

# 線上圖文互動網頁之設計-使用 HTML5 與 ASP.NET

張逸中 蔡宗達 曾冠傑 彭紹剛  
台灣首府大學數位娛樂與遊戲設計學系  
ycc@tsu.edu.tw

## 摘要

超文件標示語言 HTML5 新增之 canvas 畫布物件提供了使用 JavaScript 程式在網頁上繪製圖形、文字、甚至貼上影像的功能，搭配滑鼠事件即可讓使用者在網頁上自由繪圖，甚至編輯影像。本研究進一步搭配 ASP.NET 的伺服器程式與 AJAX 定時更新網頁的功能，令使用者的繪圖動作、發言與上傳影像訊息都能經過網站公用變數的交換，與其他線上使用者使用圖文互動。最終完成一個可以經由網頁選擇通訊對象，使用影像、繪圖與文字交談，如同網頁白板的網頁程式。其重要性是在此設計架構下，無需使用 W3C 規範之外的任何嵌入式網頁程式元件，安全性無虞且執行效能甚佳。在應用潛力上依此架構也可以完整地設計出各種網頁遊戲或遠距教學互動網頁。

**關鍵詞：**ASP.NET, AJAX, HTML5, JavaScript, 白板

## Abstract

The HTML5 provides a new tag type “canvas” which can be used to draw graphs, texts, or images by JavaScript language in the web page. Adding the mouse event response functions, a freely interactive drawing web program could be done. In this research, we further employ the server side program of ASP.NET and some AJAX techniques to make the actions of drawing, texting and image uploading could be transported to another online user. Thus an interactive web page like an online whiteboard was done. Under this design framework, no embedded program element should be inserted into the web page, everything follow the regulations of W3C. Under this framework, we can also design web-based online games or distant interactive learning pages.

**Keywords:** ASP.NET, AJAX, HTML5, JavaScript, Whiteboard.

## 1. 前言

隨著網頁互動需求的提高，網頁已經日漸脫離靜態呈現資料的角色，網頁即時通[1]、聊天室[2]、塗鴉牆[3][4][5]乃至互動遊戲[2]近年都逐步成為網際網路世界的一部份。其中即時通與聊天室等互動功能主要是以伺服端的網站公用程式與變數代替傳統視窗網路程式軟體之伺服器功能，讓使用網頁瀏覽器上網的人也可以經過共用的網站交換訊息。

但以互動程式在網頁內繪圖一直是 HTML 語言的一大困難，在 HTML4 之前的版本中，需要繪圖功能的塗鴉牆或者網頁互動程式通常都要仰賴嵌入式的元件。譬如先使用 Flash、Java、Basic 或 C 語言等程式設計具有繪圖功能的程式，完成後編譯成可以嵌入網頁的元件，如 swf, Java Applet[6]或 ActiveX[7]等格式的檔案，最後將它們以外掛的方式嵌入網頁。當然這樣做的主要缺點是程式內容不透明安全性堪慮；而且它們並非網頁程式架構中的正規成員，與網頁中其他元件及來源網站之間的互動設計較為困難。

新版的 HTML5 有鑑於此，提供了新的 canvas 標籤物件[8][9]，作為網頁繪圖的基礎，搭配符合 W3C[10]安全規範的 JavaScript 語言，在網頁原始碼內就可以直接設計出內容豐富的靜態乃至互動繪圖的程式。其中互動繪圖以 JavaScript 的滑鼠事件，搭配在 canvas 上的繪圖動作即可完成。

另一方面，目前在 Facebook[2]等社群網站中常見的聊天室程式可以使用伺服端的網站公用程式與變數，搭配 AJAX[11]技術中定時更新網頁的功能達到使用者之間即時交換訊息的目的。本研究的主要動機就是藉由類似的訊息交換機制，除了交談文字之外也同步交換繪圖動作與使用影像的訊息。使得在本機瀏覽器中執行的繪圖與上傳影像資訊也可以傳遞到交談對象的瀏覽器中，並以 JavaScript 程式繪圖重現遠端使用者的繪圖動作與影像。

