

以 JavaScript 動態物件製作網頁射擊遊戲介面之研究

張逸中* 洪龍成**

*台灣首府大學數位娛樂與遊戲設計系副教授 (ycc@tsu.edu.tw)

**台灣首府大學資訊與多媒體設計系助理教授 (hlc-zen@tsu.edu.tw)

摘要

目前多數商業運轉之網頁遊戲多以 Flash 或 Java 等程式語言獨立設計，經過編譯之後以外掛程式的方式嵌入網頁運作。有鑑於安全性的疑慮以及過度使用網路資源及耗電的缺點，外掛程式的設計方式已經受到嚴厲的質疑與挑戰。取而代之的技術趨勢預期將是以標準網頁設計工具：HTML、CSS 與 JavaScript 語言來設計網頁遊戲。然而根據一般認知，JavaScript 的程式設計環境與預期功能都遠遜於一般的視窗程式語言，恐難以完成精緻網頁遊戲之設計。本研究的內容即為探討如何以 JavaScript 程式設計需要高速移動，與隨時增減砲彈物件的射擊遊戲介面，並獲得具體成效，以極精簡之 JavaScript 程式完成需要動態物件的遊戲程式。

關鍵字：網頁、JavaScript、射擊遊戲、動態物件

The Design of Web Shooting Game Using JavaScript Dynamic Object Technique

Yet-Chnug Chang* and Lon-Chen Hung**

*Taiwan Shoufu University, Dept. Digital Entertainment and Game Design ,

**Taiwan Shoufu University, Dept. Computer Science and Multimedia Design ,

ABSTRACT

Most commercial web games now prevailing are designed by programs like Flash or Java, which were designed independently first, later compiled and implemented into normal web pages. Such way of designing was criticized for their security problem and consumed too much network resource and electric power. An alternate framework for this is to design web game purely by standard web tools: HTML, CSS and JavaScript. However, in most programmers' opinion, the designing environment and functionality of JavaScript are worse than most independent programming languages. In this paper, we try to employ the JavaScript dynamic object operating techniques to design a web shooting game, which needs many fast movements and object creations and omissions. It is proved the standard web design tools can do as well as implemented programs in designing and executing efficiency.

Key Words: Web, JavaScript, Shooting game, Dynamic object

一、介紹

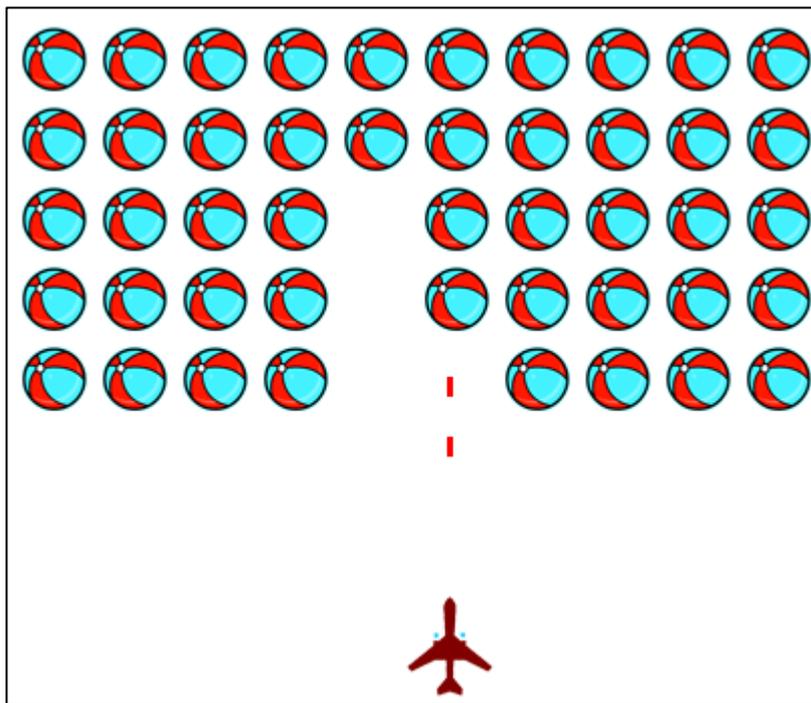
網頁遊戲受限於 HTTP 通訊協定的運作方式，一般被認為較難以預設的標準網頁設計工具(HTML、CSS 與 JavaScript 語言)直接設計，尤其是需要物件高速運動與動態變化的射擊類遊戲。因此多數商業運轉的網頁遊戲都以外掛程式的模式設計，亦即使用非網頁專屬的程式語言，如 Flash 或 Java 語言獨立設計、測試，最終編譯為執行檔，嵌入一般網頁之中執行。這種設計模式因為系統與瀏覽器程式都無法檢視其程式內容，隱藏著安全性的疑慮，且許多研究顯示它們有過度消耗網路資源以及設備電力的傾向，對於目前盛行用於上網的智慧型手機與平板電腦等設備相當不利。

