Exploration of Virtual Reality and Deferred Immediate Mode

Jonathan Sumner Evans, Robinson Merillat, and Sam Sartor

Department of Computer Science, Colorado School of Mines

Goals

We set out to find a system that provides a modern, fast, and practical approach to virtual reality development. Specifically, we needed a framework which met the following criteria:

Performant VR requires at least 90 frames per second to run smoothly. Low frame rates can cause users to experience headaches and nausea faster than when at high frame rates [1]. This requires VR programs to be highly optimized and multi-threadable.

Natural VR enables new user interfaces where components are organized within a 3D space. We wanted such components to be first class.

Flexible We need a general purpose user interface toolkit designed specifically for VR.

Modular We need a toolkit which does not include unnecessary features, but is extensible with modular components.

Exploration

Over the course of many months, we explored several existing VR frameworks and application architectures. We determined that none of them meet our criteria for VR development.

We created our own user interface and rendering framework which addresses the problems that we encountered during our exploration.

We initially made use of the immediate mode program architecture because of its flexibility and extensibility, but repeatedly ran into the following problem:

There are some questions about the state of the system which cannot be answered until all system elements have "reported" their state.

To solve this problem, we created a powerful new program architecture called **Deferred Immediate Mode (DIM)**. DIM adds *deferrability* to the classic immediate mode architecture while retaining flexibility and solving our issues with interdependent user interface elements.

The DIM architecture enabled us to meet all of our goals for effective VR application development.

Flight

Our virtual reality toolkit, called flight, features high level abstractions for interacting with VR hardware, a state-of-the-art real-time rendering engine, asset loading tools, and a selection of built-in UI elements. Flight is designed from the ground up to be performant, general, and modular.

