora, hemos definido los vértices por sus coordenadas 3D
- Pero podemos asignar más información a los vértices
  - colores
  - normales
  - ...
- Vamos a ver cómo asignar colores a los vértices por medio de atributos.

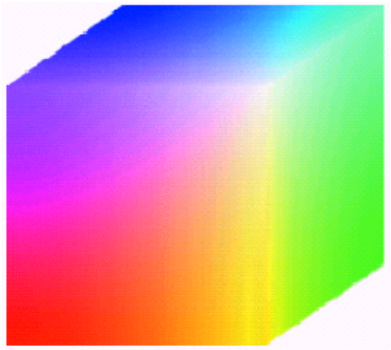
Color y THREE.js

```
var colorRed = new THREE.Color("rgb(255, 0, 0)");
var colorMagenta = new THREE.Color("hsv(200,178, 122)");
var colorYellow = new THREE.Color(0xFFFF00);
```

- Hay diversas maneras de representar el color
  - RGB
  - CYMK
  - HSV
  - ...
- El esquema RGB es el más usado
  - Esquema *aditivo*
  - Tres componentes (canales): *Red*, *Green*, *Blue*
    - Vector de tres elementos $\in [0, \ldots, 1]$
    - Número hexadecimal

Cubo RGB





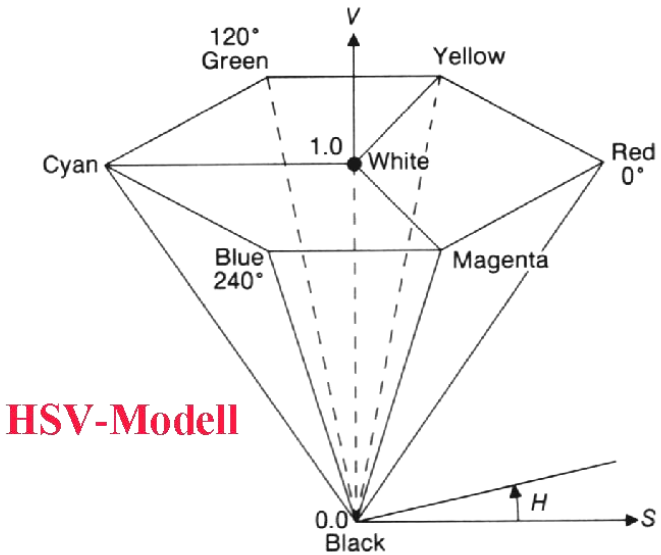
Color	Vector	Núm. hexadecimal
Negro	$(0,0,0)$	`0x000000`
Blanco	$(1,1,1)$	`0xFFFFFF`
Rojo	$(1,0,0)$	`0xFF0000`
Verde	$(0,1,0)$	`0x00FF00`
Cyan	$(0,1,1)$	`0x00FFFF`
Gris	$(0.5,0.5,0.5)$	`0x808080`

Color RGB: representación hexadecimal

- Usada por *Three.js*
- Representación compacta de los tres canales
- Formato hexadecimal: 0-9, A (10), B(11), C(12), D(13), E(14), F(15)
- Dos dígitos por canal. Mínimo `0x00`, máximo `0xFF` (255)
- Ejemplo: `0x1280FF`
  - El prefijo `0x` indica un número hexadecimal
  - Canal R: `12` hex == 18 decimal (1 * 16 + 2)
  - Canal G: `80` hex == 128 decimal (8 * 16 + 0)
  - Canal B: `FF` hex == 255 decimal (15 * 16 + 15)
- `0x1280FF` = $(\frac{18}{255}, \frac{128}{255}, \frac{255}{255}) = (0.07, 0.5, 1) \rightarrow$ Color amarillo

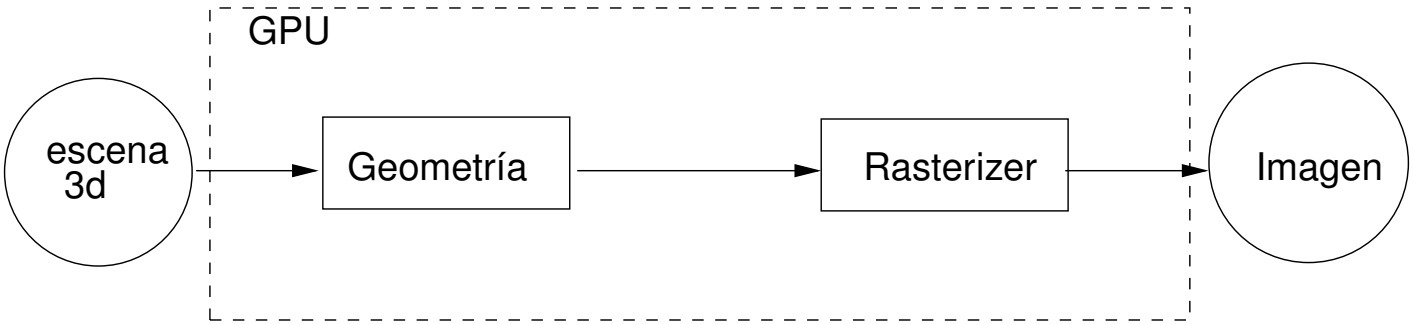
Color HSV

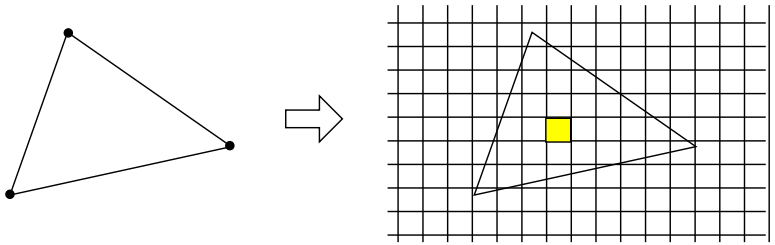
- Más intuitiva que RGB
- Se representa como un cono tridimensional
- Tres factores:
  - H (*hue*): el color como tal. Representado como un ángulo.
  - S (*saturation*): distancia desde el eje. En el extxemo estan los colores completamente saturados.
  - V (*value*): la intensidad. El eje vertical del cono.



Recordando el *rendering pipeline*

- *pipeline* simplificado:
  - La aplicación envía un triángulo a la GPU.
  - La GPU determina si los vértices del triángulo estan dentro de la pantalla.
  - Cada pixel del triángulo es coloreado.
- Pregunta clave: Cómo calculamos el color de cada *pixel*?





Triangulo coloreado en THREE.js

- Asignar un color differente a cada vértice
  - en forma de atributo.
- Dos elementos de `THREE.js`:
  - Asignar `vertexColors` a las caras de la geometría:

```
geometry.faces[0].vertexColors = [ red, green, blue ];
```

  - la propiedad `vertexColors` de los materiales.

```
var material = new THREE.MeshBasicMaterial(
                {vertexColors: THREE.VertexColors} );
```