

# 臺灣二〇〇八年國際科學展覽會

科 別：電腦科學

作品名稱：打破語言隔閡-以圖點餐之數位餐廳

得獎獎項：第三名

學校 / 作者：國立臺中第一高級中學  
國立臺中第一高級中學

林冠錚  
張方駿

# 摘要

本次研究利用Web Service軟體系統和Microsoft SQL Server的全文搜尋 (Full-Text Search)，並藉由使用者選擇的圖片之組合排列從系統資料庫中檢索比對，篩檢出正確或者最相似的餐點。Web Service是建立於網際網路上之系統架構，且為一個具開放性、分散式的軟體元件，其基礎建立在HTTP、XML、SOAP、WSDL和UDDI等標準協定上，此研究即是透過Web Service系統與資料庫伺服器的連結進行。很多數位化資訊是以沒有結構化的文字資料，存成純文字檔及格式化的文件。這一類的文字資料儲存在檔案系統，而許多組織現在將這一類資料儲存到關聯式資料庫。這樣做就可以廣泛利用資料庫的好處——包括高效能存取、查詢能力、對使用者而言簡單的應用程式使用者介面，以及安全的遠端存取。運用此功能即可在配對完成時搜尋最佳的結果並加以排名。「打破語言隔閡——以圖點餐之數位餐廳」不僅消除語言的隔閡，更讓點餐手續方便許多，並讓錯誤的可能性降低。另外圖片資料庫的增減簡單快速、介面明瞭易懂。希望這些應用及創新能讓人們用餐方便及餐廳事業蓬勃發展。

關鍵詞：Web Service、SQL Server

# ABSTRACT

This research uses Web Service software system and Microsoft SQL Server. To choose the correct or the most similar meals by checking the combination and arrangement of the pictures chosen by users from the system data banks. Web-Service is a system structure establishes at the Internet. And its an open, dispersed software part. Its foundation is established at the standard agreement of HTTP, XML, SOAP, WSDL, and UDDI. The research proceeds by linking Web Service system and data-bank server. Lots of digital data are stored as words files and stylization documents without any structural words data. This kind of word data are stored at file systems, and many organized systems now store them at connected data banks. By doing this, people can make the most advantage of data banks—including efficient access, inquiry abilities, simple applies users' interface, and safe remote access. People can find the best results and place them when this function finishes matching. It can not only eliminate language barriers but also make order procedure more convenient. It can also lower the mistake possibilities. In addition, we can increase or decrease pictures of data banks easily and fast. We can make the interface understood easily. We hope the applications and innovations can make it more convenient for people to order meals. And the business of restaurants can be developed successfully.

Keywords: Web Service, SQL Server

# 自我介紹一



我叫林冠錚，家住彰化，上有兩個哥哥，爸爸是商人，觀念也比較不會古板，因此常常鼓勵我們要發掘新的興趣，長大之後可以照著自己的興趣工作，才是最好的，之後我大哥果然靠著奧林匹亞保送了交大，而二哥也選了有興趣的高醫生物。

從小我哥就教我C語言，當時因為完全不懂英文，只能用硬背的背下來，但是到了國中英文基礎漸漸有了，於是對C語言越來越有興趣。到了高中遇到了老師，有了這個難得的機會參加國際科展。起初題目換來換去，最後終於決定了「打破語言隔閡——以圖點餐之數位餐廳」，上課想、下課做，因此成績也一落千丈，但是為了做科展也值得，得知初選通過時我們非常的高興，但是緊接著又是另一個階段。而我負責的程式部分也遇到不少困難，幸好我讀交大資工的哥哥一步一步帶著我，教我有那些元件、功能，和許許多多的小撇步。

這段時間雖然學校成績掉了許多，但是爸媽都非常支持我繼續做科展，說這是一個難得的學習機會，一定要好好的把握。這段時間雖然辛苦，要寫報告、要做調查、要寫程式.....生平第一次做這麼長期的研究、寫這麼周詳的報告，但是我覺得非常的值得，因為做科展真的真的很好玩。

## 自我介紹二



我叫張方駿，來自於彰化，對於各方面的事情都有極高的好奇心，而電腦則是我  
最感興趣的一環，但起步較晚，從高中開始才接觸程式語言，從c語言開始學習，而冠  
錚是指導我的良師，對我亦師益友，而練習了一段時間之後才開始接觸VB。

我的父親從商，因此也對從商有很多的憧憬，所以在學校第一次的分組選擇了社  
會組，但是分組後聽了多方的意見和想法，覺得需要有專業的才能再從事商業會有比  
較好的發展，而因緣際會下，國中同學冠錚邀請我製作電腦國際科展，讓我有機會更  
接觸電腦這門專業的領域，並藉由這段製作作品的時間練習c語言和VB的能力，我對  
於平常制式化的讀書方式沒有很大的興趣，所以全力投入這次的研究計畫，與冠錚這  
個同窗好友一起付出，常常做到半夜甚至通宵，也會因為理念不合而衝突，但我們都  
希望可以把作品做好。指導老師梁老師也不遺餘力的幫助我們，而我媽媽是英文老師，  
也提供了我們英文上的指導。

因為投入了很多時間在這次的作品上面，成績方面也滑落了很多，但父母親不但  
沒有責罵，更多了許多的支持與鼓勵，希望這次的結果也不要讓他們失望。

# 一、前言（含研究動機、目的）

2004年夏天，我們倆一同隨班上到歐洲（德國）參加管樂比賽，並且享受當地的人文風采和美麗景色，但是在十幾天快樂的旅途中，語言的隔閡造成我們許許多多的不方便，連在台灣學習的英文也派不上用場（因當地以德文為主），因此常常只能以比手畫腳代替語言的溝通。尤其當想要盡情品嚐當地美食的時刻，卻只能看著完全不懂的德文菜單望「單」興嘆，連平常所熟知的德國豬腳，或是當地人的主食麵包、培根都不知道德文長什麼樣子呢！這次旅行因語言隔閡所造成無法享受美食的深刻印象，讓我們思考，如何能突破語言的藩籬，即使身在異國的餐廳，也能享受美味的食物，而不是看著完全不懂的菜單憑空猜測，使我們產生解決「語言」這個問題的動機，全心投入這次的研究計畫。

今年剛好因指導老師的引介有機會參加這次的國際科展，所以把四年前遇到的問題拿出來解決，因此想到了以圖的表示代替文字，因為圖不會隨著國家而改變，而文字卻會因為國家而不同，藉由圖的表示，使人們能直接看到食物真實的樣子，而決定是否點餐，使顧客可以享用自己喜歡的且已經在圖片上瀏覽過的餐點。使顧客誤點的機會降到最低，且不用冒點到自己不敢吃的食物的風險。

這次研究是利用使用者選圖及烹飪方式配對並使用Web Service軟體系統作訊息上的連結。以往用菜單列表示的點菜（或者用PDA）用圖片完全取代，就連烹飪方式都以圖示的方法呈現，讓使用者一目了然，選完後即用全文搜尋（Full-Text Search）的方式搜尋資料庫，找到正確或者最相似的餐點，因為當配對的餐點無法或者太困難準備時，則提供次要的選擇並加以排名，讓使用者可以照自己的意思點餐，就算沒有他想要的餐點時，也不會差異太多，而做完點餐手續後即把訊息傳訊至廚房，讓廚房的人員可以直接的知道要準備什麼食材，讓以往服務生拿著菜單忙的七手八腳，甚至出現出錯菜的情形大幅降低。甚至當使用者對於要吃什麼沒有想法時，介面裡還有提供隨機的選擇，以隨機交替的圖片供使用者點選。

