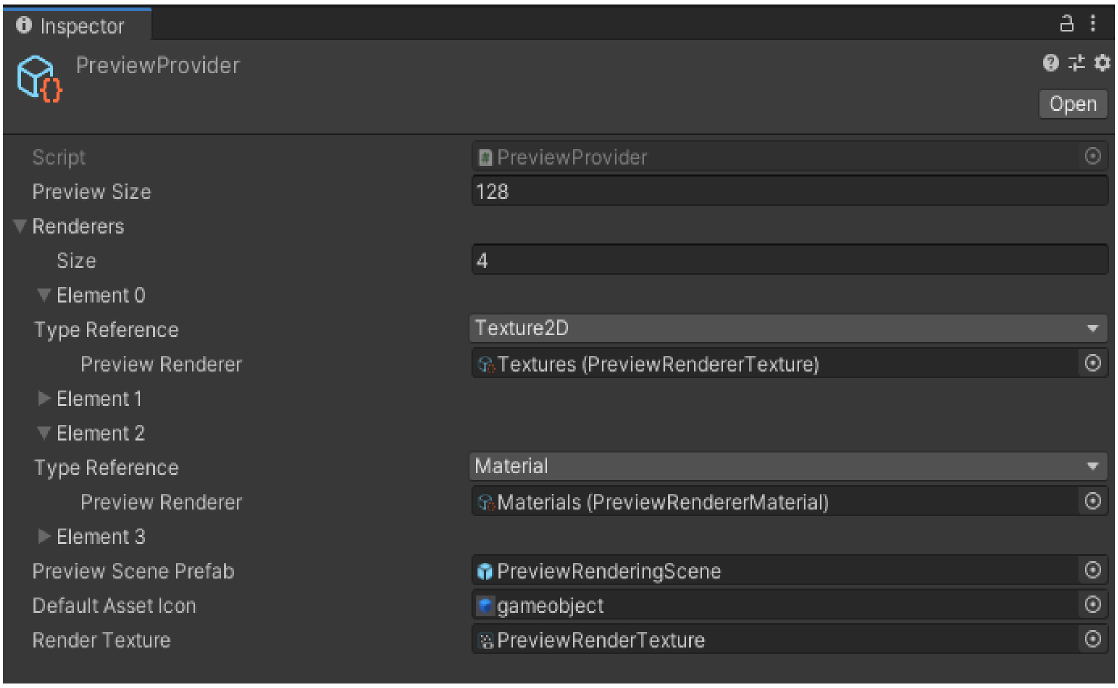
Architecture

Preview provider arhitecture is pretty simple. Object rendering manager and few easy extendable type Renderers, like Material, GameObject and other renderers.

When requesting a preview image from manager it puts request into a request queue, when time has come, it selects proper renderer based on object type. Almost any object type is supported as long as it has respective renderer to render itself onto image.

After image is generated it gets into the cache where it can be accessed later to avoid object rerendering when requested again.



Script	PreviewProvider
Preview Size	128
▼ Renderers	
Size	4
▼ Element 0	
Type Reference	Texture2D
Preview Renderer	Textures (PreviewRendererTexture)
▶ Element 1	
▼ Element 2	
Type Reference	Material
Preview Renderer	Materials (PreviewRendererMaterial)
▶ Element 3	
Preview Scene Prefab	PreviewRenderingScene
Default Asset Icon	gameobject
Render Texture	PreviewRenderTexture

Preview Provider Window

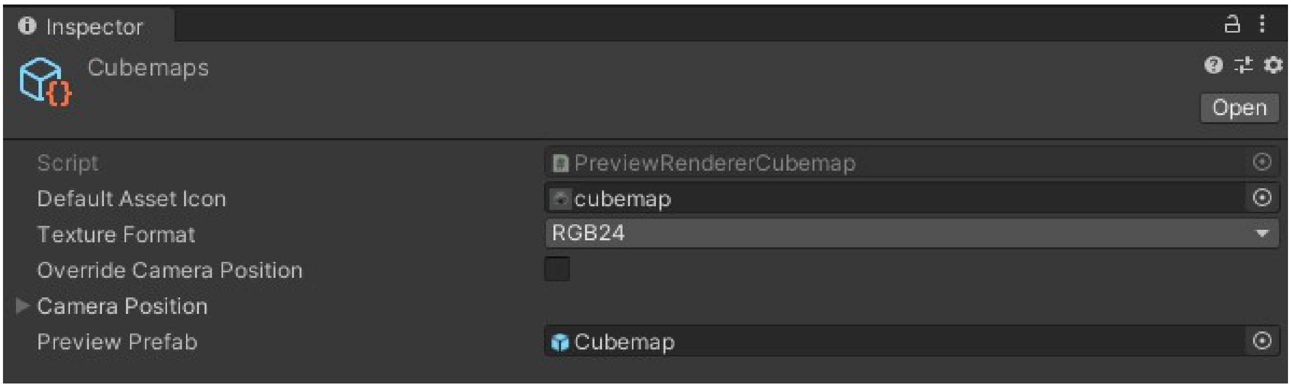
- **Preview Size**. Default Size of Render Texture used to render all objects.
- **Renderers**. List of renderers responsible for setting up and rendering preview object for respective object type.
- **Preview Scene Prefab**. Game Object with Camera and Lighting setup for previews rendering.
- **Default Asset Icon**. In case object rendering failed, it gets default asset icon.
- **Render Texture**. Custom Render Texture instead of default one. Note: if Render Texture is assigned directly no default texture is generated so preview size has no effect.

