**Drawcall**: llamada que dice “pinta”. Al pintar, se puede hacer en varios sitios. Hay distintos buffers. Yo puedo estar pintando en 2 buffers a la vez.

Un dibujo de un pizarrón blanco

Descripción generada automáticamente con confianza baja

Vertex Shader – Pixel Shader --- test

Al pasar de PS a Test se usa el RGBA, pero si hay varios buffers podemos tener RGBA1, RGBA2, etc. Uno para cada buffer, y tienen la misma resolución. Así:

Una caricatura de un pizarrón blanco

Descripción generada automáticamente con confianza baja

Donde yo pinto se puede usar como inputs de otros momentos. Puedo mezclar el buffer A+B, combinarlas, escalarlas y lo que yo quiera.

Al lado de cada fotograma, viene asociado un Z-buffer. Es un buffer donde el HW almacena cual es la Z más cercana. Ej: yo tengo una cámara que mira hacia una esfera y un plano

Un dibujo de una persona

Descripción generada automáticamente con confianza baja

El HW sabe que debe pintar primero el plano y luego encima la pelota. Cada pixel sé a qué distancia está de la cámara.

Hace una pregunta a las 18:19 preguntando qué es óptimo. ¿¿?? No lo escuché bien

Maneras de pintar una escena con luces

* **Forward Rendering**: técnica hace años, q esta en móviles sobre todo. Para pintar un fotograma, empezamos desde 0. Pitnamos el suelo y mientras lo estamos pintando, nos fijamos en la luz. Sacamos la normal del suelo, y nos fijamos en la luz y calculamos q es un vertice nos llega una luz de 0.7 por ejemplo.

Un letrero de color blanco

Descripción generada automáticamente con confianza baja

Y si tenemos varias luces pues las tenemos todas en cuenta y se suman los resultados. Si tenemos 150 luces esto no se podría… Además muchas están lejísimos no tendría sentido calcularlas.

Entonces lo q acabamos haciendo es q **para cada luz, para cada objeto, pinta**.

Si tengo 1000 objetos y una luz, 1000 drawcalls.

Si tengo 1000 objetos y 20 luces, acabo con 20000 drawcalls

Entonces los de PC vienen aquí y dicen no puede ser esto.

Hacen una fase de pintado que se llama **GBuffer**.

Para cada objeto pinta. Me guardo en el primer fotograma el albedo, en el 2 la normal, en el 3ero el specular y roughness y en el cuarto la Z. No se nada de luces. Me cuesta lo mismo si hay una luz o 50

Y luego para cada luz, pinta luz. Solamente miro los pixeles afectados por la luz, no todos.

Si tengo 1000 objetos y 20 luces, ahora tengo 1020 drawcalls.

Se retrasa el calculo de la luz al final.

Un dibujo de un pizarrón blanco

Descripción generada automáticamente con confianza baja

**ZPrePass**: pinto todos los objetos leyendo solo la Z. Esto lo que hace es que si está detrás dejo de calcular muchos objetos. EL Unreal lo hace por defecto. Para transparentes, primero se pintan los solidos entonces y luego los transparentes.

Znear - 0.01

Zfar – 10000

16

Web para ver como renderiza un frame del Deus Ex

<https://www.adriancourreges.com/blog/2015/03/10/deus-ex-human-revolution-graphics-study/>

24 bits de Z buffer. Desde el 0 al 16M.

**Stencil buffer**: 8 bits. Se usa para marcar. Esto es una cámara, un coche, lo que quiera. Cuando seleccione un objeto pues le decimos al esténcil que pinte la marca.

Tenemos unos cuantos pixels con el bit de stencil a 1 (si está seleccionado).

Filtro **sobel**: cogemos un pixel y vemos los 4 vecinos y generamos una respuesta diciendo q el color final sea la suma de los 4 vecinos. Lo q buscamos es si el del centro es muy distinto de los demás, ¿? Algo no se

Entonces q pasa, que cuando miro el negro y miro sus vecinos negros pues se queda igual. Pa los blancos balncos, peeero en los bordes digo uy cuidao. Entonces es ahí donde yo puedo mirar q esos pixeles los pinto amarillos por ejemplo para marcar el borde.

Frustum culling: descarta todo aquello que no se está viendo.

<https://www.adriancourreges.com/blog/2015/06/23/supreme-commander-graphics-study/>

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

**Doom**:

Mega-Texture: las texturas están partidas en cachitos.

Ventaja:

* + hay un solo material (una textura). Ahí está todo. No hay que cambiar de material al pintar. No hay que cambiar de textura, pintar, cambiar de textura pintar. Aquí es: coge esta textura y pinta todo.f

Velocity Map: para saber hacia dónde se están moviendo los objetos

<https://www.adriancourreges.com/>