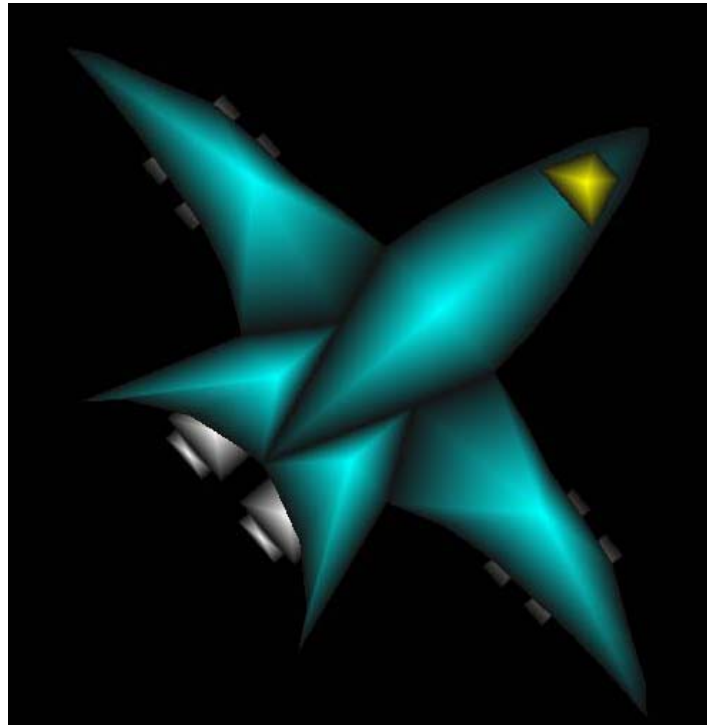


The Last Starfighter Manual/Report

CSS 450 Final Project

December 9, 2008

Joseph Chouinard
Dan Heringer



The Goal of the Game

This program is a one or two player space fighting game. It consists of multiple Ship designs with each ship having different capabilities. It also spans two scenes for game play.

At all times the players have the option to choose the type of boundary they want to move around in, as well as the type of ship they wish to use. During two player control the players can choose between a fixed boundary (the default) and a world wrapped boundary that will allow movement without changing the size or position of the world model. While in one player control, the player also has the option to scroll the world and allow for larger “searching” of the world model.

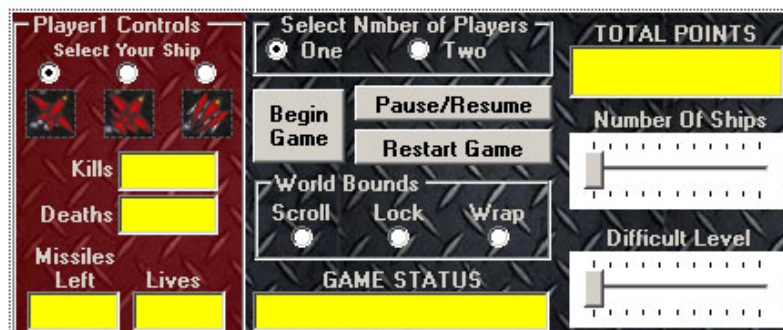
There are two small windows located at the bottom corners of the screen. During play the player1 ship is in focus on the left side and the current player1 target (whether player2 or the currently selected AI ship).

One Player Functionality (see Fig. A)

When in single player mode, the player is faced against pre-programmed AI ships that will follow and attack the player. You are awarded 100 points for every ship you kill with a multiplicative bonus for how many ships you have killed since your ship died. There are 2 levels which alternate as the number of enemies you face and the challenges of the environment increase. Once you have advanced through 8 stages you begin to face addition waves of ships for every stage. You start with 2 lives and are awarded a life for every stage that you clear.

