

Game Design Using A 3D Engine

Paul Bartholomew
Jason Ruvinsky

CS 475 Senior Project
May 2, 2008



Summary

Motivation: We wanted to apply our programming knowledge to create a major project. We both desired to design and build a video game. We also wanted to learn how to use a 3D Engine.

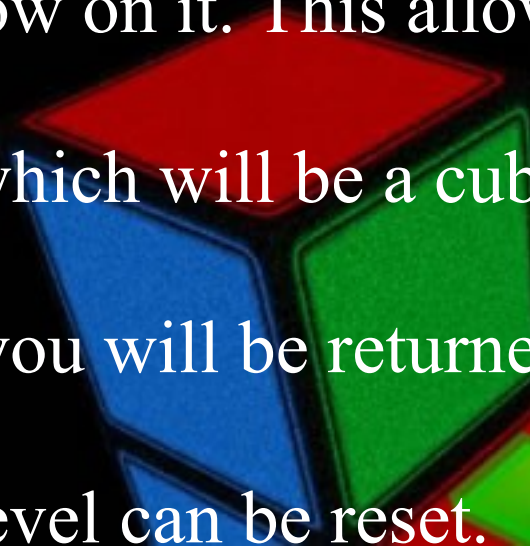
Problem: Designing and developing a game that is unique, easily understood, and fun.

Solution: We designed Cube World.

