

BumpTop Desktop

Student: Faham Negini
NSID: fan780
Course: CMPT 811, HCI
Instructor: Carl Gutwin
Milestone: 2 - Initial Demo

Overview

BumpTop Desktop is about a 3D desktop environment to be used instead of the traditionally 2D desktop. BumpTop Desktop features some real world physical relationships between documents on a desk like weight, volume, friction and etc.

Language and Tools

I'm using C++ as the main programming language of this project, OGRE as the 3D rendering engine, Bullet as the physics simulation engine and OgreBullet as a think wrapper which connects OGRE to Bullet and OIS as the input library. KDevelop4 is used as the main IDE and CMake scripting language is used to setup the projects configuration files.

Software Architecture

It is decided for the project to be mainly created on top of OGRE library that it features lots of basic needed subsystems like resource management, scene management, material scripting language, log system and etc. Publisher/subscriber relationship is widely used for handling subjects like input events by OIS, frame rendering events by OGRE and window manager events like WindowResized. There is a thin wrapper layer called OgreBullet, used for synchronization between visual and physical representation of the scene. according to demanded features, this is not only the physical calculations which are applied to visual, but also sometimes new transformation state is decided by visual layer and then is applied to the physics, then this is a 2 way synchronization.

Main Modules

There is a Desktop module which runs the main rendering loop and also is a frame and input listener, it receives different frame and input events (i.e. FrameStarted, FrameEnded, FrameRenderingQueued, MousePressed, KeyPressed and etc.) and implement some general features of the application like the Camera system, selection and transformation of objects. Object is another core module which abstracts any kind of objects which are going to be spawned into Desktop, this module encapsulates both visual and physical features of an object and talks to operating system whenever the object is executed (like when it is double clicked). Pile module is going to provide pile operation capabilities, user create and destroy piles of objects through different scenarios like Drag-N-Cross.

Risk Areas

The most risky operation seems to be applying visually calculated transformations onto physics world, this is in contradiction with what the physics engine desire and may lead to some unknown circumstances.

Timeline

Main features yet remained to be implemented are communication between Objects and OS, and the Pile system, and I hope to implement them until Nov. 18, to get ready for evaluation.