

# 缺字系統整合動態組字之應用

<sup>1</sup>黃俊瑋      <sup>2</sup>林金龍      <sup>2</sup>黃國倫

<sup>1</sup>元智大學資訊管理系

<sup>2</sup>中央研究院資訊科學研究所

s932658@mail.yzu.edu.tw, {eddy, kulun}@iis.sinica.edu.tw

## 摘要

近年來，由於『數位典藏國家型科技計畫』之執行，在漢字資料處理時，因罕用字或古字無法存入系統中之「缺字問題」[10]已經有許多單位研究多年，目前在典藏單位中多數以中央研究院資訊科學研究所文獻處理實驗室發展出的漢字構形資料庫[7,9]為基準。而現今網際網路發展蓬勃，使用網際網路來閱讀書籍已經相當平凡，古文中的缺字問題則是解決閱讀障礙的重要一環。

本文延用漢字構形資料庫的概念，加入「動態組字」[5]的方法，著重在瀏覽器介面上顯示缺字的應用上，主要目的為解決目前漢字構形資料庫中尚未支援的缺字。此架構可以利用於文獻典藏、缺字顯示、以及教學等方面。得以改善少數尚未編碼之缺字的顯示問題。

**關鍵詞：**數位典藏、缺字顯示、動態組字

## 1. 前言

電腦的編碼系統原本是採用一個字對一個碼，才能夠在電腦上顯示出來，人類雖然看得懂字形，但是電腦只能看懂內碼，且能夠被放到電腦的數量，受限於編碼空間的限制，必須挑選適當的文字或符號放到電腦，但是使用其他電腦時，該電腦的造字區必須跟您的造字區一致：同樣的字形圖得對到一樣的碼，否則會得到錯誤的顯示，這樣的方式使得彼此的造字難以分享。

當部分沒有被放到字符集的文字，使用者又需在電腦上顯示，將無法找到字形所對應的字碼，造成無法在電腦上表達，這些文字稱為缺字。

為了應付缺字問題，一般治標的方法是：在交換碼的使用者造字區內，選一個碼位，並造上所缺的字形。這種做法固然可以在該電腦上顯示出該字形，但是付出的代價巨大，且沒能真正解決問題。茲將這種做法衍生的問題略述如次：

1. 大幅增加了資料登錄的工作；  
當鍵入資料時的成本，尤其古籍中之缺字數量多。
2. 造字的管理不易；  
如果所缺的字不多，登錄工作以人工方式尚可解決，可是，若古籍之量龐大，缺字數動輒上千，再加上所造的字無法依交換碼的字序排列，

那麼，這些新造的字是極不易查核比對的。

3. 造字的空間不足；  
通常，交換碼中允許造字的空間都在數十字至數百字之間，不會太大。以五大碼而言，造字空間是最大的了，也只有5809字。超越此數後，勢必造成碼位的重疊或衝突。
4. 造成資訊共享的障礙；  
使用者造字所訂的單行碼，破壞了交換碼的通用性，而使電子文件無法與大家共享。

且目前資訊科技與網際網路的快速發展，人們更常利用網際網路取得資訊內容，相較於中央研究院資訊科學研究所文獻處理實驗室的漢字構形資料庫單機版軟體，如果能將其應用在網際網路上，更能發揮其推廣效果，也能讓學習者隨時隨地與不限時空的利用網際網路來學習、了解漢字構形的發展。

目前中央研究院資訊科學研究所文獻處理實驗室所發展漢字構形資料庫是由人工一個字一個字所建立，無法即時完成所有缺字之建造，造成目前仍然有部分缺字在瀏覽器介面上無法顯示。

在本文中，我們利用漢字構形資料庫整合由Java所開發的動態組字系統，將其應用在網頁缺字顯示上，動態組字在此缺字系統上扮演重要的角色，解決使用者閱讀上的問題。此方法在效能上也有良好的表現，不會另外造成網頁顯示的相關問題。在整合於缺字系統後，更能凸顯與其他缺字顯示的優異之處。