事實上由於標準化網頁設計工具以及瀏覽器軟體的持續進步，目前在程式功能擴展及執行效能方面都已經有了長足的進步，直接以 JavaScript 程式設計動態之網頁遊戲已非難事。但是由於此類實作案例較少，如何使用 JavaScript 程式進行類似一般視窗程式擅長的動態物件設計？如拖曳物件、發射子彈或碰撞反應等動作，參考資料仍然極少。

有鑑於此，本研究嘗試匯集、整理與測試 JavaScript 程式對於物件動態設計的技巧，以一個簡易的網頁射擊遊戲為例，展示如何使用網頁設計的標準工具：HTML、CSS 與 JavaScript 就能設計出此類需要高度動態功能之遊戲介面。

二、設計目標

本文將介紹之簡易射擊遊戲網頁畫面如下圖：



圖中上半部大量密集的海灘球圖案是射擊的目標，將由程式迴圈動態大量的產生；下方飛機圖案代表射擊用的槍枝，可以使用鍵盤的左右鍵移動位置，並以向上鍵發射砲彈，圖中較小的紅色矩形是由程式動態產生，且移動中的砲彈。砲彈擊中目標之後還必須能被偵測，讓砲彈與目標一起被刪除，當然砲彈飛出畫面之後也應該被刪除，以免持續增加記憶體垃圾。

如果以 VB 或 C#等標準化的視窗程式設計此類遊戲時，因為有屬性與功能相當完整的控制項物件，不論是撰寫程式碼或以圖形化介面設計遊戲畫面都比較方便。但是 JavaScript 可以

操作的只是 HTML 網頁元件，要讓它們具有一般視窗程式中控制項物件一樣的操控性，在程式技術上較為繁複。主要是必須配合 CSS 的功能強化 HTML 原始物件的定位功能，讓物件可以經由程式控制而運動；當然動態產生與刪除物件在 JavaScript 語言中也是比較少被使用的功能，值得在此深入介紹。

三、動態產生網頁物件

以下是本研究以 JavaScript 撰寫用於動態產生物件的函數：

```
function newObj(x, y, w, h, bgC, img, tt) {
    var b = document.createElement("div");//建立目標物件
    b.style.position = "absolute";//依據座標定位
    b.style.posLeft = x;//X座標
    b.style.posTop = y;//Y座標
    b.style.posWidth = w;//寬度
    b.style.posHeight = h;//高度
    b.style.backgroundColor = bgC;//背景色
    b.style.backgroundImage = "url(" + img + ")";//影像
    b.title = tt;//替代文字
    return b;//回傳新物件
}
```