本文內容將以圖文說明程式運作機制與通訊流程，並於網址：<http://ycc.tsu.edu.tw/webboard/>[12] 展示設計成果，並提供完整之原始程式專案下載，歡迎各界免費合法參考使用，本文中將不會詳列程式碼與使用者介面設計之細節。此外，本文介紹之相關技術大致已經納入本人[線上遊戲設計](#)課程[13]內容，其中[互動式網頁白板](#)[14]介紹互動繪圖程式設計；[網頁五子棋](#)[15]則為依此技術設計之簡易網頁線上遊戲。除進一步證實此技術的實用性，也歡迎各校師生作為教學或自學之用。

## 2. 主頁面之設計



圖 1 主頁面設計圖

本研究之網頁程式使用 Microsoft Visual Web Developer 2010 Express 程式軟體設計，使用之網站範本為 ASP.NET 網站，網頁格式是副檔名為 aspx 之 Web Form 網頁，設計階段主頁面之控制項擺設內容概略如圖 1。

### 2.1 程式基本功能

如圖 1 之設計頁面所示，此程式主要功能是用戶先輸入自身名稱按下『上線』按鈕宣告上線，名稱將隨即出現於右側的線上名單；隨後於線上名單選擇交談對象。若自身宣告名稱與線上名單重複會在黑底標籤顯示警告，登入將無法成功。交談過程中，使用者可在標示『上傳影像視窗』的區塊內自由繪圖，或先用下方的介面上傳需要分享討論的影像檔案。『交談區』功能與一般網頁聊天室功能相同，在下方文字方塊輸入內容後按下『發言』鈕即可送出，雙方交談內容將持續顯示於上方較大的文字方塊看板。

### 2.2 AJAX 功能相關物件

在操作介面上方的 ScriptManager 物件可視為匯入 AJAX 機制所需的函式庫。主介面下方的 Timer 與圖中無法顯示的多個 UpdatePanel 物件都是與 AJAX 功能相關的物件，都需要此函式庫的支援。此外 Timer 物件將用於控制客戶端與伺服端的定時通訊頻率與內容，這是交談中的客戶端網頁內容可以即時同步更新的主要控制元件。

至於 UpdatePanel 的功能是控制網頁更新內容的範圍，譬如我們可能會定時用 Timer 機制從伺服器取得某些新資訊，但這些資訊只與網頁上的某一元件有關，此時就可以 UpdatePanel 包覆需要更新的元件，之後我們在 Timer 程式中就可以指定何種時機應該更新哪一個 UpdatePanel 的內容。這個機制十分重要，如果任由網頁定時的全面更新，那麼繪圖時可能會清除交談內容，或交談時繪圖內容卻

被清除，這些都不是我們希望的執行效果。

### 2.3 資訊交換物件

最下方的三個 HiddenField 物件顧名思義在執行階段是不可見的元件，這點與前述的幾個 AJAX 功能相關元件一樣。HiddenField 的主要作用是擔任伺服器程式(本研究採用 VB 語言)與客戶端網頁內繪圖用的 JavaScript 程式之間的資訊交換平台。

此種物件可以讓伺服器與客戶兩端的程式自由讀寫，因此繪圖動作資訊可以經由客戶端的 JavaScript 程式寫入此物件，再由伺服端的 VB 程式讀出，讓伺服器程式取得使用者的繪圖資訊，再以網站公用變數傳遞給交談對象的遠端使用者。相反的，伺服器程式收到遠端使用者傳來的繪圖動作時，也需要將資訊寫入此物件，讓客戶端的程式取得相關資訊後自動執行遠端使用者的繪圖動作，顯示在資料接收者的網頁上。

## 3. 繪圖動作的通訊流程

此程式中最重要的程式機制是繪圖動作的通訊流程。如圖 2 所示，事實上每個客戶執行的程式實體都包含伺服器與客戶端兩部分。基本上繪圖程式必須以客戶端的 JavaScript 撰寫，使用者之間的資訊傳遞則必須使用伺服端的 VB 程式處理。