## 二、研究方法或過程

### 2.1 研究方法

#### 2.1.1 SOA

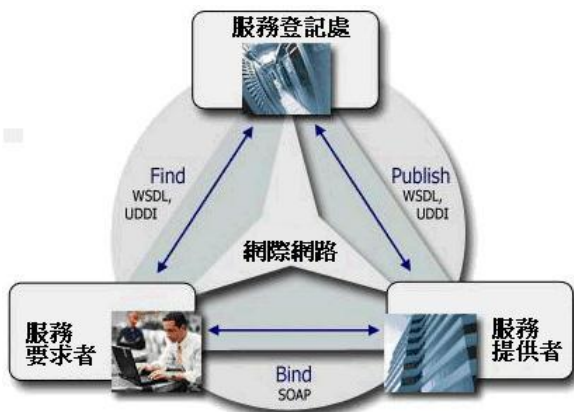


圖 1 SOA關係圖

SOA (Service Oriented Architecture, 服務導向架構)，是一種應用程式架構的概念，將應用程式及資源以可重複使用的「服務」方式呈現，使用標準化的介面相互溝通，藉此提供更高彈性、更高效率、及資訊整合的IT環境。雖然這些概念已存在數十年，但直到出現以標準為基礎的整合技術（如Web Services和XML）之後，SOA才被加速採用。當時各廠商所提出的開發技術和系統環境各異，企業進行系統整合工程時會遇到高成本和開發的難題，技術與成本形成難以跨越的障礙；但隨著現今技術的演進，開放且標準化的XML及Web Services建立後，去除了廠商產品技術間的藩籬，符合企業降低成本、提高效率的目標，SOA也因此逐漸受到各方青睞。

SOA將是企業下一波的IT導入重點：據專家指出，2008年前全球將有超過60%的企業會以SOA作為建立核心應用程式和商業流程的指導原則；未來3年，創新科技、標準化地開發軟體將會增加企業IT整合的需求，而SOA的採用率也將持續成長。

業務和資料服務是服務導向的架構（SOA）中的可重用組件。這些服務可以提供眾多功能，包括資料讀取、基礎架構和業務級功能。它們提供了更進階別的程式設計抽象，可極大地改進效率和生產率，但設計可通過網路讀取的可重用元件並非易事。它

要求進行認真的介面設計，既要著眼於實作服務重用，又要確保網路資源的高效使用，同時最大限度減少服務與使用這些服務的應用程式之間的依賴性。

正因為企業內部的多數商業流程都與科技有關。採用SOA，一個組織即可對員工、顧客和商業夥伴提供快速、良好的服務，無須像從前的專屬方案那樣投入大量的時間與金錢；由於SOA依循共通標準，企業能夠快速反應、彈性應變將更具有競爭力。在整個企業部署SOA，會釋出IT資源，有助於確保科技投資集中於帶動業績成長的核心。

SOA這種與Web Service息息相關的觀念，將在站在未來的市場上頗大的地位。而接下來我們將繼續介紹這次研究所用到的觀念——Web Service。

## 2.1.2 Web Service

Web Service運用服務導向的架構（SOA）的技術，通過標準的Web協議提供服務，目的是保證不同平台的應用服務可以互操作。而Web Service這種技術也讓SOA快速的發展。

Web Service是一種讓「可自我描述的軟體元件（self-describing components）」透過「開放的網際網路與標準協定（例如XML）」而進行「動態連結與協同工作（dynamic component connectivity and interoperability）」的標準方法。然而，Web Service亦可定義成一種分散式運算技術，其利用網際網路技術構成操作介面以便完成分散式運算中的元件連結或其他要求。

Web Service讓應用程式之間可透過共同的網路標準互相連結使用，利用此軟體元件讓軟體系統的建置易於彈性的組裝，隨時經由網路增減元件以調整系統功能，建構出「可程式化網路（programmable web）」，並因此降低了系統整合的複雜性。另外，Web Service利用網際網路為系統的基礎平台，讓軟體開發人員更有效率地進行應用程式的開發與整合，並且使系統達到跨作業平台、跨程式語言，並可以跨越任何終端設備。由於HTML與HTTP為開放且全球性的標準與協定，並且有非常高的普及率，因此讓網際網路成為Web Service最安全且低成本的連結平台。



Web Service之所以能跨越網路以及各種不同的平台結構，是因為具備以下幾項的相關技術：

### 1 XML (Extensible Markup Language, 可擴展置標語言)

可擴展置標語言 (XML)，又可稱為可擴展標記語言，是一種置標語言。通過此種標記，電腦之間可處理包含各種信息的文章。定義這些標記，可以選擇國際通用的標記語言，如HTML，亦可使用像XML這種由相關人士自由決定的標記語言，這就是語言的可擴展性。XML也被廣泛用來作為跨平台之間交互數據的形式，主要針對數據的內容，通過不同的格式化描述手段可以完成最終的形式表達。

### 2 SOAP (Simple Object Access Protocol, 簡單物件存取協定)

Simple Object Access Protocol (SOAP, 簡單物件存取協定) 是一種標準化的通訊規範，主要用於Web Service中。SOAP的出現是為了簡化網頁伺服器 (Web Server) 在從XML數據庫中提取資料時，而無需花時間去格式化頁面，並能夠讓不同應用程式之間透過HTTP通訊協定，以XML格式互相交換彼此的資料，使其與程式語言、平台和硬體無關聯。

### 3 WSDL (Web Service Description Language, Web服務描述語言)

應用程式的開發工具可以依據WSDL的定義，自動地產生伺服器及客戶端的程式碼，也可以依據伺服器的應用程式來產生 WSDL的定義，讓傳統的應用程式可以用新的XML和SOAP等標準的方式在網路上提供服務。WSDL延伸了XML與SOAP等技術，強化了應用程式彼此之間的溝通介面，且讓伺服器和客戶端的程式不一定要在相同的平台或者相同的平台環境下，皆可以互相溝通與協調，因此目前亦成為電腦界和網路廣泛接受的標準。

技術上來說，Web Service是一組協定，建構在SOAP、XML與HTTP等基礎上，繼承這些協定的優點而發展的技術。若以運作程序解釋，Web Service是一種提供應用軟體可透過網際網路互相尋找、連結的程序。而Web Service亦是一個去中心化 (decentralized)、鬆散的關聯結構 (loosely coupled) 與協同作用的 (synergistic) 網路建構模式。

