Flat Hunt User Guide

Ursina Caluori
ucaluori@student.ethz.ch

September 26, 2005

Contents

Flat Hunt is an application that is used to teach you programming, along with another application named *Touch* [1]. *Flat Hunt* is a "Scotland Yard"-like game that will mainly appear in the assignments for the Introduction to Programming course. It is based on *TRAFFIC* [2] for modeling the city where the game takes place and on *EiffelMedia* (formerly known as *ESDL* [3]) for visualization.

This document describes how to use *Flat Hunt*.

1 Introduction

Welcome to *Flat Hunt*!

Flat Hunt is a simple adaptation of the well-known board game "Scotland Yard" (see Figure 1). Instead of some agents hunting Mr. X all around London, it is about a group of students starting off at ETH Zurich. To make their student life a bit more pleasant, they are desperately trying to find a flat in this little big city. But to get a flat, they must first meet the estate agent, who is running all around Zurich showing his flats to other people. . .
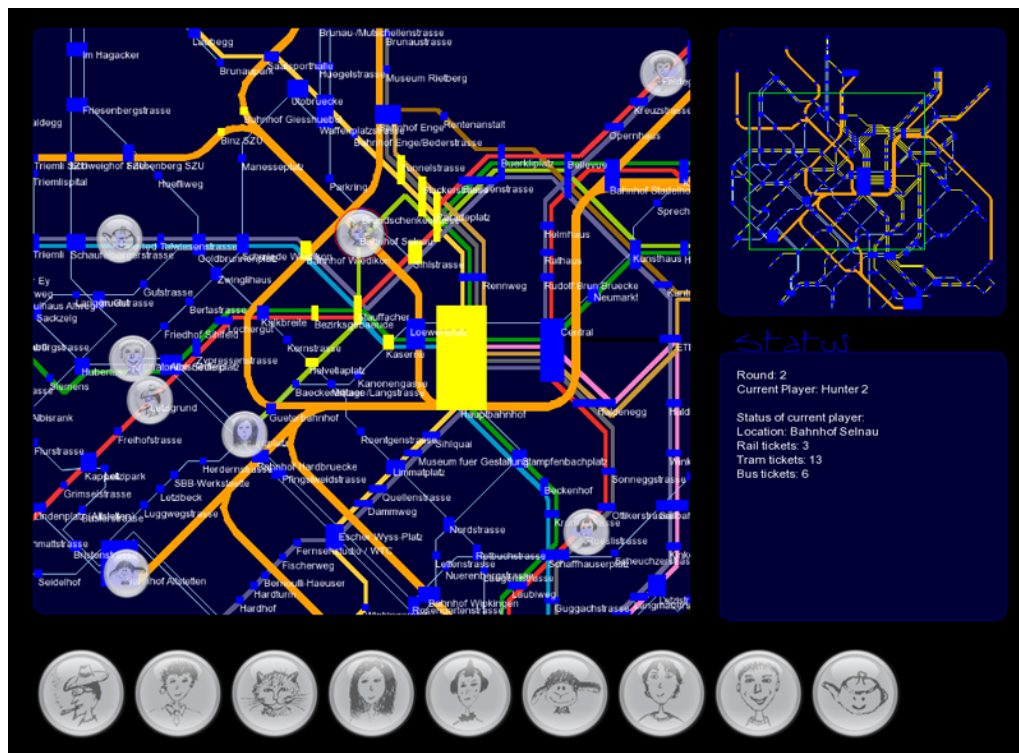

Figure 1: Screenshot of *Flat Hunt* in action

2 The Story

As the title suggests (and the introduction mentions), it is all about finding a flat in Zurich. . .

However, this is not so easy... There is this guy, the estate agent, who is renting flats. The problem is that he is always busy showing flats to other customers, and even in his office they don't really always know where exactly he is. The only thing they know is what kind of transport he is moving around with. This is because the estate agent is taking part in a new VBZ-project called "Customer tracking".

In collaboration with ETH, they equipped some volunteers with transponders. These transponders gather information like current position and type of transport, and send it in real-time to the office. However, for privacy reasons, only the type of transport can be accessed all the time.

Once in a while, the estate agent (Figure 2) calls his office to tell the secretary which flat he is currently visiting. So sometimes, the people there in the office can tell the flat hunters (Figure 3) where to look for him...

