

# Segundo Avance PF CVI

Marco Alejandro Ramírez Camacho

14 de mayo de 2025

## 1. Introducción

En esta segunda iteración del proyecto final, he realizado cambios significativos siguiendo las recomendaciones del profesor. He migrado la base del proyecto del tutorial 23 al tutorial 22, lo que ha permitido implementar un sistema de renderizado híbrido (el grueso del tutorial 22) que combina rasterización tradicional con ray tracing en tiempo real, mejorando considerablemente la calidad visual del paisaje urbano generado en la anterior entrega del PF.

## 2. Cambios Implementados

Siguiendo las indicaciones recibidas, he realizado la transición del tutorial 23 al tutorial 22 como base del proyecto. Esta migración no solo ha mantenido la funcionalidad previa de generación con slider de edificios (muchos o pocos), sino que también ha incorporado las ventajas del renderizado híbrido, permitiendo efectos visuales más realistas sobre el agua como reflejos en tiempo real y sombras precisas sobre el agua y sobreapilamiento de sombras de edificios.

## 3. Nuevas Funcionalidades

Respecto a la anterior iteración ahora hay Ray Tracing y Rasterización en todos los objetos de la escena, esto se logró creando las siguientes funciones en el tutorial 22.

### 3.1. CreateBuildingMaterials

Esta función permite la creación de materiales específicos para los edificios, definiendo texturas, colores base y muestreadores. Gracias al sistema híbrido, estos materiales pueden ahora interactuar correctamente tanto con la rasterización como con el ray tracing, permitiendo efectos de reflexión realistas en las fachadas de los edificios.

```
1 void Tutorial22_HybridRendering::CreateBuildingMaterials(
2     uint2 &BuildingMaterialRange,
3     std::vector<HLSL::MaterialAttribs> &Materials,
```

```

4     UInt32 SamplerInd)
5 {
6     // Definicion de materiales para edificios
7 }

```

### 3.2. CreateTexturedBuildingMesh

Esta función genera la geometría 3D para los edificios, creando cubos con normales y coordenadas UV correctas para cada cara. La función ha sido adaptada para que los modelos generados sean compatibles con las estructuras de aceleración necesarias para el ray tracing. Esta parte fue complicada porque, no sabía que si se repetía una cara, el renderizado fallaba y se veía horrible

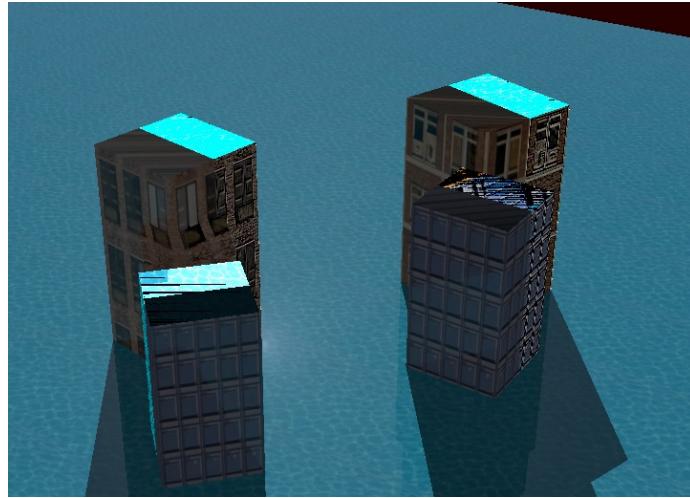


Figura 1: Ejemplo de error con la creacion de los edificios

```

1 Tutorial122_HybridRendering::Mesh
2 Tutorial122_HybridRendering::CreateTexturedBuildingMesh(
3     IRenderDevice *pDevice,
4     float2 UVScale,
5     float3 Dimensions)
6 {
7     // Creacion de geometria compatible con ray tracing
8     // ...
9 }

```

### 3.3. RecreateSceneObjects

Esta función permite la reconstrucción dinámica de los objetos de la escena sin necesidad de reiniciar la aplicación, esta función es la que permite que el slider de edificios funcione, actualizando la escena cuando se utiliza el slider.