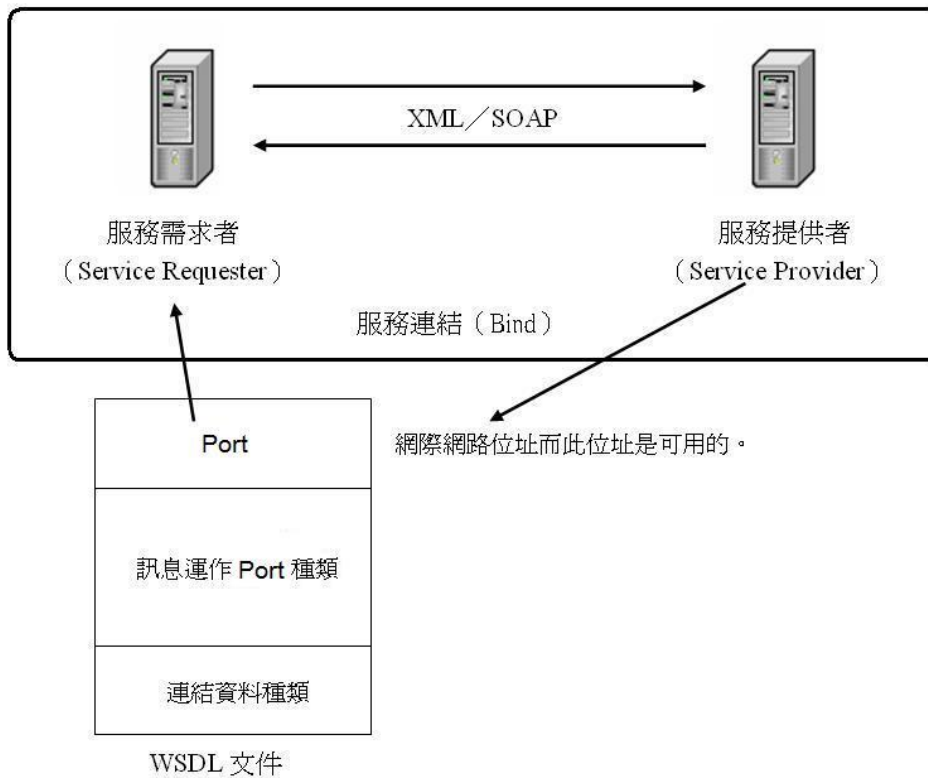


圖 2 Web Service運作方式

### 2.1.3 Microsoft SQL Server

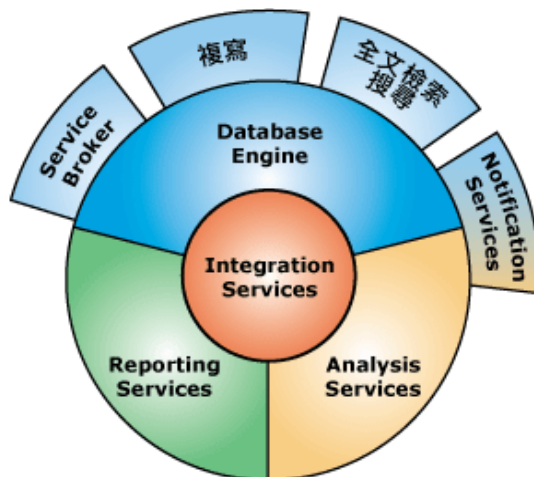


圖 3 Microsoft SQL Server 2005 所含元件

Microsoft SQL Server 2005 是大型線上交易處理 (OLTP)、資料倉儲和電子商務應用程式的資料庫平台；它也是資料整合、分析和報表方案的商務智慧平台。

SQL Server其中一項元件即是我們這次研究計畫應用到的全文檢索搜尋，而以下是對於全文檢索搜尋的介紹。

- 全文檢索搜尋：

全文檢索查詢的功能。全文檢索查詢可以包含單字和片語，或多種格式的單字或片語。「全文檢索搜尋」能夠針對儲存在 Microsoft SQL Server 資料庫中的文字資料進行快速且彈性的關鍵字查詢。

使用「全文檢索搜尋」同時在多個資料表的多個欄位中搜尋一般以字元為主的資料。查詢大量非結構化文字資料時，特別能展現全文檢索搜尋的效能。例如，針對上百萬列的文字資料執行 Transact-SQL LIKE 查詢可能需要數分鐘才能傳回結果；而針對同一份資料執行全文檢索查詢可能只需要幾秒或更少的時間。您可在下列資料上建立全文檢索索引：儲存在 char、varchar 或 nvarchar 資料行中的資料，或儲存在 varbinary (max) 或 image 資料行中的二進位格式資料。

很多數位化資訊是以沒有結構化的文字資料，存成純文字檔及格式化的文件。這一類的文字資料儲存在檔案系統，而許多組織現在將這一類資料儲存到關聯式資料庫。這樣做就可以廣泛利用資料庫的好處——包括高效能存取、查詢能力、對使用者而言簡單的應用程式使用者介面、以及安全的遠端存取。全文搜尋讓開發人員可以在他們的應用程式中以統一方式提供結構化的與未結構化的資料。

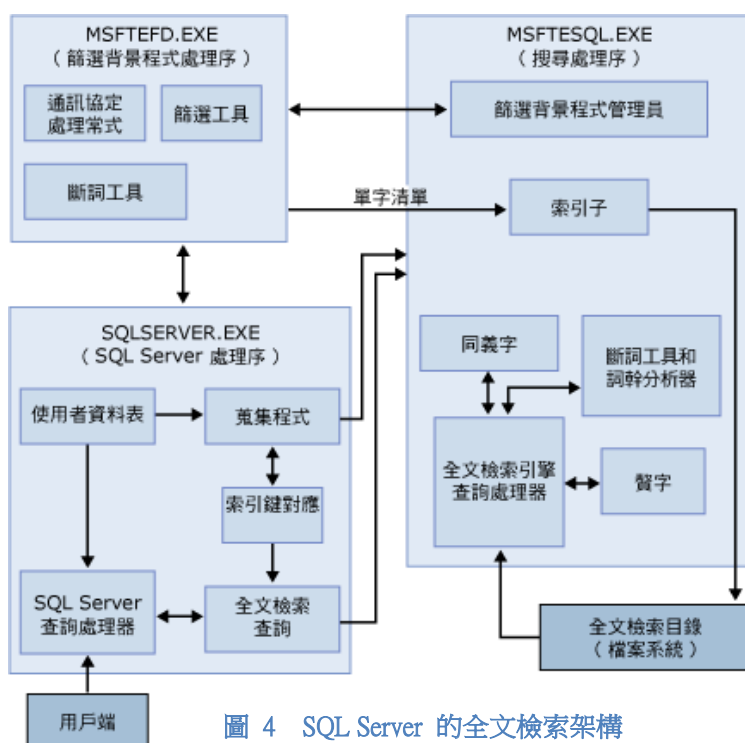


圖 4 SQL Server 的全文檢索架構

## 2.1.4 3-tier

使用者介面處理主要負責兩項工作：從使用者端讀入資料，還有把資料顯示到使用者端。而資料庫存取介面則是對於後端資料庫伺服器作存取的介面。至於應用程式處理邏輯則是整個程式的核心，它會藉由使用者介面來與使用者互動，並視需要，經由資料庫存取界面對後端資料庫做適當的操作。

3-tier把一個完整的網路應用程式分為三層：使用者介面處理（presentation）、應用程式處理邏輯（application）以及資料庫存取介面（data processing）。如圖5：