此函數匯入的引數依序是：物件在網頁上的水平與垂直像素點座標，寬度與高度，背景色，背景影像與提示文字。進入程式首先是宣告一個 div 網頁物件，接著一一使用樣式表 style，也就是所謂 CSS 的附屬屬性功能，將這些匯入引數設定為新物件的屬性。與一般視窗程式不同之處是：div 物件本身預設的 HTML 屬性不足以讓物件產生類似視窗程式控制項的程式控制需求，所以多數與位置相關的屬性都必須變成「物件.style.屬性」這樣的 CSS 子屬性。

此處非 CSS 屬性的設定只有一個 title，在此例中我們其實是用它做為物件種類的標籤。本範例中我們會動態產生的物件有射擊目標及砲彈兩種，後續程式中我們會需要知道被處理物件的種類，所以在此會將目標物件的 title 標註為"X"，砲彈則是空字串。同時間，砲彈不設背景圖案，只設定背景為紅色，目標物件則是一個海灘球。下面是網頁載入會執行的程式碼：

```
function init() {
    for (var i = 0; i < 10; i++) {
        for (var j = 0; j < 5; j++) {
            var b = newObj(40 * i + 4, 40 * j + 15, 32, 32, " 'bb.gif', 'X');
            document.body.appendChild(b); //物件加入網頁
        }
    }
}
```

在此是用兩層迴圈產生 5x10 的射擊目標群，類似打磚塊遊戲中的一面磚牆。必須注意的是：以 newObj 建立好的網頁物件必須加入網頁的 body 區塊(使用 document.body.appendChild 方法)，才算完成建置，讓物件顯示於網頁上。

四、移動槍枝與發射砲彈

槍枝移動與射擊是由鍵盤事件驅動，首先應該在 body 標籤中宣告鍵盤事件副程式，當然前一節介紹的目標建置副程式 init 也應該一起在此宣告，HTML 原始碼如下：

```
<body onkeydown="kd()" onload="init()">
```

鍵盤事件的副程式內容如下：

```

function kd() {
  switch (event.keyCode) {
    case 38://向上鍵→開槍
      var b = newObj(0, 0, 3, 10, 'red', "", "")
      document.body.appendChild(b); //物件加入網頁
      shoot(b); //發射子彈
      break;
    case 37://向左鍵→左移槍枝
      G.style.posLeft -= 5; break;
    case 39://向右鍵→右移槍枝
      G.style.posLeft += 5; break;
  }
}

```

在此地的 G 物件是我們在網頁設計一開始就以靜態設計方式完成的槍枝物件，我們使用它的 style 屬性就可以移動它的位置。開槍的動作則由一個 shoot 自訂副程式來處理其飛行與碰撞事件。

五、控制砲彈飛行

控制砲彈飛行的程式架構概略如下：

```

function shoot(q) {
  with (q.style) { //有關於子彈的樣式
    posLeft = G.style.posLeft + (G.width - posWidth) / 2; //水平位置
    posTop = G.style.posTop; //垂直位置
    var v = 10; //速度10
    //設定運動計時器
    var T = setInterval(function () {
      posTop -= v; //往上移動
      if (posTop < 0) { //超出畫面上端
        clearInterval(T); //停止運動
        q.removeNode(); //刪除子彈
        return; //離開副程式
      }
      ..... //碰撞偵測
    }, 50); //50毫秒移動一次(每秒20次)
  }
}

```

首先是將砲彈移至槍口位置作為運動起點，接著設定一個類似一般視窗程式中 Timer 物件的副程式，每 50 毫秒將砲彈上移 10 點，這樣砲彈就會向上跑了！如果砲彈超出畫面時應該停止這個計時器並且刪除砲彈本身這個物件。網頁物件的自我刪除可以使用 removeNode 這個方法，但是經過測試在 IE 瀏覽器可以正確執行，在 Google Chrome 則無效。可見同樣的 JavaScript 原始程式由不同瀏覽器解讀還是會有差異，而且刪除物件這種功能真的很少被使用，才會讓主流瀏覽器之間都有這種明顯落差。