You can bypass the regular difficulty progression by adjusting the single player slider bars. The slider controls allow the player to choose the number of ships to face at any time. There is also a difficulty slider that controls the amount of difficulty the player wants to face within the level. (Note the sliderbar value changes do not take effect until the environment is reloaded by clicking the “begin button.”) Echo areas display the number of kills, deaths, lives, and missiles left for the player. Game ends when player life reaches zero.

Figure A: GameControls for one player game



Two Player Functionality (see Fig. B)

When in two player mode, the players face off against each other, where the goal is to destroy the opponent starship. Players begin with six lives and the game will end when one of the players has lost all of their life. Echo areas display the number of kills, deaths, lives, and missiles left for each player.

Figure B: GameControls for two player game



Game Controls

The game includes Begin Game, Pause/Resume Game, and Restart Game functionality. By default a game is ended when the number of players is changed.

BeginGame Button: Will start a new game and if in on-player mode, will update the game using the slider bar values. This button will not change the level.

Pause/Resume Button: Will pause and resume game play.

Restart Button: Will restart a currently running game and will set all the settings to first level. And change the scene.

Select Your Ship Radio Buttons: Each player has the option of three ships to select with varying movement controls and weapon capability.

Select the World Bound Options: The game allows for three options for world bounds during on-player mode, and two options for two-player mode.

Players have the option to use either the keyboard or USB controller functionality for controlling the ships. The mouse functionality is only used for accessing the game controls located within the form. During one player mode the player can change choose different AI ships to target by pressing the “Tab” key on the keyboard, or the “Y” button on the controller.

Player 1

Model Response	Keyboard	Controller
Accelerate Ship	“W” Key	Right Trigger
Reduce Ship Speed	“S” Key	Left Trigger
Turn/Rotate Left	“A” Key	Left Stick Left
Turn/Rotate Right	“D” Key	Left Stick Right
Fire Primary Weapon	“Spacebar” Key	“X” button
Fire Secondary Weapon	“F” Key	“B” button
Resize Player Window		Right/Left BTN
Pause/Resume Function		Start Button
Select New Target	“Tab”	“Y” Button

Player 2

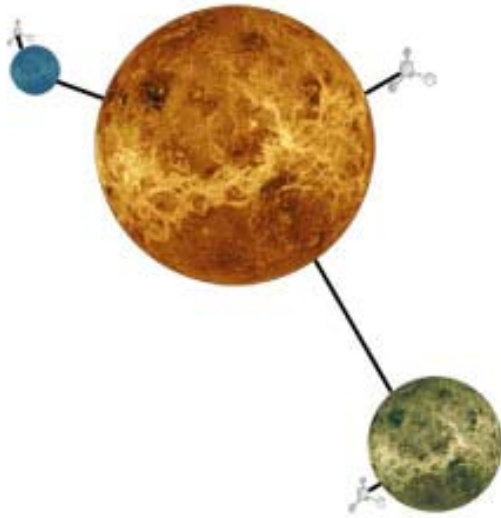
Model Response	Keyboard	Controller
Accelerate Ship	“Up” Key	Right Trigger
Reduce Ship Speed	“Down” Key	Left Trigger
Turn/Rotate Left	“Left” Key	Left Stick Left
Turn/Rotate Right	“Right” Key	Left Stick Right
Fire Primary Weapon	“Delete” Key	“X” button
Fire Secondary Weapon	“Insert” Key	“B” button
Resize Player Window		Right/Left BTN
Pause/Resume Function		Start Button
Select New Target	NA in Verses	“Y” Button