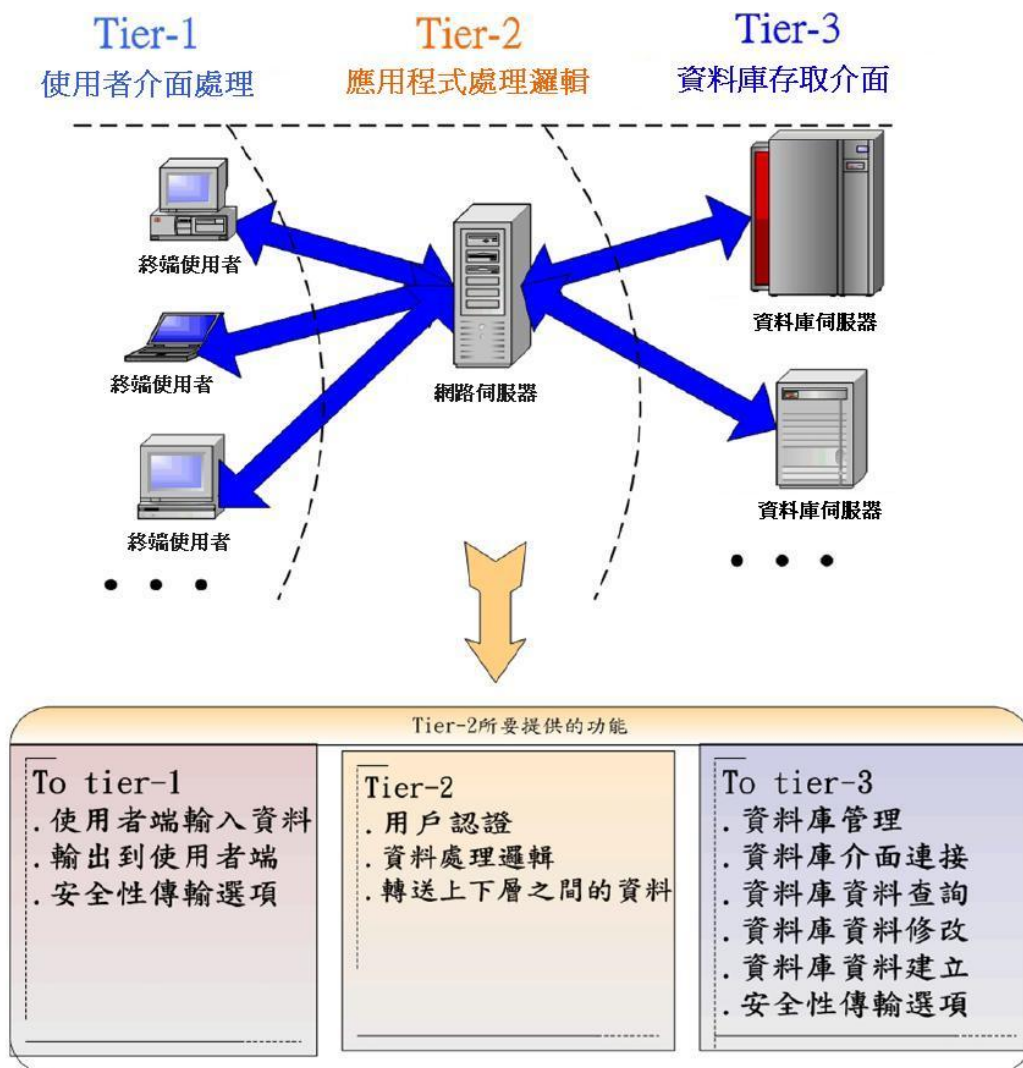


圖 5 三層式架構中第二層與另兩層之關係圖

## 2.1.5 數位影像簡介

在人類的文化發展過程中，影像（Image）就一直扮演著非常重要的角色。人類不僅使用語言、文字，更學會使用圖像來彼此溝通、紀錄各種歷史。直到今天，資訊科技的突飛猛進加上網路的普及，使得數位化的影像資料充斥著整個網路世界。人類對於這麼龐大的影像資料，運用的需求是一天勝過一天。

影像（Image）是現實世界的視覺圖像在平面上呈現出來的結果；人類的視覺圖像、繪畫照片、電影及電視畫面等，都屬於影像。廣義的「影像」可定義為「依某人或某物的外型作成的複製品」，它不僅涵蓋動態影像（Moving Images），也涵蓋靜態影像（Still Images）。而一般靜態影像的表現形式又可分為二維平面（Two-Dimensional）與三維立體（Tri-Dimensional）。狹義的靜態影像所指的是二維平面的靜態圖像，其涵蓋的範圍相當廣泛，例如文件影像、照片、影像、插畫、人工繪圖、電腦繪圖…等。如圖所示。

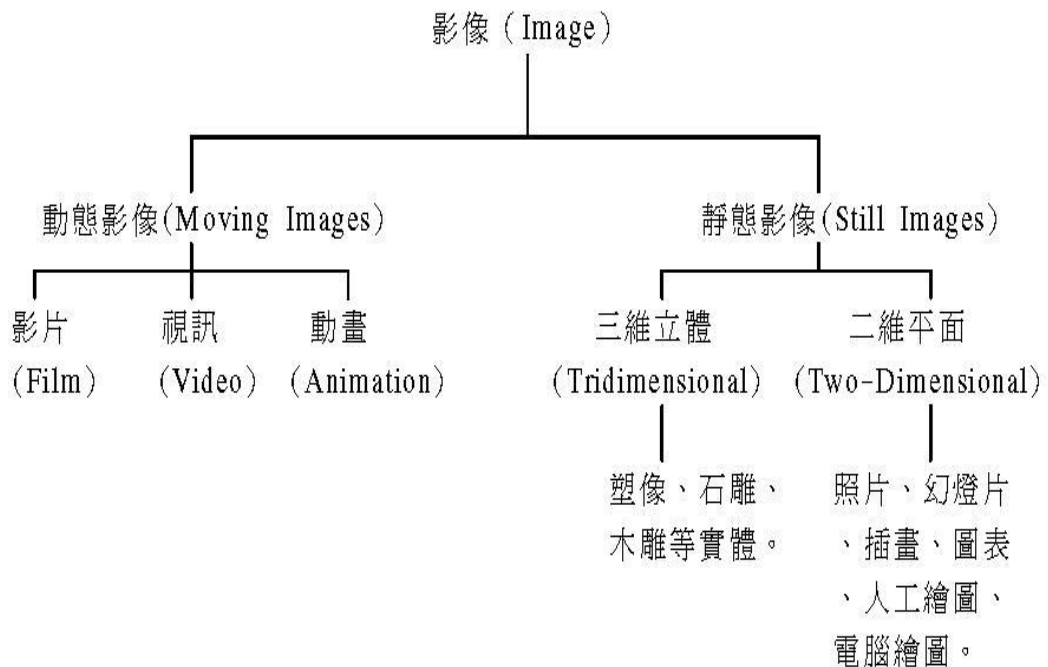


圖 6 影像的定義

## 2.1.6 影像檢索系統

所謂的影像檢索(Image Retrieval)，簡單地說就是在一個影像資料庫中尋找出自己所需要的影像，通常一個影像檢索系統大多含有三個基本的概念，查詢(Query)、比對(Match)和結果(Results)。

影像資訊的檢索技術目前發展出兩種主要的查詢介面，其方法如下：

1. 文字查詢 (Query by Text)
2. 特徵查詢 (Query by Feature)

以關鍵字為主的檢索技術，其優點為：

1. 文字易於表達人類高階的認知與抽象的概念。
2. 相較於內涵檢索，文字檢索較簡單，技術也均已成熟，系統容易實現。

而以關鍵字為主的檢索技術，其缺點為：

1. 較費時費力，資料庫中所儲存的是圖片本身相關的代碼或文字片語資料，這些資料必須人為方式事先建置。
2. 文字敘述有時會流於主觀，不同人會有不同的認知或描述，可能會難以搜尋特定的影像。

內容式的影像檢索技術，其優點為：

1. 電腦直接萃取出影像的特徵元素，如形狀、顏色、紋理等資訊，作為相似比對的內容，不需太多的前置作業、人力的介入。
2. 減少人為的主觀差異。
3. 方便使用者搜尋影像，並毋須語言介入。

而內容式的影像檢索技術，其缺點為：

1. 目前技術尚未成熟，檢索出的圖形，往往無法讓使用者感到滿意。
2. 低階的技術特性，很難突破與人類較深層抽象意境的差距。
3. 抽取特徵或進行特徵比對時，需花較多時間作繁雜的運算。

## 2.2 研究過程

這次研究我們所使用的軟硬體：

1. 筆記型電腦二台  
(CPU：Intel Core2 Duo L7500 1.60GHz，記憶體：4GB)  
(CPU：Intel Core2 Duo T7100 1.80GHz，記憶體：1GB)
2. 程式軟體：Microsoft Visual Basic 2008 Express
3. 資料庫軟體：Microsoft SQL Server 2005 Express