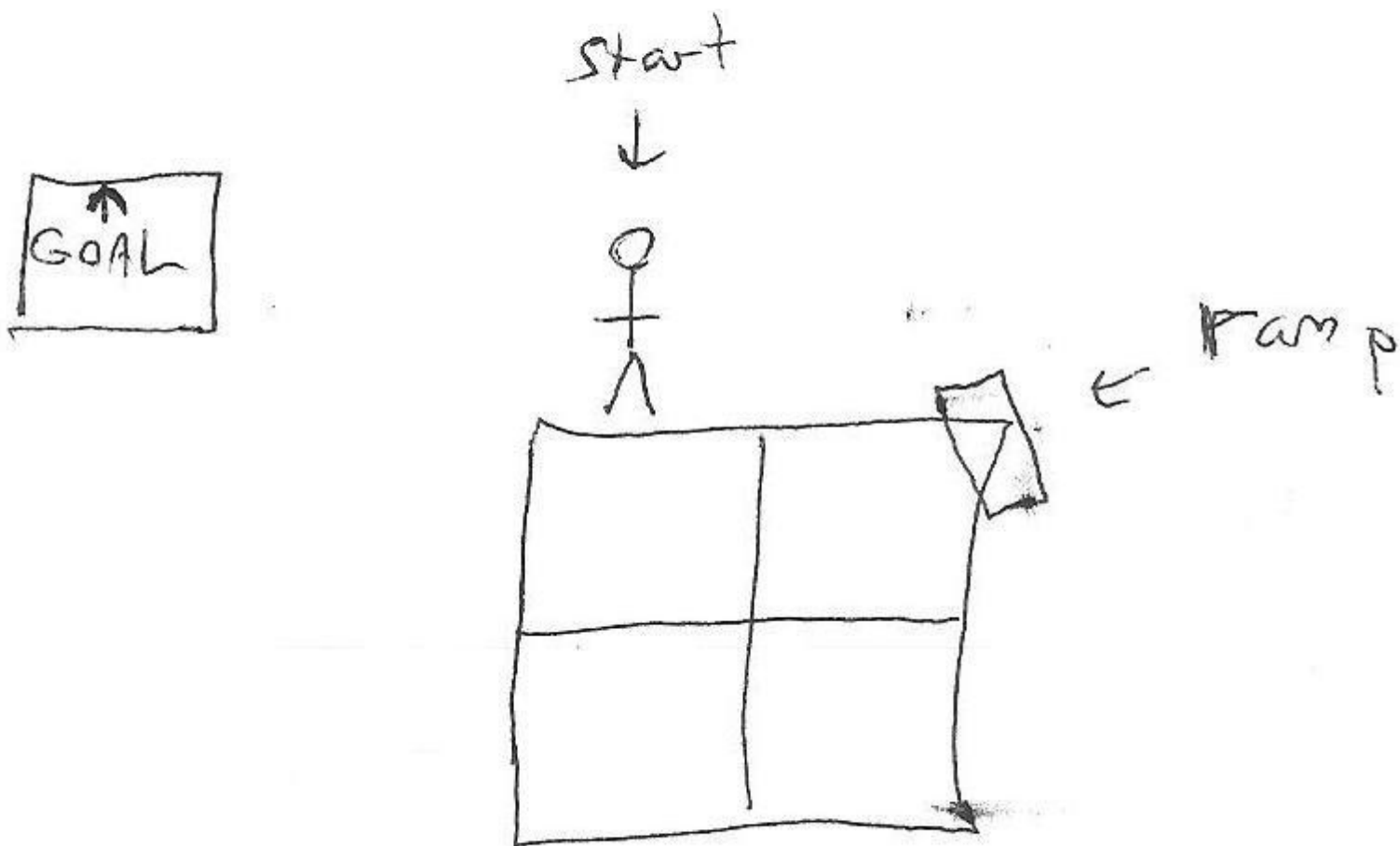


Our Game Design: Cube World

- 3-dimensional platformer / puzzle game with a 3rd person perspective.
- Game world consists of cubes that may be moved by the player character by stomping them downwards.
- The direction of gravity can be changed by walking over a special ramp block, which has an arrow on it. This allows the player to stand on another cube face.
- The object will be to reach a goal, which will be a cube that will have a star on its face.
- No death. If you fall into oblivion, you will be returned to the last place you were standing.
- One move can be undone, and the level can be reset.