## 4. Mejoras Visuales

La implementación del renderizado híbrido ha proporcionado las siguientes mejoras estéticas:

- **Reflejos realistas en el agua:** Ahora el plano de agua muestra reflejos precisos de los edificios gracias al ray tracing.
- **Sombras suaves y precisas:** El sistema híbrido permite calcular sombras más realistas con bordes suaves.
- **Iluminación global aproximada:** Se logra un efecto de iluminación más natural con rebotes de luz entre superficies.

## 5. Imágenes de demostración

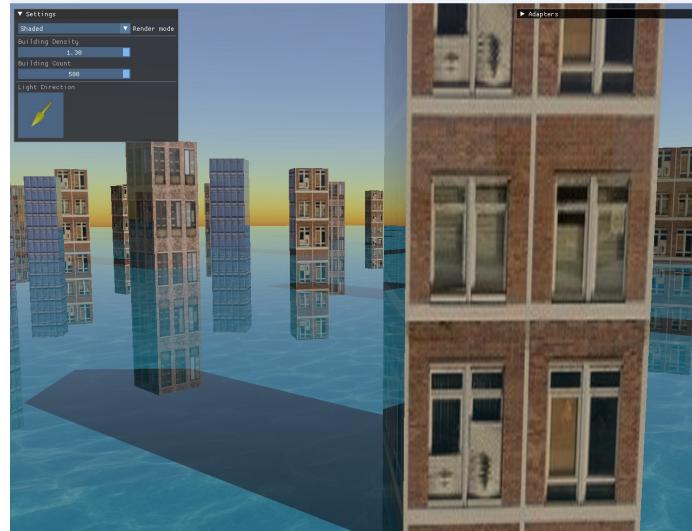


Figura 2: Muestra de sombras de edificios sobre los edificios, y de los edificios contra el agua

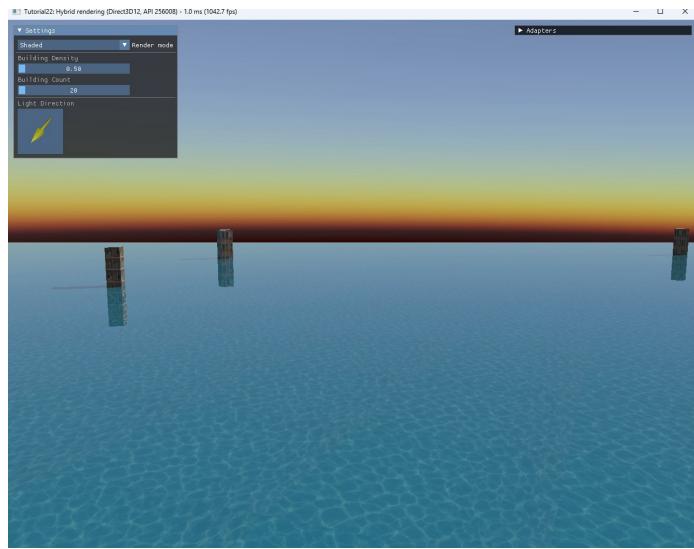


Figura 3: Pocos edificios en la escena

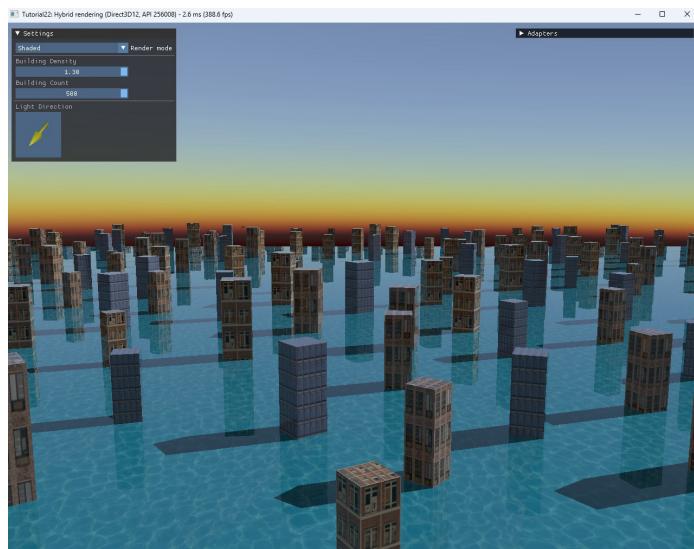


Figura 4: Muchos edificios en la escena