

(由本局填寫)

承辦人代碼：
大 類：
I P C 分類：

A6  
B6

本案已向：

美 國 ( 地 區 ) 申 請 專 利 ， 申 請 日 期 ： 1999.02.26 案 號 ： 09/258,687 ，  有  無 主 張 優 先 權

中國 優先權不予受理

有關微生物已寄存於：

， 寄存日期：

， 寄存號碼：

無

(請先閱讀背面之注意事項再填寫本頁各欄)

裝

訂

線

經濟部智慧財產局員工消費合作社印製

## 五、發明說明 ( 1 )

發明背景

發明之領域

本發明係關於一種基於代理人的系統，特別是關於一種自目標項目瀏覽資訊而自動獲取資訊的系統。

相關技術之描述

隨著與用戶互動作用的應用設計的出現，各種應用透過執行各種基於電腦的前景與背景模式的任務來實現，關於代理人的技術變得越來越重要。代理人軟體由電腦程式構成，這些電腦程式為用戶執行常規的、冗長的、費時的各種任務。為了更有利於用戶，代理人必須針對用戶個人的目的、習慣、和偏愛進行個性化設計。因此，為使代理人高效率、高效益的從用戶處獲得用戶的個人資料，並將其用於執行為用戶設計的任務，一些基本的要求必須考慮到。

代理人的概念，或者用戶代理人的概念，已經非常普及。代理人就是一個人被另一個人授權，典型的是指被委託人為該委託人辦事。根據這種方式，委託人授權代理人執行委託人不願或不能執行的任務。例如，保險代理人為委託人處理所有的保險需求，或者一個人才代理人為演奏者安排音樂會的時間。

隨著電腦的普及，一個新的代理人領域來到了。專家系統的突出優點使電腦程式能夠為用戶執行常規的、冗長的、和其它費時的任務。這些電腦程式被稱為"軟體代理人"。

## 五、發明說明 ( 2 )

另外，近來電腦和通信網絡激增。這些網絡允許用戶本質上不受任何地理限制而獲得大量的資訊和服務。因此，軟體代理人在為用戶完成大量的任務方面具有廣闊的應用前景。例如，代理人可以訂飛機票，購票以及將機票直接交到用戶手中。類似的，代理人也可以瀏覽網際網路，獲取從最新的體育節目或新聞到應用物理的一篇畢業論文。現有的解決方法無法將代理人技術應用於從目標文章中獲取資訊，並根據目標文章的資訊，應用通信網絡，指示用戶行動。

### 發明概要

從本發明較佳實施例的一個大的方面來說，一個系統協助從一篇感興趣的文章中獲取資訊，並根據目標文章的資訊，利用這個資訊接受用戶的直接行動。系統從包含會議時間的一個日曆程式中，以字符形式獲得象徵目標會議的輸入正文。當會議時間到來時，透過查詢日曆程式獲得目標事件的正文，並將該資訊用作代理人系統的輸入。然後，代理人系統分析輸入會議正文，摘取它的不同的元件諸如名稱、內容、與會者、地點、時間等等。系統還執行模式匹配以辨別特定的相應資訊。這一資訊用來查詢網路上不同的資訊源，獲得當前會議相關的說明，發回日曆程式。

### 較佳實施例之詳細說明

依照本發明之較佳實施例的系統非常適合用於個人電腦如IBM的個人電腦、蘋果電腦、或UNIX工作站的語境。典

## 五、發明說明 ( 3 )

型的硬體環境如圖1所示，該圖說明了較佳實施例的一個工作站的典型硬體配置，其中包含一個中央處理機110（如一個微處理機）、與一些由系統匯流排112連接的其它單元。圖1所示的該工作站包括一個隨機存取記憶體（RAM）114；唯讀記憶體（ROM）116；輸入／輸出配接器118將週邊設備如磁碟儲存單元120連接到匯流排112；用戶界面配接器122將鍵盤124、滑鼠126、揚聲器128、擴音器132、及／或其它用戶界面設備例如觸摸屏（未圖示）連接到匯流排112；通信配接器134將工作站接到通訊網絡（如一個資料處理網絡）；以及顯示配接器136連接匯流排112和顯示器138。工作站所採用的典型的操作系統如微軟視窗NT或視窗／95操作系統（OS）、IBM OS／2操作系統、MAC操作系統、或UNIX操作系統。熟悉本技藝的人士會瞭解本發明也可以用於以上沒有提到的其它平臺和操作系統。

較佳實施例使用JAVA、C、和C++語言並採用物件導向程式規劃（OOP）方法寫成。物件導向程式規劃在開發複雜應用程式方面用得越來越多。由於OOP已成為軟體設計和開發的主流，各種軟體解決方法都應該適應這一轉變，盡可能的利用OOP的優勢。有必要將OOP的原則應用於電子資訊傳遞系統的資訊傳遞界面，這樣可為資訊傳遞界面提供一批OOP的類與物件。

OOP是一個使用物件開發電腦軟體的過程，它由以下步驟構成：分析問題、設計系統、編寫程式。物件是包括資

## 五、發明說明( 4 )

料及相關的結構與過程集合的一個軟體包。由於它包括資料及結構與過程的集合，顯而易見，它是自滿足元件，不需要其它附加的結構、過程或資料來完成它內部的任務。因此，OOP將電腦程式看作稱為物件的高度自治的元件的一個集合，每個物件回應特定的任務。這種將資料、結構和過程組合到一個元件或模組當中的概念叫做封裝。

通常，OOP元件是可再度利用的軟體模組，它具有一個與物件模組的界面，並透過元件整合結構在程式運作時接受取用。元件整合結構是一系列的構造機制，它們允許軟體模組在不同的過程空間裡使用各自的權能或功能。這一般是透過假定一個公共元件物件模型並在其上設計這一結構來作到的。

這裡有必要區別一下物件和物件的類。物件是物件的類的一個單獨的事件，物件的類通常也叫類。物件的類可以認為是一個藍圖，許多物件從中產生。

OOP允許程式師創建一個物件，作為另一個物件的一部分。例如，代表活塞發動機的物件可以說與代表活塞的物件存在包含關係（composition-relationship）。實際上，一個活塞發動機包含一個活塞，閥門和許多其它元件；活塞是活塞發動機的一個元件這個事實可以在OOP中用兩個物件在邏輯上和語義上表示出來。

OOP還允許從一個物件當中衍生出一個物件。如果有兩個物件，一個代表一臺活塞發動機，另一個代表活塞是用陶瓷製成的活塞發動機，那麼這兩個物件的關係是包含關

## 五、發明說明 ( 5 )

係。陶瓷活塞發動機並不是活塞發動機的構成部分。不如說它只是一種比活塞發動機多了一項限制的活塞發動機，它的活塞是用陶瓷製成的。在此情況下，代表陶瓷活塞發動機的物件叫做衍生物件，它繼承了代表活塞發動機的所有特性，並做了進一步的限制或者說將它具體化了。代表陶瓷活塞發動機的物件"衍生"自代表活塞發動機的物件，這種物件之間的關係叫做繼承。

代表陶瓷活塞發動機的物件和類繼承了代表活塞發動機的所有特性，從而它也繼承了活塞發動機類中定義的標準活塞發動機的熱力特性。但是，陶瓷活塞發動機物件重寫了陶瓷活塞的特殊熱力特性，這與金屬活塞的特性明顯不同。它拋開原來的功能，使用與陶瓷活塞相關的新的功能。不同的活塞發動機有不同的特性，但是，它們可能具備相同的相應底層功能（如，發動機當中有多少活塞、點火次序、潤滑系統等等）。為了在任何一個活塞發動機物件當中取用這些功能，程式師應以相同的名稱調用相同的功能，但是，每種活塞發動機可能包含不同的／重新定義的具有相同名稱的功能實現。這種將不同的功能實現冠以相同名稱的能力叫做多態性，它極大的簡化了物件之間的通訊。

有了包含關係、封裝、繼承、和多態的概念，一個物件可以表示現實世界的任何事件。實際上，在物件導向的軟體中，我們對現實的理解是決定什麼類型的事情可以成為物件的唯一限制條件。一些典型目錄如下所示：

## 五、發明說明 ( 6 )

- 物件可以表示物體，如交通模擬中的汽車、電路設計程式中的電氣元件、經濟模型中的國家、或飛機交通控制系統中的飛機。
- 物件可以表示電腦用戶環境當中的一些元素，如視窗、項目單、或圖形物件。
- 物件可以表示一個目錄，如個人文件或城市緯度和經度的列表。
- 物件可以表示用戶定義的資料類型，如時間、角度、複數、或平面上的點。

由於物件所具有的可以表示任何邏輯上獨立的事件的這一強大的功能，OOP允許軟體開發者設計和實現一個電腦程式作為一些現實特徵的模型，而無論這些現實是一個物理實體、一個過程、一個系統、或是一個事件的組合。因為物件可以表示任何事情，軟體開發者可以創建一個物件作為將來更大的軟體設計的元件。

如果90%的新OOP軟體程式由經過驗證的、現存的從原有的可再度利用物件中產生的元件所構成，那麼僅有剩下的10%的新軟體設計需要編寫和測試。由於90%來自經過測試的可再度利用的物件，代碼錯誤的可能的範圍就被限制在10%的程式當中。這樣，OOP就使得軟體開發者能夠從其它現存的物件當中創建新的物件。

這個過程非常類似複雜的機械由元件和子元件構成。因此，OOP技術使得軟體工程非常類似硬體工程，軟體是從已存在的元件中設計起來的。這些元件可以作為物件供給

## 五、發明說明 ( 7 )

開發者使用。所有這些都提高了軟體的質量和開發速度。程式規劃語言正開始全面支援 OOP 的原則，如封裝性、繼承性、多態性、和包含關係。隨著 C++ 語言的出現，許多商業軟體開發者都採用 OOP 的方法。C++ 是一種提供了快速機器可執行代碼的 OOP 語言。此外，C++ 適合於商業應用和系統程式規劃設計的開發。現在，C++ 已成為許多 OOP 程式師的流行選擇，但是，還有許多其它的 OOP 語言，如 Smalltalk、公用資料處理物件系統 (CLOS)、和 Eiffel。另外，OOP 的功能還被用於一些常用的傳統程式規劃語言如 Pascal。

物件類的優點可以總結如下：

- 物件和它們相應的類把複雜的程式規劃問題分成許多小的、簡單的問題。
- 封裝性透過將資料組織成小的獨立的可相互通訊的物件，增強了資料的抽象性。封裝性保護資料不受意外破壞，但允許其它物件透過調用物件的功能和結構與資料互動作用。
- 子類和繼承透過從系統已有的標準類當中衍生出新的物件類型，使擴充物件和修改物件成為可能。這樣不需要改寫就可以創造出新的功能。
- 多態性和多重繼承使不同的程式規劃人員可以混合、匹配許多不同的類的特徵，創建出特定的物件，同時這些物件還可以與相關的物件以可預知的方式工作。
- 類的層次和容器層次提供了一個靈活的機制塑造現實世

## 五、發明說明 ( 8 )

界的物件和它們之間的關係。

- 可再度利用類庫在許多情況下都是很有用的，但是它們也存在一些局限。如：
- 複雜化。在一個有幾十、上百個類的複雜系統中，相關類的類的層次會變得異常混亂。
- 控制流。由類庫協助寫成的程式仍然負責控制流（如，它必須控制一個特定庫中產生的所有物件的互動作用）。程式師必須決定何時為何物件調用何種功能。
- 效果的複製。雖然類庫允許程式師使用和再度利用許多小的代碼塊，但是每個人以不同的方式組合這些代碼片段。兩個不同的程式師使用同樣一組類庫來編寫實現相同功能的兩個程式，它們的內部結構（如設計）可能會完全不同，這取決於每個程式師程式規劃過程中所作出的上百個決定。不可避免的，相似的代碼段以稍微不同的方式做相似的事情，它們配合起來就不會如它們應該達到的那麼好。

類庫是非常靈活的。由於程式變得越來越複雜，許多程式師被迫一次一次為基本問題構造基本的解決方法。類庫概念的一個相對新的擴展就是要有一個類庫的架構體系。這一架構體系更加複雜並且由類的龐大的集合構成。它可以獲取小模式和大多數機制兩種方式，來實現特定應用域的一般需求和設計。它們首先用於把程式師從瑣碎的工作中解放出來，這些工作包括顯示項目單、視窗、對話框、和其它個人電腦的標準用戶界面元素。

## 五、發明說明 ( 9 )

這個架構體系還代表了程式師對他們編寫的代碼和其他人編寫的代碼之間的互動作用的想法的轉變。在早些時候的過程化程式設計當中，程式師調用操作系統提供的庫來執行特定的任務，但是基本上是從程式頭執行到程式尾，而程式師自己負責控制流。這適合於列印支票、計算數學表格、或使用僅以一種方式工作的程式來解決其它一些問題。

圖形用戶界面的發展打破了結構化程式設計的限制。這些界面允許用戶而不是程式邏輯，來驅動程式並決定特定動作何時發生。今天，大多數個人電腦軟體透過事件循環來實現這一功能，這一事件循環用來監視滑鼠、鍵盤、和其它外部事件，並且根據用戶平臺的行為來調用程式代碼相應的部分。程式師不再需要決定事件發生的順序。相反，程式被分成獨立的片段並在不可預知的時間以不可預知的順序調用。由於透過這種方式將控制權下放給了用戶，開發人員就創建了更加易用的程式。雖然如此，但是程式師寫的單獨的程式塊仍然要調用操作系統提供的庫來完成特定的任務，程式師在事件循環調用之後還要決定每塊程式的控制流。應用程式代碼仍然"設計在系統頂端"。

事件循環程式甚至要求程式師編寫大量代碼，對於每個應用這些代碼不需要單獨編寫。應用架構體系的概念進一步加深了事件循環的概念。程式師不再需要處理在構造基本的項目單、視窗、對話框、以及將這些要素連接起來配合運作時所遇到的所有細節問題。程式師使用應用架構體

## 五、發明說明 ( 10 )

系直接從編寫應用代碼和基本用戶界面元素開始。接著，他們根據具體應用需要，將架構體系的通用權能換成具體應用的特定的權能。

應用架構體系減少了程式師需要改寫的大量的程式代碼。但是，由於架構體系實際上是一個用來顯示視窗、支援複製和黏貼等通用的應用的，程式師也可以在比事件循環程式所允許的更大程度上下放控制權。架構體系代碼幾乎完成了所有的事件處理和控制流，程式師的代碼僅在架構體系需要時才被調用（例如，創建和處理一個自定義資料結構）。

程式師在編寫一個架構體系程式時，不僅將控制權下放給用戶（這對於事件循環程式也是如此），而且把程式具體的控制流交給了架構體系。使用這種方法可以創建一些更為複雜的系統，它們以非常有趣的方式一起運作。而包含用戶代碼的孤立程式為了解決類似的問題要一次又一次的反覆編寫。

因此，如上所述，架構體系基本上是一個相互配合的類的集合，它們構成一個給定問題域可再度利用的設計解決方法。典型的，它包括一些提供預設值行為的物件（如項目單和視窗），程式師透過繼承一些預設值行為並重寫其它一些行為來使用它，這樣，架構體系就可以在相應的時間調用相應的程式代碼。

架構體系與類庫有三個主要的區別：

- 行為對應規約。類庫是一些行為的基本的集合，當你在

## 五、發明說明 ( 11 )

你的程式中需要那些單獨的行為時可以進行調用。另一方面，架構體系不僅提供了行為，還提供了規約或決定行為配合方式的一系列的規定，這包含程式師對應架構體系所提供者而應提供者的規則。

- 調用對應覆蓋(versus override)。利用類庫、代碼，程式師將物件實例化，並調用其成員函數。利用架構體系，也可以相同方法實例化和調用物件（如：將架構體系作為一個類庫），但是，為了最大程度上利用架構體系的可再度利用設計，程式師一般用覆蓋的方法編寫代碼並由架構體系調用。架構體系在它的物件之間安排控制流。編寫由架構體系調用的單獨的不同的程式塊，要勝過程式師自己決定不同程式塊如何相互配合。
- 實現對應設計。使用類庫，程式師僅能再度利用工具，而利用架構體系，他們可以再度利用設計。架構體系賦予一族相關程式或程式塊工作的方式。它代表了一個通用設計方法，這一方法可用於給定域的不同的問題。例如，單獨一個架構體系就可以將用戶界面的工作方法實例化，即使由同一個架構體系創建的兩個不同的用戶界面也能解決完全不同的界面問題。

因此，透過對不同問題和程式任務的架構體系的開發，軟體設計和開發的難度將得到極大的降低。該發明的較佳實施例使用超文字標記語言（HTML）和用於客戶機和主機（Newco）之間的傳遞媒體的安全通信規約將文件檔放在網際網路上。HTTP或其它規約可以很好的用於HTML而沒有

## 五、發明說明 ( 12 )

不適應經驗。這些產品的資訊可以從以下內容獲得：T. Berners-Lee, D. Connolly, "RFC 1866：超文字標記語言-2.0" (1995年11月)；以及R. Fielding, H. Frystyk, T. Berners-Lee, J. Gettys和J.C. Mogul, "超文字傳輸公約—HTTP/1.1：HTTP工作組網際網路草案" (1996年5月2日)。HTML是一種簡單的資料格式，可用來創建跨平臺移植的超文字文件檔。HTML文件檔就是一般語義的SGML文件檔，適用於表示非常廣闊的領域中的資訊。HTML最初從1990年用於全球資訊網的全球資訊。HTML是ISO8890標準的應用：1986資訊處理語言和辦公系統；標準通用標記語言(SGML)。

到現在，網路開發工具在創建跨越客戶機和伺服器以及與現有的電腦資源互動作用的動態網路應用時仍然受到它們能力的限制。直到現在，HTML在開發基於網際網路的解決方法時，仍然是占主導地位的技術。但是，可以證明HTML在以下領域存在不足：

- 性能較差；
- 對用戶互動能力的限制；
- 僅能創建靜態網頁；
- 與現有應用和資料缺少互動作用；並且
- 不能擴展。

Sun Microsystem的Java語言透過以下方法可以解決許多客戶端問題：

- 提高客戶端運作性能；

## 五、發明說明 ( 13 )

- 使創建動態的即時網路應用成為可能；並且
- 提供創建各種用戶界面元件的能力。

使用Java，開發者可以創建健全的用戶界面（UI）元件。創建用戶的"小元件"（如即時股票接受器、動態圖標等等），並且可以提高客戶端的運作性能。不像HTML，Java支援客戶端確認的概念，它可以向客戶機下載相應的處理過程以提高運作性能。可以創建動態的即時網頁。同時，使用上面提到的用戶UI元件也可以創建動態網頁。

Sun的Java語言是作為"網際網路程式規劃"的工業標準語言而出現的。Sun是這樣定義Java的："簡單的、物件導向的、分佈式的、解釋性的、健全的、安全的、中性結構的、可移植的、高性能的、多線程的、動態的、容錯的、通用的程式規劃語言，Java支援使用獨立於平臺的Java小應用程式進行網際網路程式規劃"。Java小應用程式是符合Sun的Java應用程式規劃界面（API）的小的特定的應用，它允許開發者在網際網路文件檔中加入"互動內容"（例如簡單的動畫、頁面裝飾、基本的遊戲等等）。小應用程式透過將代碼從伺服器複製到客戶機上，在Java兼容的瀏覽器（如網景的Navigator）中執行。從語言的立場來看，Java的核心特性設計在C++的基礎上。Sun的Java語言宣稱Java基本上是"C++的，並包含適用於更多動態方法解決方法的Objective C的擴展"。