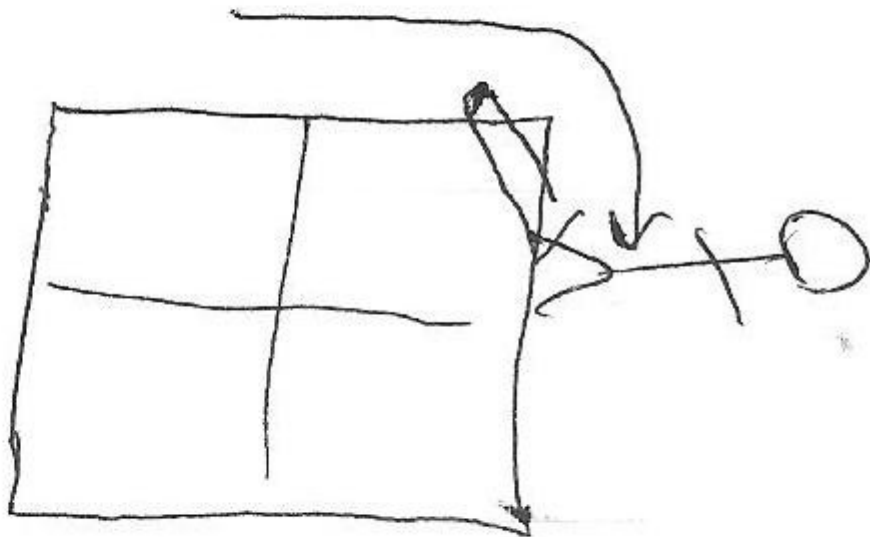
①



* can't reach goal

2

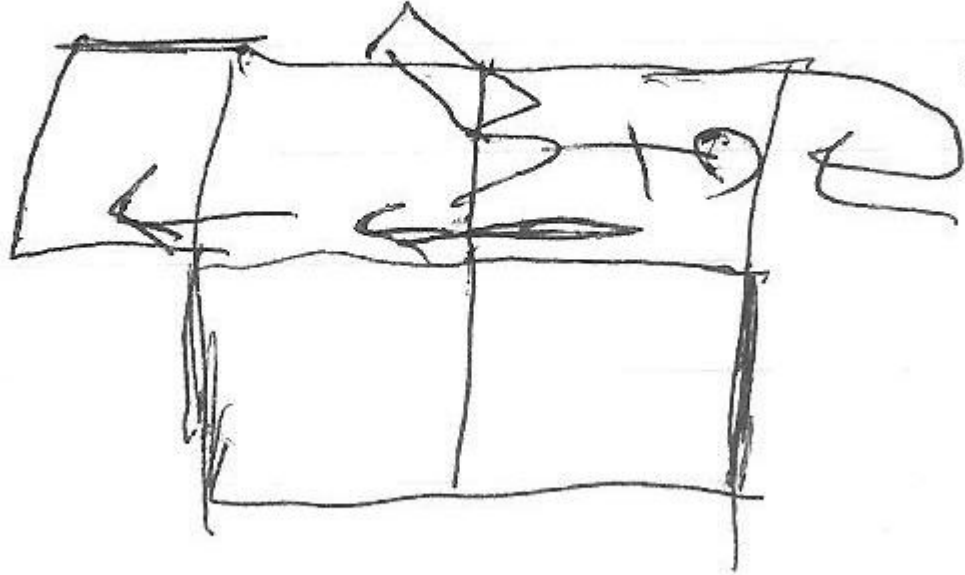
↑
God



* ramp changes gravity
when walked over

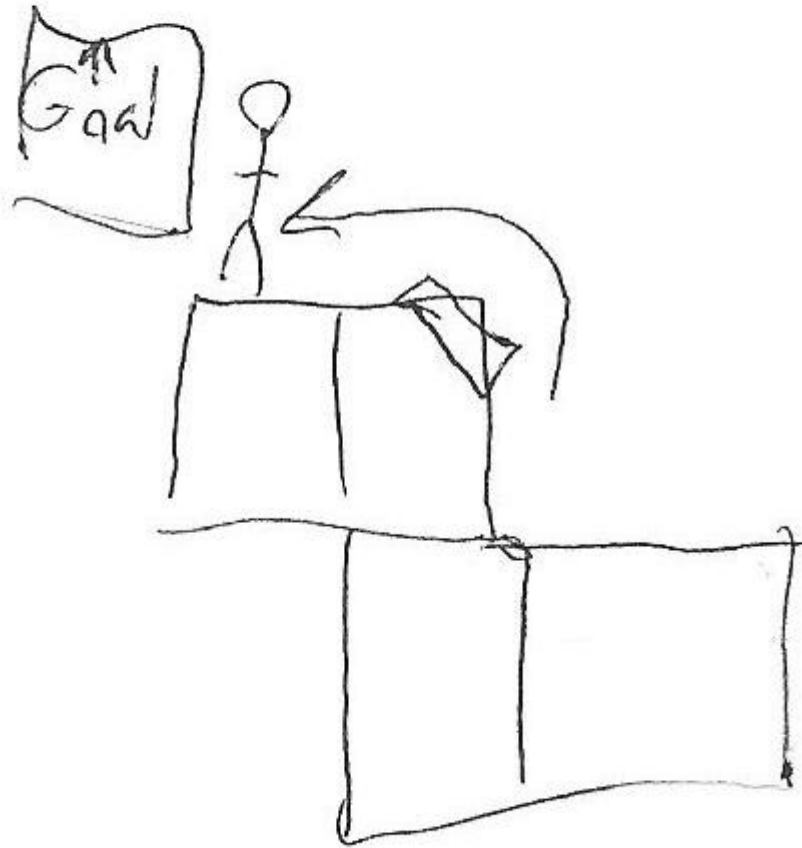
3

↑
Goal!



↑
Kamp

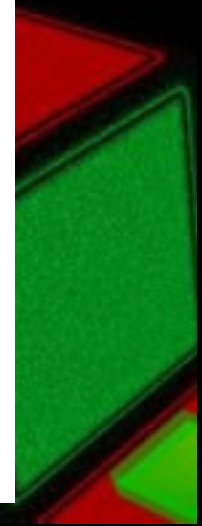
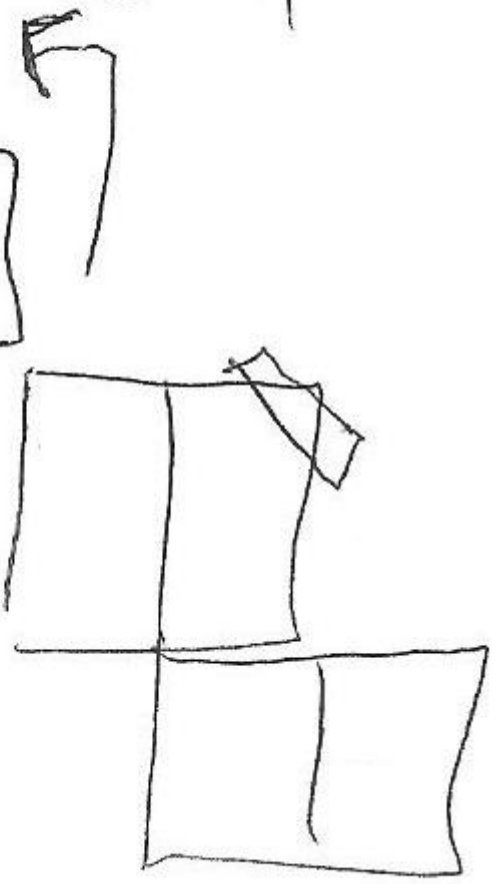
4