我們假設餐廳管理者所儲存之圖片檔名皆以此圖片之描述命名，如：



檔名：蝦

而完成圖以「肉類名稱、蔬菜名稱、烹飪方式」順序命名，而肉類與蔬菜不可同時不存在，如：



檔名：青椒炒

我們利用Microsoft的Visual Basic 2008 Express製作此次研究計畫的介面以供使用者點選欲選之餐點，運用圖片取代書面文字的表示，並使用Imagelist元件和Listview元件的連結。主菜單是在該Listview元件對應之Imagelist元件並循序輸出，Imagelist資料來自於Microsoft SQL Server 2005 Express所建立之資料庫，而Visual Basic 2008 與SQL Server 2005利用Web Service作連結，近年來無線網路的蓬勃發展，運用網路連結，能讓每台平板電腦不需有獨立資料庫，爾後資料庫若有增減，不需每台更新。

我們也結合了文字查詢（Query by Text）與特徵查詢（Query by Feature）的主要優點；並盡量減少其缺點，管理者使用文字來描述圖片的內容，易於表達圖片的內涵，而顧客運用點選圖片，圖片內容並不需要語言溝通，因此使用者在操作本系統時，會類似與特徵查詢（Query by Feature）的「以圖找圖」，而在系統背後是以文字查詢（Query by Text）的搜尋，我們綜合了以下的優點和改進各種方法的缺點：

綜合兩者的優點：

1. 文字檢索較簡單，技術也均已成熟，系統容易實現，因此修改不難。
2. 文字易於表達圖片的意涵，而食物圖片的內容對於管理者不難。
3. 方便使用者搜尋影像，並毋須語言介入。

改進兩者的缺點：

1. 剔除了特徵查詢往往會讓使用者不滿意。
2. 免除了抽取特徵或進行特徵比對時的長時間。



以下是系統流程圖：

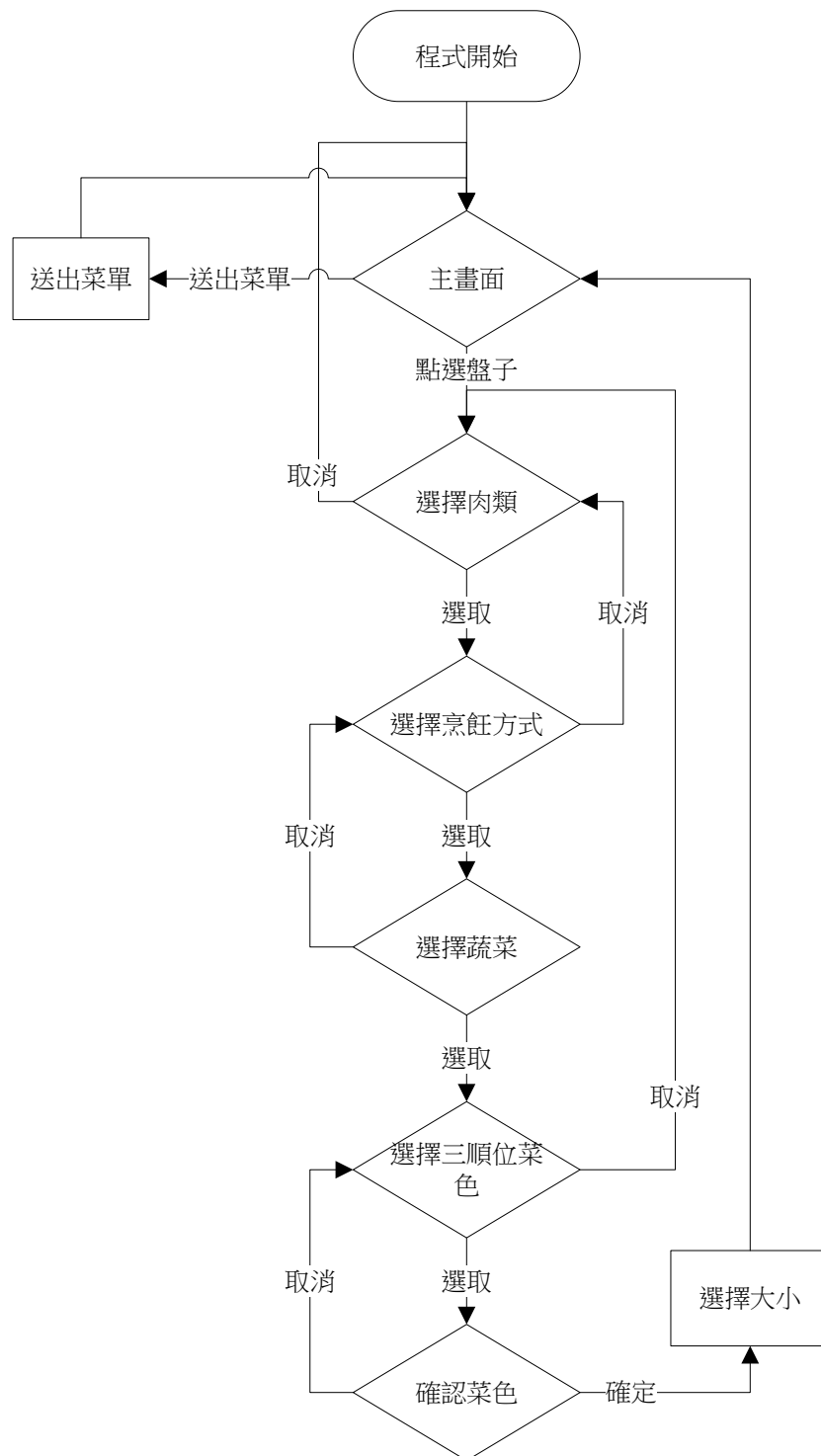


圖 7 點餐流程圖

以下是在肉類、烹飪、蔬菜的選擇介面轉換的程式碼：

```
Public page As Integer = 0      '步驟，開頭為
Public ordered(3) As String    '所點選三樣食材名稱

Private Sub 選擇介面_Click(ByVal sender As Object, ByVal e As System.EventArgs)
Handles 選擇介面.Click

    Dim selected As System.Windows.Forms.ListViewItem
    Dim selectedname As String

    selected = Me.選擇介面.FocusedItem
    selectedname = selected.ImageKey      '取得使用者點選的名稱
    ordered(page) = selectedname

    page = page + 1

    Call 選擇介面顯示(page)
    Back.Enabled = True

End Sub
```

以下是「選擇介面顯示」的程式碼內容：

```
Public Sub 選擇介面顯示(ByVal m)

    Dim ws As New localhost.ImageWebService
    Dim skipenable As Boolean

    If page = 3 Then      '三個步驟是否已結束

        順位.dishname(0) = ordered(0)
        順位.dishname(1) = ordered(2)
        順位.dishname(2) = ordered(1)
        順位.Show()
        順位.TopMost = True
        Me.隨機.Enabled = False
        Me.Enabled = False

    Else

        ordered(page) = ""

        Me.選擇介面.Items.Clear()
```

```

If page = 0 Then      '如果page=0，即使用者正在選擇肉類

    type = 開頭.肉類

    Me.選擇介面.LargeImageList = type
    For n As Integer = 0 To type.Images.Count - 1
        Me.選擇介面.Items.Add("", type.Images.Keys(n))
        Me.選擇介面.Items(n).Group = Me.選擇介面.Groups(0)
    Next

ElseIf page = 1 Then  '如果page=1，即使用者正在選擇烹飪方式

    skipenable = False

    type = 開頭.烹飪

    Dim method() As String
    Dim strcmd As String = "select 烹飪from dbo.所有菜where 肉='" &
ordered(0) & "'"
    method = ws.GetTheNameWeHave(strcmd, skipenable)

    Me.選擇介面.LargeImageList = type
    For n As Integer = 0 To method.Length - 1
        Me.選擇介面.Items.Add("", method(n))
        Me.選擇介面.Items(n).Group = Me.選擇介面.Groups(0)
    Next

Else      '如果page≠0也≠1，即使用者正在選擇蔬菜

    skipenable = False

    type = 開頭.蔬菜

    Dim vegetable() As String
    Dim strcmd As String = "select 蔬菜from dbo.所有菜where 肉='" &
ordered(0) & "'and 烹飪='" & ordered(1) & "'"
    vegetable = ws.GetTheNameWeHave(strcmd, skipenable)

    If vegetable.Length = 0 Or (vegetable.Length = 1 And skipenable =
False) Then '如果蔬菜只有一種選擇方式（沒蔬菜或只能點選一種蔬菜）此步驟便跳過

        page = 3
        If vegetable.Length = 1 Then
            ordered(2) = vegetable(0)
        End If