另一項提供類似Java功能的技術是由微軟和ActiveX技術提供的，它供給開發者和網頁設計者所有創建網際網路和

## 五、發明說明 ( 14 )

個人電腦的動態內容的所有工具。ActiveX包括用於開發動畫、三維虛擬實境、影像、和其它多媒體內容的工具。這些工具使用網際網路標準，跨平臺工作，並且被100家公司所支援。這些構件群叫做ActiveX控制元件，它們是一些小的、快速的元件，可以使開發者將軟體的一些部分記錄超文字標記語言（HTML）的頁面中。ActiveX控制元件可以與眾多程式規劃語言共同使用，這些語言包括微軟Visual C++、Borland Delphi、微軟Visual Basic等程式規劃系統，以及將來微軟的Java開發工具"Jakarta"。ActiveX技術還包括ActiveX伺服器架構體系，允許開發者創建伺服器應用。根據該技術的一個一般技巧，可以輕鬆的用ActiveX替代Java來實現本發明而不需不適應實驗。

依照較佳實施例，背景搜索器（BF）用作負責個人會議準備工作的代理人，它幫助他／她從各種資訊源當中檢索會議的相關資訊。BF以字符形式接受指示目標會議的輸入文字。輸入文字依照較佳實施例透過包含會議時間的日曆程式產生。當會議時間接近時，查詢日曆程式獲得目標事件的文字，此資訊作為輸入傳給代理人。然後，代理人分析輸入的會議文字，摘錄會議的各種要素，如名稱、內容、與會者、地點、時間等等。系統還會執行模式匹配來辨認會議文字的特定會議域。這個資訊用來查詢網際網路上的各種資訊源並獲得當前會議的相關內容，將其傳回日曆系統。例如，如果有人得悉網景與微軟有一個會議討論他們的糾紛，想要從日曆系統得到這一內部消息。它就透

## 五、發明說明 ( 15 )

過分析文字的語義得知參加會議的公司是"網景"與"微軟"主題是"糾紛"。然後，系統查詢網路與主題相關的資訊。於是，根據本發明的一個目的，系統更新日曆系統，並且用戶最終得到了它為用戶搜集的目標會議的最好的資訊。依照較佳實施例，資訊儲存在一個文件中，它可以透過選擇一個記錄日曆程式的鏈接得到。

### 程式組織

依照較佳實施例設計的電腦程式分成五個獨立的模組：BF.Main、BF.Parse、Background Finder.Error、BF.PatternMatching、和BF.Search。還有一個frmMain提供了一個僅用於除錯目的的用戶界面。較佳實施例的可執行程式執行時永遠沒有用戶界面，並且僅能透過微軟的Winsock控制返回日曆程式。系統的較佳實施例以兩種不同的方式執行，這可以由日曆程式發給他的命令行來定義。當系統以簡單模式運作時，它由外部搜索引擎執行關鍵詞查詢。當它以複雜模式執行時，系統在它形成發往搜索引擎的查詢時，首先執行模式匹配。

### 資料結構

依照較佳實施例的系統使用了三個用戶自定義結構：

1. TMeetingRecord；
2. TPatternElement；和
3. TPatternRecord。

用戶自定義結構tMeetingRecord用來儲存一個會議的所有相關資訊。這些資訊包括用戶ID、對會議的一個初始說

## 五、發明說明 ( 16 )

明、會議名稱的關鍵詞的摘要列表、以及會議內容等等。必須注意，依照較佳實施例系統的每一個實例僅創建一個會議記錄。因此，會議記錄根據當前檢索到的會議創建。ParseMeetingText加入這一會議記錄並將會議的有關資訊提供給其它功能。

如果GoPatternMatch能夠對特定會議域賦值，會議記錄相應的入口也會更新。依照較佳實施例，以下圓括號中給出了tMeetingRecord結構中每一個域的說明。

A.1.1.1.1.1	公共類型 ( Public Type )
	tMeetingRecord
sUserID As String	( Munin 給出的用戶 ID )
sTitleOrig As String	( 我們需要保留以發回到 Munin 的可用詞表的初始會議名稱 )
sTitleKW As String	( 僅有關鍵詞的禁用詞表的會議名稱 )
sBodyKW As String	( 僅有關鍵詞的禁用詞表的會議內容 )
sCompany() As String	( 透過模式匹配從名稱和內容中識別的公司名 )
sTopic() As String	( 透過模式匹配從名稱和內容中識別的會議主題 )
sPeople() As String	( 透過模式匹配從名稱和內容中識別的與會人名 )
sWhen() As String	( 透過模式匹配從名稱和內容中

## 五、發明說明 ( 17 )

識別的會議時間)

sWhere() As String (透過模式匹配從名稱和內容中  
識別的會議地點)

sLocation As String (Munin的會議地點)

sTime As String (Munin的會議時間)

sParticipants() As String (Munin的會議參與者)

sMeetingText As String (無用戶 ID 的初始會議正文)

End Type

還有兩個其它的結構表示用於模式匹配的單獨模式。tAPatternRecord記錄是一個包含一個模式的所有元件／元素的數組。tAPatternElement類型是一個包含代表一個模式中每個元素的字符串的數組。由於每一個元素都會有許多"代用品"，我們需要一個字符串數組跟蹤這些代用品。依照較佳實施例，tAPatternElement和tAPatternRecord類型表示如下。

Public Type tAPatternElement

elementArray() As String

End Type

Public Type tAPatternRecord

patternArray() As tAPatternElement

End Type

#### 公用用戶定義常數

依照較佳實施例，每個程式的說明部分定義的許多常數在系統運作過程當中需定期更新。這些常數是可存取的，

## 五、發明說明 ( 18 )

這樣就允許系統透過動態配置隨時更新以維護代碼。

以下表格包含了每個模組中我認為需要隨時更新的常數列表。儘管如此，仍然有一些程式中的常數未包括在下面的列表中。這並不意味著沒有包含在內的常數就不會更改。而只是說它們更改得相對不太頻繁。

主程式塊 (BF.Main)：

常數	當前值	用途
MSGTOMUNIN_TYPE	6	定義用於識別 BF 和 Munin 之間的消息的消息數量
IP_ADDRESS_MUNIN	"10.2.100.48"	定義 Munin 和 BF 運作的機器 IP 地址以便它們透過 UDP 傳遞資料
PORT_MUNIN	7777	定義我們運作的遠方端口
TIMEOUT_AV	60	定義 inet 控制中設置時間結束的常數
TIMEOUT_NP	60	定義 inet 控制中設置時間結束的常數
CMD_SEPARATOR	"\"	定義區分 Munin 命令的哪一部分代表我們輸入的會議正文的定義符
OUTPARAM_SEPARATOR	"::"	定義區分輸出的不同部分的定義符。這一分割符用來定義消息類型、用戶 ID、會議名稱、以及檢索到的實際內容的起始處。

搜索模組 (BF.Search)

常數	當前值	用途
PAST_NDAYS	5	定義你希望回過來看 AltaVista 報告的天數。現在這並不要緊因為我們並不真的在 alta vista 中進行新聞搜索。我們需要所有的資訊。
CONNECTOR_AV_URL	"+AND+"	定義如何連接關鍵詞。我們要求所有的關鍵詞出現在字符串中，所以使用 AND。如果需要 OR 或別的什麼，僅需改變連接符。
CONNECTOR_NP_URL	"+AND+"	定義如何連接關鍵詞。我們要求所有的關鍵詞出現在字符串中，所以使用 AND。如果需要 OR 或別的什麼，僅需改變連接符。

## 五、發明說明 ( 19 )

NUM_NP_STORIES	3	定義從 NewsPage 發回到 Munin 的內容的數量
NUM_AV_STORIES	3	定義從 NewsPage 發回到 Munin 的內容的數量

語法分析模組 ( BF.Parse ) :

常數	當前值	用途
PORTION_SEPARATOR	":"	定義 Munin 中的會議正文的不同部分的分隔符。例如，"09::Meet with Chad::about life::Chad   Denise:::"，":"是會議正文不同部分的分隔符。
PARTICIPANT_SEPARATOR	" "	定義初始會議正文的與會者列表部分中的每一個與會者之間的分隔符。參考以上例子。

模式匹配模組 ( BFPatternMatch ) : 這一模組沒有需要頻繁更新的常數。

### 通用流程

說明流程和功能協調的最好方式就是從圖2到圖6的五張流程圖。圖2說明了較佳實施例的整體流程。整個過程從圖的頂端功能塊200開始，該功能塊在程式開始時運作。依照較佳實施例，當應用開始時，如功能塊210所示，命令行透過語法分析刪除對應的會議正文以開始背景查詢操作。如功能塊220所示，當目標已經決定後，一個全局禁用詞表就產生了。接著，如功能塊230所示，用於模式匹配操作的所有模式就產生了。然後，根據圖表，功能塊200進入GoBF功能塊240，該模組負責打包 ( wrap ) 根據特定的目標搜索引擎得到的正確的搜索查詢 ( query ) 資訊的邏輯處理過程。例如，從功能塊240到功能塊250，接著它調用功能塊260所示的GoPatternMatch，要看GoPatternMatch的流程，我們轉到

## 五、發明說明 ( 20 )

名為"BF模式匹配單元的流程"的圖式。

需要注意的一個關鍵事情是圖中以同樣水準說明的功能塊以從左到右(從頂部到底部)的次序由它們的父函數依次調用。例如,主程式200調用功能塊ProcessCommandLine 210,接著CreateStopListist 220、CreatePatterns 230、和GoBackgroundFinder 240。圖3到6說明了整個程式的詳細邏輯、語法分析單元、模式匹配單元、和搜索單元。圖6透過BackgroundFinder詳細說明了主要資訊的資料流的邏輯判斷過程,並畫出負責創建或處理這一資訊的功能塊。

### **詳細搜索                      簡單查詢模式下的構造**

#### **ALTA VISTA 搜索**

(圖2所示功能塊270)

如圖2的功能塊270所示,Alta Vista 搜索引擎識別並返回當前會議的主題相關的一般資訊。較佳實施例的系統從初始會議正文中的名稱部分獲取所有的關鍵詞,構造一個高級查詢查詢並將其發給Alta Vista。關鍵詞按照一定邏輯組合在查詢中。查詢結果也根據同樣的一組關鍵詞進行排列。根據這項技術的一個一般技巧,會很容易理解資料約束或出版商標準可能會方便我們檢索物品。一組最高優先級的檢索內容被送回較佳實施例的日曆系統。

#### **新聞頁 (NewsPage)**

(圖2所示功能塊275)

NewsPage搜索系統負責為我們提供有關目標會議的最近新聞主題。NewsPage搜索系統從初始會議正文中的名稱部

## 五、發明說明 ( 21 )

分獲取所有的關鍵詞，構造一個查詢並將其發給 NewsPage。關鍵詞按照一定邏輯組合在查詢中。僅有最近發表的文章才會檢索到。NewsPage 搜索系統提供了一個可由用戶根據其偏愛設置的資料約束標準。最高優先級的檢索內容被送回日曆系統。

圖 3 是較佳實施例的用戶特徵資料模型。過程從功能塊 300 開始，這一模組負責從主模組調用該程式。接著，在功能塊 310，調用打包功能為功能塊 320 的關鍵詞摘要過程做準備。關鍵詞被摘取之後，處理過程進入功能塊 330 以決定分隔符是否正確放置。接著，在功能塊 340，特定字符串中的詞被計算出來，這樣，在功能塊 350 中由分隔符劃分的會議正文的特定域就被檢索出來。然後，在功能塊 380，再一次檢查字符串的分隔符以確保它們正確放置。最後，在功能塊 360，執行資訊的標題和主體的每個詞的摘取，應用功能塊 362 的邏輯發現輸入詞組的下一個最近的單字分隔符，功能塊 364 跳過詞中不必要的內容，功能塊 366 決定是否有單字在禁用詞表中，如有則返回錯誤資訊。

### 較佳實施例的模式匹配

簡單搜索方法的局限性包括以下內容：

1. 由於它依靠不可用詞的禁用詞表來從會議正文當中摘取一系列關鍵詞。這樣，在如何理解禁用詞表的問題上就存在局限性。我們應該把注意力放在會議正文中我們想要的部分，而不是我們要放棄的部分。
2. 依照較佳實施例的簡單搜索方法僅用會議名稱的關鍵詞

## 五、發明說明 ( 22 )

形成查詢並送給 Alta Vista 和 NewsPage。會議體注意到，一個用於查詢的可替換資訊源被忽略了。我們形成查詢時不能包含會議體的關鍵詞，因為這經常導致查詢查詢太長，太複雜以至於不能得到有意義的結果。

3. 我們無法說出每個關鍵詞代表什麼。例如，我們可能摘取 "Andy" 和 "Grove" 作為關鍵詞。但是一個簡單搜索無法得知 "Andy Grove" 實際上是一個人的名字。想像一下我們能夠猜測出 "Andy Grove" 是人名的可能性。我們可以發現他是否是 Andersen 人，以及他是否以前曾作過什麼設計等等。
4. 總而言之，僅僅依賴禁用詞表來分析出不必要的詞，我們就會導致 "資訊過載"。

### 克服局限的較佳實施例的模式匹配

這裡是依照較佳實施例，模式匹配系統如何解決以上相關的每個問題的方法。

1. 在模式匹配時，我們僅匹配想要的部分，並摘出那些部分。
2. 透過執行對會議體的模式匹配並僅摘取我們想要的會議體部分，會議體（會議內容）就不是完全沒用的。
3. 模式匹配設計在我們定義的一系列樣板上，允許我們從會議正文當中分辨人名、公司名等等。
4. 總之，有了模式匹配，我們不再被資訊過載困擾。當然，我們的模式匹配的效果如何是一個大問題。如果我們僅依賴於人工智慧處理，我們將無法作到 100% 的命中

## 五、發明說明 ( 23 )

率。我們能夠識別出提供給我們的所有公司的 20% 公司名字。

### 模式

較佳實施例語境中的模式就是定義我們從會議正文當中所要尋找的短語結構的樣板。較佳實施例所支援的這些模式被選擇出來，是因為它們是一些在會議正文中極有可能出現的短語的樣板。例如，當需要在日程表中鍵入一個會議時，許多人可能會輸入諸如"下星期二與Stanford大學的Bob Dutton的會議(Meet with Bob Dutton from Stanford University next Tuesday)"的東西。一個常用的樣板可能就是"with"後面跟著人名(本例中是Bob Dutton)"from"跟著一個機構的名字(本例中是Stanford大學)。

### 模式匹配術語

模式匹配的常用術語：

- ◆ 模式 (Pattern)：模式就是定義我們想要以會議正文賦值的短語結構的樣板。它包含子單元。
- ◆ 元素 (Element)：一個模式包含許多子單元。這些子單元叫做元素。例如，在模式 "with \$PEOPLES\$ from \$COMPANY\$" 中，"with"、"\$PEOPLES\$"、"from"、"\$COMPANY\$" 都是元素。
- ◆ 位置標誌符 (Placeholder)：位置標誌符是一種特殊的元素，我們可以對它賦值。如上例中 "\$PEOPLES\$" 就是位置標誌符。
- ◆ 指示器 (Indicator)：指示器是另外一種元素，是我們要

## 五、發明說明 ( 24 )

從會議正文中發現的，但是不需要對其賦值。在一個特定模式中，通常會存在不止一個指示器。這就是為什麼指示器不是一個"基元"類型。

- ◆代用品 (substitutes)：代用品是一系列的指示器，它們彼此同義。從輸入當中發現它們任何一個都行。

每個會議都包含五個域：

- ◆公司 (\$COMPANY\$)
- ◆人員 (\$PEOPLE\$)
- ◆地點 (\$LOCATION\$)
- ◆時間 (\$TIME\$)
- ◆主題 (\$TOPIC\_UPPER\$) 或 (\$TOPIC\_ALL\$)

圓括號中是我的代碼中用來代表相應會議域的位置標誌符。

每個位置標誌符有以下含義：

- ◆\$COMPANY\$：以一個以大寫字母開始的字符串賦值（如：Meet with Joe Carter of <Andersen Consulting >）
- ◆\$PEOPLE\$：以兩組大寫字母開頭的字符串賦值，其間可能以"." "and" 或 "&" 連接（如 Meet with <Joe Carter> of Andersen Consulting, Meet with <Joe Carter and Luke Hughes> of Andersen Consulting）
- ◆\$LOCATION\$：以一個大寫字母開頭的字符串賦值（如 Meet Susan at <帕羅奧多市 Square>）
- ◆\$TIME\$：以一個#:## 格式的字符串賦值（Dinner at <6:30 pm>）

## 五、發明說明 ( 25 )

◆ \$TOPIC\_UPPER\$ : 為我們的主題賦以一個大寫字母開頭的字符串 ( 如 <Stanford Engineering Recruiting> 關於新的招聘的會議 )

◆ \$TOPIC\_ALL\$ : 以一個字符串賦值而不關心是否以大寫字母開始 ( 如 Meet to talk about <ubiquitous computing> )

這裡是表示BF支援的所有的模式的表格。每一個模式屬於一個模式組。一個模式組當中的所有模式有類似的形式，它們只是在使用何種指示器作為代用品的問題上有所區別。注意塗灰的這些模式在代碼中也做了相應的注釋。BF具有支援這些模式的權能，但是在這一點上，匹配這些模式並不是基本要求。

模式組別	模式號 #	模式	舉例
1	a	\$PEOPLES of \$COMPANY\$	Paul Maritz of Microsoft
	B	\$PEOPLES\$ from \$COMPANY\$	Bill Gates, Paul Allen and Paul Maritz from Microsoft
2	a	\$TOPIC_UPPER\$ meeting	Push Technology Meeting
	b	\$TOPIC_UPPER\$ mtg	Push Technology Mtg
	c	\$TOPIC_UPPER\$ demo	Push Technology demo
	d	\$TOPIC_UPPER\$ interview	Push Technology interview
	e	\$TOPIC_UPPER\$ presentation	Push Technology presentation
	f	\$TOPIC_UPPER\$ visit	Push Technology visit
	g	\$TOPIC_UPPER\$ briefing	Push Technology briefing
	h	\$TOPIC_UPPER\$	discussion Push Technology discussion
	i	\$TOPIC_UPPER\$ workshop	Push Technology workshop
	j	\$TOPIC_UPPER\$ prep	Push Technology prep
	k	\$TOPIC_UPPER\$ review	Push Technology review
3	l	\$TOPIC_UPPER\$ lunch	Push Technology lunch
	m	\$TOPIC_UPPER\$ project	Push Technology project
	n	\$TOPIC_UPPER\$ projects	Push Technology projects
	a	\$COMPANY\$ corporation	Intel Corporation
	b	\$COMPANY\$ corp.	IBM Corp

## 五、發明說明 ( 26 )

	c	\$COMPANY\$ systems	Cisco Systems
	d	\$COMPANY\$ limited	IBM limited
	e	\$COMPANY\$ ltd	IBM ltd
4	a	about \$TOPIC_ALL\$	About 智能代理人技術
	b	discuss \$TOPIC_ALL\$	Discuss 智能代理人技術
	c	show \$TOPIC_ALL\$	Show 客戶智能代理人技術
	d	re: \$TOPIC_ALL\$	re: 智能代理人技術
	e	review \$TOPIC_ALL\$	Review 智能代理人技術
	f	議程	議程如下所示： --完成 --完成 --完成
	g	議程: \$TOPIC_ALL\$	議程： --demo 客戶智能代理人技術 --demo ecommerce.
5	a	w/\$PEOPLES\$ of \$COMPANY\$	約會 w/Joe Carter of Andersen Consulting
	b	w/\$PEOPLES\$ from \$COMPANY\$	約會 w/Joe Carter from Andersen Consulting
6	a	w/\$COMPANY\$ per \$PEOPLES\$	會談 w/Intel per Jason Foster
7	a	At \$TIMES\$	at 3:00pm
	b	Around \$TIMES\$	Around 3:00 pm
8	a	At \$LOCATIONS\$	At LuLu's 餐廳
	b	In \$LOCATIONS\$	in Santa Clara
9	a	Per \$PEOPLES\$	per Susan Butler
10	a	call w/\$PEOPLES\$	Conf call w/John Smith
	B	call with \$PEOPLES\$	Conf call with John Smith
11	A	prep for \$TOPIC_ALL\$	Prep for London 會議
	B	preparation for \$TOPIC_ALL\$	Preparation for London 會議