Renderers

Renderers are objects that know how to render respective type of object.

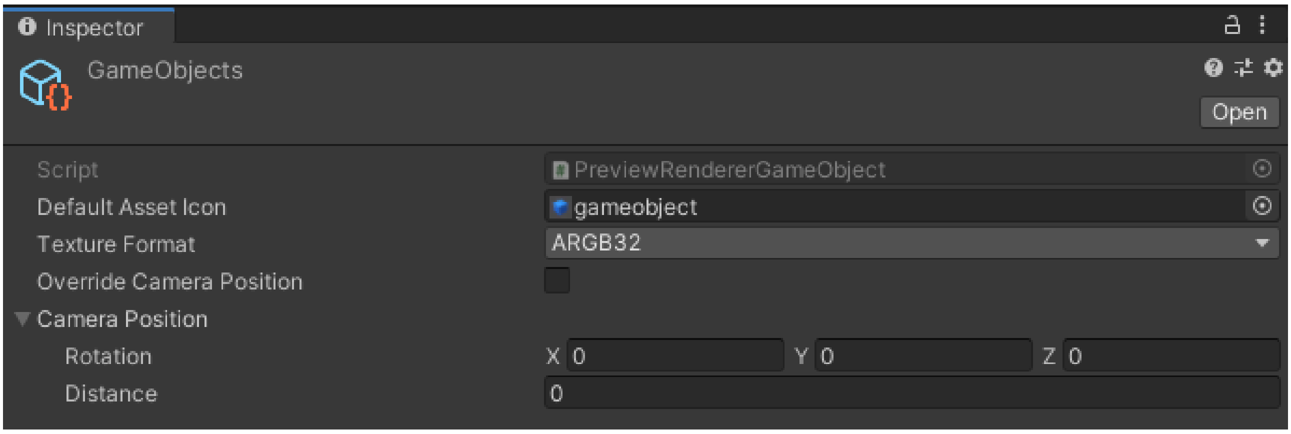
For example **Texture** renderer positions preview quad object on a rendering scene assigns respective texture to it and in case of normal map texture changes shader to properly render it.

GameObject renderer knows how to scale an object to fit preview image. Note that resizing is done based on renderer bounding box size and not on actual visuals, so in some cases object can be rendered smaller.