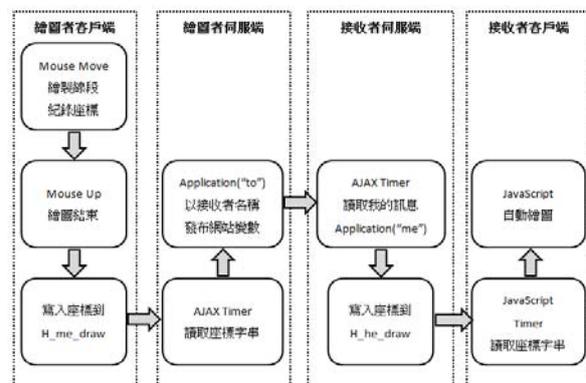


圖 2 繪圖動作通訊流程圖

### 3.1 繪圖者客戶端→繪圖者伺服器

首先繪圖者在其網頁的 canvas 物件上拖曳滑鼠(Mouse Move)時開始了繪圖動作，除了執行繪圖之外 JavaScript 也會同步紀錄已經畫過的座標，直到滑鼠按鍵被釋放(Mouse Up)時，將之前筆畫的所有座標紀錄到圖 1 中標示為 H\_me\_draw 的 HiddenField 物件之中；此期間繪圖者伺服端的 AJAX Timer 程式會持續監看記錄繪圖動作的 H\_me\_draw 物件，一旦資訊出現就會加以處理，將該資訊以網站公用變數 Application(“變數名稱”)的方式發布。處理完畢之後就會將 H\_me\_draw 物件內容清除，以避免重複處理浪費資源。

此處的“變數名稱”語意上應該使用你選擇的

交談對象名稱，但如果這個名稱是中文就會違反 Application 的命名規則。因此事實上本範例內部是使用系統時間刻度值作為使用者代號，這是一個以  $10^{-7}$  秒為單位的極大整數，每個使用者登入的時間前面加上一個“U”字元就是系統內部使用者的 ID 了，實際變數名稱類似“U123456789”這樣的字串。如此設計之後，使用者輸入的名稱就可以使用任何語文，此名稱也會記錄於網站公用變數之中，稍後顯示線上人員時當然是用這個使用者輸入的外顯名稱，前面的時間產生的 ID 僅供程式內部使用。

### 3.2 接收者伺服器→接收者客戶端

接下來，交談對象(接收者)的伺服器端 AJAX Timer 則持續監看有無以自己的 ID 命名的網站公用變數(Application)? 有的話就接收下來，寫入自己網頁中標示為 H\_he\_draw 的一個 HiddenField 物件之中；此時也必須在客戶端的 JavaScript 程式中設計一個具備 Timer 功能的程式，定時監看此 H\_he\_draw 物件；如果有訊息傳入就依據傳來的座標，重製遠端的繪圖動作，並清除 H\_he\_draw 的內容。

如圖 1 所示，我們同時也設計了修改畫筆粗細、顏色與繪圖模式的介面，這些資訊也可以隨著繪圖座標資訊一起傳遞到遠端，由對方程式重繪一模一樣的筆畫或圖形。這些參數的傳遞與使用方式當然牽涉到較細節的畫布繪圖程式的技巧，在此不做細節的敘述，可直接參考成果實作網頁[12]內部的 JavaScript 程式碼。

## 4. 線上交談的通訊機制

線上交談的機制其實與繪圖動作通訊相似，且更為簡單，流程如圖 3 所示：

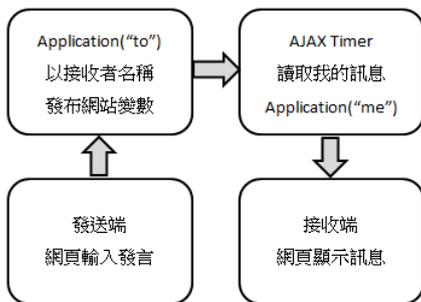


圖 3 線上交談的通訊流程圖

當然如果我們同樣以交談對象的 ID 作為網站公用變數的名稱傳遞交談訊息，對方可能會誤以為是繪圖動作，因此訊息中必須有適當的標記，也就是通訊協定以區隔傳遞的訊息種類。本例中我們是以訊息的字頭中加入“msg:”代表交談訊息。接收端收到 Application 變數時檢視字頭即可辨識資訊種類，如為“msg:”即可直接加入交談看板，無此標記則以繪圖資料處理。

## 5. 影像上傳與分享的機制

### 5.1 影像與資訊分享流程

上傳影像至網站可以使用一般 ASP.NET 網頁提供的伺服器端 FileUpload 物件。此物件可以讓使用者選擇本機檔案上傳至網站，但是要將該影像顯示於網頁上則不是伺服器端程式可以直接完成的動作。因此仍須借助圖 1 中被命名為 H\_image 的 HiddenField 物件，將上傳影像的檔名及長寬等資訊寫入，再由客戶端的 JavaScript 程式偵測取得後將該影像繪製到 canvas 畫布物件之中。

