Battle Tinker - Game Proposal Chapter

Sandro De Zanet, Fabio Zünd, Marco Rietmann

March 10, 2008

1 Introduction

It's the year 2308. Over the last centuries humanity advanced deeper and deeper into space until some day they came across the planet Nauhra, very similar to earth. Unfortunately, the Nauhraians were not very hospitable nor were the humans very friendly. War spread throughout the galaxy. Ultimately, Nauhra was destroyed by men and earth was blown up by the Nauhraians. Nothing survived... except... the human housekeeping robots [1]. Now, five years after Earth Destruction Day, the robots are bored and depressed, they have nothing to do. They start constructing starships out of left over junk parts from the earth and battle each other. It's day one of Battle Tinker Wars.

2 Game Description

2.1 Game Play

BattleTinker is a multiplayer space simulation. We intend to allow up to four players to play on one Xbox360 on a split screen. There are two play modes: Battle Arena and Training.

Play mode: Battle Arena When the game starts each player can choose a housekeeping robot out of N different types of robots (gardener, butler, driver, garbage collector, etc.) and a color. In the next step each player can choose a pre-built space ship or he can decide to build it from scratch. If a pre-built ship has been chosen, nothing has to be done, and the player will get on the battle ground. On the other hand, if a player has chosen the space-ship-from-scratch-option, he will have first to build that ship. After every player has built or selected a ship, the battle begins. Basically, the players have to shoot and destroy each other. Every time a ship is destroyed, the destroyed player can change his spaceship in the editor. He has a limited number of changes he can do to his ship.

The battle is held in a Free-For-All 'arena', in the place where earth has exploded. Possibly, the players can choose to select a different arena such as the space near Nauhra. In the background the player sees stars and planets like the Sun or Mars and Venus. The arena is restricted by a hull of debris (spaceship parts, stones, anything). If a player flies into the debris the ship will be destroyed immediately.

Play mode: Training The Training mode is intended for one player only to exercise constructing ships and shoot dummy objects. The player can also store the constructed and tested ships on the harddisk in order to use them later in Battle Arena against other players.

[1] In the recent past, housekeeping robots became more and more important and evolved to intelligent, cultivated and even philosophizing beings.

The slightly hidden but fundamental challenge the players will face in this game is to appropriately place the engines on the ship and meaningfully assign them to the controller buttons in order to be able to move the ship in all directions. Only when this premise is met, the player can learn how to fly his ship and eventually engage the other players in battle successfully.

We like to provide the game with smooth lounge music that is played in the menu and in the editor and with a chary but distinctive type of music that is played in the arena. We intent to create our own music for this purpose. The sound effects should be such as one would expect from this type of game.