六、碰撞偵測

射擊遊戲中打到目標就必須有反應，所以碰撞偵測是不可少的部分，所謂的碰撞就是用程式比較兩個物件的位置是不是有重疊？此例中因為砲彈物件較小，我們僅以它的彈頭座標為基準，如果這個座標位置落於某個目標物件的內部就是擊中目標了！程式碼如下：

```

function chkHit(B, x, y) {
  if (y > (B.offsetTop + B.offsetHeight)) return false; //偏低

```

```

    if (y < B.offsetTop) return false;//偏高
    if (x > (B.offsetLeft + B.offsetWidth)) return false;//偏右
    if (x < B.offsetLeft) return false;//偏左
    return true;//擊中
}

```

此副程式會檢查代表砲彈位置的 x 與 y 是不是位於物件 B 的範圍之內？如果是，就回傳 $true$ ，否則回傳 $false$ 。當我們要看某一個位於座標 (x, y) 的砲彈是不是碰到網頁上的某個目標時程式碼概略如下：

```

for (var i = 0; i < document.body.all.length; i++) {
    var B = document.body.all[i];//一一取出網頁物件
    if (B.title == "X") { //如果被標示為子彈
        if (chkHit(B, x, y)) { //檢查碰撞
            .....
        }
    }
}
}

```

就是使用一個迴圈，一一取出網頁內的元件，檢視它是不是標示為“X”的目標？如果是，就呼叫碰撞偵測程式，如果碰到了就執行應有的回應動作程式，在本例中就是讓被擊中的目標物件與砲彈一起消失，當然也可以在此加入計分機制。

七、討論與結論

本文介紹的遊戲網頁公布於：<http://ycc.tsu.edu.tw/share/shootgame.html>，完整的程式碼可由瀏覽器的檢視功能原始碼功能完整下載以供參考。類似以JavaScript程式設計之互動遊戲介面製作範例亦可瀏覽筆者之網頁設計課程網頁：<http://ycc.tsu.edu.tw/WebPage/>[2]。甚至只要搭配簡單的ASP.NET伺服器端網站通訊機制就可以完成網頁線上遊戲的製作[3]。

本文的研究已經證明即使動態程度需求甚高的射擊類網頁遊戲介面，也可以在標準網頁技術架構下經由JavaScript程式製作完成。事實上認真探究JavaScript程式技術，確實可以完成許多複雜且互動功能甚佳的程式，Google Maps[5]與Gmail就是極佳的範例。除了動態物件的控制，在複雜遊戲邏輯的運算管理機制方面JavaScript也可以勝任有餘，請參考筆者製作之象棋遊戲網頁：<http://ycc.tsu.edu.tw/JSchess/adv.htm>。

當然，由於JavaScript程式必須由瀏覽器解譯為機械碼的特性並未改變，目前執行效能方面仍將略遜於經過編譯之後嵌入網頁之外掛程式，但隨著電腦硬體效能及瀏覽器功能的持續提升，其間的差異已經愈來愈小。本文之意義在於凸顯：即使是需要許多動態物件操作的網頁射擊遊戲也可以直接使用JavaScript程式製作，不必仰賴外掛程式。尤其是近年網頁設計工具本身也在積極進化之中，HTML5的出現事實上已經大幅提升JavaScript程式可以設計運作的空間，預期未來HTML+CSS+JavaScript的組合將逐步演化為可以完全獨立設計各種網頁應用程式的環境。

參考文獻

- [1] 張智興，JavaScript程式設計與應用，國立清華大學出版社，2008/1
- [2] 張逸中，網頁設計，<http://ycc.tsu.edu.tw/WebPage/>
- [3] 張逸中，Visual Basic 2012網路程式設計線上遊戲實作，博碩文化出版，2013/4
- [4] 楊水清等編著，JavaScript與Ajax網頁程式設計，博碩文化出版，2008/9
- [5] Google Company，Google Maps，<https://maps.google.com.tw/>