

12 Glosario

Depth Buffer: es un búfer que es de la misma anchura y altura como su render target. Este búfer registra la profundidad de cada píxel que es renderizado. Cuando un píxel es renderizado una segunda vez, como cuando un objeto es renderizado detrás de otro, el Depth Buffer toma cualquier valor anterior de profundidad, o lo reemplaza por el valor de profundidad del segundo píxel. Qué profundidad es preservada y qué profundidad es descartada dependerá de la función de profundidad seleccionada. Por ejemplo, si `CompareFunction.LessEqual` es la función actual, los valores de profundidad son menores o iguales al valor actual son preservados. Cualquier valor mayor al valor de profundidad actual es descartado. Esto se le llama *depth test*. El *depth test* se produce cada vez que el píxel es renderizado. Cuando un píxel pasa el *depth test*, ese color es escrito en el render target y esa profundidad es escrita en el depth buffer.

La profundidad de un píxel se determina basándose en el tamaño de las matrices de vista y proyección seleccionadas para renderizar. Un píxel que toca el plano near de la proyección tiene una profundidad 0. Un píxel que toca el plano far de la proyección tiene una profundidad 1. Como cada uno de los objetos en la escena es renderizado, normalmente los píxeles que estén cerca de la cámara se mantendrán, ya que esos objetos bloquean la visión de los objetos detrás de ellos.

El depth buffer puede contener bits de stencil, por esta razón se le llama a menudo *depth-stencil buffer*. El formato de profundidad describe la composición del depth buffer. El depth buffer siempre es de 32 bits, pero esos bits pueden ser arreglados en diferentes maneras, similar a cómo pueden variar los formatos de textura. Un común formato de profundidad es `Depth32`, donde todos los 32 bits son reservados para la información de profundidad. Otro formato común es `DepthFormat.Depth24Stencil8`, donde los 24 bits son reservados para los cálculos de profundidad y 8 bits son usados para el stencil buffer. `DepthFormat.Depth24Stencil8Single` es un formato inusual donde los 24 bits del depth buffer son dispuestos como un valor de punto flotante.⁴⁵

Renderizado: es la representación final de la imagen, a la que han eliminado las partes ocultas y se le han aplicado modelos de iluminación.

Render Target: Es un búfer donde la tarjeta de video dibuja los píxeles de una escena siendo éste renderizado por un `Effect Class`.⁴⁶

Shader: procedimiento de sombreado e iluminación personalizado que permite al artista o programador especificar el renderizado de un vértice o un píxel.⁴⁷

Sprite: Es un bitmap 2D que es dibujado directamente sin usar las transformaciones del pipeline, luces o efectos.⁴⁸

Stencil Buffer: Es similar al depth buffer. De hecho, éste ocupa parte del depth buffer. El stencil buffer permite a los programadores establecer una función de stencil que probará el valor de referencia stencil, un valor global, contra el valor en el stencil buffer cada momento que el píxel es renderizado.

El resultado del test stencil determina si el valor del color del píxel es escrito en la target render, y si el valor de profundidad de ese píxel es escrito en el depth buffer.⁴⁹

Viewport: Es un rectángulo que define cómo se representa una escena 3D en una venta 2D.⁵⁰

⁴⁵ Traducción hecha a partir de la siguiente dirección: <http://msdn.microsoft.com/en-us/library/bb976071.aspx>

⁴⁶ Traducción hecha a partir de la siguiente dirección: <http://msdn.microsoft.com/en-us/library/bb976073.aspx>

⁴⁷ http://www.srg.es/files/apendice_tuberia_programable.pdf

⁴⁸ <http://msdn.microsoft.com/en-us/library/bb203919.aspx>

⁴⁹ Traducción hecha a partir de la siguiente dirección: <http://msdn.microsoft.com/en-us/library/bb976074.aspx>

⁵⁰ [http://msdn.microsoft.com/es-es/library/microsoft.windowsmobile.directx.direct3d.viewport\(VS.90\).aspx](http://msdn.microsoft.com/es-es/library/microsoft.windowsmobile.directx.direct3d.viewport(VS.90).aspx)