圖4是較佳實施例的詳細的模式匹配流程圖。處理過程從功能塊400開始，其中，主程式調用模式匹配應用並將控制權交給功能塊410以開始模式匹配處理。接著，在功能塊420，打包功能開始循環處理每一個模式，這包括決定正文字符串是否符合一個模式（如功能塊430所示）。然後，在功能塊440，不同的位置標識符被賦值（如果它們存在），在功能塊441，一系列由標點符號分開的名字被賦值，在功能

## 五、發明說明 ( 27 )

塊442，透過發現兩個大寫字母開始的單字作為一個完整的名字，並取得單字後的一個空格後面的字符，判斷其是否大寫來處理一個完整的名字。接著，在功能塊443，透過適當的方式從字符串中分析出時間，在功能塊444中分析出空格後的下一個單字。然後，在功能塊445，接下來的大寫詞組如公司、主題、和地點被提取出來，在功能塊446，依照較佳實施例，空格後的下一個單字被獲取以便進一步處理。在模式匹配處理過程後，用功能塊450定位一個代表模式頭部的指示器，如功能塊452所示，空格後的下一個單字被獲取並檢查以決定該詞是否是一個指示器（如功能塊454所示）。接著，在功能塊460，對字符串進行分析以放置指示器，並確保指示器不在模式的結尾處。不必要的空格如換行或回車後的下一個單字在功能塊462中處理。功能塊464分析這個單字以決定它是否一個指示器。然後，在功能塊470，暫時記錄被重置為空，以備處理下一個字符串，在功能塊480當中，會議記錄被更新，在功能塊482當中，在進行下一個會議記錄之前，檢查程式入口是否已經準備就緒。

### 使用可識別會議域

既然我們已經識別出我們認為重要的會議正文中的域，我們就可以利用它完成很多的事情。模式匹配的一個最重要的應用當然就是我們構造的交給Alta Vista和News Page的查詢。還有許多利用模式匹配的結果的選擇和增強功能可用於背景查詢（BF）。這些其它選擇我們在下一節中敘

## 五、發明說明 ( 28 )

述。本節的目的是幫助讀者理解模式匹配的結果如何幫助我們獲得更好的查詢結果。

圖5是較佳實施例中準備查詢並從網際網路中獲得資訊的具體過程。過程從功能塊500開始，馬上轉入功能塊510處理打包功能，為使用網路搜索引擎的網際網路搜索做準備。如果這個搜索使用Alta Vista搜索引擎，那麼在功能塊530系統從會議記錄中取得資訊，在功能塊540到560中形成一個查詢並交給搜索引擎。如果這個搜索應用NewsPage搜索引擎，那麼在功能塊520系統從會議記錄中取得資訊，在功能塊521到528中形成一個查詢並交給搜索引擎。

### Alta Vista 搜索引擎

Alta Vista搜索引擎的優勢是它提供了強大的靈活性。個人可以構造所有類型的布爾（Boolean）查詢並且根據需要決定優先級。但是，它的一個最大的缺點是它在處理一個大的查詢時並不很好，而且好像會給出不相關的結果。如果我們能夠在會議正文中識別主題和公司，我們就可以形成一個短而易於理解的查詢，從而有希望得出較好的結果。我們也要注意所找到的主題。對於用戶來說，找到一個公司的資訊也許沒有太大的價值，特別是已經非常了解該公司並與它們有過大量的會議。他們要搜索的就是這些主題。

### News Page 搜索引擎

News Page搜索引擎的優勢是如果你能給出一個合法的公司名，它就可以在搜索最近新聞方面做很多工作。因此，

## 五、發明說明 ( 29 )

當我們向網站的新聞頁 (Newspage) 提交一個查詢時，我們發出我們可以識別的公司名稱，並且僅有當我們無法找到一個公司名稱時，我們才使用會議主題形成查詢。用來形成查詢並提交 Alta Vista 的運算法如圖 7 所示。用來形成查詢並提交 News Page 的運算法如圖 8 所示。

以下列表詳細說明依照較佳實施例的每一個功能。功能出現的次序與流程盡可能的相仿。當存在一個功能被多次調用的情況時，這一功能會排在首次調用它的功能之後，對它的說明在以後對它的每次調用中不再重複敘述。

過程名	類型	被調用過程	說明
Main (BF.Main)	Public Sub	無	這是程式首先開始運作的主程式。它以適當的參數對 BF 開始 (如網際網路超時、禁用詞表...) 並調用 GoBF 運作程式的主要部分
ProcessCommandLine (BF.Main)	Private Sub	Main	這一功能分析命令行。它假設指示 Munin 輸入開始的分隔符儲存在常數 CMD_SEPARATOR 中。
CreateStopList (BF.Main)	Private Function	Main	這一功能為將來從會議正文中分析出非用單字設計禁用詞表。每個單字的兩邊有引號以便直接檢查。
CreatePatterns (BF.PatternMatch)	Public Sub	Main	這一過程的調用首次出現在 BF 進行開始，創建所有以會議正文的部分內容賦值的可能模式時。一個模式可包含它所需要任意多的元素。有兩種元素，頭一種元素是指示器。有實際的單字用來分隔後面跟隨的可能的會議域 (如公司)。許多指示器是非用 (stop) 單字，因為非用單字對所有會議正文都是共同的，所以用來構成模式是有意義的。第二種元素是代表位置標誌符的特殊字符串。位置標誌符總是以 \$*\$ 的方式出現，其中 * 可以是 PEOPLE、COMPANY、TOPIC_UPPER、TIME、

## 五、發明說明 ( 30 )

			LOCATION、或 TOPIC_ALL。一個模式可以兩種元素的任何一種開始，可以是任意長度，包含任意數量／種類的元素。這一過程為表中的每一個模式動態的創建一個新模式記錄並且還可以為一個模式中的每個元素動態創建一個新 tAPatternElements。另外，模式中還有可替代的概念。例如，模式 \$PEOPLES\$ of \$COMPANY\$ 與模式 \$PEOPLES\$ from \$COMPANY\$ 類似。"from"是"of"的替代。我們的結構應能表示這種替代。
GoBF (BF.Main)	Public Sub	Main	這是一個打包過程。它調用 BF 的分析和查詢兩個子程式。它還負責將資料發回 Munin。
ParseMeetingText (BF.Parse)	Public Function	GoBackGround Finder	這一功能獲取會議正文並定義記錄中的用戶 ID 和會議正文的其它部分，包括名稱、主體、與會者列表、地點、和時間。另外，我們調用一個幫助功能 ProcessStopList 從初始會議名稱和會議主體中消除非用詞以便僅留下關鍵詞。分析出來的資訊儲存在 MeetingRecord 結構中。注意這個功能沒有做錯誤檢查，並且多數情況下假設會議正文字符串可以由 Munin 正確的形成。重要的變量 thisMeeting Record 是當前會議所有資訊的暫存處。它最終返回調用程式。
FormatDelimitation (BF.Parse)	Private	ParseMeetingText, DetermineNum Words, GetAWordFrom String	有 4 種方法放置分隔符，我們對每一種情況進行考慮並將其化為第四種情況，僅在字符串的不同的域之間有分隔符 (如 A::B::C)
DetermineNum Words (BF.Parse)	Public Function	ParseMeeting Text, ProcessStop List	這一功能決定字串中包含多少單字 (stInEvalString)。該功能假設每個詞如 stSeparator 定義，由指定的分隔符分開。返回類型是一個整數，表示在假定每個詞由 stSeparator 分開的條件下，發現了多少單字。這一功能總是與

## 五、發明說明 ( 31 )

			GetAWordFrom String 一起使用並在調用 GetAWordFromString 之前調用。
GetAWordFromString (BF.Parse)	Public Function	ParseMeeting Text, ProcessStop List	這一功能從字符串中摘取第 i 個單字 (stInEvalString)，並假定字符串中每一個詞由包含在變量 stSeparator 中的指定的分隔符分開。多數情況下，本功能與 DetermineNumWords 一起使用。這一功能返回所要的詞。這一功能檢查並確保 iInWordNum 在限制範圍之內以便 i 不大於字符串中的所有單字數，不小於/等於零。如果它超出限制，我們返回一個空字符串以指示我們不能得到任何東西。我們透過首先調用 DetermineNumWords 以確保此類事情不會發生。
ParseAndClean Phrase (BF.Parse)	Private Function	ParseMeetingText	這一功能首先獲取單字將它交給 CleanWord 以便去掉沒用的內容。ParseWord 中的一些東西會去除這個單字，所以需要一種循環檢查方法過濾單字而不去掉整個功能，iguess 保存 CleanWord 並檢查返回值 ok，現在我有一個詞所以我需要把它交給分析鏈。該分析鏈按下列過程進行 ParseCleanPhrase -> CleanWord -> EvaluateWord。如果這個單字透過所有環節而不被去除，它將被加入我們的關鍵詞字符串的末尾。首先檢查分隔符"/"並摘取該部分內容。這個過程我叫做 "StitchFace" (Denise 為較一般，可以叫做 GetAWordFromString) 如果這樣發現了單字，那麼它們每一個都將返回分析鏈。如果它們透過整個的分析鏈沒有被去掉也沒有被加入關鍵詞中，那麼最好把它們加入關鍵詞中。
FindMin (BF.Parse)	Private Function	ParseAndCleanPhrase	這一功能包括 6 個輸入值，以及檢查最小的非零值。它首先創建一個容器 (holder) 數組以便我們對五

## 五、發明說明 ( 32 )

			個輸入值從小到大排序。這樣最小值就是數組的第一個非零值元素。如果我們檢查整個數組沒有發現非零值，我們就知道存在錯誤，該程式就會終止。
CleanWord (BF.Parse)	Private Function	ParseAndCleanPhrase	這一功能試圖從會議正文當中清洗一個單字。它首先決定字符串是否有合法的長度。然後對它進行一系列的檢驗，看它是否干淨，如果必要，它對該詞編輯並去掉不必要的字符。這些檢驗包括去掉文件擴展部分、空字符、數字等等。
EvaluateWord (BF.Parse)	Private Function	ParseAndCleanPhrase	這一功能檢查該詞是否在禁用詞表 (stoplist) 中以決定是否從初始會議正文中消除該詞。如果一個詞不在禁用詞表，它會被作為關鍵詞保留，該功能不會返回錯誤。但是，如果單字是非用詞，就會返回錯誤。我們必須正確的界定輸入測試字符串，這樣就不會去錯誤的檢索下一級字符串。
GoPatternMatch (BF.PatternMatch)	Public Sub	GoBF	當我們的 QueryMethod 被設置為複雜查詢時 (即我們希望做所有的模式匹配工作) 該過程被調用。它是一個簡單的賦值程式，對一些數組開始並接著調用對名稱和主體的模式匹配過程。
MatchPatterns (BF.PatternMatch)	Public Sub	GoPatternMatch	這個過程從模式表的每一個模式開始循環，試圖識別一個會議正文中 sInEvalString 定義的不同的域。根據除錯要求，它還統計一定的模式被觸發多少次，並將結果存入 gTabulateMatches 以觀察何種模式調用最多。GTabulateMatches 作為全球變量保存，因為我們要能夠運作一批包含 40 到 50 個測試字符串的文件，還希望知道一個模式被觸發的頻率。
MatchAPattern (BF.PatternMatch)	Private Function	MatchPatterns	這一功能透過當前模式的每一個元素。首先判斷一個元素是否位置標識符或指示器。如果它是位置標識符，就對位置標識符賦值。如果是

## 五、發明說明 ( 33 )

			指示器，我們就有對它定位。這裡有一個技巧，根據我們當前的元素是否是模式的頭部而採取不同的方法。如果我們是在頭部，我們就尋找指示器或位置標識符。如果我們不能找到它，那麼我們知道當前的模式不存在，我們就停下來。無論如何，如果它不是頭部，我們就繼續尋找，因為在別處還有一個頭部。這種情況下我們接著試。
Match MeetingField (BF.PatternMatch)	Private Function	MatchAPattern	這一功能使用一個大開關指令首先決定我們談論的是何種位置標識符，根據其種類我們有特定的要求和賦值標準，如接下來的功能調用 BindNames、BindTime 等所定義的那樣。如果賦值成功，則把它加入我們猜測的記錄中。
BindNames (BF.PatternMatch)	Private Function	MatchMeetingField	在本功能當中，我們將名字與相應的位置標識符 \$PEOPLES\$ 相匹配。名字定義為任意兩個大寫字母開頭的連續的單字。我們還要檢索一系列由 "and"，"，" 和 "&" 連接的名字直到我們看不到這 3 種分隔符的任何一種為止。注意我們不要以一個單獨的名字賦值，因為它可能太普通，因此我們不需要得到廣而無關的結果。這個功能調用 BindAFullName，它賦值了一個名字，所以從一定意義上說，BindNames 包含了所有 BindAFullName 的結果。
BindAFullName (BF.PatternMatch)	Private Function	BindNames	這個功能試圖以一個完整的名字賦值。如果 \$PEOPLES\$ 位置標識符不是模式的頭部，我們知道要轉到所測試的字符串的起始處，因為我們要刪除字符串頭部之前的資訊。如果它是頭部，我們繼續搜索直到找全一個完整的名字。如果無法找到，那麼這個模式中根本沒有這樣的模式，我們從這個模式中完全退出這將返回我們在 MatchPatterns 中的下一個模式。

## 五、發明說明 ( 34 )

GetNextWordAfterWhiteSpace (BF.PatternMatch)	Private Function	BindAFullName, BindTime, BindCompanyTopicLoc	這個功能獲取測試字符串的下一個單字。它尋找空格、@、或/的下一個單字。當我們遇到另一個空格或分隔符時該詞被定義為結束。
BindTime (BF.PatternMatch)	Private Function	MatchMeetingField	得到下一個立即詞並看它是否看來是一個時間模式。如果我們已經發現了一個時間，我們就要將其加入記錄。我們可能需要加入更多的時間模式。但是現在人們似乎並不喜歡在他們的會議名當中鍵入時間，特別是我們現在擁有如 Outlook 這樣的工具。
BindCompanyTopicLoc (BF.PatternMatch)	Private Function	MatchMeetingField	這一功能發現一個連續的大寫字符串並將其賦值給 stMatch，它從 MatchMeetingField 中調用。一個連續的大寫字符串就是一系列沒有被","之類的東西打斷的大寫單字。可能會有更多的東西需要加入中斷列表中。
LocatePatternHead (BF.PatternMatch)	Private Function	MatchAPattern	這一功能試圖定位一個指示器元素。注意這個指示器應該是在模式的頭部否則它將轉到功能 LocateIndicator 去。因此，我們一直獲取下一個單字直到沒有單字可供給我們或我們發現了一個我們要找的指示器為止。
ContainInArray (BF.PatternMatch)	Private Function	LocatePatternHead, LocateIndicator	這個功能相當簡單。它循環檢查數組中的所有元素，來發現一個匹配字符串。
LocateIndicator (BF.PatternMatch)	Private Function	MatchAPattern	這一功能試圖定位一個指示器元素。注意這個指示器不是在模式的頭部，否則它將轉向 LocatePatternHead。這樣，如果我們的模式被滿足了，我們獲取的下一個單字就可能是指示器，否則將導致錯誤。因此，我們僅獲取一個單字，檢查它是否是一個合法的指示器並返回結果。
InitializeGuessesRecord (BF.PatternMatch)	Private Sub	MatchAPattern	這一功能重新開始我們的暫時測試結構，由於我們已經把資訊傳給了永久結構，我們可以對其重新開始使其每個僅有一個元素。

## 五、發明說明 ( 35 )

AddToMeetingRecord (BF.PatternMatch)	Private Sub	MatchAPattern	這一功能僅有當我們知道 tInCurrGuesses 儲存的資訊合法時 (即它代表對會議域的合法的猜測, 該會議域即將被儲存在永久記錄 tInMeetingRecord 中) 才被調用。我們透過檢查確保我們不會儲存同樣內容, 我們還要對準備儲存的資訊進行清理使其沒有諸如標點符號之類的資訊。我們到現在為止沒有清理的原因是為了節省時間。直到我們明確知道將要永久的保存時, 我們才調用 ParseAndCleanPhrase。
NoDuplicateEntry (BF.PatternMatch)	Private Function	AddToMeetingRecord	這個功能循環檢查數組的每個元素以確保被測試字符串 aString 與數組中已有字符串不同。該功能與 ContainInArray 稍有不同。
SearchAltaVista (BF.Search)	Public Function	GoBackGround Finder	這一功能準備一個將要交給 AltaVista 搜索引擎的查詢。它提交查詢並以適當的格式分析返回的結果, 這一格式包含每段檢索到的內容的名稱、URL、和會議主體/摘要等。查詢內容的數目由常數 NUM_AV_STORIES 定義。重要的變量包括 stURLAltaVista 用來儲存要提交的查詢, stResultHTML 用來儲存 stURLAltaVista 定義的頁面的 html。
ConstructAltaVistaURL (BF.Search)	Private Function	SearchAltaVista	這一功能使用高級查詢模式為 alta vista 搜索引擎構造 URL 字符串。它包括要用到的關鍵詞、語言、和如何規定查詢級別。根據我們是否需要使用模式匹配的結果, 我們以不同方式構造查詢。
ConstructSimpleKeyWord (BF.Search)	Private Function	ConstructAltaVistaURL, Construct NewsPageURL	這一功能將儲存在 tTitleKW 或 stBodyKW 中的關鍵詞列表和會議記錄的域進行匹配並將其連成一個字符串, 每個詞之間根據輸入變量 stInConnector 由連接符分開。返回新構造的字符串。

## 五、發明說明 ( 36 )

ConstructComplexAV KeyWord (BF.Search)	Private Function	ConstructAltaVi staURL	這一功能構造發往 AltaVista 站點的關鍵詞。不同於構造簡單關鍵詞只是從名稱當中取得所有關鍵詞來構成查詢，這一功能檢查 BF 的模式匹配的結果並看是否能夠識別任何特定的公司名或主題以構造查詢。如果不能從中識別公司或主題，那麼查詢中將包括簡單查詢中識別出的公司和主題。
JoinWithConnectors (BF.Search)	Private Function	ConstructCompl exAVKey Word, ConstructCompl exNPKey Word, RefineWith Rank	這一功能將字符串中的單字之間的空格簡單的取代以由輸入決定的連接符。
RefineWithDate (NOT CALLED AT THE MOMENT) (BF.Search)	Private Function	ConstructAltaVi staURL	這一功能構造 alta vista 查詢的資料部分並以字符串返回 URL 的這一部分。它確保 alta vista 搜索過去 PAST_NDAYS 內的內容。
RefineWithRank (BF.Search)	Private Function	ConstructAltaVi staURL	這一功能構成交給 Altavista 的字符串以形成一個高級查詢。如果我們構造簡單查詢，我們要摘取名稱中的所有的關鍵詞。對複雜查詢，我們從公司和主題中摘取單字。我們在 ConstructComplexAVKeyWord 中以同樣方式構造查詢。
IdentifyBlock (BF.Parse)	Public Function	SearchAltaVista , SearchNewsPag e	這一功能摘取由開始和結束標誌標出的一個字符串的塊，而輸入從一個特定地點開始 (iStart)。這個塊的檢索不包括標記本身。如果這個塊無法由特定的分隔符辨別，我們透過參數 iReturnSuccess 返回不成功指示作為參考。返回類型是檢索到的塊。
IsOpenURLError (BF.Error)	Public Function	SearchAltaVista ,SearchNewsPag e	這個功能決定所遇到的錯誤是否超時錯誤。它將滑鼠恢復為預設值箭頭，並且如果是超時返回真，其它情況返回假。
SearchNewsPage (BF.Search)	Public Function	GoBackGround Finder	這一功能準備一個將要交給 NewsPage 搜索引擎的查詢。它提交查詢並以適當的格式分析返回的結果，這一格式包含每段檢索到的