5



Jump



storyboard.lev (the level definition)

x:0 y:-100 z:0

x:-800 y:100 z:0 goal:top

x:0 y:-300 z:0

x:200 y:-100 z:0 ramp:top-right

x:200 y:-300 z:0

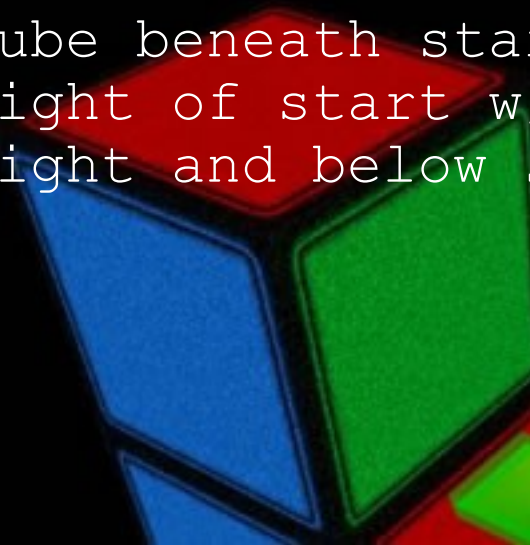
starting cube

goal cube too far to reach

cube beneath start cube

Right of start w/ rt ramp

Right and below start cube



Work Performed by Midterm

- Designed game and created rules and a conceptual story board
- Learned Ogre3D Engine by following tutorials
- Created program that initializes the 3D environment
- Created basic movement listener
- Created game options class (includes key mapping)
- Wrote documentation on how to play



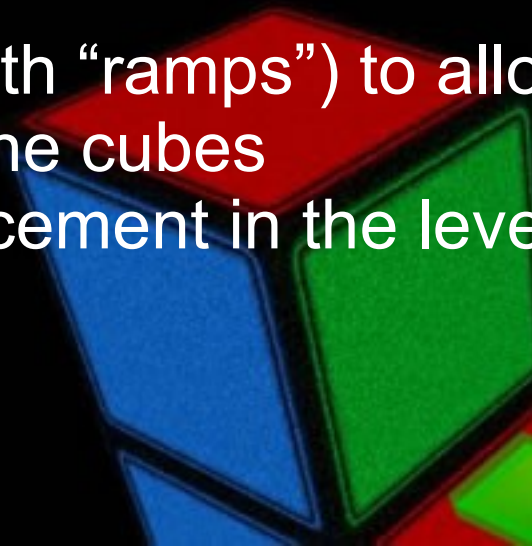
Work Performed since Midterm

- Drew textures for all game objects
- Generated cubes with proper textures
- Created a collision detection engine
- Created movement engine
- Created level loader to generate levels from a file
- Built a game engine to spec of our original design plans (see next slide)
- Built functionality for playing multiple levels
- Wrote game manual



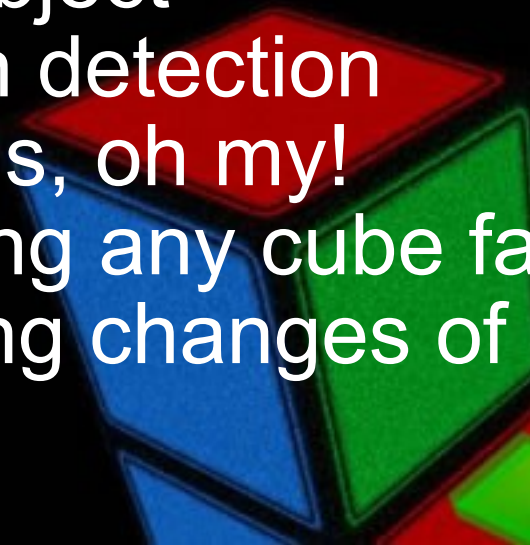
Building the Game Engine

- Loads cubes based on locations given in file
- Movement and collision detection integrated to create gameplay
- Changing orientation (via edges with “ramps”) to allow player to walk on different faces of the cubes
- Stomping cubes to affect cube placement in the level
- Winning by reaching a goal



Obstacles Overcome

- Taking off the training wheels: Getting Ogre3D to work without the “example” classes
- Will it blend?: Learning how to use Blender to model in 3d to create mesh object
- Bang head on wall: Collision detection
- Crashing, Glitches, and Bugs, oh my!
- Cube Orientation 101: Making any cube face behave the same, and allowing changes of orientation



What can be added for the next version?

- GUI including menu system and game interface
- Resolve dangling pointers created by loading a new level
- Improve collision detection to 'predict' collisions, and not break held key movement while jumping or falling
- Impose limits on camera zooming feature
- Clean up code, consolidate classes
- Eye candy! Including background
- Gamepad control option
- Music and sound effects
- Design more levels :-)



What was learned

- Implementing a game with a 3d engine
- How to work against deadlines for a large project
- Proper pacing for large projects
- Seeing a project from conception through completion
- How to divide the workload in a collaborative environment



Resources

We used the Ogre3D tutorials available at:
http://www.ogre3d.org/wiki/index.php/Ogre_Tutorials
To learn how to program with the Ogre3D Engine.

The C++ Resources Network
<http://www.cplusplus.com>

Influences:

Portal, published and developed by Valve Software
Super Mario Galaxy, published and developed by
Nintendo