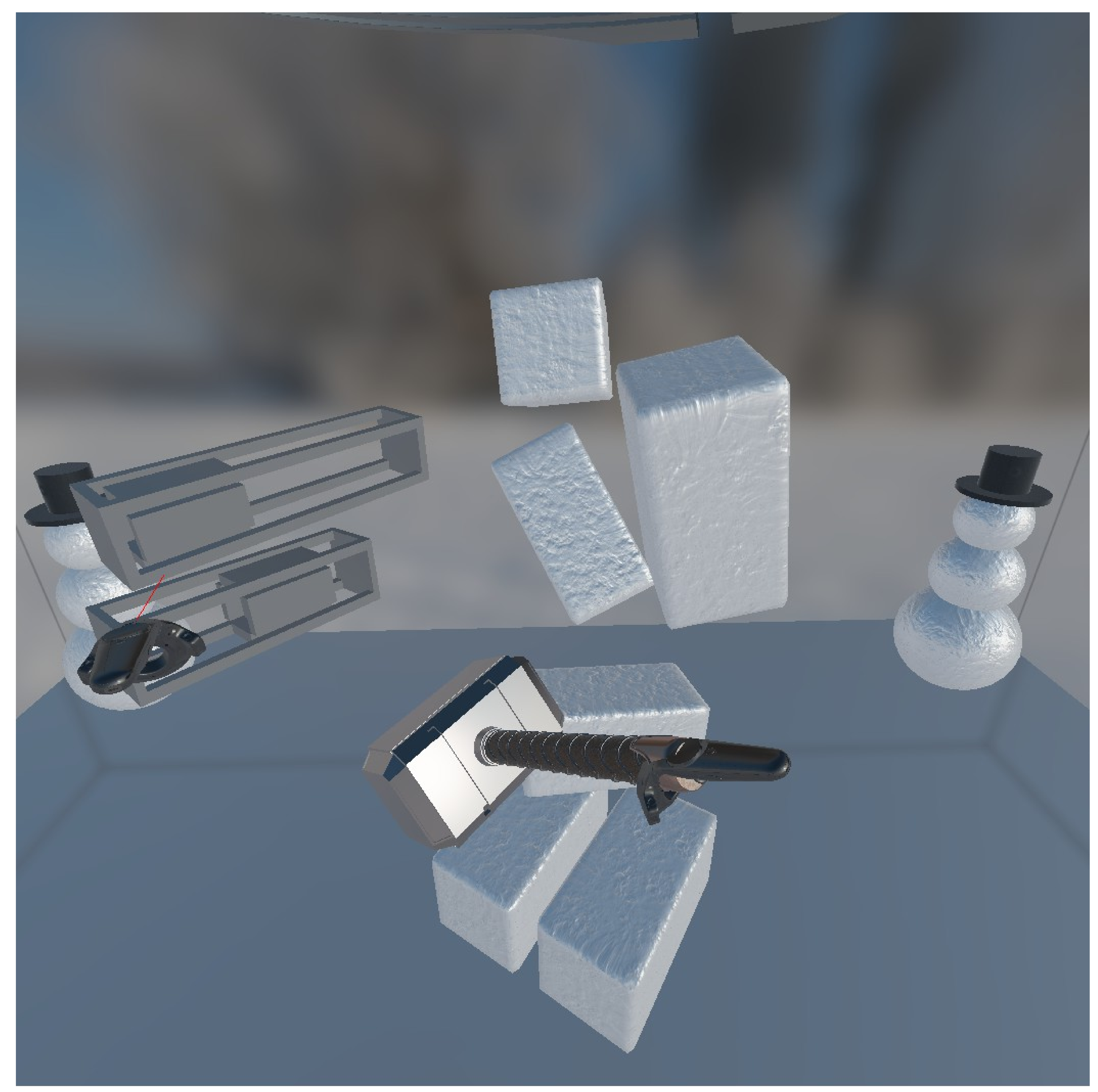


Figure: Our Final Project

References

[1] *The Importance of Frame Rates.* URL: https://help.irisvr.com/hc/en-us/articles/215884547-The-Importance-of-Frame-Rates.

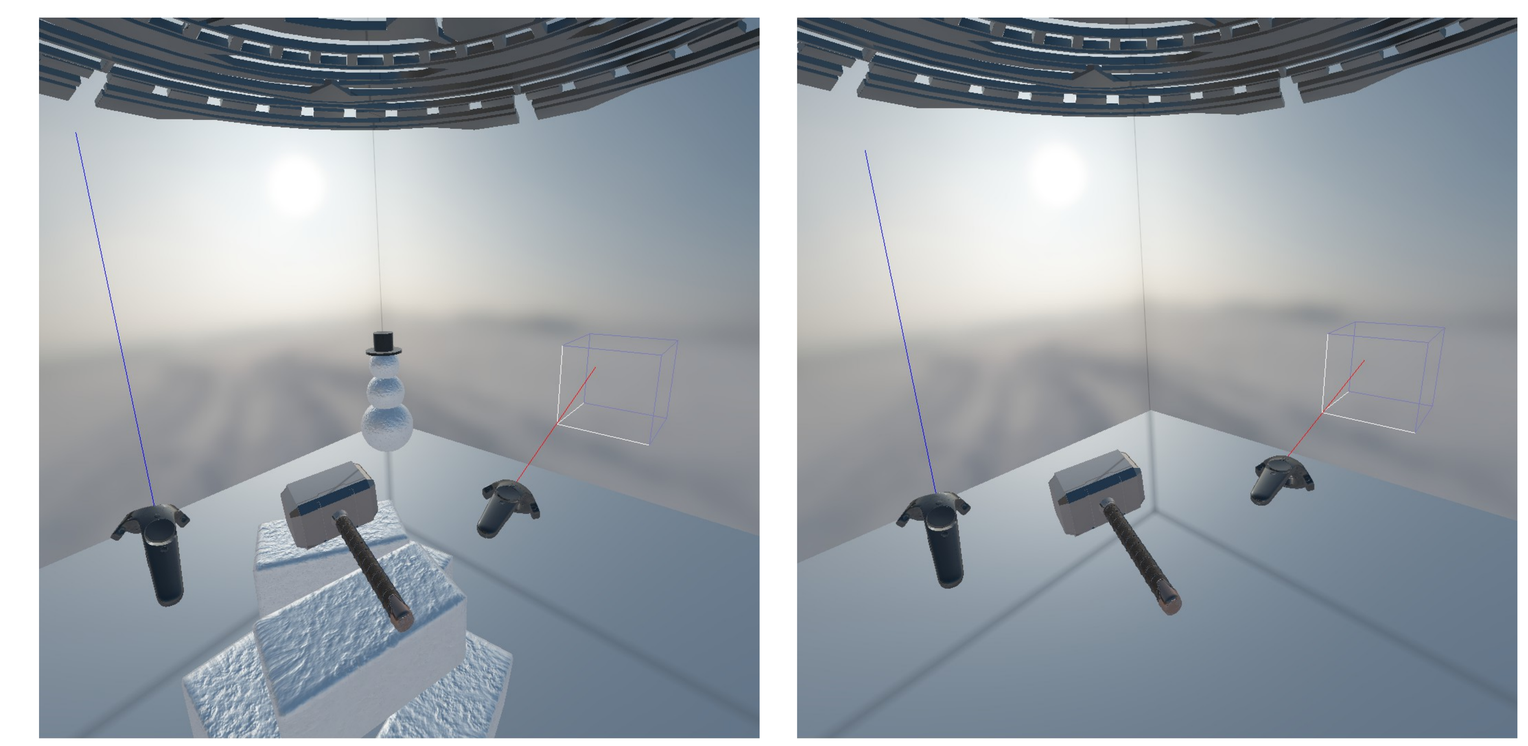
More Information

Project	github.com/CSM-Dream-Team/final-project
Flight	github.com/flight-rs/flight

Final Project

Our final project demonstrates the results of using DIM to implement a complex VR application. The environment currently provides these features:

Modularity: Each part of the application is its own isolated module that can be modified and even turned off without interfering with other modules.

