Rendering System

The rendering system is the process behind the rendering functionality of the engine. Once per frame, after the event updates, the render system will be called to render the scene.

The rendering system will initialise both renderers passing the various configuration values by retrieving them from the Resource Manager. See the resource manager document on how to store config values in the config json. The rendering system will also set both renderer’s texture unit managers and units which both renderers share from the same instance.

The render system will get a list of entities if the entity list has been updated since last frame or it will use the old list. This list is retrieved by going through each layer in the scene and getting the renderable entities. It will then get the list of render passes (per scene) and call their render function and pass them this entity list and it will do so in the order of the render passes which is insertion ordered.

The render passes which holds a framebuffer can then bind the framebuffer, set any OpenGL states and begin calling the renderers. Typically, the pass must check each entity to see if it is displayed. If so, it will call the entity’s onRender function which will let the entity, by subtype, handle what exactly gets submitted to the renderer.