Figure 2: Estate agent

Figure 3: Flat hunters

3   Getting Started

What you need for running Flat Hunt...

3.1   Requirements

EiffelStudio: http://www.eiffel.com/downloads/

TRAFFIC: http://se.inf.ethz.ch/traffic/

EiffelMedia: http://se.inf.ethz.ch/eiffelmedia/

3.2   Installation

EiffelStudio, as well as *EiffelMedia*, come with an installer. Just follow the on-screen instructions like you would when installing any other program. No magic there..

   TRAFFIC does not need to be installed, just download the zip-file and unzip it to a directory of your choice.
Flat Hunt is located in the directory `traffic/example/flat_hunt`. To get it to run, however, you'll have to compile it first.

   For that you have to complete the following steps:

1. Start EiffelStudio

2. Click on "File ->New Project...". Choose "Open existing Ace (control file)" from the dialog (see Figure 4) and click on "Next".
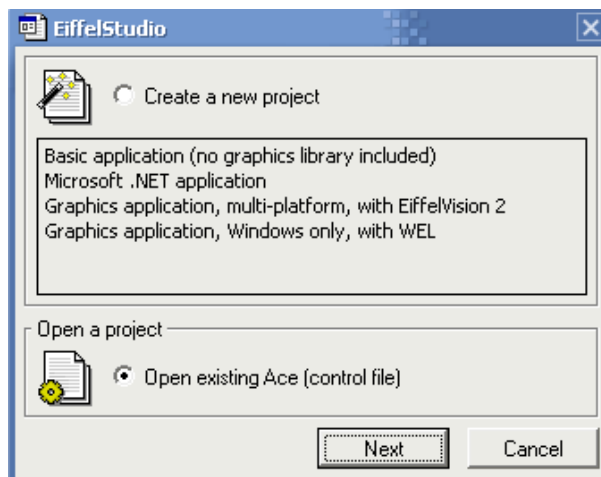

Figure 4: New Project Dialog

3. This will open a file dialog that lets you choose the Ace file. Browse to the directory `traffic/example/flat_hunt`. Depending on the operating system you are working on, choose *ise_windows.ace* or *ise_linux.ace*. Click on "Open".

4. The dialog shown in Figure 5 lets you choose the project directory. In most cases you can leave both paths (Ace file and location) as EiffelStudio proposes. Make sure that the checkbox for compiling the generated project is selected. Click on "OK". This will start the compilation of the project.
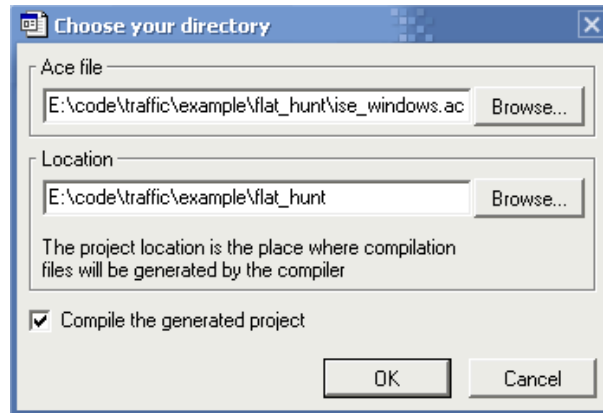
Figure 5: Project Directory Dialog

5. Once the project is compiled you can execute it by clicking on the "Launch" button in EiffelStudio or by hitting **F5**.

Now you are ready for playing Flat Hunt. Enjoy...

4   Gameplay

Playing Flat Hunt is not very difficult, especially for those that know the game "Scotland Yard"...

4.1   General Rules

The game lasts for at most 23 rounds. In these 23 rounds, the flat hunters try to find the estate agent, while he tries to avoid them (this is because he would rather rent the flats to elderly couples, since presumably they make fewer parties in the middle of the night...).

In each round, every player can make one move on the public transport system. The estate agent is the first, then it's the hunters' turn. One move is either

- one or two stops by tram (colored lines),

- one stop by train (thick orange lines),

- or one stop by bus (thin light blue lines).

A move with a certain transport can only be made if one has still enough tickets (see Figure 6), if there is a connection (obviously), and if there is no other player at that destination (and in the case of tram lines, if there is no hunter in between).