Object Interactions

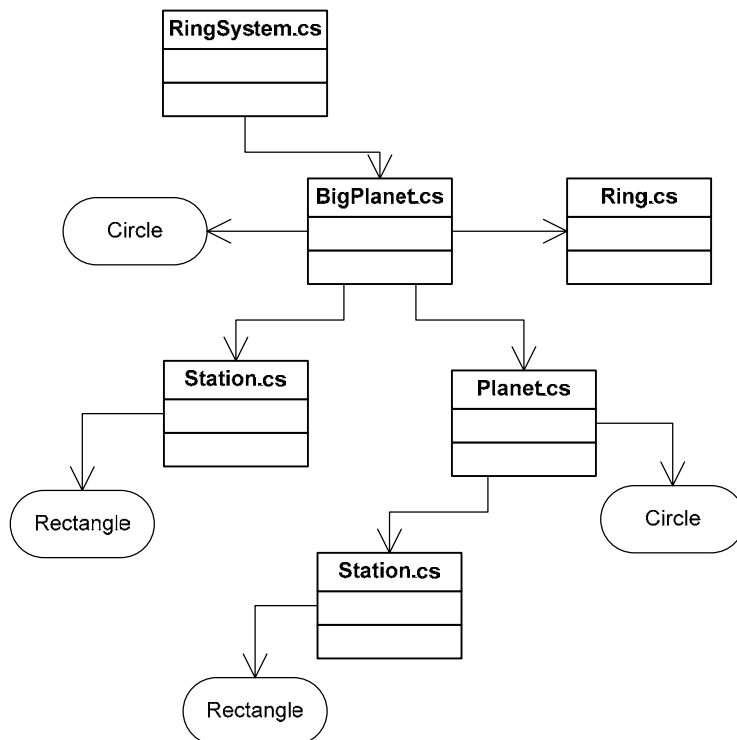
- StarShips:
 - o One or 2 Star ships can be controlled by the players. There are 3 types of ships, all of which can fire blasters and missiles at different rates
 - o StarShips are able to avoid each other in close proximity, but receive damage from colliding with blasters, missiles, planets, Stars, and black holes.
- Planets:
 - o Maintain an orbit and have a gravitational force that effect’s starships in the world. The Gravitational force is approximately equal to the Gravitational constant * Planet Area/ Distance Squared.
 - o Plants will blow up into many particles if that receive enough damage. (1000 hit points around 100 shoots to kill)
- Stars:
 - o Constantly spawn particles that do extreme damage to starships. Depending on the difficulty level stars will grow and explode, thus encouraging the user to dispatch his enemies quickly.
 - o Once a star goes supernova it turns into a black hole that sucks in starships.
- Space Station:
 - o Orbit planets and fire heat seeking missiles at the user ships. Higher difficulty levels increase the rate of missiles fired from space station.