## 五、發明說明 ( 37 )

			內容的名稱、URL、和會議主體／摘要等。查詢內容的數目由常數UM_NP_STORIE定義。
ConstructNewsPageURL (BF.Search)	Private Function	SearchNewsPage	該功能構造發往 NewsPage 站點的 URL。它使用輸入會議記錄包含的資訊來決定使用哪個關鍵詞。同時根據我們要用簡單查詢還是複雜查詢，調用不同功能構造字符串。
ConstructComplexNPKeyword (BF.Search)	Private Function	Construct NewsPage URL	該功能構造發往 NewsPage 處的關鍵詞。不同於構造簡單關鍵詞只是從名稱當中取得所有關鍵詞來構成查詢，這一功能檢查 BF 的模式匹配的結果，並看是否能夠識別任何特定的公司名或主題以構造查詢。由於當我們有一個公司名字時，News Page 工作情況最好，所以我們僅用公司名。而僅有不存在公司時，我們才使用主題。
ConstructOverallResult (BF.Main)	Private Function	GoBackGround Finder	本功能接受一個字符串 (stInStories) 數組並以儲存當前會議資訊的會議記錄作為輸入。數組中的每個元素儲存從每個資訊源檢索到的內容。這個功能簡單的構造包含返回消息類型的輸出並發往 Munin，讓 Munin 知道 BF 的回應，原始的用戶 ID 和會議名稱，這樣 Munin 就知道 BF 談論的是哪個會議了。
ConnectAndTransferToMunin (BF.Main)	Public Sub	GoBack GroundFinder	這一功能允許背景搜索器與 Munin 連接並將資訊傳到 Munin。為了正確設定遠方主機和端口，我們使用 UDP 規約而不是 TCP 規約。我們使用一個全局字符串儲存 gResult，因為雖然它對於 UDP 並不需要，但是如果我們將開關撥回且不想更改代碼時，對於 TCP 它就是必要的了。
DisconnectFromMunin And Quit (BF.Main)	Public Sub		

圖 6 是較佳實施例的用於向 Alta Vista 和 Newspaper 搜索引擎準備和提交搜索的實際代碼的流程圖。過程從功能塊 610 開

## 五、發明說明 ( 38 )

始，一段命令行用於以特定的日曆資訊更新日曆入口。根據功能塊620消息被發出，根據功能塊630一個會議記錄被創建用來儲存當前會議資訊。接著，在功能塊640，查詢被交給 Alta Vista 搜索引擎，在功能塊650，查詢被交給 Newspaper 搜索引擎。當一個消息從搜索引擎返回時，如功能塊660所示，它儲存一個結果資料結構，資訊以摘要方式處理並儲存在文件中以備會議使用，詳見功能塊670。

依照較佳實施例，圖7提供了創建查詢更多的細節。處理過程從功能塊710開始，透過分析會議記錄取得可能的公司、人員、主題、地點、和時間。接著，在功能塊720，至少有一個主題被分析出來，並且在功能塊720至少一個公司名被辨識出來，最後在功能塊740，決定什麼資料發往用戶的最終文件。

圖8是圖7所示的查詢方法的一個變形例。在功能塊800對一個會議記錄進行語法分析，在功能塊820一個公司被分辨出來。在功能塊830一個主題被分辨出來，最後，在功能塊840這個主題及／或公司被用於形成查詢。

下面討論實施方法的可替換方法，其中為以下討論的特定用戶需求加入了不同的特性。

### **增強模式匹配的目標命中率**

為提高BF的性能，更多的模式／模式組被加入"CreatePatterns"過程。現有的模式聲明代碼可用做將來模式的樣板。由於所有的東西都以動態數組保存透過剪切和黏貼可以非常方便的再度利用代碼。負責將一個值與位置標

## 五、發明說明 ( 39 )

識符相聯結的功能 BindName、BindTime、BindCompanyLocTopic可得到加強。這個加強是透過增加對一定會議域賦值的標準的設立來實現的，透過這一手段就可以增加賦值的數量。例如，BindTime現在都以##:##或#:##的形式接收和賦值，為增加我們賦值的時間，我們可能希望BindTime也可接受從1到12後面跟著更美觀的時間術語如"o'clock"的數字。基於詞彙的識別運算法賦予BF的每個猜測一個準確率，並且僅允許符合一定門檻的猜測是合法的。

取決於系統透過模式匹配辨識出來的地點或者取決於用戶指定作為會議場合的地點，一個較佳實施例的系統當它檢查到這樣的詞如：午餐／晚餐／早餐時，它就會建議大量的餐廳。我們還可以使用諸如公司搜索器的站點來確認我們得到的是公司名稱或模式匹配沒有辨認出任何公司，我們可以使用公司搜索器網站作為"字典"來決定特定的大寫單字是否代表公司的名稱。我們甚至可以顯示所辨識的公司的股票價格和簡訊。

**較佳實施例的無線交易識別**

圖9是說明設備和軟體系統的硬體和邏輯控制流的流程圖，它允許基於網路的購物和傳統的、物理的、非網路的零售環境進行比較。一個無線電話或類似的具備網際網路通訊能力的無線手提設備920與微型條碼閱讀器910（裝在電話中或一段短電纜上）配合併用於從一本書或其它產品上掃描通用產品代碼（UPC）900。這種無線裝置920透過天

## 五、發明說明 ( 40 )

線 930 將條碼傳入小型交易服務模組（運作於網路伺服器上）940，在其中將其轉化為國際標準書號（針對書籍的情況）或其它任何合適的識別符（其它產品的情況）。服務模組（Service Module）接著到合適的第三方網站尋找價格，裝運和其它可獲取的關於網路供應商各種產品的資訊 950。這些資訊被格式化並顯示於手提設備的屏幕上。IP 無線電話或其它手提設備使用無線調制解調器（modem）如 Metricom 的 Ricochet SE 無線調制解調器。應用這一設備，用戶可以在一個咖啡館使用一臺便攜電腦掛出買單，放在搖晃的小桌子上，盛有拿鐵咖啡的杯子挨著鍵盤，而用電話線直接取用網際網路。

8 盎司 Ricochet SE 無線調制解調器大小如一盒煙並且設置非常簡單，僅需將調制解調器用它的 Velcro 片接於你的便攜電腦屏幕背後，將電纜插入串行口，彈出天線即可傳輸資料。軟體設置同樣簡單：一個直接安裝程式可以加入 Ricochet 調制解調器驅動器，並將連接圖標置於你的桌面。調制解調器的功能方面與傳統電話調制解調器完全相同。

當然，無線上網的表現不像傳統的撥號電話連接可靠。當我們在視窗附近時，在舊金山我們能夠獲得很強的連接。但是，在 CNET 的磚牆指揮部，Ricochet 完全無法連接。當你在上線時，可得到大約 28.8 kbps 的性能，這個速度有些減慢。但是，即使是這個減慢的速度也並不另人失望。比較替代方式—透過單體調制解調器連接—Ricochet 快得多、更可靠、而且使用起來並不昂貴。當然，SE 無線調

## 五、發明說明 ( 41 )

制解調器是電池供電的。調制解調器具備將近12小時的連續供電能力。依照較佳實施例，在Ricochet信號開始減弱之前，我們就已經用完了便攜電腦的雙重電池。

因此，依照較佳實施例，用戶可利用無線調制解調器，使用網路伺服器軟體940辨認正確的產品950並使用相應的裝置鍵選擇供應商並進行訂貨。交易服務模組與第三方網路供應商完成訂貨960。

### **mySite! 個人網站和目的評估網絡原型**

mySite! 是富有衝擊性的基於網際網路的應用，它專注於提供服務的方法並在以顧客為中心的世界透過個人網站為每個顧客提供個性化經驗。這些服務是以直覺所組織以用來滿足用戶的需求—需多方面籌劃的基本生活需要或目的、以及對幾個領域進行協調的需求，如財務計劃、醫療計劃、個人或專業發展、家庭生活、和其它方面。每個組成部分都包含並代表用戶個人，這使得他能夠創建並瀏覽系統為他設計的特定的內容。從輸入一個產品或服務的需求開始，到完成付款，智能代理人可以用來實施研究、執行交易、並提供建議。透過進一步說明和篩選，智能代理人學習有關它所服務的用戶的知識，提高其服務水準。用戶的需要包含日常後勤管理（如電子郵件、日曆、約會、日程表、帳單、購物、和旅遊計劃）；並包括社區轉移（如尋找居住地、轉移家庭財產、計算旅遊和裝運保險總額、通知生意和個人交際、了解新社區）。從用戶的角度來說，mySite! 提供了用戶可獲得相關產品和服務並最便捷

## 五、發明說明 ( 42 )

的完成日常任務的中心。從生意的角度來說，mySite!代表了一種有效的吸引、服務、和留住顧客的增值的和創新的方法。

### 自我中心界面

自我中心界面是專門用來滿足特定用戶的需求、偏好、和當前語境的用戶界面。它使用儲存在一個中心個性資料庫的用戶個人資訊來對界面進行顧客化。用戶可以設置安全許可並選擇界面元素和內容。整合在自我中心界面的內容根據用戶相關資訊被顧客化。當顯示內容時，自我中心界面會包含內容和用戶的關係，並顯示這些內容如何與用戶相關。例如，當顯示關於一個即將到來的用戶已經訂好的滑雪度假時，該界面就會包含用戶個人日曆和約會表中的事件資訊，如滑雪度假期間將要出現的其他的人。這用來向用戶個人熟悉的語境加入新的資訊。

圖10B是用於在自我中心的界面中創建網頁的邏輯的流程圖。該環境假定一個伺服器和一個瀏覽器透過TCP/IP網絡連接，如透過公共網際網路或私人的內部網路(Intranet)。可能的網路伺服器包括微軟Internet Information Server、網景Enterprise Server、或Apache。可能的網路瀏覽器包括微軟Internet Explorer或網景Navigator。客戶機（如網路瀏覽器）向伺服器1001（如網路伺服器）創建一個對特定網頁的要求。這通常是由用戶點選一個按鈕或網頁中的一個鏈接來完成的。當鍵入一個獨一無二的儲存在客戶機（如網路瀏覽器）和用戶特性資料庫1003的用戶ID時，網路伺服

## 五、發明說明 ( 44 )

(profile)資訊，形成用於不同語境的設置。用戶可以創建一個角色用於為家庭購物。這一角色可能包含他的家庭住址並可能指出這個用戶在購物時尋求最好的價格。同一個用戶可能創建第二個角色，用於他的工作語境。這一角色可能存有用戶的工作地址並可能指出這個用戶選擇特定賣主或有打折活動的特定公司的工廠。當購貨與工作有關時，用戶可能會使用這個角色。角色可能也包含規定和限制。例如，工作角色可能限制用戶僅向單一的旅行社訂購機票，並根據他的雇主規定來做。

圖12說明了用戶，他的多種角色和他的多種概貌之間的關係。在用戶的層面上是用戶概貌1200。這一概貌說明用戶和他的帳目資訊。每一個用戶的帳目在資料庫中都有獨一無二的記錄。附在每個用戶之上的是多種角色1220、1230、與1240。這些角色用於將多種概貌組合成實用的語境。例如，試想一個用戶住在舊金山、在帕羅奧多市工作，但是在Lake Tahoe山上有個小屋。他就有三種不同的取用他的站點的語境。一個語境是與工作相關的。另外兩個是與家庭生活相關的，但是在不同的地點。用戶可以創建一個工作的角色1220、一個家庭的角色1230、和一個小屋的家中的角色1240。每個角色都參考不同的總體概貌1250、1260、和1270，它們包含位置地址。因此，有三個總體概貌。每個角色還參考兩個旅行概貌中的一個。用戶保持一個工作旅行概貌1280，包括與訂票和預訂機票相關的工作上的準則。這一概貌可能規定，例如，此人僅能因工作需

## 五、發明說明 ( 45 )

要而旅行或工作是第一位的而且他的選擇僅能是國家航空公司 (United Airlines)。工作角色就參考這種工作的旅行概貌。用戶可能還有一個家庭旅行概貌 1290，規定他選擇坐馬車旅行並且要尋得非償還市集，因為他們通常較便宜。不論家的角色還是小屋的家的角色都指向家庭旅行的用戶概貌。

圖 13 說明了支援角色概念的資料模型。用戶表格 1310 包含了在系統裡有帳戶的每個用戶的一個記錄。該表格包含一個用戶名和一個口令 1320 和一個獨一無二的標識符。每個用戶可擁有多種角色 1330，作為包含大量叫做概貌 1340 的容器。用戶概貌在概貌域 1350 記錄中包含詳細的個人資訊。附於每個概貌上的是一系列概貌約束 1360 記錄。它們每個都包含一個名字 1370 和一條規則 1380，用來定義這個約束。規則以類似 (如果 x 則 y) 的形式出現，允許規則限於特定的用途。在一個例子中，用戶概貌約束可以是一條規則，指示用戶不能訂購包括在列表中的航空公司的機票。例如，這條概貌約束可以包括在由用戶的雇主設定的 "工作" 角色中的 "旅行" 概貌中。每個概貌域還包含一系列位於記錄中的許可 1390。這些許可指示對於特定概貌域的資訊，什麼人擁有什麼樣的取用權。

### 目的中心界面

為滿足用戶的目的，如退休計劃或重新定居需要一個特定的界面。用戶的目的需要全面的計劃和許多領域的協調配合，從財務安全、住房和交通到醫療衛生、個人和職業

## 五、發明說明 ( 46 )

發展，以及娛樂和它的東西。滿足目的需要一個補充的業務網絡，與其它各種產業共同滿足顧客要求。

目的中心界面是一個用來幫助用戶管理個人計劃的界面。從已有的觀點來看，界面內容被顧客化僅顯示與特定目的相關的內容。目的中心界面允許用戶管理用來滿足特定目的的這個過程。這包括一系列離散的步驟和一系列用戶可以取用的內容區域。從任何一點來說，用戶都可以切換界面來管理不同的目的，這樣做會改變界面的內容使其僅包含與滿足最近選擇的目的相關的內容。

圖14提供了用於支援需求中心界面的資料模型的詳細說明。每一個用戶角色1410（見圖13對角色資料模型的詳細說明）有任意個有效的用戶目的1420。每一個用戶目的給出一個暱稱1430，也就是用戶在屏幕上看到的名字。每個有效的用戶目的還包含一些資料域1440，它們包含與用戶互動過程中獲取的任何用戶資料。例如，如果用戶填好了屏幕上的一個表格，其中一項是社會保險號碼，相應的資料域就可能包括名字="SSN" 1450，數值="999-99-9999" 1460。每一個用戶目的還追蹤目的步驟1470完成的情況。完成域1480指示用戶已經完成了這個步驟。每個用戶目的都是一個通用目的1490的用戶專用化的版本，這個通用化目的透過用戶規則1411和1412進行顧客化，並被附於目的的子步驟上。顧客規則是一些說明系統如何使用單個用戶的概貌資訊為每個單獨用戶單獨定制目的的。

### 統計代理人

## 五、發明說明 ( 47 )

一個代理人跟蹤每個用戶的關鍵統計資料。這些統計資料用來以與電子寵物雜虛擬實境寵物玩具類似的方式鼓勵用戶的特別的行為。所記錄的統計資料是登入的頻率，內容評估的頻率如新聞條目，代理人的活動，這根據一定期間執行的任務的數量來量測。這個資訊由系統用來從情感上吸引用戶以鼓勵特定的行為。

圖 15 說明產生顯示於代理人當前統計資料的頁面的過程。當用戶從客戶瀏覽器中要求代理人統計資料頁 1510 時，伺服器從用戶的概貌資料庫 1530 中檢索用戶的統計資料 1520。伺服器接著執行必要的數學計算來創建標準的統計資料 1540。伺服器接著從內容資料庫 1560 中檢索公式 1550，該內容資料庫用來計算用戶中心模式的統計資料。使用一般的公式和用戶的統計資料圖形就產生了 1570。這些圖形被插入樣板來創建統計資料 1580 頁。這一頁接著返回用戶 1590。

### **個性化產品報告服務**

系統提供類似顧客報告的服務，它根據用戶概貌針對每個顧客進行顧客化。系統對於產品質量和可取性從多方面為用戶記錄並提供評估。該系統和傳統產品質量測量服務之間的區別是該系統返回顧客的評估是個性化的。這一服務的工作方式是透過發現與用戶概貌最匹配的人並事先對所要求的產品評估。使用這種運算法有助於確保發回用戶的產品報告僅包含與用戶相似的人們的統計資料。

圖 16 說明了決定對一個用戶的個性化產品評估等級的運

## 五、發明說明 ( 48 )

算法。當用戶要求產品X的一個產品報告1610時，該運算法從包含已經評估過該產品的用戶的概貌資料庫1630（包括產品評估等級）中檢索用戶概貌1620。然後系統檢索從內容資料庫1650中匹配概貌的預設門檻值1640。接著它繪製概貌匹配運算法1660中規定的幾個方面的所有的用戶短表。用n（已定義為門檻值變量）決定最近的鄰居數，利用概貌匹配運算法的結果，透過測試決定它們是否在設置1670中的用戶概貌的距離y（也已定義為門檻值變量）之內。如果它們不在門檻值之內，門檻值變量被釋放1680，再一次進行測試。這一過程反覆進行直到測試返回真。透過n個最近鄰居的較小的設置得到產品的評估並將其用於決定幾個方面的產品統計資料1690。這些統計資料插進產品報告樣板1695，並作為產品報告返回用戶1697。

### 個人概貌和服務無處不在

這個系統提供了一個個人概貌的中心存放處。這一存放處是一個透過公共網際網路可得到的伺服器，它可以被任何與網際網路相聯的設備透過適當的途徑取用。由於隨處都可以獲得用戶的概貌，大量取用設備可用來根據用戶的概貌將用戶的服務顧客化。例如，商品的網站可使用這個概貌向用戶提供個性化內容。可取用網際網路的個人數字助理（PDA）可以將PDA上的個人日程表、電子郵件、聯繫簿、任務表和通知與存在網際網站的版本進行同步。這使得此人僅保有這個資料的一個版本，以便可以以任何它需要的方式隨時獲得。

## 五、發明說明 ( 49 )

圖 17 給出了取用這一中心儲存概貌的許多不同方法的詳細邏輯。概貌資料庫 1710 是用戶概貌資訊的中心儲存處。概貌通道伺服器 1720 接收對概貌資訊的所有請求，無論是用戶本人還是商店主想向用戶提供一個服務。概貌通道伺服器負責保證僅有當概貌擁有人給出許可資訊才會發出。任何可以透過 TCP/IP (一種標準網絡通訊協議) 取用公共網際網路 1730 的設備都可以從概貌伺服器中以智能 HTTP 請求資訊。顧客能夠從諸如電視機 1740、行動電話、智能卡、煤氣表、水表、廚房器具、安全系統、桌面電腦、袖珍管理器、PDA、他們的汽車、及其它的設備中得到獲取服務的途徑。類似的，店主能夠獲取這些概貌 (取得擁有每個概貌的顧客的許可)，並能夠因此向消費者提供顧客化和個性化的服務。

無處不在的概貌的一個可能的應用是在連鎖旅館。顧客可以攜帶一個具有可以獨一無二的辨認出他的數字證書的智能卡。該智能卡的數字證書由系統處理，它將他的概貌資訊記入概貌資料庫。用戶把此卡帶到連鎖旅館並經過檢查獲準進入。旅館雇員刷卡，顧客輸入他的 Pin 號碼，對數字證書解鎖。該證書送往概貌開關伺服器 (利用安全傳輸規約) 並被授權。旅館獲準讀取事先定義的顧客概貌的特定部分。接著旅館檢索所有顧客的帳目資訊和對旅館房間的偏好等等。旅館還可以讀取顧客對電影和飲食的偏好並提供這兩項的顧客化項目單。旅館可提議向顧客的配偶發電子郵件讓他 / 她知道此人已登記住進旅館並且是安全