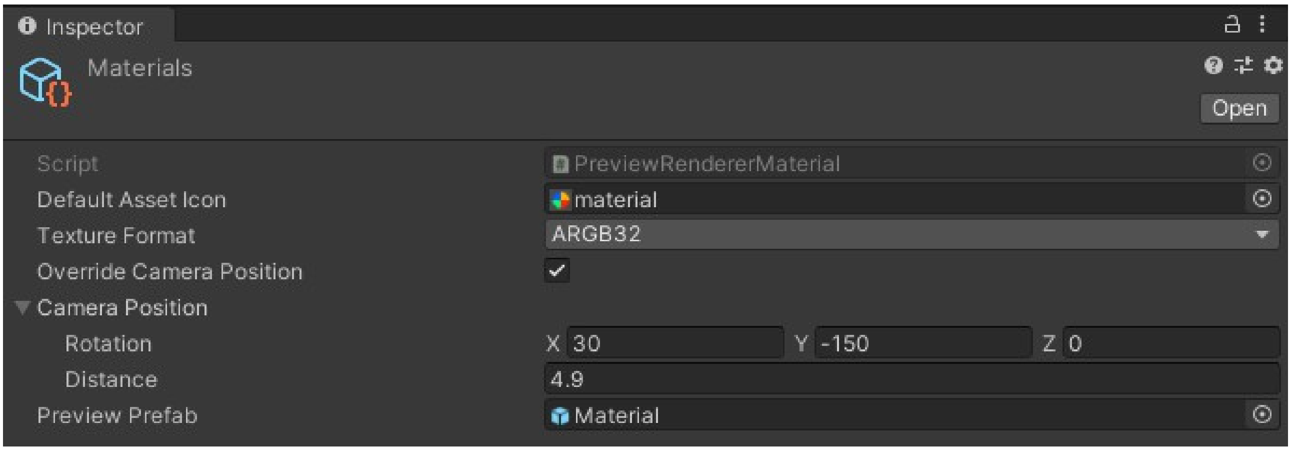
Cubemaps

Script	PreviewRendererCubemap
Default Asset Icon	cubemap
Texture Format	RGB24
Override Camera Position	□
▶ Camera Position	
Preview Prefab	Cubemap



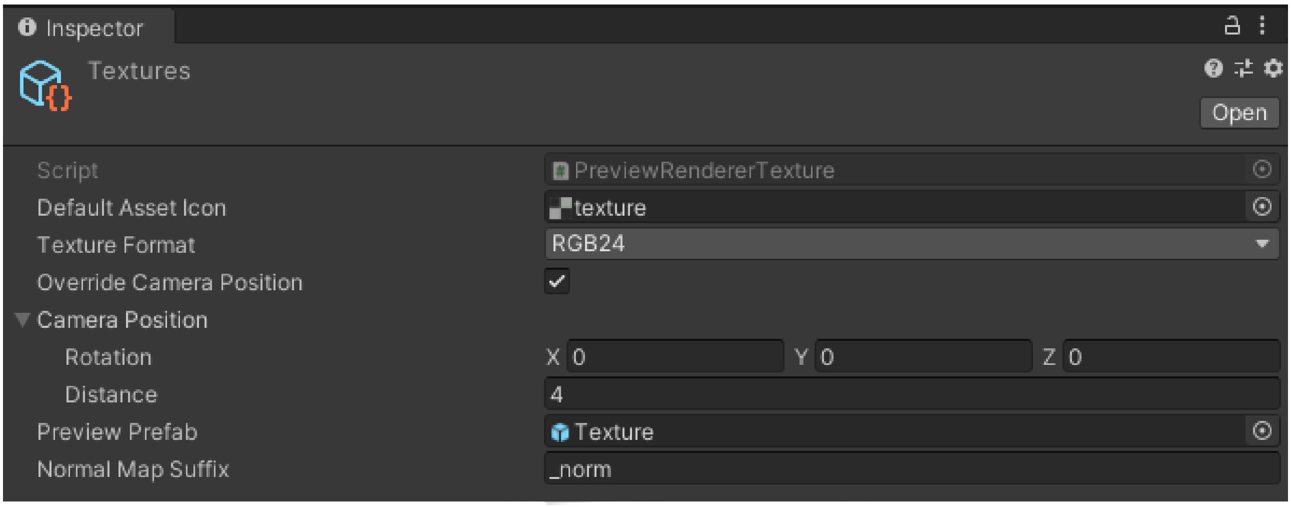
GameObjects

Script	PreviewRendererGameObject
Default Asset Icon	gameobject
Texture Format	ARGB32
Override Camera Position	□
▼ Camera Position	
Rotation	X 0 Y 0 Z 0
Distance	0



Materials

Script	PreviewRendererMaterial
Default Asset Icon	material
Texture Format	ARGB32
Override Camera Position	☑
▼ Camera Position	
Rotation	X 30 Y -150 Z 0
Distance	4.9
Preview Prefab	Material



Textures

Script	PreviewRendererTexture
Default Asset Icon	texture
Texture Format	RGB24
Override Camera Position	☑
▼ Camera Position	
Rotation	X 0 Y 0 Z 0
Distance	4
Preview Prefab	Texture
Normal Map Suffix	_norm

Most properties of renderers are shared:

- **Default Asset Icon**. Default asset icon assigned in case rendering failed.
- **Texture Format**. In case you need preview with transparent background set format to RGBA32 here.
- **Override Camera Position**. Set custom camera position for objects of this type here.
- **Camera Position**. Rotation angles around preview object and distance to it.
- **Preview Prefab**. Game object to use as preview.
  Note texture renderer has Normal Map Suffix property to determine if supplied texture is normal map. See source code for how it works.

Implementing Custom Renderer

To be continued...