## 2. 相關理論與技術

本章節將針對本研究所使用之相關理論與技術做一個簡單的介紹。

### 2.1 漢字構形資料庫

目前電腦處理漢字的諸多缺失，例如缺字、異體字等問題，主要的原因在於電腦裡的漢字知識嚴重不足。有鑑於此，中央研究院資訊所文獻處理實驗室自1993年起，即先由字形著手，建置漢字構形資料庫。

目前漢字構形資料庫是由甲骨文、金文、楚系文字、小篆及楷書構形資料庫組合而成，如圖1。從圖1可看到每個構形資料庫都有各自的字集、部件集、字根集、異體字表及電腦字型，各個字集間彼此也有銜接。簡單的說，漢字構形資料庫的主要

特色有如下：

1. 銜接古今文字以反映字形源流演變。
2. 收錄不同歷史時期的異體字表，以表達不同漢字在各個歷史層面的使用關係。
3. 記錄不同歷史時期的漢字結構，以呈現漢字因義構形的特點。
4. 使用構字式及風格碼來解決古今漢字的編碼問題。

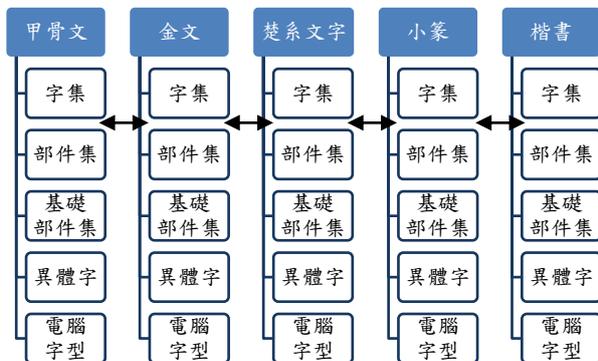


圖 1. 漢字構形資料庫的組成

## 2.2 構字式

為了徹底解決現行漢字交換碼不足所造成的缺字問題，中研院資訊所文獻處理實驗室從漢字字形結構的拆分與分析中，利用有限的部件及字根的組合方式來表達任一漢字，此稱為構字式[8]。例如『穎』，以構字式拆解的話，可拆分成「景」與「頁」兩個部件，其中為了表示部件與部件的連接關係，故定義了三類共計十三個的「構字符號」，故『穎』的構字式為『景 $\Delta$ 頁』。因此構字式是由部件和構字符號組成，且「構字符號」也是一般文字和缺字的辨識依據。

因此以構字式為基礎下，將收錄的漢字建立以 Big5 為編碼系統的『漢字構形資料庫』。目前漢字構形資料庫已收錄了楷體字形 62,366 個、小篆 11,100 個、金文 20,069 個，甲骨文 2,197 個，所以當各典藏單位面對數位化所遭遇到缺字問題時，若使用漢字構形資料庫做為缺字的解決方案未嘗不是個成本較低、功能又較完備的好方法。且目前漢字構形資料庫仍然持續在資料量上擴充。

構字符號種類如表 1、表 2 所示。利用簡單的連接符號為主來架構出中文字體，部分較複雜之字形則以起始符號、終止符號來包夾表示。

表 1. 構字式之連接符號與部件序

類別	符號	說	明	構字式範例
連接符號	$\Delta$	當部件的連接順序由左至右		順 = 川 $\Delta$ 頁 唄 = 口 $\Delta$ 貝
	$\triangle$	當部件的連接順序由上至下		含 = 今 $\triangle$ 口 員 = 口 $\triangle$ 貝
	$\triangleleft$	當部件的連接順序由外至內		圓 = 口 $\triangleleft$ 員
部件序	$\square$	按部件書寫順序輸入，前後以起始符號 ( $\square$ ) 和終止符號 ( $\square$ ) 包夾。		解 = $\square$ 角 刀牛 $\square$