## 五、發明說明 ( 50 )

的。當旅館登記這個顧客之後，所有的交易資訊都可以裝入顧客的概貌之中。這將允許旅館的合夥人利用該旅館搜集的該顧客的資訊（再一次的，徵得該顧客的允許）。

### 需求價值網絡

在一個需求價值網絡之中，總體整合系統為顧客協調產品的傳遞和服務。整合器管理一個網絡，它用來協助供貨商根據反映在用戶概貌中的用戶偏好，從現實和虛擬兩方面向顧客提供產品和服務。整合器管理供貨商和消費者之間的關係，並協調供貨商滿足顧客的需求。它是透過供給顧客關於產品和供貨商的資訊，以及提供客觀建議和其它手段，來作到這一點的。

圖18揭示了在一個消費者和有關於一個供應商的整合器之間的詳盡的互動過程。用戶取用一個網路瀏覽器1810並向整合器詢問產品和價格資訊。該詢問從用戶的瀏覽器發往整合器的網路／應用伺服器1820。用戶的偏好和個人資訊從整合器的顧客概貌資料庫1830獲得並返回網路／應用伺服器。所詢問的產品資訊從供貨商的产品資料庫1840中摘取，並為特定顧客進行顧客化。網路／應用伺服器用顧客的詢問資訊更新供貨商的顧客資訊資料庫1850。產品和價格資訊接著格式化進網頁1860並返回顧客的網路瀏覽器。

### 摘要代理人

一個運作在應用程式和網路伺服器上的軟體代理人套裝在程式規劃時注意了用戶的重複的或現實的任務。代理人

## 五、發明說明 ( 51 )

根據用戶設定的規則工作，並僅允許執行用戶詳細定義的任務。代理人特別注意為用戶支付帳單，過濾內容和電子郵件，並提供關於任務和代理人活動的摘要瀏覽。代理人的用戶界面可以透過修改以適合特定的用戶。

圖19揭示了較佳實施例的代理人為用戶產生一個口頭摘要的邏輯過程。當用戶要求摘要頁1900時，伺服器從用戶概貌資料庫1930中給出用戶代理人的偏愛1920，如代理人類型、規則、和摘要等級。伺服器從內容資料庫1950中取得這些內容1940，如電子郵件、將完成事項列表、新聞、帳單。代理人使用儲存在概貌資料庫中的規則分析所有內容，並對內容1960摘要。內容根據一個樣板格式化進網頁1970。使用內容資料庫的內容1990和儲存在資料庫中的講話樣板，產生代理人講話的正文1980。講話正文插入網頁1995，該頁被返回用戶1997。

### 可信賴的第三方

根據已發表的政策，以上的方法要求網站對資訊的隱私權的保證。這個系統是顧客的可信賴的第三方，考慮它在各種情況下的利益，在資訊的隱私性方面可能的失誤要比在刺激商業機會方面可能的失誤大的多。可信賴的第三方具有一系列的程式正好保證了它與規定政策的同一性。

### 個性化電子商務"meCommerce"

這個單字是單字"eCommerce"的擴展，意思是"個性化電子商務"。

圖20說明了較佳實施例的一個登入顯示。該顯示用於包

## 五、發明說明 ( 52 )

含一個代理人2000的應用，它指引用戶透過與系統的互動過程並顧客化和個性化各種系統元件以搜集資訊並與用戶的個人需要進行互動。用戶在2010鍵入名字，在2020鍵入口令並選擇一個按鈕2040來開始登入程式。如標識2030所建議的，將電子商務進行個性化處理，所以叫做"我的"商務。

圖21說明了依照較佳實施例管理日常後勤的顯示。用戶透過一個動畫代理人2100以個性化資訊2190得到問候。用戶可以從基於需求的不同的活動中進行選擇，包括旅行2110、家務2120、財務2130、和逛商場活動2140。圖標2142表示常規任務如電子郵件、日程表、和文件比較，它們也可以用來方便從一個活動轉向另一個活動。直接鏈接2146也允許新聞和其它興趣項的傳遞。根據用戶的位置可以選擇不同的概貌。例如，工作、家庭、或假期。當用戶為另一個地點要求一個新的概貌時，可以添加概貌2170。不同的個人資訊2180從用戶中選擇出來以支援不同的目的。另外，還為各項內容2180設置許可2150以保證資訊的及時性和不過時。

圖22說明了較佳實施例的一個用戶的主畫面。根據用戶的選擇提供世界2200和本地新聞2210。用戶還可以選擇個人不動產2230作為主畫面顯示的直接資訊。另外，根據用戶的偏好，可提供不同的代理人2220。

圖23說明了較佳實施例的一個代理人互動。代理人2310與用戶交流資訊2300，指示用戶的壽險需要更改，並指給用戶經過最好總結的用戶的資訊圖表。特別暗示2395用於

## 五、發明說明 ( 53 )

提供更多的基於當前用戶統計資料的更詳細的資訊。用戶的壽險需求的圖表2370也被高亮顯示在中央，幫助用戶採取合適的行動。按鈕2380用來幫助改變策略，一組按鈕2390用來幫助用戶選擇用戶保險需求的不同內容。

### 事件背景器

事件背景器是在事件發生之前發往用戶的即將來臨的事件的簡短說明。事件背景器根據最近的與事件相關的資訊不斷更新。相關資訊，如旅程和家務都包括在其中，其它有用的資訊，如用戶認識的可能在同一地點出現的人，也包括在其中。事件背景器的目的是為用戶提供最新的資訊，依靠一些資訊源，如公共網站和用戶日程表和聯繫表，使用戶在給定情形下得到最好反映。

### 鄰近朋友尋找程式

當一個朋友、家庭成員、或熟人出現在或將要出現在用戶的區域時，這一軟體尋找機會告訴這個用戶。這個軟體掃描用戶日程表尋找即將來臨的事件。接著它使用圖形化的地圖比較這些日程表事件和列在他的熟人列表中的人的日程表事件。接著如果有滿足匹配的條件，它通知用戶並告知某人預計在特定時間在附近出現。

### 資訊過載

資訊過載要從它的定義和含義及後果兩方面理解。人們在一個時間內僅能把注意力放在有限的事情上，但是今天對用戶注意力的競爭越來越激烈。簡單的說，太多的資訊和太少的時間是今天大多數知識工作者生活複雜化的主要

## 五、發明說明 ( 54 )

因素。

動態處理資訊過載的第一個嘗試主要集中於資訊的智能篩選，這樣資訊的數量就會減少。大多數這些方法不是簡單的去掉隨機的資訊，而是試圖在最終交給用戶什麼資訊的問題上進行智能化處理。透過根據用戶興趣評估每份文件並去掉較少相關的文件，就完成了這個過程。因此，接下來，它的質量也得到了提高。

在這個新的年代，資訊篩選只是資訊處理的第一步。可論證的是，與文件質量同樣重要的是它是容易取用的。當你參加一個會議時，一份包含會議主題的標準資訊的文件送到你的辦公室並沒有太大價值。由於在聯絡技術的推動下，業務進行的速度越來越快，無論何時何地都能收到合格的資訊變得相當重要。這個新的方法叫做智能資訊傳遞，並且是新的資訊時代的先驅。

較佳實施例示範了以上說明的智能資訊傳遞理論，不僅降低了資訊過載，還能不論何時何地僅要用戶需要，就能傳遞高質量的資訊。換言之，這個系統在合適的時間合適的地點向合適的人傳遞合適的資訊。

### 活動知識管理系統說明

圖24是依照較佳實施例的活動知識管理系統的方塊圖。這個系統以下部分組成：接於一個或一個以上伺服器的後備2400、個人行動無線客戶機（認知機器）2430、2436、公共客戶機（魔術牆）2410、2420、網路客戶機2446、2448、e-mail客戶機2450、2460。

## 五、發明說明 ( 56 )

- 兼容當前網路標準的多媒體輸出
- 聲音識別
- 觸摸輸入
- 智能代理人以可發聲動態字符形式出現

魔術牆如下操作：

1. 如果一個用戶在魔術牆附近出現，感應器組觸發"用戶在這裡"事件，將包括用戶 ID 和位置的環境線索傳給智能代理人協調員。
2. 根據感應器組傳回的資訊辨識用戶。
3. 魔術牆切換到"鎖定用戶"模式。如果另一個用戶進入，系統提示他或她在當前用戶接受服務時，系統不能為別的用户服務。
4. 通知智能代理人協調員用戶的存在。
5. 智能代理人協調員決定用戶和魔術牆將要顯示的本地時間相關資訊是否相關（如交通報導、會議提示）。如果這樣的資訊存在，就準備傳遞。如果不存在，控制權交給資訊最佳化子系統。
6. 資訊最佳化子系統根據用戶的個人概貌、更新的資訊、和智能代理人協調員以前的建議，決定什麼資訊與用戶最相關。
7. 顯示此時和此地被辨識出來與用戶最相關的資訊頁。資訊傳遞的方式也可以包括從智能代理人代表那裡發出的動畫和講話。
8. 如果用戶希望，他／她可以要求魔術牆顯示一個特殊

## 五、發明說明 ( 57 )

頁。魔術牆辨認講話片段，並顯示所要求的頁面。

9. 當用戶從魔術牆區域離開，感應器組觸發"用戶離開"事件。

10. 魔術牆切換到等待狀態。

### 其它客戶

網路客戶是一個在一批網頁中導航的標準瀏覽器，它允許用戶看到透過魔術牆同樣可以看到的相同的資訊。

電子郵件客戶是任何標準電子郵件程式。

### 智能代理人協調員說明

這一段代碼是活動知識管理系統的協調代理人（或介代理人(meta-agent)）。這意味著系統和每一個用戶之間的通信，以及不同下級代理人之間的通信都由智能代理人協調員處理（協調）。這些下級代理人的例子有：

- 背景查詢器—為每個用戶分析會議正文以決定重要的關鍵詞並分析和發現會議的背景資訊的一個代理人
- 交通查詢器—根據用戶住在何處為每個用戶查詢交通資訊的一個代理人
- 負責每個用戶的概貌中資料的統計資料分析和該資料相關域的更新的幾個其它代理人

圖25的智能代理人協調員2580也是用戶對系統的"界面"，每當用戶與系統互動時，不管是GUI還是其它終端用戶界面，它們最終要與智能代理人協調員打交道（詢問問題或發出指令）。智能代理人協調員有四個基本的責任：1）監視用戶行動、2）處理資訊查詢、3）保持每個用戶的概

## 五、發明說明 ( 58 )

貌、和4) 將資訊向或從用戶路由以及將資訊向或從其它相應的代理人路由。

### 監視用戶行動

任何時候一個用戶觸發一個感應器，智能代理人協調員收到一"環境提示"。這些線索不僅使智能代理人協調員能夠獲知傳遞資訊的用戶方位，還可以得到每個人生活的標準模式（到達時間、離開時間等等）。這些模式不斷的更新以求提高系統傳遞資訊的智能性。例如，今天一個人擁有幾個電子郵件帳戶（工作的、家庭的、行動的等等），和幾臺不同的電腦用於檢索所有這些帳戶的現象已不少見。因此，智能代理人協調器要想成功的向正確的地點傳遞資訊，它必須考慮所有這些因素，以及用戶可能取用它們的時間以便最大可能的使用戶得到資訊。這將在下一節進一步討論。

### 處理資訊查詢

為了對每個用戶要求的資訊個性化，以及更準確的反映每個用戶給出的資訊中的用戶的興趣，智能代理人協調員處理來自其它代理人的查詢要求。這些查詢要求通常與用戶概貌有關。例如，如果一個代理人為用戶準備一份交通報告，它可能向智能協調員要求該用戶的交通區域（搜索字串）。所有對用戶概貌資料的取用都透過這種方法存取。

### 保持用戶概貌

用戶概貌包括用戶的各方面的資訊。這個資訊是用戶定

## 五、發明說明 ( 59 )

義資料和智能代理人協調員從每個用戶的資訊和活動中學習和外推得到的資訊的一個混合體。為了保護在概貌中的資料，智能代理人協調員必須處理所有要求的用戶資訊。智能代理人協調員透過觀察用戶的行動和努力學習他們的生活模式，來不斷的修改和更新這些概貌，以便幫助處理更加常規的、重複的任務。智能代理人協調員還雇用其它代理人搜集每個用戶的日常活動的意圖。這些代理人發掘這一資料來發現當前興趣、長期興趣的指向，以及每種資訊傳遞時間的偏好。智能代理人協調員的另外一個重要方面就是它還根據路由的目的試圖決定每個用戶一天當中的實際位置。

### 資訊路徑

大多數人一天當中是移動的。智能代理人協調員透過在觀察（非指導性學習）和提取環境線索兩方面決定用戶在／或可能在什麼位置，來試圖感覺到這一情況。這對於向何處發送用戶的資訊肯定是重要的，同時對於以何種方式發送資訊也是重要的。例如，如果用戶在他的桌子前使用網路瀏覽器，智能代理人協調員可能會從她的PC機不斷接到她活動的指示，並由此知道將必要資訊發向那裡。另外，由於桌上型PC一般功能很強大，特性完備，應該發送圖形質量更高的版本。但是，考慮可能的情形：智能代理人協調員接到你剛剛離開樓房的指示（透過退出邊上的卡片閱讀器）。幾分鐘後，智能協調員又收到你接到一條緊急郵件的通知。智能協調員知道你已經離開樓房並且沒有

## 五、發明說明 ( 60 )

收到任何其它指示，假設透過你的手提設備（它也知道其權能）可以找到你，它就將這條緊急資訊的正文發往那裡，而不是一個面向圖形的版本。

### 特有的創新

活動知識管理系統代表了知識管理和人機互動方面的一些最先進的想法一些主要的革新如下：

- 上面說明的智能代理人協調員。
- 智能資訊傳遞理論的發展、示範、和實現。
- 支援幾個資訊傳遞渠道，所有的傳遞渠道使用共同的後備。例如，如果一個用戶位於魔術牆前面，資訊將以多媒體方式提出。如果系統決定用戶是流動的，資訊將以標準正文送到他們可用的機器中。無論何時何地僅要用戶需要資訊，它都可幫助資訊的傳遞。
- 資訊的個人化不僅設計在靜態用戶概貌上，還考慮了用戶互動的歷史資料和包括"何人、何處、何時"等的當前即時情況。
- 應用快速、可量測的資訊排序子系統，它考慮了智能代理人的意見用戶的選擇和用戶互動的歷史。它將世俗決定從智能代理人部分去除掉，因此允許代理人更複雜精確而不影響其可量測性。
- 語音識別和語音綜合配合智能代理人使敘述更加活潑，觸覺輸入提供了有效的直覺的和人情味的與系統的界面。

### 較佳實施例的支援代碼

## 五、發明說明 ( 61 )

下面的代碼是依照較佳實施例，在微軟 Active Server Pages環境中編寫和執行。它主要由包含一些記錄代碼中的資料庫調用的微軟 Jscript組成，可向資料庫查詢和儲存資訊。

### 目的中心的界面

#### 創建一個目的 ASP 頁 ("intention\_create.asp")

```
<%@ LANGUAGE = "JScript" %>
<%
Response.Buffer = true;
Response.Expires = 0;
%>

<html>
<head>
  <title>Create An Intention</title>
</head>

<body bgcolor="#FFE9D5" style="font-family: Arial" text="#000000">

<%
//Define some variables

upl = Server.CreateObject("SoftArtisans.FileUp")
intention_name = upl.Form("intention_name")
intention_desc = upl.Form("intention_desc")

//intention_name = Request.Form("intention_name")
//intention_desc = Request.Form("intention_desc")

//intention_icon = Request.Form("intention_icon")
submitted = upl.Form("submitted")
items = new Enumerator(upl.Form)
%>

<%
//Establish connection to the database
objConnection = Server.CreateObject("ADODB.Connection")
objConnection.Open("Maelstrom")
%>
```

## 五、發明說明 ( 62 )

```

<%
//Check to see if the person hit the button and do the appropriate thing
if (submitted == "Add/Delete")
{
    flag = "false"

    //loop through all the inputs
    while(!items.atEnd())
    {
        i = items.item()

        //if items are checked then delete them
        if(upl.Form(i) == "on")
        {
            objConnection.Execute("delete from user_intention where intention_id =" + i);
            objConnection.Execute("delete from intentions where intention_id =" + i);
            objConnection.Execute("delete from tools_to_intention where intention_id =" + i)
            flag = "true"
        }
        items.moveToNext()
    }

    // if items were not deleted then insert whatever is in the text field in the database
    if(flag == "false")
    {
        intention_name_short = intention_name.replace(/ /gi,"")
        objConnection.Execute("INSERT INTO intentions
(intention_name,intention_desc,intention_icon) values('" + intention_name + "','" +
intention_desc + "','" + intention_name_short + ".gif' + "'")")
        Response.write("the intention short name is " + intention_name_short);
        upl.SaveAs("E:development/asp_examples/" + intention_name_short + ".gif")
    }
}

// Query the database to show the most recent items.
rsCustomersList = objConnection.Execute("SELECT * FROM intentions")
%>


```

## 五、發明說明 ( 63 )

```

<tr><td><font face="Arial">Description:</font></td><td><TEXTAREA
name="intention_desc"></TEXTAREA></td></tr>
<tr><td><font face="Arial">Icon Image:</font></td><td><INPUT TYPE="file"
NAME="intention_icon" size=40></td></tr>
<tr><td colspan="2"><INPUT type="submit" name="submitted"
value="Add/Delete"></td></tr>
</TABLE>
<HR>
<font face="Arial" size="+1"><b>Current Intentions</b></font>
<TABLE>
  <tr bgcolor=E69780 align="center">
    <td>
      <FONT color="white">Delete</FONT>
    </td>
    <TD>
      <FONT color="white">Intention</FONT>
    </TD>
    <TD>
      <FONT color="white">Description</FONT>
    </TD>
    <TD>
      <FONT color="white">Image</FONT>
    </TD>
  </tr>

  <%
  // Loop over the intentions in the list
  counter = 0;
  while (!rsCustomersList.EOF)
  {
  %>
    <tr bgcolor="white" style="font-size: smaller">
      <td align=center>
        <INPUT type="checkbox" name="<%=rsCustomersList("intention_id")%>">
      </TD>
      <td>
        <%= rsCustomersList("intention_name")%>
      </td>
      <td>
        <%= rsCustomersList("intention_desc")%>
      </td>
      <td>
        ">
      </td>
    </tr>

  <%

```

## 五、發明說明 ( 64 )

```

counter++
rsCustomersList.MoveNext()
%>
</TABLE>
<hr>
Available Tools
</form>
</BODY>
</HTML>

```

**檢索目的列表 ASP 頁 ("intentions\_list.asp")**

```
<!-- #include file="include/check_authentication.inc" -->
```

```

<HTML>
<HEAD>
  <TITLE>mySite! Intentions List</TITLE>

<SCRIPT LANGUAGE="JavaScript">
  function intentionsList () {

    this.internalArray = new Array();

    <%
    // establish connection to the database
    objConnection = Server.CreateObject("ADODB.Connection");
    objConnection.Open("Maelstrom");

    // create query
    intentionsQuery = objConnection.Execute("SELECT * FROM intentions ORDER
    BY intention_name asc");
    %>

    // write out the options
    <%
    numOptions = 0
    while (!intentionsQuery.EOF) {
      intentionName = intentionsQuery("intention_name");
      intentionIcon = intentionsQuery("intention_icon");
    %>

      this.internalArray[<%= numOptions%>] = new Array(2);
      this.internalArray[<%= numOptions%>][0] = "<%= intentionName %>";
      this.internalArray[<%= numOptions%>][1] = "images/<%= intentionIcon %>";
    %>
      numOptions++; intentionsQuery.MoveNext(); %>

    <% } %>
  }
  numIntentions = <%= numOptions%>;