2.2 Editor

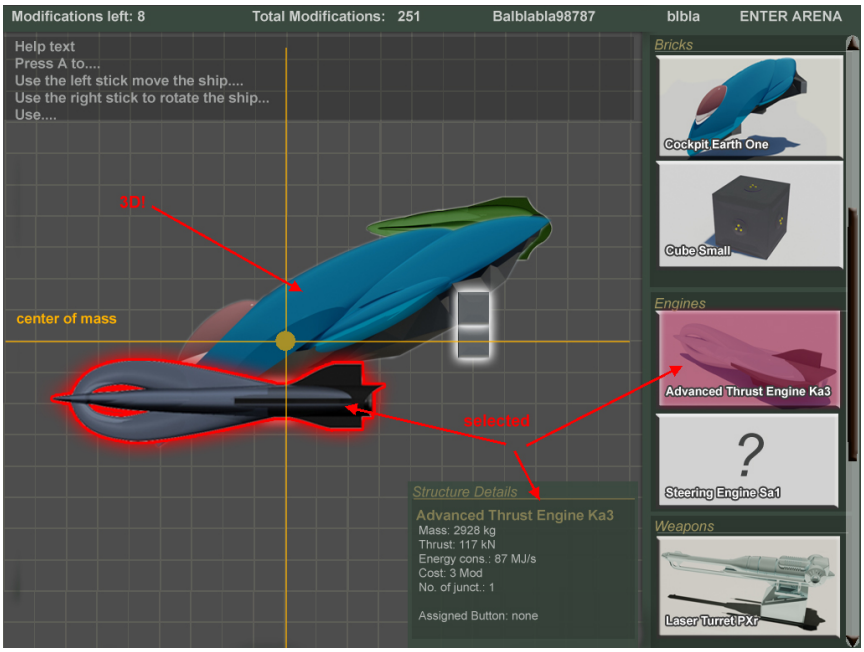

Figure 1: The Editor

In order to modify or build a ship, a player has to make use of the ship-editor (see fig. 1). In this editor a player can see his ship on the left side and all available structures on the right side. He can rotate and move the ship around for a better view. An info box displaying all attributes of a structures appears if the players selects a structure. To enhance the fine tuning and aligning of the structures, the centre of mass is always displayed as a dot with adjacent grid lines. The player can perform only a predefined number of modifications to the ship, depending either on the initial number of possible modifications when the player is building a new ship, or on a battle ground defined number.

The construction of a ship is subject to the following points:

1. Every ship consists (only) of structures.

2. A structure can be any kind of cockpit, weaponry element, defense element, etc.
3. A ship has to have at least a cockpit.
4. Every additional structure will be added directly to the cockpit or to another already added structure.
5. Each structure has at least one juncture to be attached to a juncture of another structure.
6. A structure can be attached to another structure only with the junctures facing each other directly.
7. A structure can be turned along the axis through the two junctures.

The editor offers the following construction structure types:

Cockpit: The cockpit represents the main unit of the space ship. If the cockpit is destroyed, the space ship is destroyed. (see fig. 2)

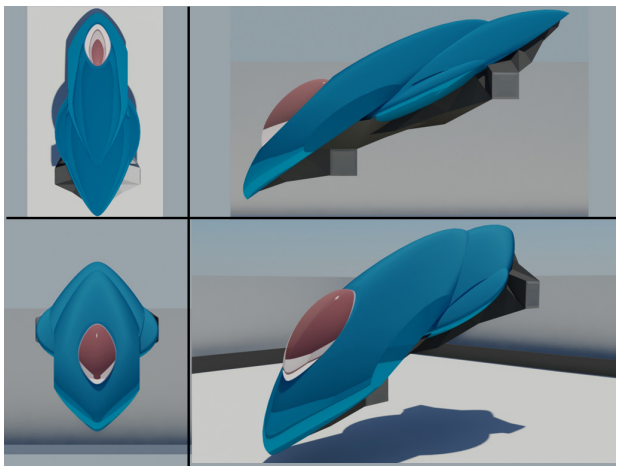

Figure 2: Cockpit Earth One

Brick: All parts of the ship, that can't be assigned to a controller button and hence can't perform any actions are called Bricks. Their mere purpose is to act as support parts that glue all the other parts together.(see fig. 3)

Engine: An engine accelerates the ship in the opposite direction it points to. We envision two different kinds of engines: Large engines that are mainly used to accelerate the ship in one direction and small 'steering' engines to rotate the ship. (see fig. 4)

Weaponry: Laser cannons, particle cannons, rocket launchers... (see fig. 5)