表 2. 構字式之方便符號

類別	符號	說	明	構字式範例
方便符號	$\circ$	二個相同部件直連		炎 = $\circ$ 火
	$\circ$	三個相同部件直連		
	$\infty$	二個相同部件橫連		朋 = $\infty$ 月
	$\infty$	三個相同部件橫連		
	$\circ$	三個相同部件呈三角狀排列		焱 = $\circ$ 火
	$\infty$	四個相同部件橫連		
	$\circ$	四個相同部件直連		
	$\circ$	四個相同部件呈四角狀排列		燚 = $\circ$ 火

## 2.3 全字庫

全字庫[1]是目前政府機關解決缺字問題的主要方法之一，目的是為了解決自造字交換問題，機關、企業、團體內部同字不同碼問題，協助機關、企業、團體整合及管理個人電腦上的中文字集。在網頁的解決方面只有較為陽春的解決方案，且未能方便的推廣使用。全字庫目前收錄八萬餘字，文字來源為各政府部門，如戶政、地政等機關，且沒有包含漢字的構形知識，也無法在編碼架構上反映異體字。

我們可以由「政府機關中文資訊應用概況調查分析結果」[3]中，顯示半數以上機關皆有缺字的困擾，而機關遇到中文缺字時的解決方案有 39% 會選擇自行造字，即顯示全字庫在政府機關之間的使用尚未普及化。該調查亦顯示出一般機關的自造字用途分別為人名、地名、專有名詞、廠商名稱的使用率最高。但目前全字庫大部分只應用在政府公家機關，例如戶政事務所，一般民眾以及學術單位使用量極少，較為不普及。

## 2.4 動態組字原理

英文字母，一套字型只要做好 26 個字母就好了，漢字的字型卻要有數千到數萬不等才能供應基本訊息交換，這製作、運作的成本都遠高於表音文字，這對於漢字的文化發展仍相當的不利。

鑑於此兩岸各有不同的人發展了動態組字技術，動態組字本質上就是解除西方電腦系統對漢字的束縛，在記憶體中，只要存有一千多個字根部件，漢字即可由這些部件以遞迴的方式作即時二維組字（英文是一維），來顯示到螢幕上來，使用者也可以應各式的需求自行組新字。動態組字除了容易組字的好處以外，組字的過程可以反向操作，也就是拆字的動作，就可以作漢字理解之用[5]。如圖 2，拆字程式允許單一文字輸入，進入字形檔所構成的字碼表取得該字的部件組成，判斷是否為末級部件，若為末級部件，就不拆字直接輸出，若不是末級部件，從字碼表內取得的子組件逐一轉換為文字後輸出。

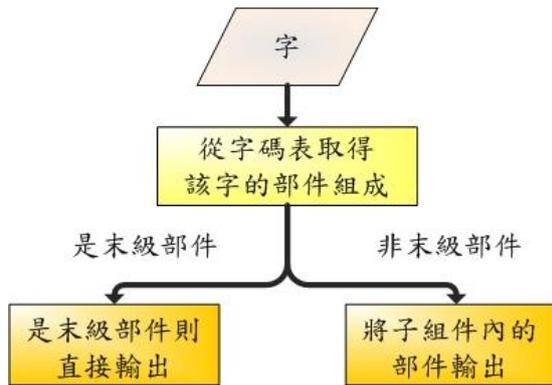


圖 2. 反向拆字流程

此外 Unicode 標準從 3.0 開始，開始有表意文字序列 (Ideograph Description Sequence)，為採用表意文字組合符 (Ideograph Description Character) [12]與文字部件的組合，為缺字描述定下一個標準基礎，只要加上解譯描繪的機制，也是一種動態組字的實現，如表 3。

由剎那搜尋工坊[2]所發展動態組字的結構可分為輸入與輸出[4]，如表 4。其該字形檔的資料只有兩種，分別是「複合部件」與「末級部件」，如表 5。在複合部件上的表示方式分別代表著部件、異體字編號、左上角 x 座標、左上角 y 座標、寬度、高度，在末級部件上的主要則是由三種指令來代表，分別為 MoveTo、LineTo、Curve。由此結構所發展出的字形檔容量小很多，在應用與實作上是未來的趨勢之一。動態組字單機版執行畫面如圖 3[6]。