```

## 五、發明說明 ( 65 )

```

intentionArray = new intentionsList().internalArray;
function selectIntention () {
    for (i=0;i<numIntentions;i++) {
        if (IntentionsListSelect.options[i].selected) {
            intentionNameTextField.value = intentionArray[i][0];
            //intentionPicture.src = intentionArray[i][1];
            break;
        }
    }
}
</SCRIPT>

</HEAD>

<BODY BGCOLOR="<%=Session("main_background")%" style="font-family: Arial">

<CENTER>
<!-- <FORM NAME="intention_list"> --->
<TABLE FRAME="BOX" border=0 CELLPADDING="2" CELLSPACING="2">

<TR><TD COLSPAN="3" STYLE="font: 20pt arial" ALIGN="CENTER"><B>Add a
mySite! Intention</B></TD></TR>

<TR><TD COLSPAN="3">&nbsp;</TD></TR>

<TR>
    <TD width="100"><font size="-1">Please Select An Intention You Would Like to
Add to Your List</font></TD>
    <TD colspan=2>
        <SELECT ID="IntentionsListSelect" NAME="IntentionsListSelect" SIZE="10"
style="font: 9pt Arial;" onClick="selectIntention()">
            <%
            intentionsQuery.moveFirst();
            for(j=0;j<numOptions;j++) { %>
                <OPTION VALUE="<%= intentionsQuery("intention_id") %%" <% if (j == 0)
{ %> SELECTED <% } %>>
                <%= intentionsQuery("intention_name") %%">
                <% intentionsQuery.moveNext()
            }
            intentionsQuery.moveFirst();
            %>
            </SELECT>
        </TD>
    </TR>
</TR>

```



## 五、發明說明 ( 67 )

```

+ "</SCRIPT><TABLE cellpadding='0' width='100%' cellspacing='0'>");

    while (!intentionsQuery.EOF)
    {
    %>
    <TR><TD><a href="javascript:changeIntention('<%=
intentionsQuery("user_intention_id") %>','<%=numintentions%>')"
onmouseover="mouseOverTab()" onmouseout="mouseOutOfTab()"><font color="Black"
face="arial" size="-2"><%= intentionsQuery("intention_custom_name")
%></font></a></TD><TD><IMG align="right" SRC="images/delete.gif" alt="Delete this
intention" onClick="confirmDelete(<%= intentionsQuery("user_intention_id")
%>)"></TD></TR>
<%=numintentions++; intentionsQuery.moveToNext();
%>

    <% }
Response.Write("<SCRIPT>numintentions="+numintentions + "</SCRIPT>");
%>
<tr><td colspan="2"><hr></td></tr>
<TR><td colspan="2"><a
href="javascript:changeIntention('add ...',<%=numintentions%>);"
onmouseover="mouseOverTab()" onmouseout="mouseOutOfTab()"><font color="Black"
face="arial" size="-2">add ...</font></a></td></TR>
    </table>
</body>
</DIV>
<DIV style="position: absolute; top:0; left:-5; width: 230; height:105; z-index:1; "
onmouseout="intentionlist.style.visibility='hidden'"
onmouseout="intentionlist.style.visibility='hidden'"
onmouseover="intentionlist.style.visibility='hidden'"></DIV>
</DIV>
</DIV>

```

**Ilink 系統**

較佳實施例的一個例子可以用來協助輔導一個一般熟練人員作出並使用本發明而沒有不適應經驗。設想一個用戶瀏覽雜誌的章節，這時她發現一個很好的OMEGA手錶的廣告。她想起她的男朋友最近提起他想要一支新手錶。因為她還沒有為他即將到來的生日準備禮物，她決定好好看看這支手錶。她往頁面右下角看，注意到著名的iLink登入標

## 五、發明說明 ( 68 )

誌。圖27說明了較佳實施例中一個用戶查看一本包含一個iLink登入標誌的雜誌。這個iLink登入2700在雜誌廣告2710的右下角。

圖28說明了較佳實施例中一個用戶利用iLink設備閱讀條碼。用戶使用手持電腦，如個人數字助理(PDA)掃描iLink條碼2810，幾秒鐘之後設備彈出網路瀏覽器以Valery選擇的語言顯示OMEGA"速度主人"手錶。她還看到一條消息說如果她決定立刻在線上購買，她將獲得10%的折扣和免費送貨。還有更重要的是顯示屏告訴她手錶可以送貨並將在三天之內到達準時趕上她朋友的生日。

透過簡單的選擇buy! 按鈕，她買下了這支手錶。這次交易的有關資料自動的傳到她的設備上。移動資料iLink設計提供了紙張和電子媒體之間迄今為止未知的協同作用，這可協助用來吸引和保持顧客。利用他們的手持電腦，透過掃描一個iLink標記顧客可以取用詳細的、相關的、和及時的資訊—不論這個條碼是印在廣告裡、物品上、盒子上、海報上、傳單上、或在這產品上。電腦技術的一個一般熟練人員很容易的理解視覺特徵識別(OCR)設備可以替代條碼系統而不會破壞所聲明的發明。顧客可以使用他們熟悉的標準界面迅速得到動態、個性化的反饋，它透過網路推到顧客的瀏覽器。

如果一個用戶在有線雜誌上看到一個手錶廣告，相應的，有人在福布斯雜誌看到同樣的手錶廣告。即使廣告是同樣的，到達這兩個人的資訊是不同的。根據目標人群的

## 五、發明說明 ( 69 )

統計分析，對這個資訊進行格式處理並包含挑選出來用來吸引讀者的詞彙。iLink可以加入任何印刷物品當中。在商店，顧客可以掃描產品包裝上的代碼讀到更多有關產品的內容或看到競爭對手的資訊。使用網絡連接如網際網路—不論是從報紙或雜誌自己的站點還是從廣告商的網頁，其讀者都可以掃描頁面上的條碼，來獲得當前的詳細資訊。

廣告商可以得到目標讀者的更好的分佈圖，每個掃描條碼的顧客都會通知廣告商，在何時、何處、以及什麼時間一本出版物觸發了一個反應。PDA提供了你想要了解的資訊—不論何時何地你僅要掃描iLink。今天公司專注於合併或將傳統媒體的出版物擴大到電子媒體上。直到現在，大多數公司還沒有找到真正把兩種媒體結合起來的方法。iLink是代表如何利用現有技術的一個概念：

- 提供一個區別於競爭對手的服務
- 透過加強顧客的互動提高顧客關係，從而吸引新的顧客與保持老的顧客
- 降低印刷費用
- 提高市場知名度
- 降低由手冊翻譯和印刷延遲帶來的產品供貨延遲
- 剪裁資訊以適合顧客
- 以即時的最新的語言可選的資訊替代要印刷維護的手冊
- 在產品頂部加入建議

Valery坐在街邊咖啡屋，讀一本英語彩色雜誌

當她瀏覽時，她發現一個很好的OMEGA手錶的廣告。她

## 五、發明說明 ( 70 )

想起她的男朋友最近提起他想要一支新手錶。因為她還沒有為他即將到來的生日準備禮物，她決定好好看看這支手錶。她往頁面右下角看，注意到著名的iLink登入標誌。

立刻，她取出她的iLink設備，掃描iLink條碼，幾秒鐘之後設備彈出網路瀏覽器以Valery選擇的語言顯示OMEGA"速度主人"手錶。她還看到一條消息說如果她決定立刻在線上購買，她將獲得10%的折扣和免費送貨。還有更重要的是顯示屏告訴她手錶可以送貨並將在三天之內到達一準時趕上她朋友的生日。

透過簡單的選擇buy! 按鈕，她買下了這支手錶。這次交易的有關資料自動的傳到她的設備上。移動資料iLink設計提供了紙張和電子媒體之間迄今為止未知的協同作用，這可協助用來吸引和保持顧客。利用他們的手持電腦，透過掃描一個iLink標記顧客可以取用詳細的，相關的和及時的資訊—不應這個條碼是印在廣告裡、物品上、盒子上、海報上、傳單上、或這在產品上。電腦技術的一個一般熟練人員很容易理解視覺特徵識別(OCR)設備可以替代條碼系統而不破壞所聲明的發明。顧客可以使用他們熟悉的標準界面迅速得到動態，個性化的反饋，它透過網路推到顧客的瀏覽器。

一個目標資訊的例子可以是，如果一個用戶在有線雜誌上看到一個手錶廣告，相應的，有人在福布斯雜誌看到同樣的手錶廣告。即使廣告是同樣的，到達這兩個人的資訊也是不同的。

## 五、發明說明 ( 71 )

一個 iLink 登入可以加入任何印刷物品當中。在商店，顧客可以掃描產品包裝上的代碼讀到更多有關產品的內容或看到競爭對手的資訊。報紙或雜誌的讀者可以掃描頁面上的條碼，來透過網路獲得最新的或有深度的資訊—不論從報紙或雜誌自己的站點還是從廣告商的網頁。廣告商可以得到目標讀者的更好的分佈圖，每個掃描條碼的顧客都會通知廣告商，在何時、何處、以及什麼時間一本出版物觸發了一個反應。

PDA 提供了你想要了解的資訊—不論何時何地你僅要掃描 iLink。今天公司專注於合併或將傳統媒體的出版物擴大到電子媒體上。直到現在，大多數公司還沒有找到真正把兩種媒體結合起來的方法。iLink 將 novel 技術用於：

- 提供一個區別於競爭對手的服務
- 透過加強顧客的互動提高顧客關係，從而吸引新的顧客與保持老的顧客
- 降低印刷費用
- 提高市場知名度
- 降低由手冊翻譯和印刷延遲帶來的產品供貨延遲
- 剪裁資訊以適合顧客
- 以即時的最新的語言可選的資訊替代要印刷維護的手冊
- 在產品頂部加入建議

### 依照較佳實施例的詳細說明

iLink 設備採用一些新的條碼內容 ( IP 地址 + 記錄條碼的介資訊 ( metainformation ) )、條碼掃描器、PDA ( 掌上導航

## 五、發明說明 ( 72 )

器)、網路瀏覽器、無線TCP/IP連接。這些技術今天已經存在，但是是以分立的形式，我們把它們整合到我們的工作原型中。今天已經有這四種技術的結合(條碼、條碼閱讀器、PDA掌上導航、和無線資料)，但是在條碼中加上IP地址和介資訊的這五種技術的獨特的應用，以及個人概貌資訊和網路瀏覽器的整合創造了一個獨特的全新的能力。

本發明的較佳實施例使用軟體(我們用C代碼開發)取得IP地址和附加的介資訊(來自符號設備的整合條碼閱讀器)，以及由iLink設備的消費者/用戶所配置的本地iLink申請的個人概貌資訊，並從傳來的字串中分析出IP地址，設計一個IP地址和全局資源鏈接(URL)/路徑，然後送往本地網路瀏覽器，該瀏覽器連接在網際網路(或內部網路)的伺服器上，接著瀏覽器譯出URL/路徑，向iLink設備發回獨一無二的資訊，該資訊的目標是顧客的概貌，掃描的廣告，掃描的雜誌和其它相關資訊。

如果用戶在瀏覽關於影印機的服務手冊，試圖找到相關的資訊來修復機器，他發現說明對於實際產品已經過時，並且因為英語不是這個用戶的母語，文件檔的一些詞和表達容易混淆，那麼用戶就可以拿起一個iLink設備，把它指向影印機的iLink登入，點選iLink按鈕。屏幕上一個標準的網路瀏覽器以用戶的母語顯示了一個網頁，提供了你的印刷手冊中沒有的最新資訊，並建議與你準備修理的型號機器的專家電話聯繫。

圖29到37說明了較佳實施例的帶屏幕顯示的手持電腦。

## 五、發明說明 ( 73 )

用戶與顯示屏進行互動，選擇可掃描標記並將其輸入網路立即查詢以便說明及／或購買目標項目。圖38說明了較佳實施例的體系結構和流程。用戶透過將一個條碼閱讀器指向掌上導航器或其它電腦設備3320從產品中讀取iCode，可以從一系列產品中挑選一件產品3310。iCode用來準備透過TCP/IP或其它通信協議3330向網際網路3350遞交的查詢，以獲得第三方網站3340的資訊。這個資訊返回用戶的電腦設備，並提供用戶說明的資訊，或允許用戶購買目標項目。

以上所述者，僅為了用於方便說明本發明之較佳實施例，而並非將本發明狹義地限制於該較佳實施例。凡依本發明所做的任何變更，皆屬本發明申請專利之範圍。

## 圖式之簡單說明

本發明之上述及其他目的、優點和特色由較佳實施例之詳細說明中並參考圖式當可更加明白，其中：

圖1是依照較佳實施例之代表性硬體環境的方塊圖；

圖2是依照較佳實施例之系統的流程圖；

圖3是依照較佳實施例之系統的語法分析模組的流程圖；

圖4是依照較佳實施例之模式匹配的流程圖；

圖5是依照較佳實施例之搜索單元的流程圖；

圖6是依照較佳實施例之整個系統過程的流程圖；

圖7是依照較佳實施例之主題處理的流程圖；

圖8是依照較佳實施例之會議記錄處理的流程圖；

圖9是依照較佳實施例之袖珍交易檢索流程的方塊圖；

## 五、發明說明 ( 74 )

圖 10A 和 10B 是說明依照較佳實施例之創建顧客化內容網頁的邏輯過程的方塊圖和流程圖；

圖 11 是說明依照較佳實施例之與檢索用戶中心內容相關的詳細邏輯的流程圖；

圖 12 是依照較佳實施例之用戶概貌的資料模型；

圖 13 是依照較佳實施例之角色資料模型；

圖 14 是依照較佳實施例之購買意圖資料模型；

圖 15 是依照較佳實施例之產生代理人當前統計資料過程的流程圖；

圖 16 是依照較佳實施例之決定用戶的個性化產品等級的邏輯的流程圖；

圖 17 是依照較佳實施例之取用中心儲存資料庫的邏輯的流程圖；

圖 18 是依照較佳實施例之用戶與一個特定供應商整合器的互動邏輯的流程圖；

圖 19 是依照較佳實施例之產生口頭摘要的代理人得流程圖；

圖 20 說明依照較佳實施例之登入畫面顯示；

圖 21 說明依照較佳實施例的日常後勤管理顯示；

圖 22 說明依照較佳實施例的用戶主顯示畫面；

圖 23 說明依照較佳實施例的代理人互動作用顯示；

圖 24 是依照較佳實施例之活動知識管理系統的方塊圖；

圖 25 是依照較佳實施例之後備伺服器的方塊圖；

圖 26 是依照較佳實施例的魔術牆的方塊圖；與

## 五、發明說明 ( 75 )

圖27說明較佳實施例中用戶閱讀具有Icode的雜誌；

圖28說明較佳實施例中用戶掃描一個目標項目標記；

圖29到37說明較佳實施例中具有屏幕顯示手持電腦；和

圖38說明較佳實施例的結構和流程。

## 符號說明

- |      |               |
|------|---------------|
| 110  | 中央處理機         |
| 112  | 匯流排           |
| 114  | 隨機存取記憶體 (RAM) |
| 116  | 唯讀記憶體 (ROM)   |
| 118  | 輸入／輸出配接器      |
| 120  | 磁碟儲存單元        |
| 122  | 用戶界面配接器       |
| 124  | 鍵盤            |
| 126  | 滑鼠            |
| 128  | 揚聲器           |
| 132  | 擴音器           |
| 1030 | 目的資料庫         |
| 1040 | 內容資料庫         |
| 1050 | 供貨商概貌資料庫      |
| 1060 | 用戶概貌資料庫       |
| 1080 | 產品訊息資料庫       |
| 1090 | 產品訂貨資料庫       |
| 1091 | 用戶訊息資料庫       |
| 1070 | 供貨商的WEB伺服器    |

## 五、發明說明 ( 76 )

- 1010 用戶的WEB瀏覽器
- 1020 綜合器的WEB伺服器
- 2400 伺服器
- 2410, 2420 魔術牆
- 2430, 2436 感知機器
- 2446, 2448 Web客戶機
- 2450, 2460 電子郵件客戶機
- 2500, 2502, 2504 智能代理
- 2510 企業內部網 (Intranet)
- 2580 智能代理協調器 (Munin)
- 2590 Munin資料庫
- 2570 通信傳輸輔助
- 2550 電子郵件
- 2560 蓮花公司群件系列 (Lotus Notes)
- 2530 訊息優先排序子系統
- 2540 訊息通道資料庫
- 2542 用戶概貌資料庫
- 2610 擴音器
- 2614 話筒
- 2620 觸摸屏顯示
- 2630, 2632, 2634 感應器
- 2640 電腦

四、中文發明摘要(發明之名稱：利用網路技術進行高級資訊收集之系統、方法和物品)

一種基於代理人的系統，協助從一篇感興趣的文章中獲取資訊，並根據目標文章的資訊，利用這個資訊接受用戶的直接行動。系統從包含會議時間的一個日曆程式中，以字符(character)形式獲得象徵目標會議的輸入正文。當會議時間到來時，透過查詢日曆程式獲得目標事件的正文，並將該資訊用作代理人系統的輸入。然後，代理人系統分析輸入會議正文，摘取它的不同的元件諸如名稱、內容、與會者、地點、時間等等。系統還執行模式匹配以辨別特定的相應資訊。這一資訊用來查詢網路上不同的資訊源，獲得當前會議相關的說明，發回日曆程式。

英文發明摘要(發明之名稱：A SYSTEM, METHOD AND ARTICLE OF MANUFACTURE FOR ADVANCED INFORMATION GATHERING UTILIZING WEB TECHNOLOGY)

An agent based system assists in obtaining information from an article of interest and utilizes the information to take user directed action based on the information from the target article. The system obtains input text in character form indicative of the target meeting from the a calendar program that includes the time of the meeting. As the time of the meeting approaches, the calendar program is queried to obtain the text of the target event and that information is utilized as input to the agent system. Then, the agent system parses the input meeting text to extract its various components such as title, body, participants, location, time etc. The system also performs pattern matching to identify particular appropriate information. This information is utilized to query various sources of information on the web and obtain relevant stories about the current meeting to send back to the calendaring system.