Attention: If you are at a bus-only stop, and you run out of bus tickets, you will get stuck there forever, so be careful...
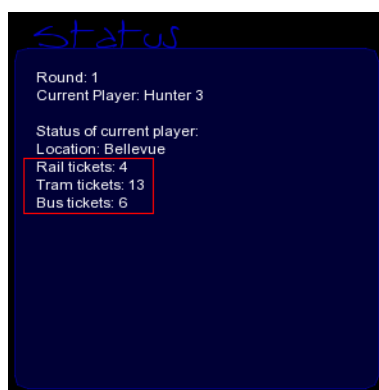

Figure 6: Ticket status

The possible places you can move to are colored yellow (see Figure 7). To make a move, just click on one of those highlighted places. The red circle centers on the player whose turn it is, and in the status box at the right, the game status and information about the current player get displayed. If you want to know the status of another player just click on his picture at the bottom. Click again to close the just opened status box.
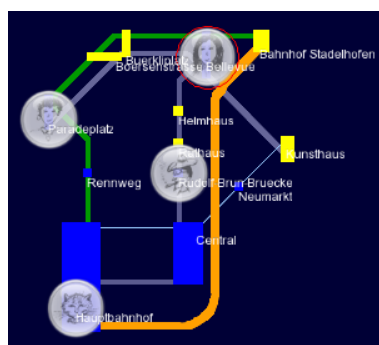

Figure 7: Highlighted places

The game is over when

a) the hunters could not find the estate agent within 23 rounds,

b) one flat hunter moves onto the place where the agent currently is,

c) or the hunters encircle the estate agent so that he cannot move anymore.

In case a), the winner is the estate agent (he does not have to rent his flat to students), whereas in b) and c) it is the hunters that win, as they get to meet the estate agent on time and thus manage to find a flat.

4.2 Game Modes

There are four modes to play *Flat Hunt*: *Hunt*, *Escape*, *Versus* and *Demo*. Depending on the mode, zero (*Versus*), one (*Hunt/Escape*) or two (*Demo*) parts are taken over by the computer.

Hunt This is probably the most typical situation; the player tries to find the agent, which is played by the computer. Thus, the player only knows about every fifth move where the agent just was... The agent shows himself only in rounds number 1, 3, 8, 13, 18, and 23. In these rounds, the exact route of the agent is displayed under *History* in the status box at the bottom right corner and in the estate agent's own status box if opened. In all other rounds you only see the detailed history up to the round the estate agent last showed himself. As soon as the agent has come out of hiding for the first time, a dimmed version of his picture will always be shown at the location he was last sighted.

Escape This is the exact opposite of *Hunt* mode: The agent is played by you, and the hunters are played by the computer. The hunters always move as close in your direction as possible, as they somehow manage to decode your transponder signal, and thus always know your precise location (so much for privacy...). You just have to try to avoid them as long as possible...

Versus This is the multiplayer mode. One of the players is the agent; the other plays all the hunters. While the player of the agent is making a move, the player of the hunters is supposed to look away...

Demo This mode is more or less the opposite of the buzzword "interactive", but is about as entertaining as watching fish in an aquarium. The computer is playing against himself, trying to catch the agent as fast as possible.

