3-D 2048 Final Specifications

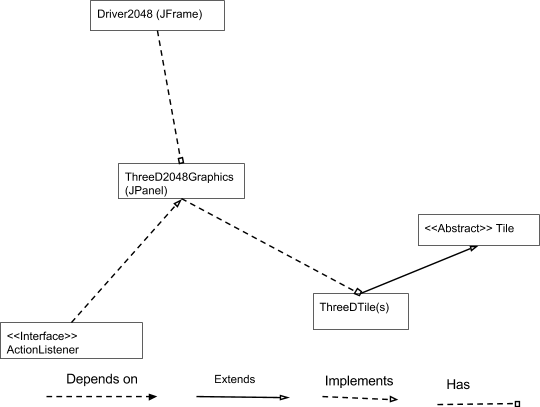
Project Overview:

Ever hear of that addictive game that everyone is playing? Imagine if you could play it in 3-Dimensions. This version of 2048 will do just that. It’s not like the other 3D versions out there which are just 3-D behind the scenes. Arrow keys will be used to move left, right, up, and down and the letters q and e will be used to move tiles into and out of the screen respectively.

Structural Design:

ThreeD2048 will be implemented with a 3D array that holds all of the active ThreeDTile objects. The addition of numbers between tiles will work the same as it does for the 2-D version, but there will more directions for the tile values to move around within the array. Each value in the 3-D array of tiles, will have a corresponding location on the Java3D canvas.

Object Oriented Design:





GUI Interface Description:

The 3-D version of the game will be run on a JPanel which will be added to a JFrame. The Java3D class will handle the heavy lifting in terms of graphics implementation. On the screen, will be 64 blank spaces for tiles and 2 instances of 3D-tiles to begin with. As the game progresses, more tiles will be added from different sides. There will be timer on the JFrame to keep track of game progress.

User Interaction:

The player will be able to translate the 2048 and 2048-3D tiles across all 2 and 3 axes respectively. The player can move cubes up, down, left and right with the arrow keys, and in and out with Q and E, accordingly in the 3D-version.

Testing Mechanism:

We will create JUnit tests for Tile3D, ThreeD2048Graphics, and Driver2048.