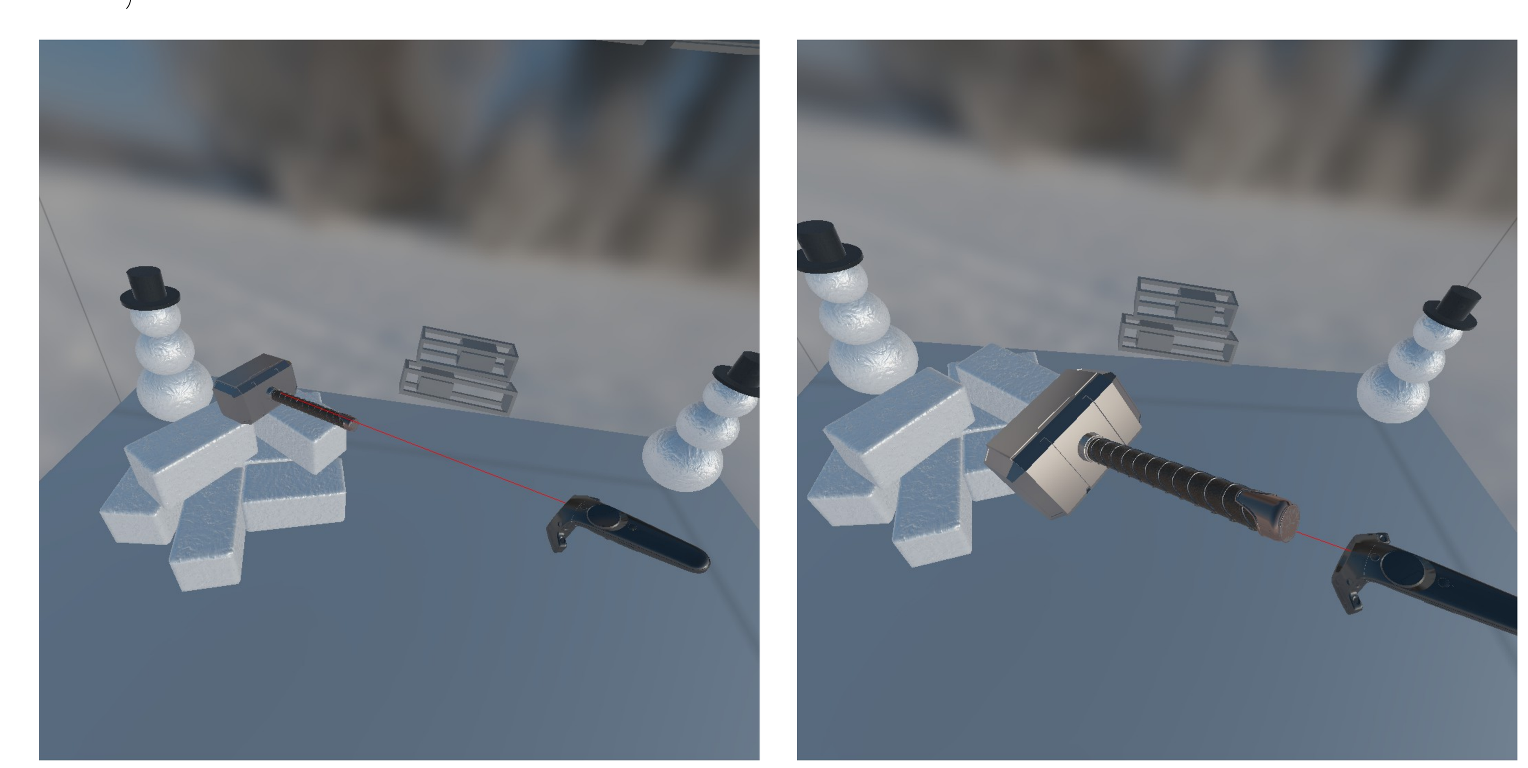


(a) Before (b) After

Figure: Application Toggles

Inter-Application Physics: Althogh each application is in a seprate module, DIM allows them to interact seemlessly. For example, Mjolnir can hit snowblocks.

Yanking, Grabbing, and Pointing: All elements which can be grabbed, yanked, or pointed at are manipulated using a common, intuitive user interface scheme.



(a) Before (b) After

Figure: Yanking Mjolnir from a Distance