4.3 Other

When you run *Flat Hunt*, the first you'll see is a menu (see Figure 8). You can either let the default options in place and just select *start game* or you can adjust the settings to your needs. *Game mode* is explained in subsection 4.2, *number of hunters* and *map size* should be self-explanatory and *characters* specifies which pictures to use for the players. To toggle between the settings menu and the normal menu press *tab*.
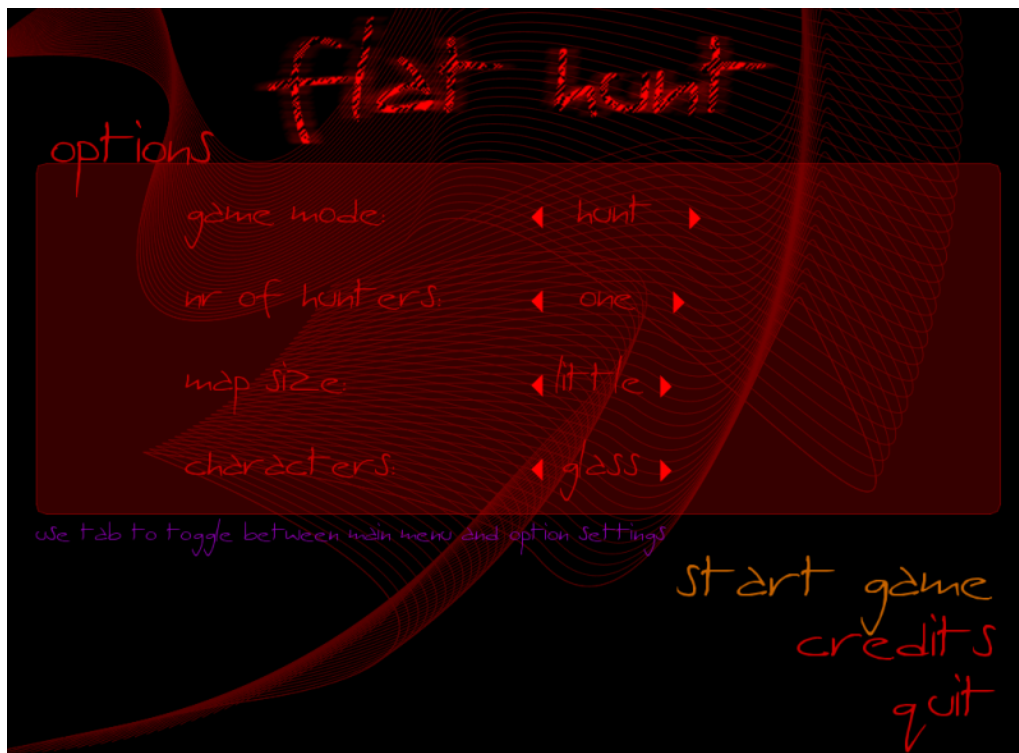

Figure 8: Screenshot of the start menu

During the game when you press *p*, the pause menu is shown. *Continue* makes the pause menu disappear and lets you resume the game, *new game* takes you to the start menu and *quit* quits the application.

The game over menu is similar to the pause menu, only there is no *continue* in this one.

5 Special Features

5.1 Map Control

Map control is fairly simple: you got two maps in a game scene, one of which only is a smaller version of the other one. The big map is on the left and that's where the action takes place, the little map on the right is meerely a navigation tool. To control the big map, use your mouse as follows:

left click: only has an impact if clicked on a highlighted place

right button down + move mouse: moves map in the direction of your mouse movement

middle button down + move mouse up: zoom in

middle button down + move mouse down: zoom out

When you **left click + move mouse** in the little map, a red rectangle is drawn between the point where your left mouse button is pressed down and the point when its released. As soon as you release the mouse button, the map segment that is inside this rectangle gets displayed on the big map.

5.2 Music Player

Flat Hunt comes with an integrated music player and some default background music. Since not everyone likes the same sound, there is also the possibility to play your own.
Just put your `.ogg`-files in the directory `${FLAT_HUNT}/resources/sound` before you start the Flat Hunt application. Flat Hunt will then automatically load all the `.ogg`-files from this directory and play them in alphabetical order (unless you enable shuffle, obviously). Music player control: see subsection 5.3.

5.3 Keyboard Shortcuts

During the game, the following shortcuts are available:

p: pause the game and show pause menu

s: music player toggle shuffle

v: music player decrease volume

shift + v: music player increase volume (at startup the volume is already at maximum)

page up: music player next song

page down: music player previous song

6   Legal Stuff and Thanks

This document is based upon its prior version, which was written by Michela Pedroni and Marcel Kessler (thanks!). All graphics for the game were designed by me and Photoshop.

Thanks to Michela Pedroni for her assistance, all my predecessors for their work, Till G. Bay (and others) for the *EiffelMedia* (formerly *ESDL*) Library and Bertrand Meyer for the *Eiffel* language.

References

[1] Roger Küng. *Touch User Guide*. ETH Zurich, 2005.
    http://se.inf.ethz.ch/projects/roger_kueng

[2] Sibylle Aregger. *Redesign of the TRAFFIC library*. ETH Zurich, 2005.
    http://se.inf.ethz.ch/projects/sibylle_aregger

[3] Till G. Bay. *Eiffel SDL Multimedia Library (ESDL)*. ETH Zurich, 2003.
    http://se.inf.ethz.ch/projects/till_bay