圖式

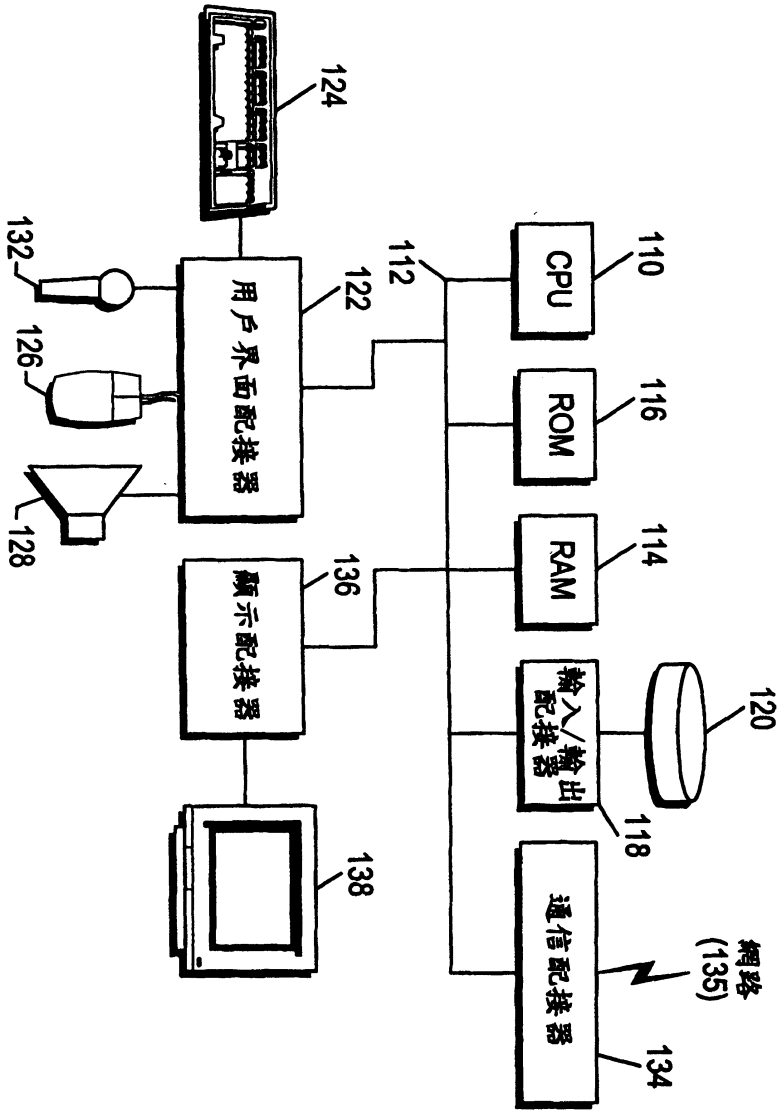


圖 1

圖式

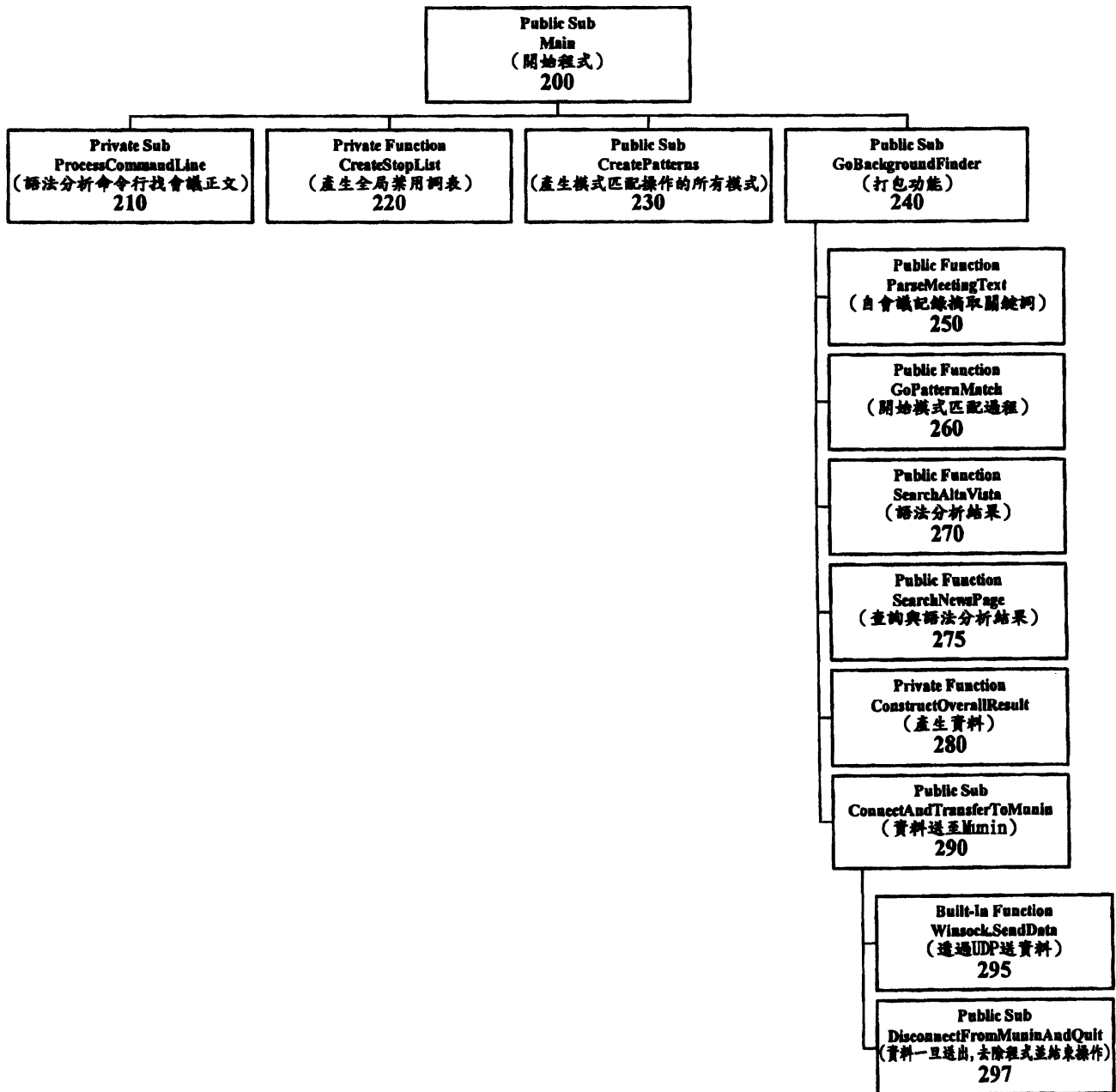


圖 2

圖式

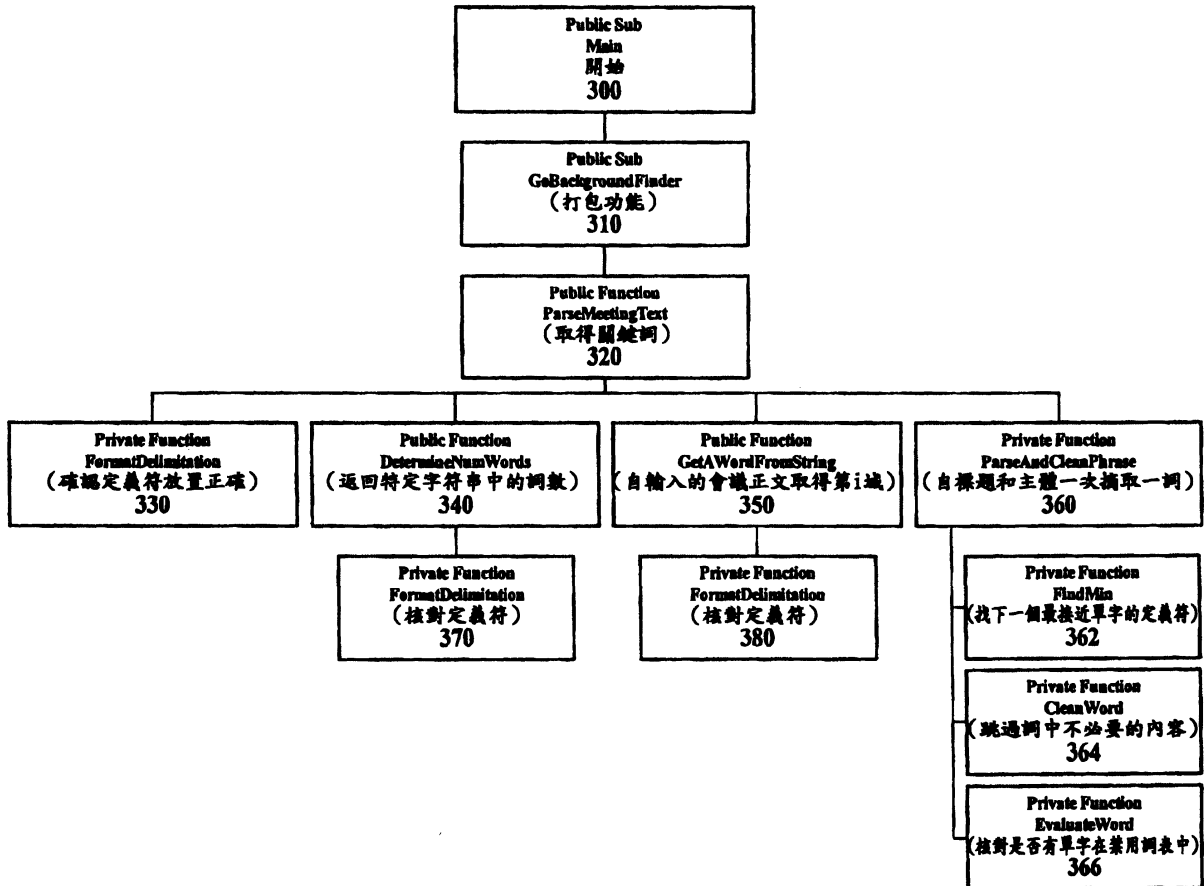


圖 3

圖式

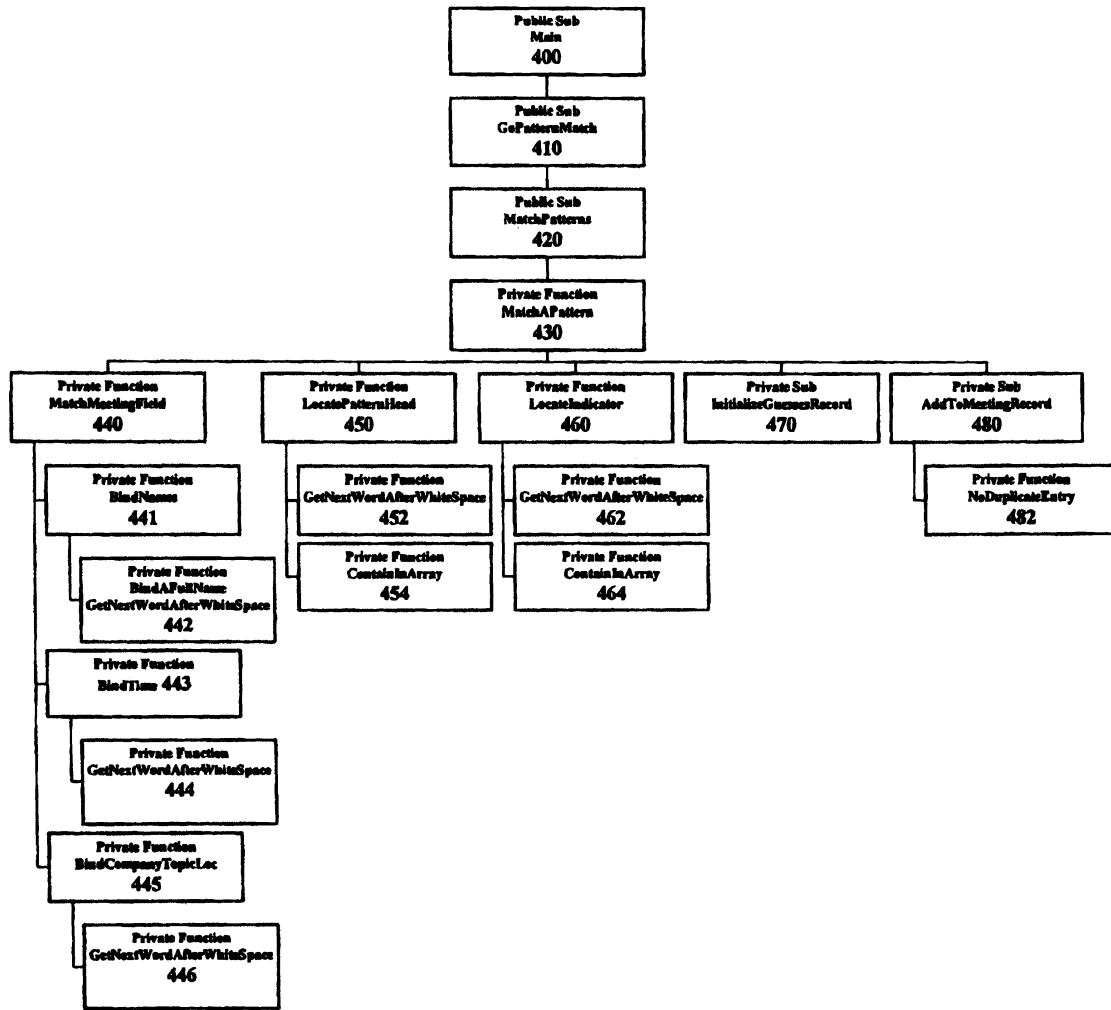


圖 4

圖式

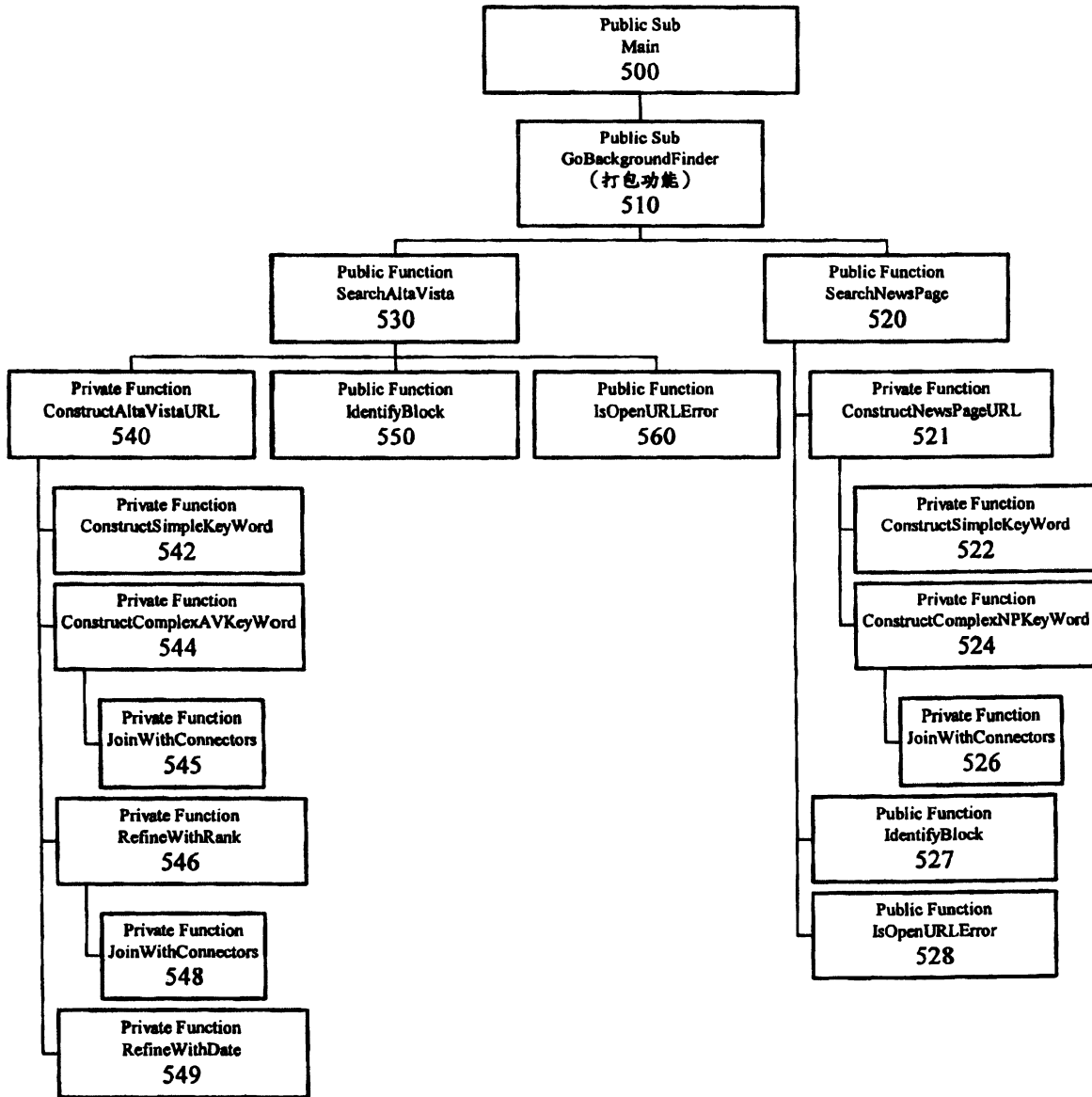


圖 5

圖式

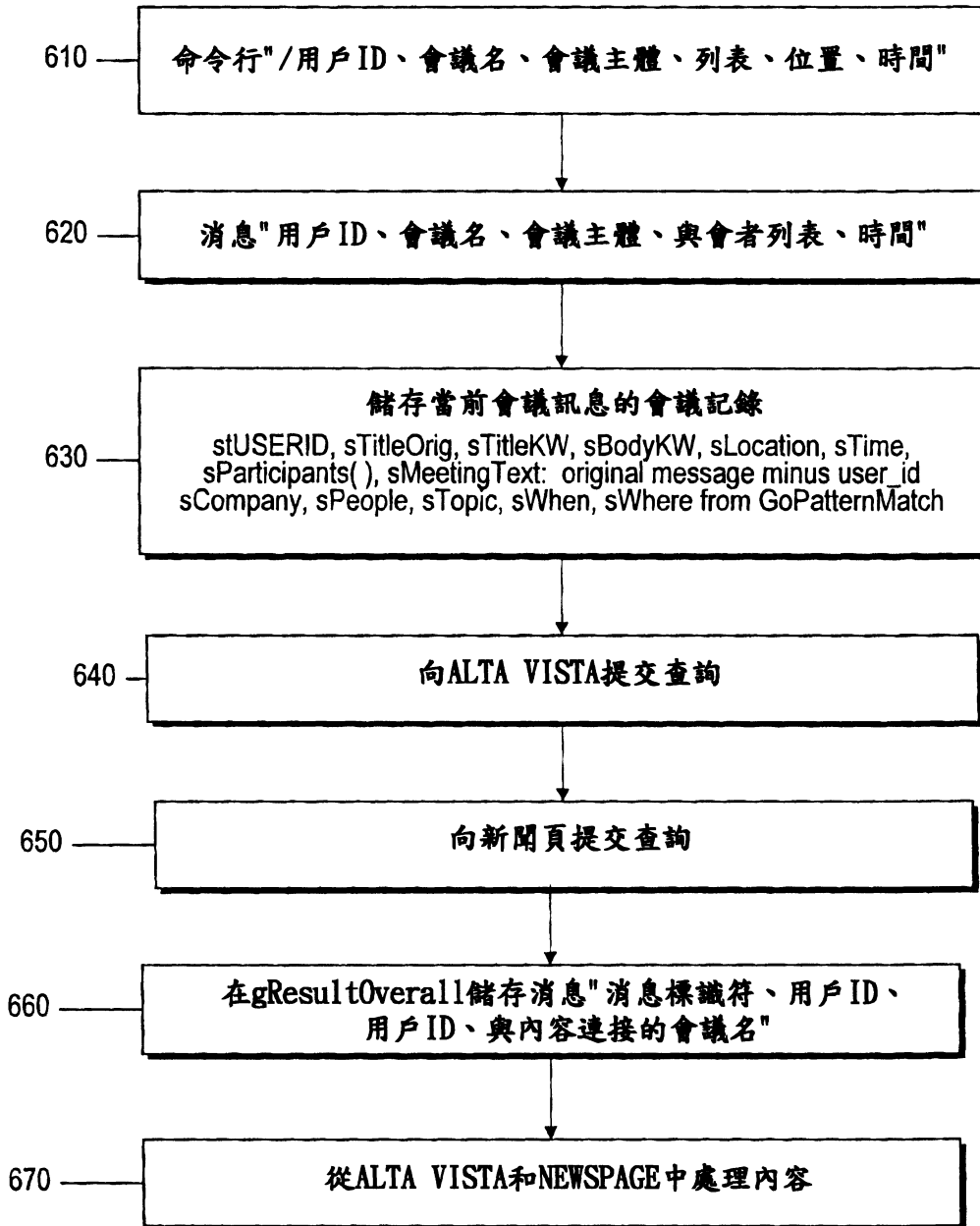


圖 6