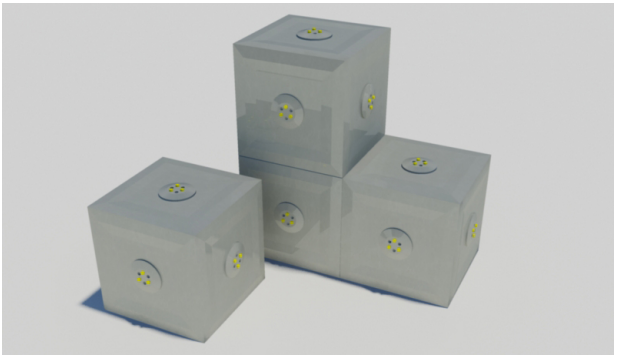

Figure 3: Cube Brick Small

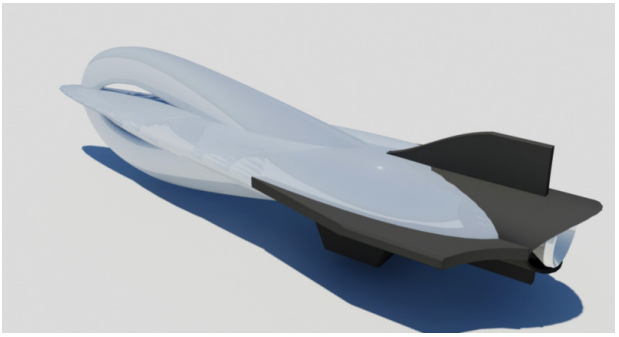

Figure 4: Advanced Thrust Engine KA1

Energy Supply: All the behavioral structures need energy. The energy supply unit stores energy and refills itself gradually.

Defense Mechanisms: Defensive structures can either be very robust structure parts that protect the ship from physical attacks or shield generators that engulf the ship with a shield to reflects and absorbs weapon fire - especially energy weapon fire.

To control the various parts of the ship, the player has to assign the controller buttons to the parts. The behavior of the structures will change depending on the 'analogousness' of the controller button. If for instance a stick or trigger is associated to an engine, the engine can accelerate arbitrarily. But if it is associated to a button, the throttle can only be full or off. On Fig. 6 you can see a example: Button A is assigned to a single steering engine on one side of the cockpit. If the button is pressed the engine is launched and the ship is propelled around the center of mass.

After constructing a ship, a player can choose to store it on the hard disk for later use.

2.3 Battle

Once a player has finished constructing a ship or selected a pre-built ship, he enters the arena (see fig. 7). The player can control the behavioral parts of the ship as he assigned it