表 3. 動態組字所使用的表示法

符號	說明	範例
	兩個部件關係是以左右組合	順=川頁
	兩個部件關係是以上下組合	員=口貝
	兩個部件關係是由外包圍內	圓=口員

表 4. 動態字形產生器結構

輸入	Unicode 內碼，構形(含比例參數，字框)，字級，字體等
輸出	Bitmap 字形或向量字形(座標)

表 5. 動態組字字形資料結構

複合部件	明 00=日 0191645B0 月 0661183DF
末級部件	日 00=00003C1A00013CE900003C280001C1280001C1E800003C770001C17700003CCD0001C1CD



圖 3. 動態組字單機版執行畫面

## 2.5 Unicode

由於 Unicode[13]在其編碼中同時容納了全世界各種語言的字元和符號，因此已成為國際常用的交換碼標準。目前 Unicode 在漢字的支援方面目前已經定義超過七萬多個字元，收納的字遠多於 Big5 (約收藏一萬三千多個字)[11]，且收納字的範圍還在繼續增補中，因此也的確解決了某些層次字形編碼不足的問題。並且在許多系統支持下，在資訊交換上也的確有其便利性。因此若在數位典藏中使用 Unicode，可以降低缺字發生的可能性。但當面對大量的古書資料時，由於包含的古文字數量相當驚人，並且在不同的時間存在著同義異形的相異字形(如：楷書、小篆)，而 Unicode 所收錄的是以常用字為主，因此根本無法包含所有的古文字，即使將每一個字皆擴充進去，那所佔的編碼空間將會相當可觀。再著，對於有著同樣意義的字，分別指派給不同的編碼，則未來在進行資料檢索時，將無法搜尋到所有的同義字，因此系統設計上將會增添許多新的難題。

## 3. 系統架構

目前在缺字的使用者方面，以歷史學家、漢學家、考古學家、博物館管理者、圖書館管理者、一般網路使用者以及政府相關機構為主，所以主要設計目標為方便使用者容易使用，雖然缺字系統部分有單機解決的方案，但並非所有使用者都有安裝以及長期使用缺字系統軟體，若能將其應用在網頁上則更為方便。此外還有缺字的可交換性、瀏覽器彼此相容、系統整體的維護與延展上，這些也是設計目標之一。

由於我們所使用的漢字構形資料庫是以人工方式處理，且缺字數量日增月益，仍有許多缺字尚未建立於資料庫內，造成在瀏覽器顯示上未能顯示出該字形或字形圖片。我們在設計的方法上，整合剎那搜尋工坊所發展的動態組字方法，稱之為可攜式造字引擎，將其使用在 Web 應用上，得以解決前述所提到的問題。

以下章節主要說明如何在 Web 應用上整合漢字構形資料庫與可攜式造字引擎，包含運作過程與完整的系統架構。

### 3.1 漢字構形資料庫整合動態組字

在尚未整合動態組字之前，原本的處理過程中，若漢字構形資料庫中無該構字式的資料，則會以空白顯示取代原有的構字式，使用者必須經由網頁原始碼才可以得知該構字式的表示法。

以下所提出的架構主要目的是將漢字構形資料庫尚未新增之字形轉換為動態組字，解決即時新增字形之問題，只要該構字式表示法正確，即可以顯示出字形圖片。完整的系統流程如圖 4 所示，當使用者開啟瀏覽器瀏覽網頁時會觸發 JavaScript 程式來分析該網頁中是否有構字式表示法，再將其傳入資料庫查詢模組內，若該構字式在資料庫中無此資料，則會進行正規化的動作，以便於判斷是否是因為表示法的差異導致資料庫查詢不到，如果在正規化動作後依然沒有找到該表示法，則將此構字式送入動態組字模組，再由該程式去進行判斷與產生圖片的程序。在此架構中與之前的架構比較下，得以解決現階段的缺字尚未造字完成而無法顯示的問題。