```

```
    順位.dishname(0) = ordered(0)
    順位.dishname(1) = ordered(2)
    順位.dishname(2) = ordered(1)
    順位.Show()
    順位.TopMost = True
    Me.隨機.Enabled = False
    Me.Enabled = False
    Exit Sub
```

```
End If
```

```
Me.選擇介面.LargeImageList = type
For n As Integer = 0 To vegetable.Length - 1
    Me.選擇介面.Items.Add("", vegetable(n))
    Me.選擇介面.Items(n).Group = Me.選擇介面.Groups(0)
Next
```

```
Skip.Enabled = skipenable
```

```
End If
```

```
End If
End Sub
```

接下來的圖片即是使用者介面及其點餐的過程。



圖 8 開頭顯示畫面，可瀏覽已選餐點

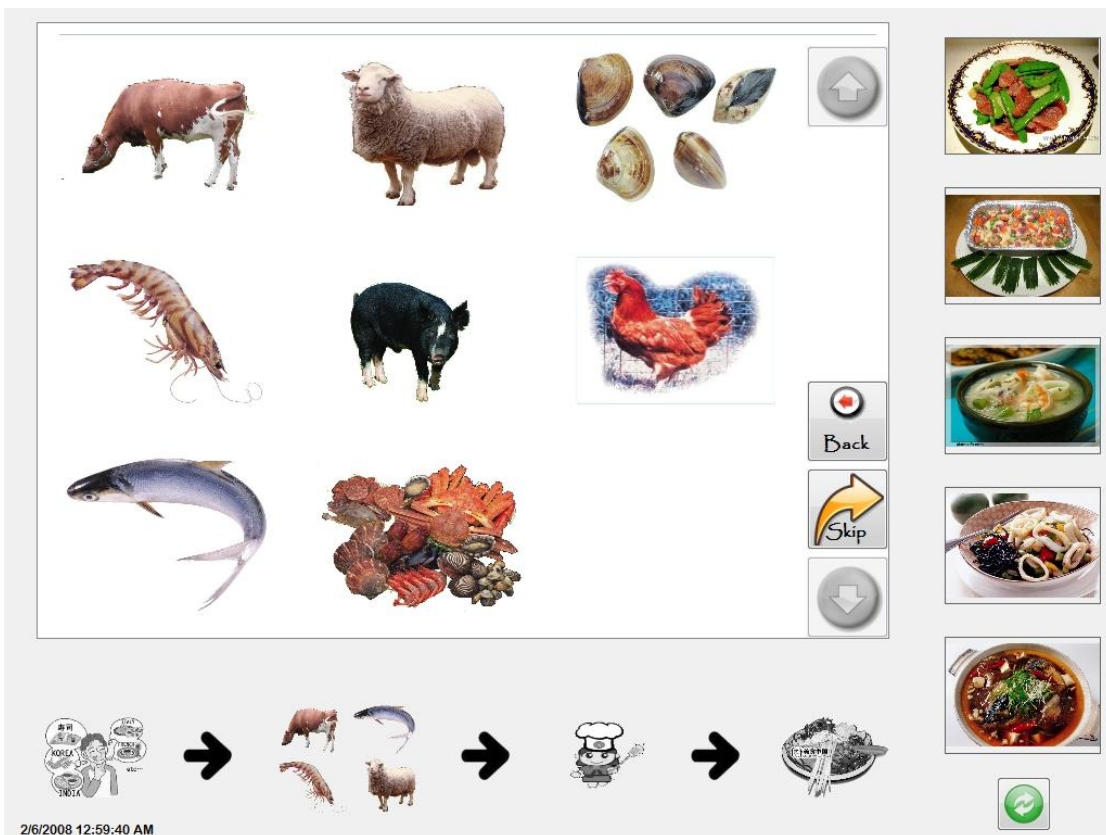


圖 9 點選肉類、烹飪方式與蔬菜介面

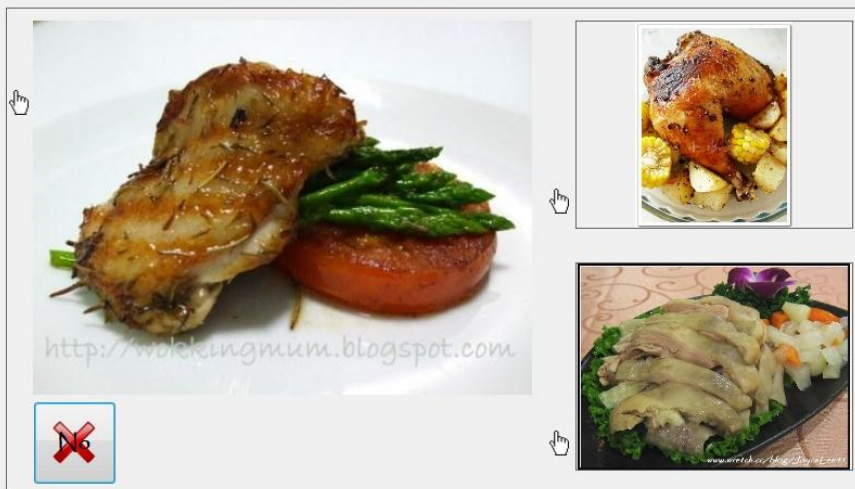


圖 10 排列順位畫面

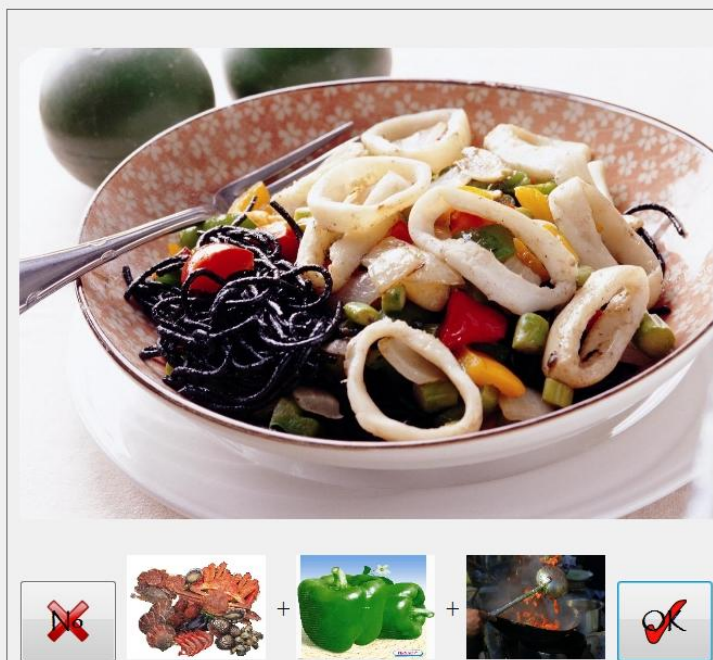


圖 11 確認畫面

我們將使用者所選取之肉類、烹飪方式、蔬菜各個圖片的檔名組合，如使用者肉類選擇牛肉，烹飪方式是炒，蔬菜則選擇青椒，而程式自動將此三種檔名組合，形成「牛肉青椒炒」，接著透過SQL的全文搜尋功能，找到完成圖之檔名，並排出順位，而第二順位若有三個以上（含），則會在若干個第二順位中隨機挑選兩個，第三順位亦是如此。

