

VR Framework based on Open Source Toolkits: Lessons Learned from Decades of Experience

Thomas Wischgoll

Commercial software environments for operating a virtual reality laboratory can provide a significant amount of flexibility and ease of entry. However, these closed source environments can come with their disadvantages, such as price, lack of support for older hardware after software updates, and limitations or lack of flexibility in supporting some hardware or software configurations. The Appenzeller Visualization Laboratory at Wright State University supports different commercial environments, including Unity, Unreal, and Virtualis' Visionary Renderer. However, open source solutions can provide more flexibility when supporting a wide range of display systems and pushing the envelope on supporting rare or home-grown hardware.

VRUI[1] is one example of such an open source framework for supporting virtual reality applications. It supports a wide range of display systems and can be configured for various different configurations both in terms of hardware support as well as display configuration. We expanded VRUI's capabilities in various ways by adding support for touch-enabled devices and combining it with additional open source toolkits to expand on VRUI's capabilities. This includes integration of VTK[2] as well as OpenSceneGraph[3] extended by using the bullet physics engine to support physical behavior of objects within the virtual environment.

This presentation will outline the range of display systems[4] supported by this open source-based software environment, including CAVE-type systems such as the DIVE[5] and Virtualis ActiveCube, as well as provide more details on the specifics of the software framework and its integration of the various components. Some use cases of the open source framework will be presented to outline its flexibility and utility as well as the pros and cons.

References:

- [1] Oliver Kreylos. "Environment-independent VR development." In *International Symposium on Visual Computing*, pp. 901-912. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008.
- [2] William J. Schroeder, Lisa Sobierajski Avila, and William Hoffman. "Visualizing with VTK: a tutorial." *IEEE Computer graphics and applications* 20, no. 5 (2000): 20-27.
- [3] Don Burns, and Robert Osfield. "Open scene graph a: Introduction, b: Examples and applications." In *Virtual Reality Conference, IEEE*, pp. 265-265. IEEE Computer Society, 2004.
- [4] Thomas Wischgoll: *Display Systems for Visualization and Simulation in Virtual Environments*, Visualization and Data Analysis, pp. 78-88, 2017.
- [5] Thomas Wischgoll, Madison Glines, Tyler Whitlock, Bradley R. Guthrie, Corinne M. Mowrey, Pratik J. Parikh, John Flach: *Display infrastructure for virtual environments (DIVE)*, Journal of Imaging Science and Technology, 61(6), pp. 60406-1-60406-11, 2017.