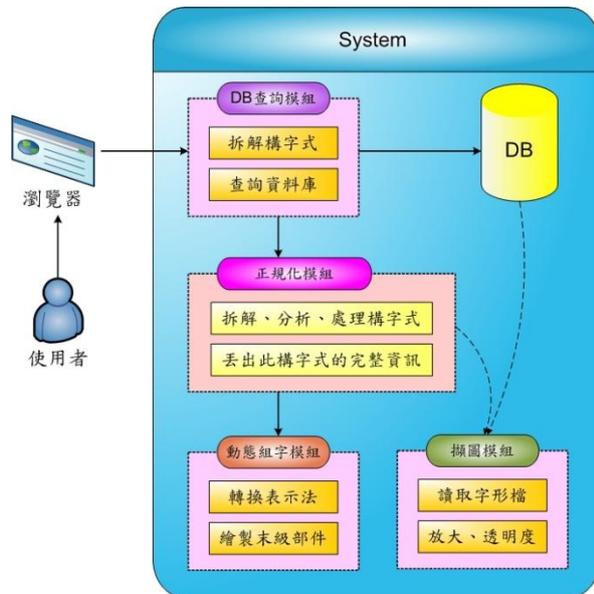


圖 4. 系統流程圖

由於漢字構形資料庫的編碼方式與可攜式造字引擎並不相同，漢字構形資料庫是以構字式、部件序列所表示，而可攜式造字引擎則是利用構形碼 Unicode + IDS 所組成，但在彼此仍有一定的相容性。以資料結構方面來看，有類似以下的區別：

- 漢字構形資料庫 → 中置式表示法
- 動態組字 → 前置式表示法

因此一特性的關係，在轉換結果中有少數較為複雜的字，會導致轉換後組合順序不相同，所以額外增加括號的功能，提供調整組合順序之用途。

轉換流程如圖 5，由於括號是調整判斷過程的順序，所以要先判斷是否有括號再將其分段，對於連接符號給定不同的優先權(Priority)，再逐一擷取每一個字符，利用堆疊(Stack)的方式來處理轉換流

程，經由堆疊判斷且轉換後再取出(Pop)的符號會與漢字結合成為轉換後的動態組字表示法(構形碼)，待每一個字皆判斷完畢則為完整的轉換後構形碼。圖 6 為文字「孔孟好學」轉換結果的例子。此轉換方式有幾點規則如下：

- 由後往前逐一讀取
- 漢字存入字串，符號另判斷。
- 取出之括號本身無需加入轉換後構字式
- 轉換後之字串由前面插入。

而連接符號所給定的優先權規則如下：

- 括號本身無優先權，但遇“左括號”時，則取出堆疊內的符號，直到取出“右括號”。
- 括號內的符號優先權高於括號前後的構字式
- 由後往前優先權逐漸變小
- 堆疊中只能以優先權高的符號，壓在優先權低的上面。若優先權低的壓在高的上面，則取出優先權高的符號。

以「孔孟好學」此字為例，我們將其原本的構字式中的漢字以  $W_x$  表示，符號則以  $C_x$  表示，作為範例說明，如表 6，即「 $W_1C_1(W_2C_2W_3C_3W_4)C_4W_5$ 」之構字式以「 $W_1C_1(W_2C_2W_3C_3W_4)C_4W_5$ 」表示，經由上述所提到的相關規則與流程轉換後，得到「 $C_1'W_1C_4'C_2'W_2C_3'W_3W_4W_5$ 」之表示法，即為「 $C_1'W_1C_4'C_2'W_2C_3'W_3W_4W_5$ 」，則為轉換後的正確結果。圖 7 為將轉換後表示法傳入動態畫字模組內所輸出的字形圖片。