Program/Hero Design

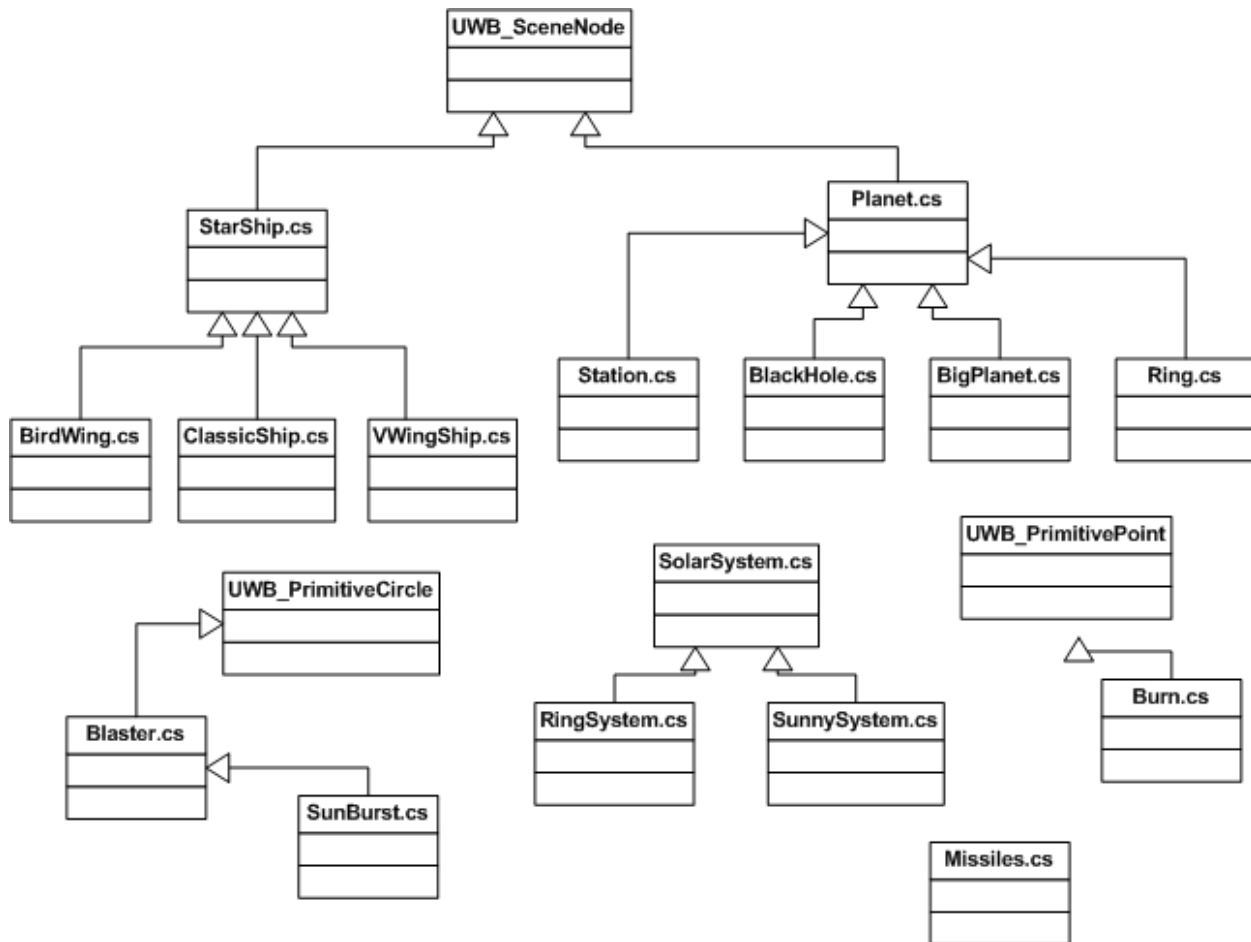
Layout of the Single SceneNode Instance Geometry



Single Instance of the SceneNode Data Structure



Program Structure



System Evaluation

Program Limitations

- We ran into trouble with game play frame rate multiple times
 - Amount of primitives drawn on the screen
 - Size of texture files
 - Size of sound files
 - Displaying multiple views
 - To solve this problem while still displaying multiple views, we used viewports instead of additional panels.
- The frame rate problem caused us to change implementation quite a bit, we had to lower the number of exhaust/explosion particles.
- We had to abandon the planetary asteroid ring which would constantly circle planets, due to the amount of circles/particles it required.

Known Bugs

- There seems to be a bug within the form object, resizing causes the control object to move out of the position desired.
- The AI ships have trouble navigating around a solar object when their target is on the other side.
- Sometimes the controller continues to vibrate when unnecessary.
- The Scrolling bounds can produce errors in the smaller view windows when you advance out of the screen.

Possible Features for Future Versions

- We would like to implement more world scenes. This would allow for multiple more levels and difficulties.
- We wanted to add an asteroid ring sceneNode to a planet object, but because of the frame rate issue we had to leave it out. Additionally, we wanted ship primitive objects that hit the asteroid ring to stay there in orbit permanently. The ship primitive parts were too expensive to pile up in a scene node and rotate.
- Different Alternate weapons. Missiles with user activated detonation that shoots blaster shots in all directions when they explode. A Turret Blaster that aims to where the target will be if the continues on his current velocity.

Sources Cited

- All textures were gathered from Flickr.com and manipulated by us.
- Sound effects were gathered from both <http://www.soundsnap.com/> and <http://www.freesound.org/>