經過以下在Web Service網頁上的程式碼即可取得所點選菜色之名稱並排出前三順位（假如我們搜尋「牛肉青椒炒」），而回傳值分別是「牛肉青椒炒」、「牛肉空心菜炒」、「牛肉綠花椰菜炒」、「牛肉炒」。其中「`Select * from containstable (dbo. 圖片連結,名稱,'ISABOUT (牛肉 weight (0.5),青椒 weight (0.2),炒 weight (0.3))') Order by rank desc,len([KEY])`」便是SQL語法。

```
<WebMethod(> _
    Public Function FreeText(ByVal three As String()) As
System.Data.SqlTypes.SqlBytes()

    <WebMethod(> _
        Public Function GetFirstThree(ByVal stcmd As String) As String()

            Dim connect As New SqlConnectionStringBuilder
            Dim orders(2) As String '儲存前三順位的名稱
            connect.DataSource = "(Local)\SQLEXPRESS"
            connect.InitialCatalog = "電子點餐"
            connect.IntegratedSecurity = True
            Using con As New SqlConnection(connect.ConnectionString)
                Dim CMD As New SqlCommand("Select * from containstable (dbo.圖片連結,
名稱,'ISABOUT (牛肉 " weight (0.5),青椒 weight (0.2),炒 weight (0.3))') Order by
rank desc,len([KEY])", con)
                con.Open()
                Using myReader As SqlDataReader = CMD.ExecuteReader
                    For i As Integer = 0 To 2
                        If myReader.Read = True Then
                            orders(i) = myReader.GetSqlString(0)
                        End If
                    Next
                End Using
            End Using
            Return orders
        End Function
```

確認介面也顯示出此道菜的組成食材，選擇第二、第三順位能知道其組成，可免除對於飲食有所禁忌的人（如不吃牛）擔心是否適合。

此次的研究計畫之所以選用Web Service做為連結的技術，因為Web Service架構具有以下優點：

1. 透過最大程度地降低對共用理解的要求，從而提高互通性：

Web服務描述語言（WSDL）使用基於XML的介面，是服務提供者與服務要求者之間共用理解的唯一要求。即使餐廳的平台不同的，Web Service之技術也可將之互通，使餐廳應用在軟體的成本可以大幅降低。

2. 使舊有應用程式的互通性成為可能：

WSDL 中對舊有應用程式進行包裝，Web Service 架構毫不費力地使這些應用程式之間新的互通成為可能。此外，安全、中間件和通信技術能夠預先包裝，做為環境前提參與 Web Service。這項好處一樣使餐廳對於應用程式的成本降到最低，即使用舊有應用程式，一樣也可以和新的應用程式互通，讓應用程式替代的方便性大大的提升。

然而我們為何不直接使用DCOM或CORBA連結，DCOM的技術如此成熟、CORBA開放性這麼好，這麼多優點，為何不使用呢？

我們發現，DCOM的技術固然成熟，但是由於DCOM技術只能用Microsoft開發，若使用異質平台，那麼DCOM便無用武之地。我們不能排除業者使用異質平台，因此DCOM並不適任。

而CORBA標準不統一，不一樣的廠商便有不同的標準，讓基礎成本提高許多，因為各個廠商有不同的標準，因此購買開發工具的成本便不容小覷，而開發人員少，相對的開發維護成本非常高。我們為了讓此系統能普及，以這樣的高成本，即讓許多的業者望之卻步，因此我們放棄了使用CORBA技術。



以下是我們對於Web Service、DCOM、CORBA的實驗比較表

表 1 Web Service、DCOM、CORBA比較表

	Web Service	DCOM	CORBA
傳輸速度	慢	快	快
各廠商標準	一樣	不一	不一
可否跨平台	可	不可	不可
開發成本	低	高	高

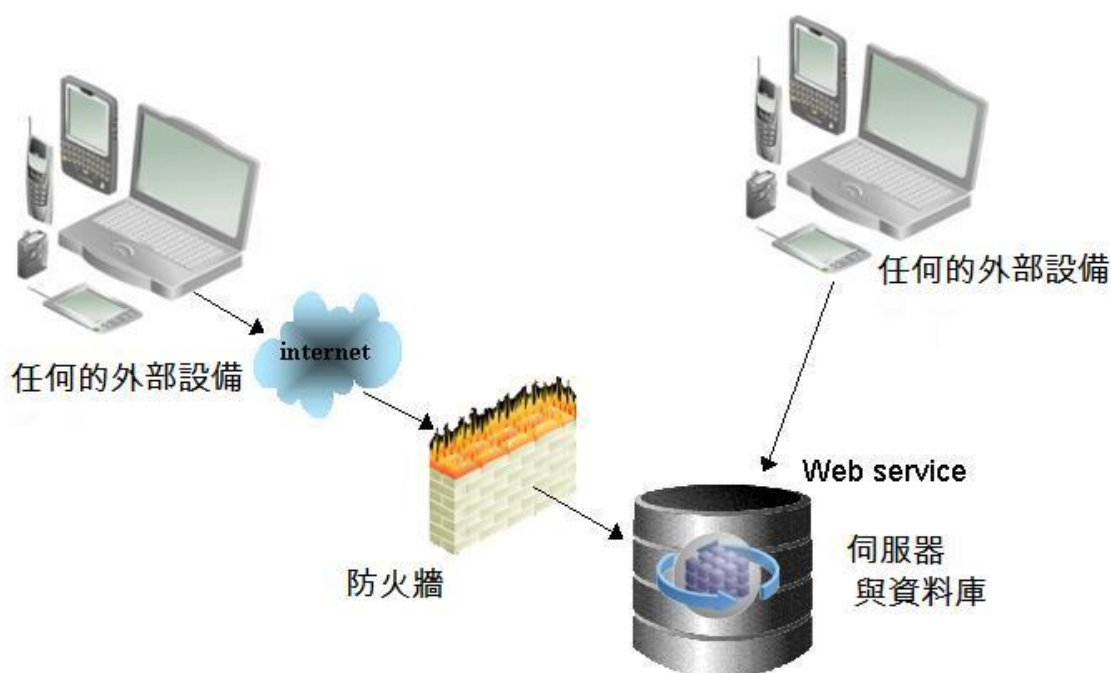


圖 12 Web Service 的跨平台可行性

而我們也參考3-tier（三層式架構）的架構，由介面（presentation）接收使用者（顧客）所點選的餐點，傳送到應用程式處理邏輯（application），資料處理過後，便產生一個字串（使用者所點選之餐點組合名稱），接著連結資料庫存取介面（data processing）運用全文搜尋，列出前三順位，再傳回應用程式處理邏輯，此層便搜尋Imagelist元件裡相符之圖片，最後，輸出到介面使用者端，讓使用者一目了然。然而之後若想要修改介面，只要把介面（presentation）修改或置換便可以完成，無須再修改程式處理邏輯部分和連結資料庫存取介面的部分，大大提升方便性。

然而經過我們多次模擬點餐之後，發現了會有太多第一順位不是客人所點選的機

會，我們思考之後，提出了兩個解決方法：

1. 先到所有菜單裡搜尋有關此肉類的所有菜名，再從烹飪方式裡搜尋有關剛剛所篩選出來的菜名。
2. 新增一個View，並運用SQL語法，直接分離出各道菜的肉、烹飪方式與菜名組成。