另外，由構字式起始符號(形)與終止符號(□)所組成的特殊構字式，由於沒有完整的規則性，所以目前在以起始符號、終止符號所組成的構字式上，只支援部分的構字式組合。

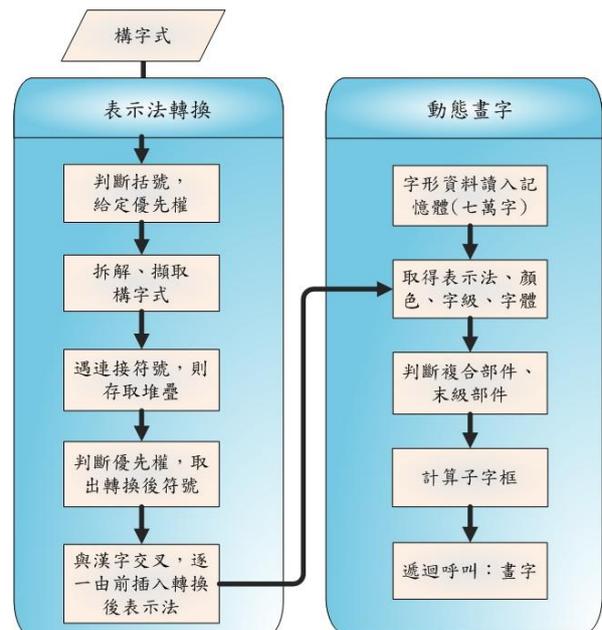


圖 5. 轉換流程圖



圖 6. 轉換釋例



圖 7. 「孔孟好學」輸出結果

表 6. 以堆疊方式之轉換過程

	讀入字元	堆疊變化	轉換後表示法	說明
1	W <sub>5</sub>		W <sub>5</sub>	讀入漢字 W <sub>5</sub> ，加入至轉換後表示法
2	C <sub>4</sub>	C <sub>4</sub>	W <sub>5</sub>	若讀入符號則存入堆疊
3	)	) C <sub>4</sub>	W <sub>5</sub>	讀入右括號存入堆疊中
4	W <sub>4</sub>	) C <sub>4</sub>	W <sub>4</sub> W <sub>5</sub>	讀入漢字 W <sub>4</sub> ，加入至轉換後表示法
5	C <sub>3</sub>	C <sub>3</sub> ) C <sub>4</sub>	W <sub>4</sub> W <sub>5</sub>	優先權 C <sub>3</sub> 大於 C <sub>4</sub> ，不需要處理堆疊內之符號
6	W <sub>3</sub>	C <sub>3</sub> ) C <sub>4</sub>	W <sub>3</sub> W <sub>4</sub> W <sub>5</sub>	讀入漢字 W <sub>3</sub> ，加入至轉換後表示法
7	C <sub>2</sub>	C <sub>2</sub> ) C <sub>4</sub>	C <sub>3</sub> 'W <sub>3</sub> W <sub>4</sub> W <sub>5</sub>	因為 C <sub>2</sub> 之優先權低於 C <sub>3</sub> 所以 C <sub>3</sub> pop 出去，經轉換後成為 C <sub>3</sub> '
8	W <sub>2</sub>	C <sub>2</sub> ) C <sub>4</sub>	W <sub>2</sub> C <sub>3</sub> 'W <sub>3</sub> W <sub>4</sub> W <sub>5</sub>	讀入漢字 W <sub>2</sub> ，加入至轉換後表示法
9	(	C <sub>4</sub>	C <sub>2</sub> 'W <sub>2</sub> C <sub>3</sub> 'W <sub>3</sub> W <sub>4</sub> W <sub>5</sub>	因遇左括號，所以到右括號之前的符號 C <sub>2</sub> 取出
10	C <sub>1</sub>	C <sub>1</sub>	C <sub>4</sub> 'C <sub>2</sub> 'W <sub>2</sub> C <sub>3</sub> 'W <sub>3</sub> W <sub>4</sub> W <sub>5</sub>	因為 C <sub>1</sub> 之優先權低於 C <sub>4</sub> 所以 C <sub>4</sub> 取出
11	W <sub>1</sub>	C <sub>1</sub>	W <sub>1</sub> C <sub>4</sub> 'C <sub>2</sub> 'W <sub>2</sub> C <sub>3</sub> 'W <sub>3</sub> W <sub>4</sub> W <sub>5</sub>	讀入漢字 W <sub>1</sub> ，加入至轉換後表示法
12			C <sub>1</sub> 'W <sub>1</sub> C <sub>4</sub> 'C <sub>2</sub> 'W <sub>2</sub> C <sub>3</sub> 'W <sub>3</sub> W <sub>4</sub> W <sub>5</sub>	構字式已經全部讀完，取出剩下的符號