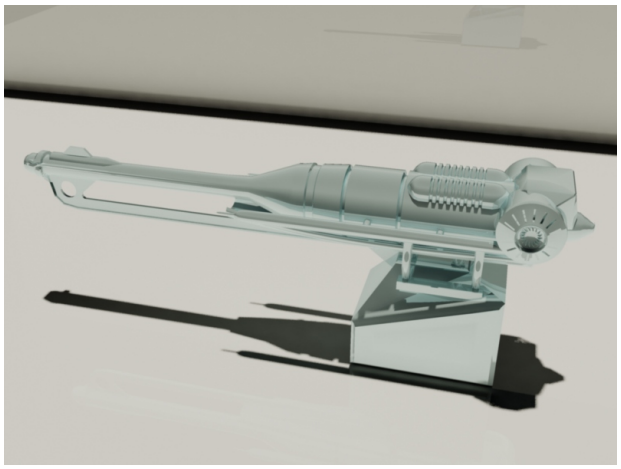

Figure 5: Laser Turret PXr

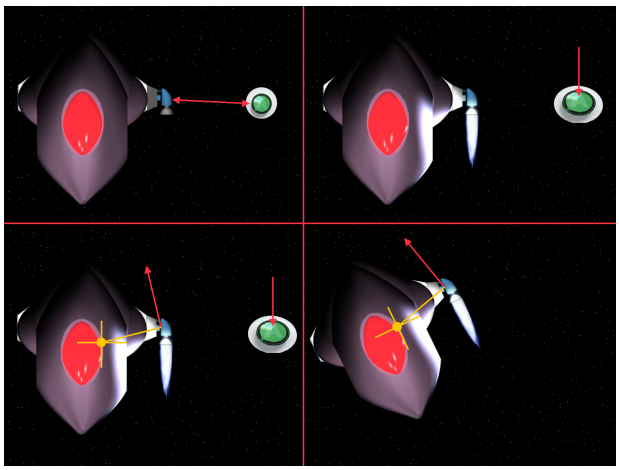

Figure 6: Assigning buttons to structures

to the controller buttons, e.g. uses the sticks und trigger to accelerate the ship and presses buttons to fire laser cannons or rocket launchers. The ship will accelerate according to its mass distribution and placement of the engines. When the ship fires weapons or is hit by bullets, laser salves[2] or rockets it will accelerate according to the law of conservation of momentum. The guns will have a delay after each shot in which it will not be able to fire. This is to prevent that the player can shoot all his energy at once.

On the head up display HUD the player sees different ship monitoring systems:

Health: A rotating model of the ship that shows the health of each structure in various colors.

Energy: Different bars that show the energy consumption of each structure and the total

[2]The mass of a laser salve is computed by its energy ($E = mc^2$)

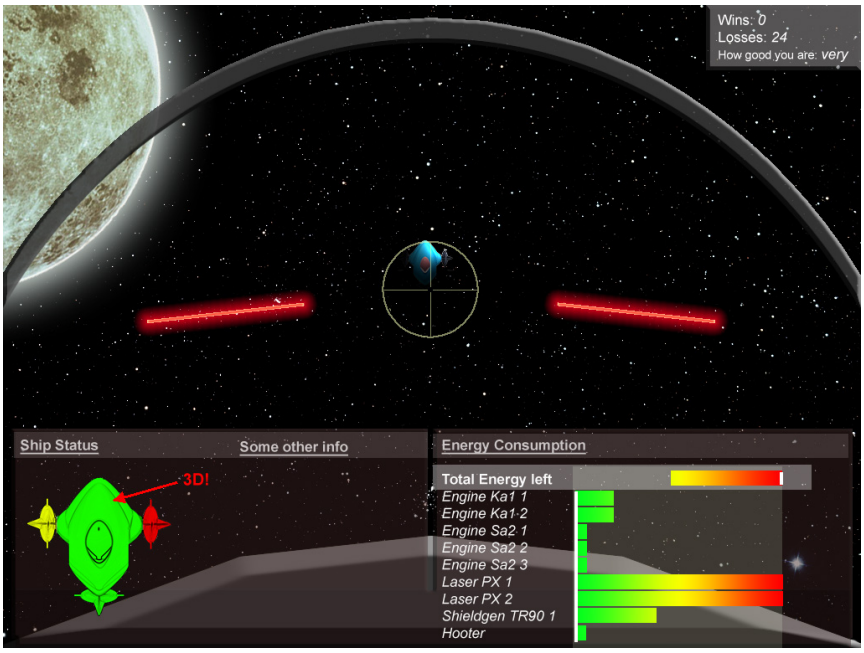

Figure 7: The Battle

energy left in all energy supply units.

Cameras: (High target) Cameras can be attached to the ship. E.g. the player can see what's happening behind or below the ship.

To enhance the navigation of the player there will be arrows pointing in the direction of the opponents.

2.4 Some Thoughts

Usually, editors allow players to build something up with parts that cost a predefined amount of money. We introduced the notion of 'modifications' instead of costs for the following reason: Obviously a death-star-laser cannon is by far more powerful than a tier 1 particle cannon. But in the editor each weapon costs only one modification to build. The point is that the death-star-laser also consumes by far more energy and is by far more large, heavy and vulnerable to attacks and therefore is not always the better choice. With this concept the player is forced to think more about his decisions instead of gradually buy the more costly parts as the game advances. Also, money and the fact that robots would buy structures instead of just collect them would fit less smoothly into the storyline.