而我們最後採用的是後者，前者雖然可以減少檔案大小，但是不斷的進進出出資料庫會拖慢整個程式，當客人在點餐時還要等程式執行完畢，是非常不合理的事，而且實際減少的大小完全不符合經濟效益。而後者藉由SQL語法的配合，會隨時同步更新並自動分離肉、烹飪方式與菜出來，這種方法只需要一次進出資料庫即可，執行速度明顯的加快很多。以下是產生此View的SQL語法：

**SELECT**

dbo.圖片連結.名稱, dbo.肉類.名稱 AS 肉, dbo.蔬菜.名稱 AS 蔬菜, dbo.烹飪方式.名稱 AS 烹飪, dbo.圖片連結.圖片

**FROM**

dbo.肉類 INNER JOIN  
dbo.圖片連結 ON dbo.肉類.名稱 = SUBSTRING(dbo.圖片連結.名稱, 1, LEN(dbo.肉類.名稱)) CROSS JOIN  
dbo.蔬菜 CROSS JOIN  
dbo.烹飪方式

**WHERE**

(SUBSTRING(dbo.圖片連結.名稱, LEN(dbo.肉類.名稱) + 1, LEN(dbo.蔬菜.名稱)) =  
dbo.蔬菜.名稱)  
AND (SUBSTRING(dbo.圖片連結.名稱, LEN(dbo.肉類.名稱)+ LEN(dbo.蔬菜.名稱)  
+ 1, LEN(dbo.烹飪方式.名稱)) = dbo.烹飪方式.名稱)

我們也對全文檢索的搜尋方式和等級排名（RANK）的產生原理，做了一些研究。首先是全文檢索的搜尋方式，若以下此資料表想要進行全文檢索搜尋，就要建立全文檢索索引，則此資料表必須有一個唯一且非Null的資料行，例如此資料表的「索引」資料行，全文檢索會指出「牛肉」位於菜名的關聯資料列索引（索引為0、1、4、5），而在處理全文檢索搜尋「牛肉」時，會從全文檢索索引取得0、1、4、5，然後SQL Server會使用這些索引傳回相符的資料列。

表 2 全文檢索搜尋方式範例資料表

索引（唯一且非Null）	菜名
0	牛肉青椒炒
1	牛肉炒
2	雞肉烤
3	豬肉青椒炒
4	牛肉空心菜炒
5	牛肉煮
6	青椒炒
7	蝦炸

而等級排名（RANK）是當使用者全文檢索搜尋「牛肉」時有此字串的菜名會獲得若干分數，青椒和炒亦是如此，最後RANK是前面分數的總和，這是未經過權重分配的，若要改變權重，只要將分數乘上分配的比例即可。

表 3 等級排名（RANK）的範例資料表

	檢索「牛肉」	檢索「青椒」	檢索「炒」	RANK
牛肉青椒炒	100	100	100	300
牛肉炒	100	0	100	200
雞肉烤	0	0	0	0
豬肉青椒炒	0	100	100	200
牛肉空心菜炒	100	0	100	200
牛肉煮	100	0	0	100
青椒炒	0	100	100	200
蝦炸	0	0	0	0

### 三、研究結果與討論

由於每個人的飲食習慣不同，因此我們做了200人的實際調查，對於「肉類」、「烹飪方式」以及「蔬菜」三類在平時飲食點餐最考慮的因素，並採單選選擇，結果如下：

表 4 實際調查結果表

肉類	烹飪方式	蔬菜
46%	35%	19%

因此，對於第二、第三順位的條件，改變全文搜尋的權重，讓「肉類」佔了50%、烹飪方式30%、蔬菜20%，以這樣的權重，最符合我們實際調查的想法。

我們利用這段SQL語法來修改權重，以達成我們實際調查的結果：

```
Select * from containstable (dbo.圖片連結,名稱, 'ISABOUT (牛肉 weight (0.5),青椒 weight (0.2),炒 weight (0.3))') Order by rank desc, len([KEY]) 其中'ISABOUT (牛肉 weight (0.5),青椒 weight (0.2),炒 weight (0.3))'便是設定搜尋權重的字句。
```

而在經過實際操作程式以及偵錯數次，並修改程式碼，經過多次的修正，搜尋的結果與我們預期的答案符合的幾乎無誤，成功率達100%，讓使用者操作本程式時可完全放心，絕對可以無誤地找到自己所點的餐點和經過搜尋順位排列的結果。

## 四、結論與應用

本次研究利用Microsoft Visual Basic 2008做了點餐的介面及全文搜尋技術進行點餐的手續，希望可讓未來餐廳內的點餐流程更加簡單方便，而資料庫的使用使得圖片的增加或者減少更加容易，更使得圖片資料原來要占用的大量空間大幅降低，一目了然的介面希望能夠完全的消除不同國家的語言隔閡，整個作業流程都是利用點選圖示的方式進行。而台灣面板的技術非常成熟，將來可以和觸控式螢幕結合，並使用Microsoft .NET CF 2.0開發讓本程式能安裝在使用Windows Mobile作業系統的平台如e-Box上，只需百元主機配合觸控螢幕即可，方便性更是大大提升。

運用隨機的功能也是研究計畫中另一個優點，讓消費者不僅可以自己配菜選擇，更可以點選螢幕上隨機交替的食物圖片，省掉自己慢慢點餐的功夫。而Web Service的技術希望可以讓更餐廳的應用程式互相整合，不會因為使用的軟體不同而不能利用，再運用無線網路的技術，便成為名符其實的數位餐廳。也使得連結更加迅速、安全、簡單，且失誤率幾乎等於零。

綜合以上之討論，本研究未來須改善的部份可歸結出以下幾點：

1. 介面的設計可以更人性化，可以更一目瞭然
2. 資料庫所容納的資料是否可以更多，能否更加方便
3. 可以做到與網際網路連結，讓在遠距離的人們也可以使用
4. 全文搜尋的使用或者排名可以更加精準完善
5. 再擴展到其他的應用，如：服飾、居家設計

## 五、參考文獻

1. 吳冠龍、尤孝庭，灰階影像內容之檢索技術，台灣 2005 年國際科學展覽會研究報告
2. 程一正，2002，校務行政系統開放源碼軟體平台之研製
3. 陳慶蔚，2002，網路系統架構之競爭策略—以 Web Services 與 Microsoft .NET 為例
4. 章立民研究室，Visual Basic 2005 檔案 IO 與資料存取秘訣，碁峯資訊股份有限公司，2006
5. [http://zh.wikipedia.org/w/index.php?title=Web\\_service&variant=zh-tw](http://zh.wikipedia.org/w/index.php?title=Web_service&variant=zh-tw)
6. <http://zh.wikipedia.org/w/index.php?title=SOAP&variant=zh-tw>
7. <http://zh.wikipedia.org/w/index.php?title=XML&variant=zh-tw>
8. <http://www-128.ibm.com/developerworks/tw/event/soa/soa.html>
9. <http://www.microsoft.com/taiwan/msdn/library/2003/Jun-2003-tw/service04172002.htm>
10. [http://www-07.ibm.com/software/tw/sf/archive/vol2/hn/what\\_s\\_soa.html](http://www-07.ibm.com/software/tw/sf/archive/vol2/hn/what_s_soa.html)
11. <http://technet.microsoft.com/zh-tw/library/ms172075.aspx>

## 評語

此專題的關鍵技術為「全文式檢索原理」。用這樣的技術在此主題上並不是很有效率的作法，例如菜單的實作成本太高。

應可想出更好的；更具效率，更低成本的演算法來解決此主題。