### 3.2 相關探討

表 7. 各種缺字概況比較

	全字庫	漢字構形資料庫	可攜式造字引擎 (動態組字)	本研究
表示法	無特殊表示法	構字式、部件序列	構形碼 Unicode+IDS	構字式、IDS
字集 (編碼空間)	一個字形檔 8,836 字，可再擴充	一個字形檔 6,217 字，可再擴充	沒有限制	沒有限制
相容性	主要做為 Windows 環境之方案	主要做為 Windows 環境之方案	兼容各作業系統， 資料庫與平台無關。	兼容各作業系統
字體	楷體字形、 宋體字形	甲骨文、金文、小 篆、楷書、細明體。	黑體、細線體、圓 體，美觀上待改善。	甲骨文、金文、小 篆、楷書、動態組字
呈現方法	需安裝全字庫應用 程式，線上則以圖 片方式	需安裝漢字構形資 料庫，不支援資料 庫沒有的字形。	只需要有造字引擎 程式產生字形	可使用應用軟體方 式，或是直接使用字 形圖片，且支援資料 庫沒有的字形。
遇到新缺字 的解決情況	必須至窗口申請新 增字，流程複雜， 處理時間久。	以古籍為主，亦可 申請造字，處理時 間較全字庫快。	即時產生	直接以動態組字產 生，可以即時解決遇 到新缺字的問題。

雖然造字工程已經在各缺字服務單位執行多年，但缺字數量仍然日益增加，並且各單位所著重的方向不同，比如全字庫是以政府機關的缺字為主，漢字構形資料庫初期是以古籍為主，可攜式造字引擎是著重在直接產生字形，表 7 所示的為此三種解決方案與本研究架構在各項屬性的比較表，雖然動態組字在目前也許不是永久的解決方案，因為其產生出來的字形圖片在文字之美上仍稍嫌不足，但卻可以即時的提供解決方案，在使用者申請其他造字的同時，可以先利用整合後的系統來顯示缺字，而不是將缺字空著，造成他人無法了解文意。

#### 4. 應用實作

本系統 Server 端採用 Linux 作業系統，Web Server 採用 Apache Tomcat 4.1，動態組字核心程式採用 Java 所開發，改版為 Java Bean 版本便於線上使用。圖 8 為提供構字式轉換顯示圖片之頁面，此例為「招財進寶」利用構字式組合後再經由動態組字系統所繪出，另提供調整顯示字形圖片大小、字體型式、字形圖片格式、字體顏色之功能。圖 9 是利用所開發的 JavaScript API 去抓取網頁中的構字式，再將構字式傳送到漢字構形資料庫系統抓取圖片，圖片中的表格第一列為標準構字式所產生的字形圖片，此構字式在資料庫第一次查詢即可查詢到正確資料，第二列為經過正規化後才能找到資料的字形圖片，第三列為經判斷後資料庫確定無該字形，所以另外送到動態組字模組來分析，產生圖片後所傳回圖片格式的字形。