但僅只如此並不能讓交談的對象看到同一影像，因此還需使用前面繪圖與交談的通訊機制，將影像資訊也用網站變數 Application 傳遞到遠端使用者。對方的伺服器端程式收到訊息後也是將資訊寫入 H\_image 物件之中，對方的客戶端程式就可以接手顯示出同一影像了。詳細流程如圖 4 所示：

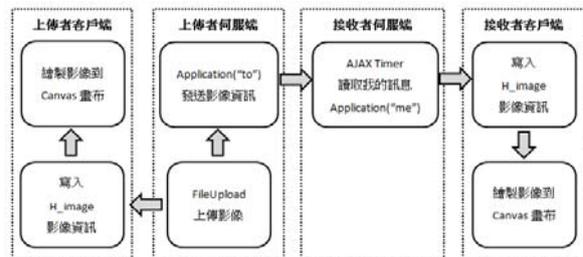


圖 4 上傳與分享影像的通訊流程圖

與線上交談機制類似的狀況，此地依舊使用交談對方的 ID 為網站公用變數名稱，為避免處理方式錯誤，我們也設計了一個訊息字頭“img:”代表此訊息為影像分享訊息。接收端收到之後就可以據此判斷這是影像分享的訊息，正確寫入 H\_image 物件。

### 5.2 避免圖檔衝突與清理舊檔的機制

在此還有一個技術問題，就是網站空間是公用的，如果不同客戶先後上傳同名但是內容不同的圖檔，就有可能將舊檔覆蓋，產生檔案的衝突與混淆。另一方面，使用者上傳網站的圖檔在雙方討論完畢之後就不會再使用，如果沒有定時清理的機制，有可能很快的耗盡網站的磁碟空間。

為了同時解決這兩個問題，我們設計了一個使用時間替代檔名的機制。在此也是如使用者 ID 一樣以單位極小( $10^{-7}$  秒)的系統時間取代原來上傳的檔名，圖檔進入網站後就以此時間為新檔名運作。因為時間單位極小，同時間上傳產生一樣檔名的機率極低；另一方面，我們也可以據此設計按照檔案上傳時間清理舊檔的機制。實際上我們是在每個客戶進入網站時檢查一次網站中來自客戶上傳的所有檔案，如果已經超過一定的時間，譬如一天或數小時，就直接刪除，如此就可以保持資源不致耗盡，也不致刪除到使用中的檔案了。

## 6. 線上名單的處理

### 6.1 網頁無連線狀態的問題

線上名單是一個動態的集合，在此也是一個 Application 網站公用變數，任何線上成員都看得到也可以更改它的內容。有人宣告上線時應增加成員，離線時則移除成員。比較困難的是運作網頁的 http 協定是一個無連線狀態的協定，也就是客戶自網站取得網頁後就已經與網站實質斷線了！如果稍後使用者關閉瀏覽器甚至關機，網站程式都無法得知，除非使用者按下某個特定按鍵主動傳訊告知伺服器。這是與一般 TCP 連線模式的線上程式不同之處，在那些程式中連線是實質且持續的，客戶離線或意外斷線都會立刻被伺服器程式偵測得知。

### 6.2 定時打卡的設計

為解決網頁的無連線狀態問題，我們採用了一個定時打卡的機制，線上名單內除了前述的使用者 ID 與名稱之外還包括一個時間值，使用者程式會經過 AJAX Timer 定時更新我在線上的最後時間，就如同打卡表示我還在線上。同時我們在網站公用程式區也建立了一個類似 Timer 的獨立程式，會每秒定時檢查所有線上成員的最後打卡時間，超過 5 秒未打卡就視同離線，此成員會從線上名單中移除。

以程式設計的角度說明，我們是使用 Hash Table(中文翻譯為雜湊表)型態的集合物件紀錄線上名單，每筆資料包含：(Key, Value)兩部分，Key 可視為每筆資料的名稱，Value 就是資料內容。如前所述的使用者 ID 在此作為 Key 值，Value 則是使用者輸入的名稱加上打卡時間的字串。