3 Assessment

The main strength of the game will be that it will encourage the creativity and ingeniousness of the players to create the best space ship compared to the other players. The target players are people that have a basic (intuitive) knowledge of physics and can imagine a working space ship. They should be able to quickly learn their self-assigned controls and be able to adapt their behavior, if some parts of the ship are destroyed. The game will also automatically support some kind of paper-scissor-stone-principle, since a space ship will be strong only against some of the other player's ships, depending on their strategies. We consider the game to be a success if the players can be creative and try out several different strategies to beat the opponents, constantly adapting to each other.

4 Concept Art

Figure 8: Housekeeping Robot

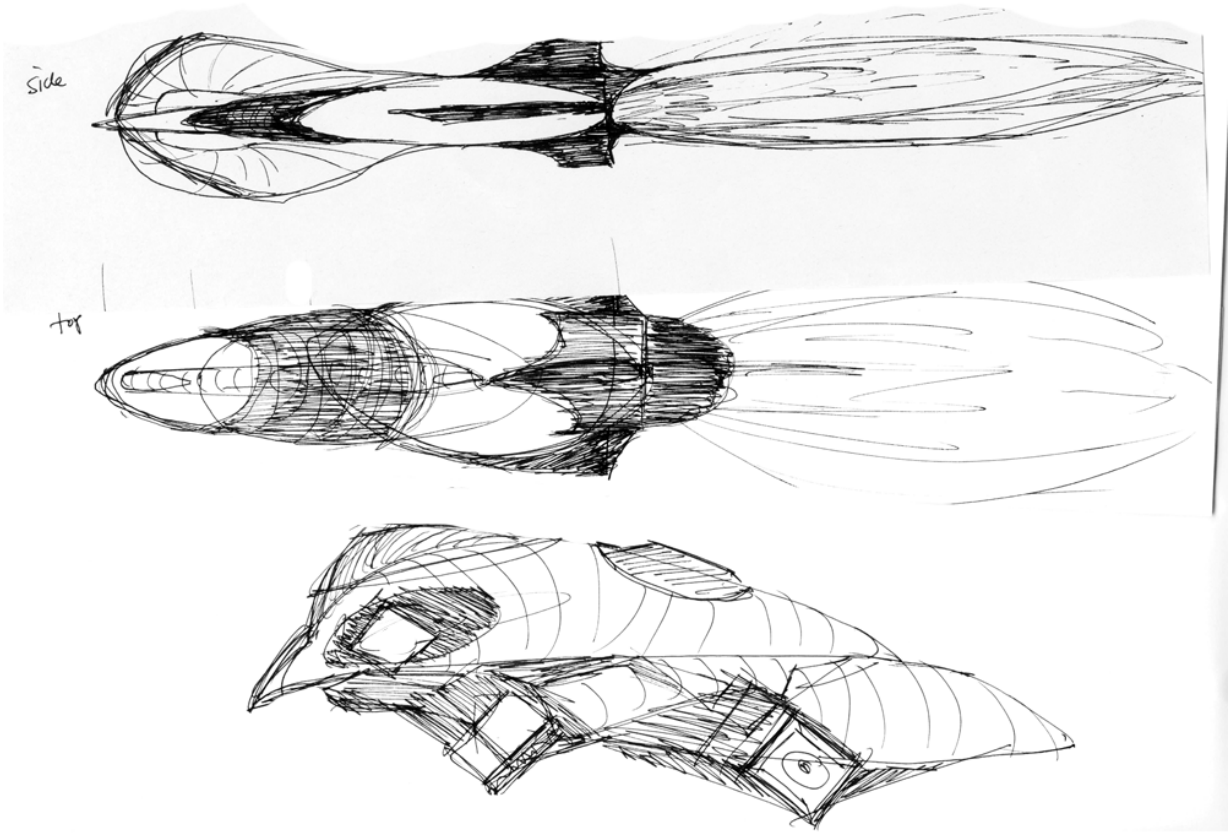

Figure 9: Cockpit and Engine

Figure 10: 3D Housekeeping Robot

Figure 11: 3D Housekeeping Robot with hair