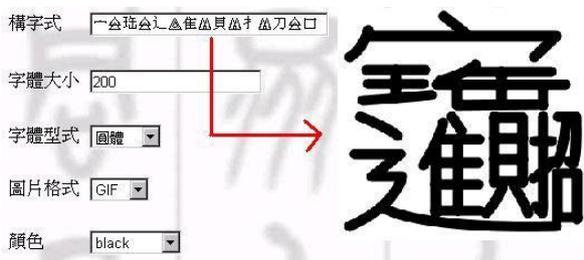


圖 8. 構字式轉換成動態組字畫面

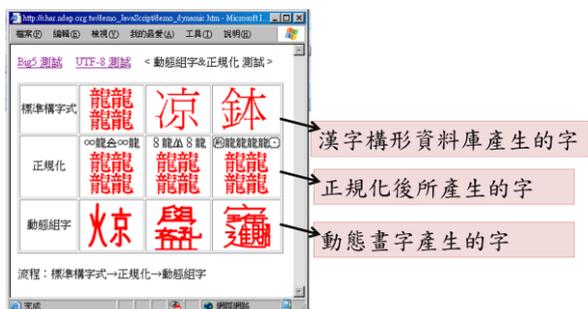


圖 9. 整合後之頁面

實作完成網頁如下：  
<http://char.ndap.org.tw/Help/system.htm>

#### 5. 結論與未來研究方向

現今漢字研究已經經過長期的時間了，而由於電腦時代的進步，所以漢字研究也必須跟進能在電腦上處理，若不能解決缺字在電腦上的顯示處理問題，則漢字的保存以及流傳上必定受到障礙。在本文所建構的漢字構形資料庫整合動態組字的方法，解決了以前礙於漢字構形資料庫字形尚未齊全的問題，更增加了實用性，使得尚未造字的字形得以利用動態組字模組來顯示缺字圖形，雖然動態組字目前的實作顯示圖片未能達到美觀的要求，但總比顯示成空白或是亂碼來得好。

我們也可以將這些構字式概念、動態組字成果應用於教學上，可以讓學生了解基礎的文字架構、文字由來、異體字等，而在網頁的顯示上，由於較之前更為健全，可便於大眾閱讀瀏覽，實用度上必定增加。

在未來發展上，可以將原本的漢字構形資料庫完整的線上化，讓使用者可以直接透過網際網路存取漢字構形資料庫。也可提供甲骨文檢索與辨識的功能，方便一般使用者在看到甲骨文的時候以圖形辨識的方式，可以即時得知其文字演變及意涵。在動態組字上，則可著重於各個部件字形彼此間的比例問題。目前也有其他有興趣在動態組字方面的人員在開發此方式的輸入法，便於利用在 3C 產品上。

#### 參考文獻

- [1] 全字庫, <http://www.cns11643.gov.tw/>
- [2] 剎那搜尋工坊, <http://www.ksana.tw/>
- [3] 政府機關中文資訊應用概況調查分析結果, <http://61.60.106.73/eng/reports.jsp>
- [4] 動態字形產生器之原理與實作, [http://www.eforth.com.tw/CT/yap\\_speak.htm](http://www.eforth.com.tw/CT/yap_speak.htm)
- [5] 動態組字, <http://zh.wikipedia.org/wiki/動態組字>
- [6] 動態組字參考實作, <http://rt.openfoundry.org/Foundry/Project/?Queue=690>
- [7] 莊德明, 謝清俊, “漢字構形資料庫的建置與應用”, 漢字與全球化國際學術研討會, 2005.1
- [8] 黃國倫, 蕭人豪, 李家豪, 陳心渝, “數位典藏系統缺字處理及應用” 第三屆數位典藏技術研討會
- [9] 漢字構形資料庫, <http://ckip.iis.sinica.edu.tw/CKIP/tool/>
- [10] 謝清俊, “電子古籍中的缺字問題”, 1996.08
- [11] Big-5, <http://61.60.106.73/web/big5/index.html>
- [12] Ideographic Description Characters, <http://unicode.org/charts/PDF/U2FF0.pdf>
- [13] Unicode, <http://www.unicode.org/versions/Unicode5.0.0/>