Jetpack Joyride

平台移植

製作：黃鳳翊

一、導論

1.動機

(1)製作容易

(2)我想玩

2.目的

將”Jetpack Joyride”移植到PC平台(原本的目標)，但是由於沒有原始素材以及VS平台的限制，所以只剩下遊戲方式相同而已

二、系統概述

1.背景

Jetpack Joyride是由Half Brick所製作的一款手機遊戲，其內容為主角會不斷地向前跑，途中還必須閃躲障礙物以及收集金幣

2.系統介紹

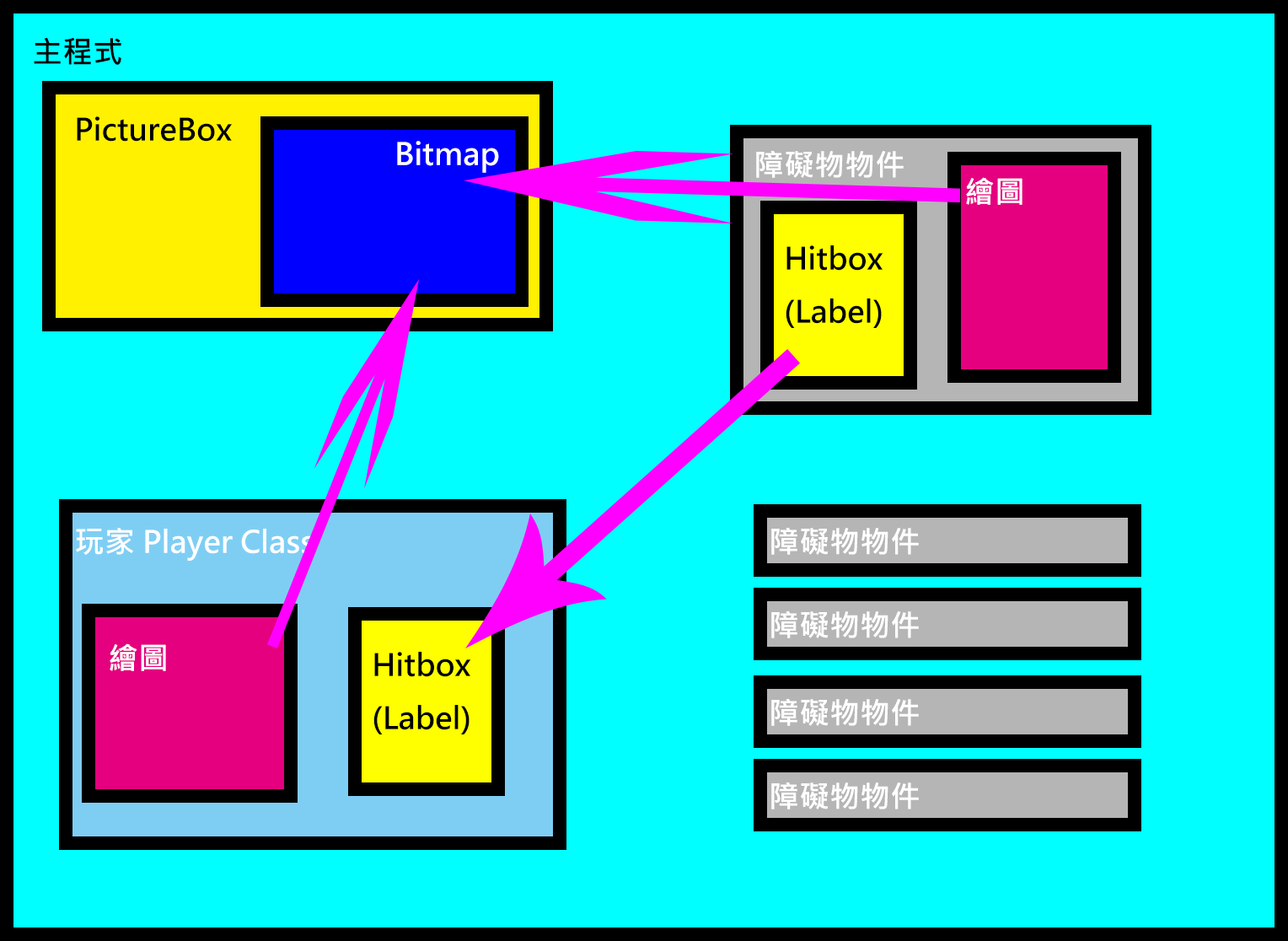
(1)一張會不斷移動的地圖

(2)看起來正在往前跑實際上原地不動的人

(3)各種障礙物，人物碰到就遊戲結束

(4)錢，可以蒐集

三、系統設計

3.1 初始設定

如圖(箭頭以外部分)

以上為會利用到的物件

3.2 函式概述

主程式

private void animatetimer\_Tick(object sender, EventArgs e)

進行繪圖(及遊戲Timer)控制

void score\_MouseClick(object sender, MouseEventArgs e)

“再玩一次”的相關設定，與記分板寫在一起

private void draw\_back()

背景繪圖控制

private void Form1\_KeyDown(object sender, KeyEventArgs e)

private void Form1\_KeyUp(object sender, KeyEventArgs e)

按下空白鍵的處理

player.cs

public void draw\_player()

玩家角色繪圖控制

void flytimer\_Tick(object sender, EventArgs e)

“飛行”動作控制

public void pressfly()

public void stopfly()

調整”飛行中”flag

ice.cs

public void draw\_ice()

障礙物繪圖以及移動

四、系統製作

1.使用者輸入設定

按空白鍵往上飛，放開則下降

2.重要使用函數與類別說明

(1)繪圖原理

利用每次刷新的Bitmap當作畫布

依照背景、人物、障礙物的順序製作遊戲畫面

繪製完成後把這張Bitmap當成Image給PictureBox

(2)碰撞判定

將顯示與實際碰撞範圍分開處理，由障礙物去偵測是否撞到玩家，可避免”看起來沒撞到卻死掉”的情形

五、系統評估

1.達成目標

(1)有背景

(2)有人物

(3)有障礙物

(4)用分數代替距離

其他額外目標：解決畫面閃爍問題

2.未達成目標

原始遊戲內的東西：

1. 錢
2. 交通工具(蒐集到會變換移動方式，並且抵銷一次死亡)

額外目標：讓畫面流暢

3.心得

由於繪圖原理的關係，光是人物+背景就會造成一定的延遲，再加上障礙物就可以感受到明顯的lag，網路上google出來的結果是VS本身的極限，如果再繼續做下去的話有可能會幾乎不能玩，所以我只好在這裡收手，如果有機會的話，應該會用其他軟體把它完成。