VR Experiment Sandbox (VRES)

Library for building and experimenting with different VR concepts and rendering techniques

Table of Contents

[2 Graphics 1](#_Toc390624344)

[2.1 Requirements 1](#_Toc390624345)

[2.2 High Level Architecture 1](#_Toc390624346)

# Graphics

## Requirements

The graphics system should be flexible and decoupled, to allow maximum experimentation and implementation freedom without modifying the test applications and scenes. The following are the basic requirements:

1. Abstract away specific shaders, and use a material description format instead. The backend will have a way to map these to available shaders.
2. Scenes will be comprised of various model objects, each describing collections of grouped geometry and materials.
3. Each scene will also include lighting information, with various types of lights supported (ambient, directional, point, spot, and possibly more eventually)
4. In a later version, but not required for v1, the scene will support describing particle systems.
5. Finally, a VR camera will be associated with the scene at runtime. This VR camera models the various data required to properly render the scene in stereo on an HMD.
6. VRES is a closed system, where the test app only implements logic and doesn’t own creation/management of the graphics device or resources.
7. The graphics system will also need to support basic font rendering for things like UI and on screen text.

## High Level Architecture

At the core of the system are the System, Application and Renderer types. These encapsulate all of the main system management, application base class functionality, and graphics resource creation & rendering. The other common types are Transform, Scene, Model, Material, Texture, and Light.