### 6.3 上線與離線

使用者宣告上線時，會先取出所有線上使用者的名單，核對線上是否有同名的人，如果有，就會拒絕登入並提出警示，讓使用者另行命名；如果名稱沒問題就建立一個 ID 加入這筆新使用者資料。必須提及的是網頁上顯示的線上名單是使用者自定的名稱，但是為了這些人被選為對話對象時可以取得他的系統 ID 以正確通訊，各使用者的 ID 也必須隱藏在網頁元件內的屬性值中，在本例中就是 ListBox 物件中各個 Item(項目)的 Value 值。

離線時如果是使用者按下『離線』按鍵當然很好處理，用程式控制直接移除該 ID 的資料即可。如果是使用者直接關閉瀏覽器的離線狀況就必須依賴前述的網站公用程式，每秒檢查一次各使用者最後的打卡時間了！

## 7. 成果與結論

依據本文敘述之技術方法，我們已經同步完成一個實體網站[12]，圖 5 就是網頁程式在線上執行時擷取的畫面，線上交談的雙方皆可自由上傳圖片進行繪圖與文字的討論。使用者須先輸入名稱並確

認上線，再由線上顯示名單中擇一線上使用者進行交談，網址為：<http://ycc.tsu.edu.tw/webboard/>[12]。

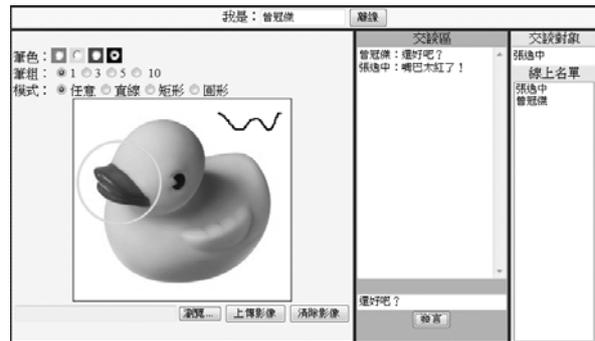


圖 5 本程式範例執行畫面圖

目前此網頁定位為技術示範性質的網頁，並無太多的細節功能與安全機制，使用者登入後如果未宣告離線即關閉瀏覽器，5 秒之後自動顯示離線。進行之交談內容與登入資料也不會留存，上傳圖檔超過一小時就會被刪除。此網頁並提供完整的原始程式設計專案下載，歡迎合法免費參考使用。

從本研究實作與展示的範例中可以看出，使用 HTML5 的 canvas 畫布元件配合 JavaScript 程式，不僅可以在客戶端順暢的執行網頁繪圖的動作，搭配伺服器程式，也可以完成相當複雜的伺服器、客戶端乃至線上使用者之間的網頁通訊。以此為基礎應可提供許多新的設計方案更多想像與發揮的空間。包括遠距互動教學網頁、線上會議或網頁線上遊戲[15]等等。

## 參考文獻

- [1] MSN 即時通訊網頁版, <http://webmsn.msn.com.tw/>
- [2] Facebook, <http://www.facebook.com/>
- [3] paintweb - Online painting application - Google Project Hosting, <http://code.google.com/p/paintweb/>
- [4] AWW - a webwhiteboard, <http://awwapp.com/>
- [5] Twiddla - web-based meeting playground : <http://www.twiddla.com/>
- [6] Java Applet, <http://zh.wikipedia.org/zh-tw/Applet>
- [7] ActiveX, <http://zh.wikipedia.org/zh-tw/ActiveX>
- [8] HTML5 canvas - the basics - Dev.Opera, <http://dev.opera.com/articles/view/html-5-Canvas-the-basics/>
- [9] Canvas tutorial - MDN, [https://developer.mozilla.org/en/Canvas\\_tutorial](https://developer.mozilla.org/en/Canvas_tutorial)
- [10] World Wide Web Consortium (W3C), <http://www.w3.org/>
- [11] Garrett J. J., "Ajax: A New Approach to Web Applications" <http://www.adaptivepath.com/publications/essays/archives/000385.php>
- [12] HTML5 網頁互動白板, <http://ycc.tsu.edu.tw/webboard/>
- [13] 線上遊戲設計課程網頁, <http://ycc.tsu.edu.tw/onlinegame/>
- [14] 網頁白板, <http://ycc.tsu.edu.tw/onlinegame/WebDaub.htm>
- [15] 網頁五子棋, <http://ycc.tsu.edu.tw/onlinegame/webgobang.htm>