圖式

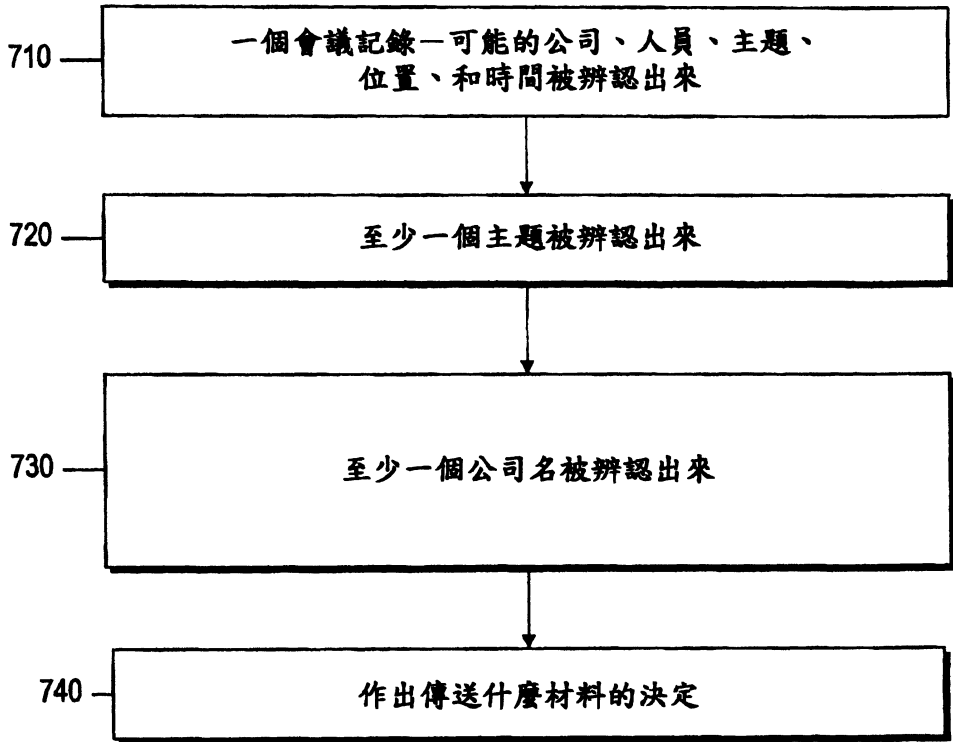


圖 7

圖式

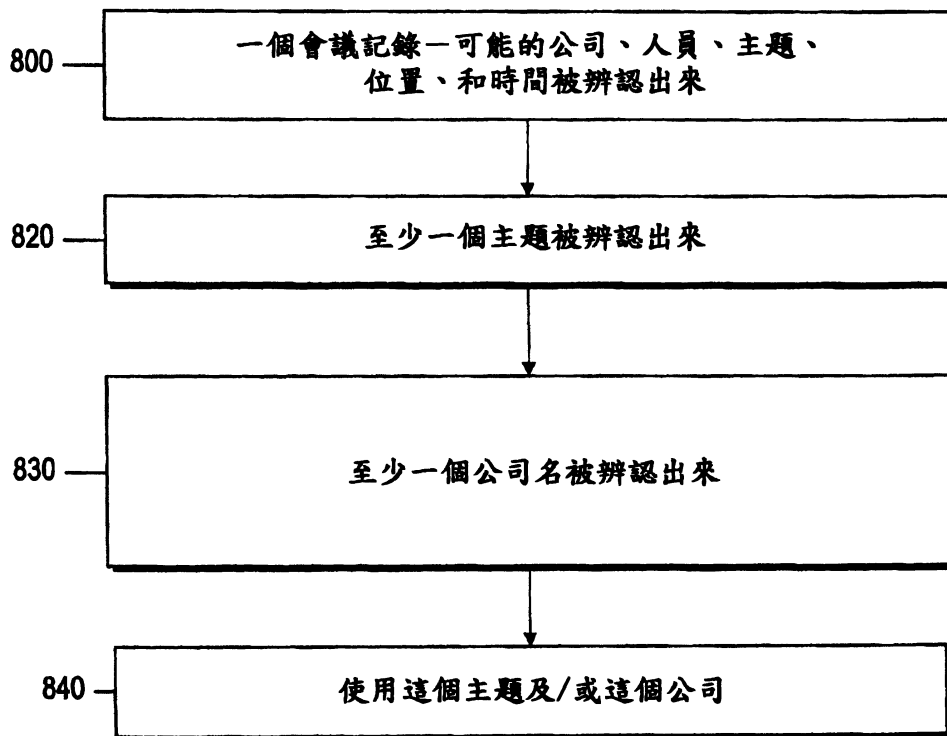


圖 8

圖式

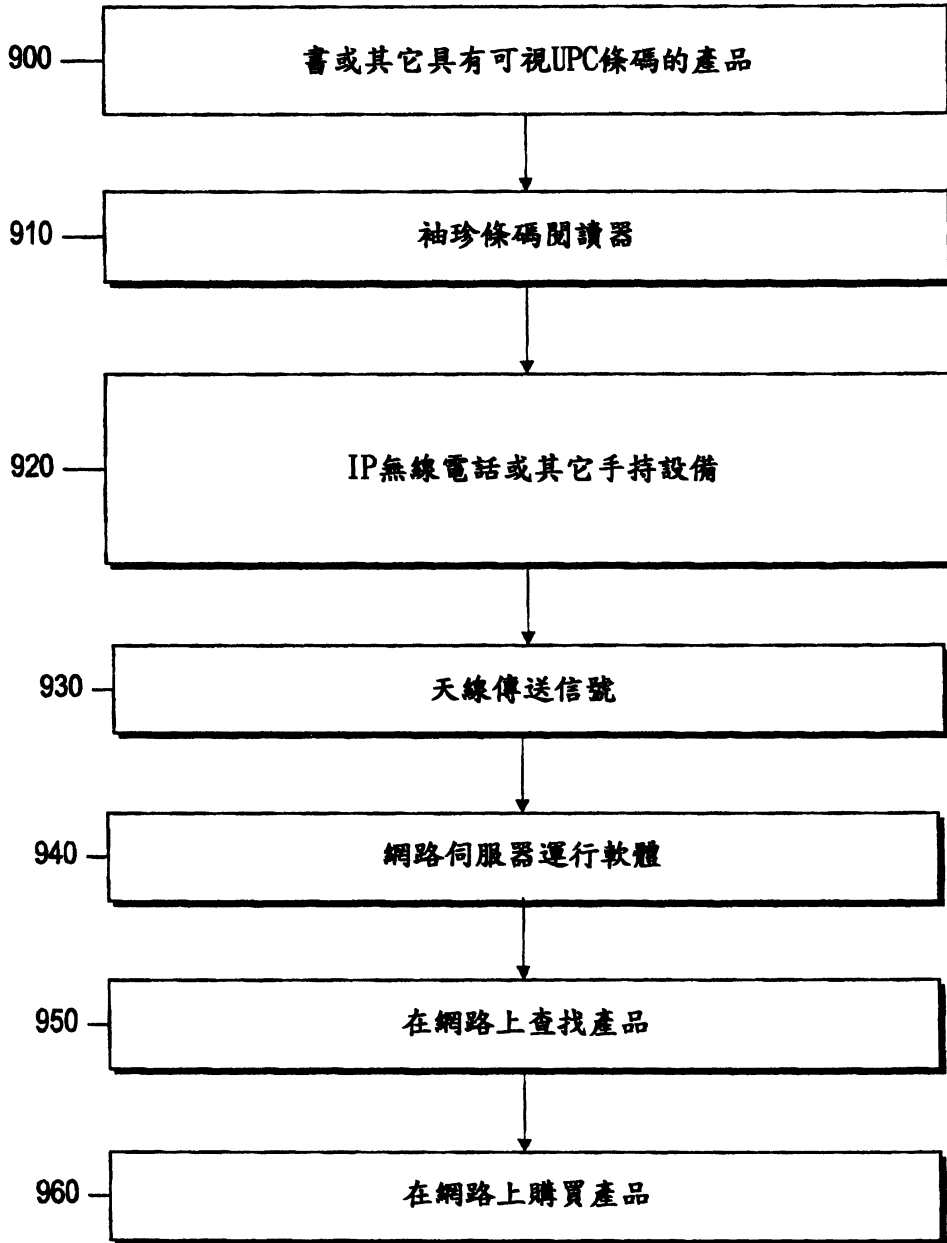
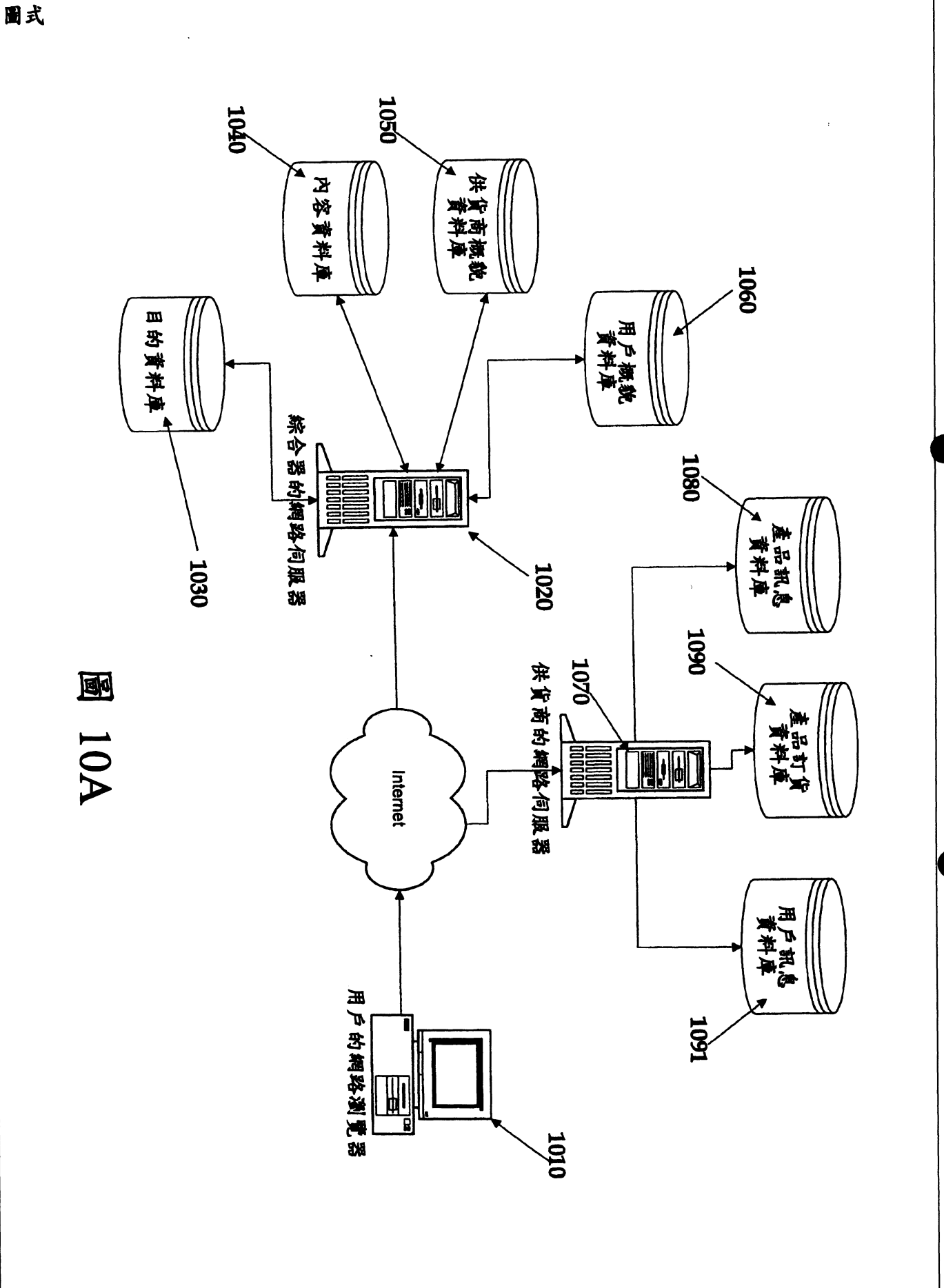


圖 9



圖式

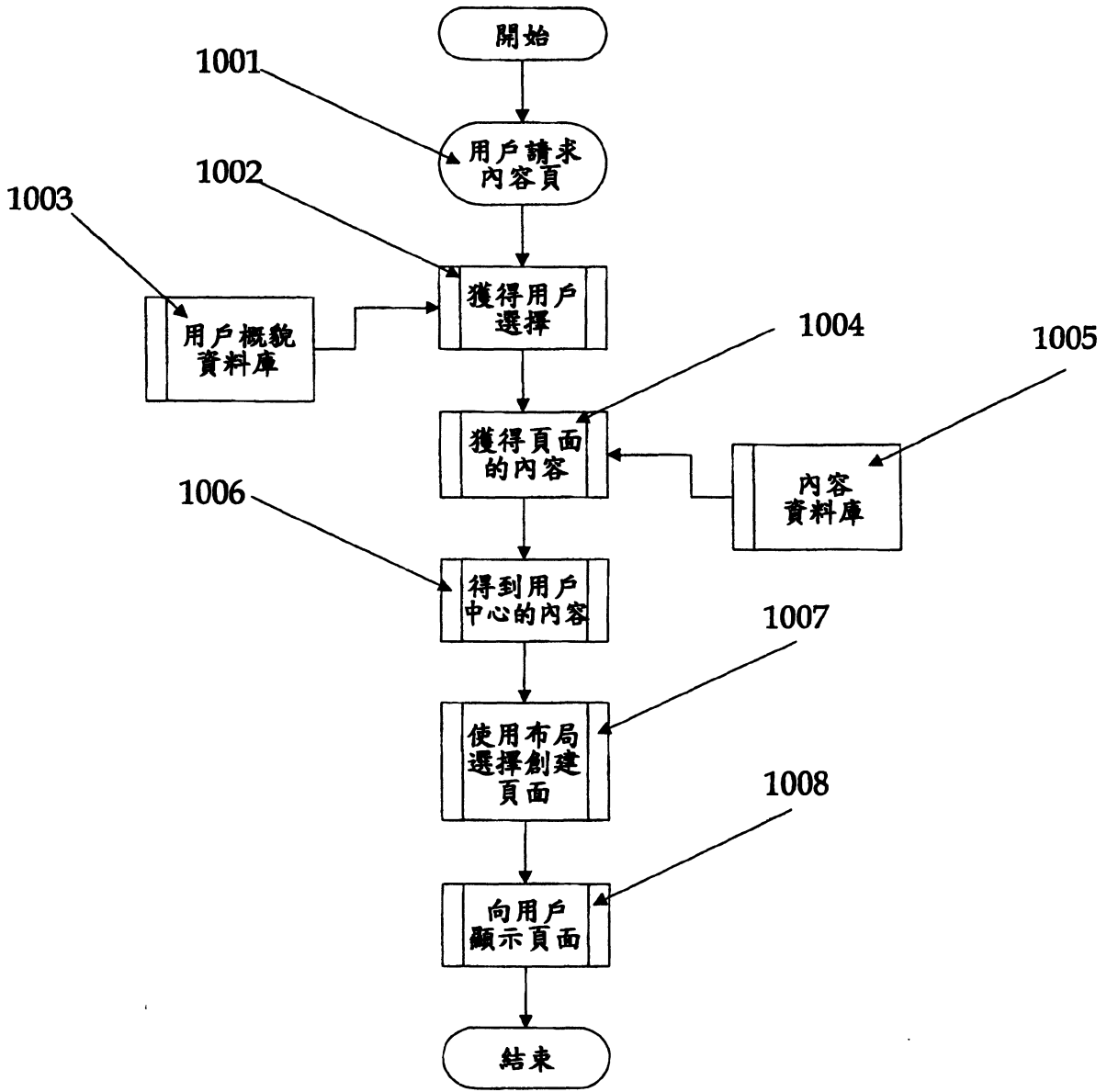


圖 10B

圖式

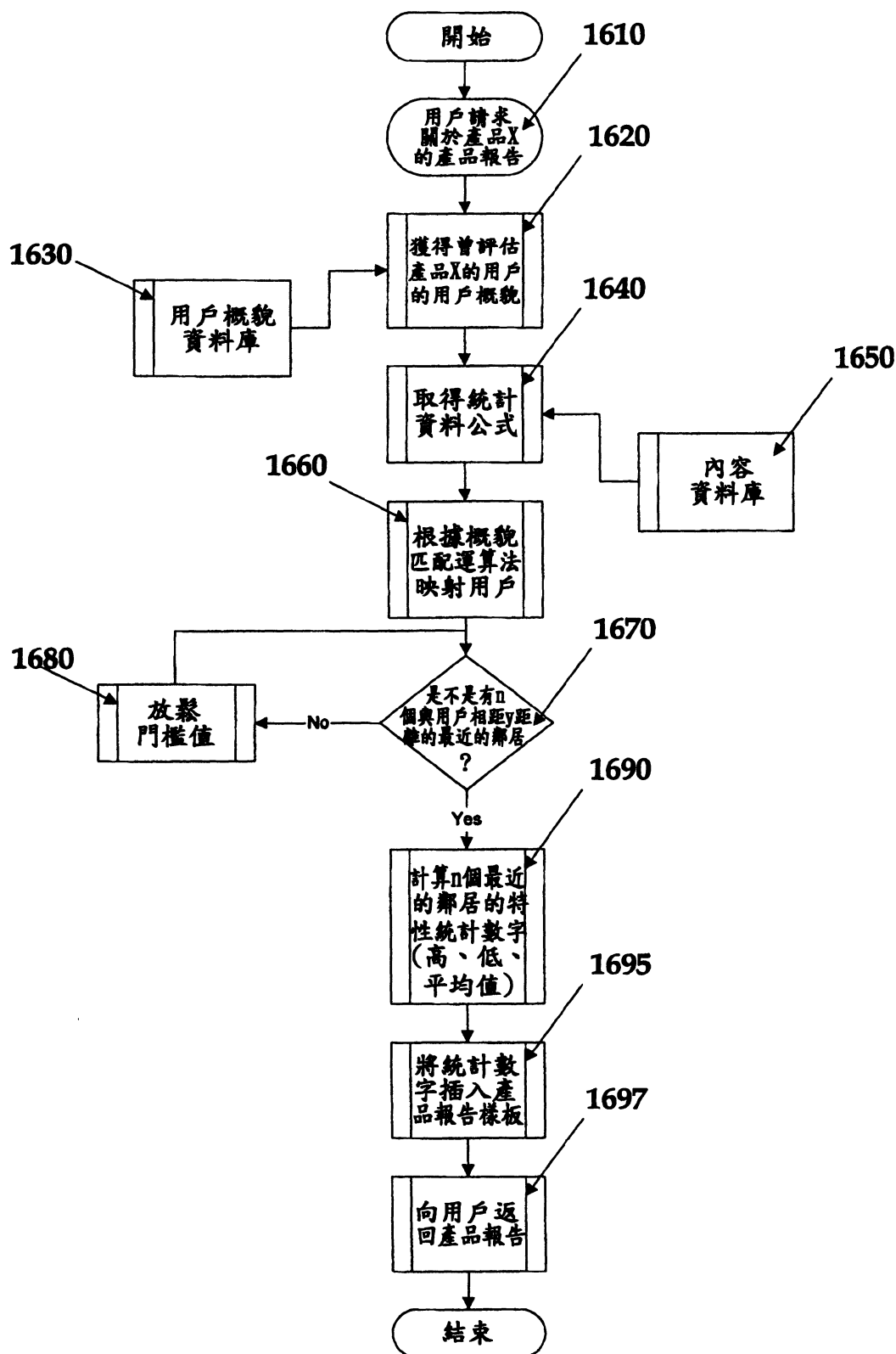


圖 16

圖式

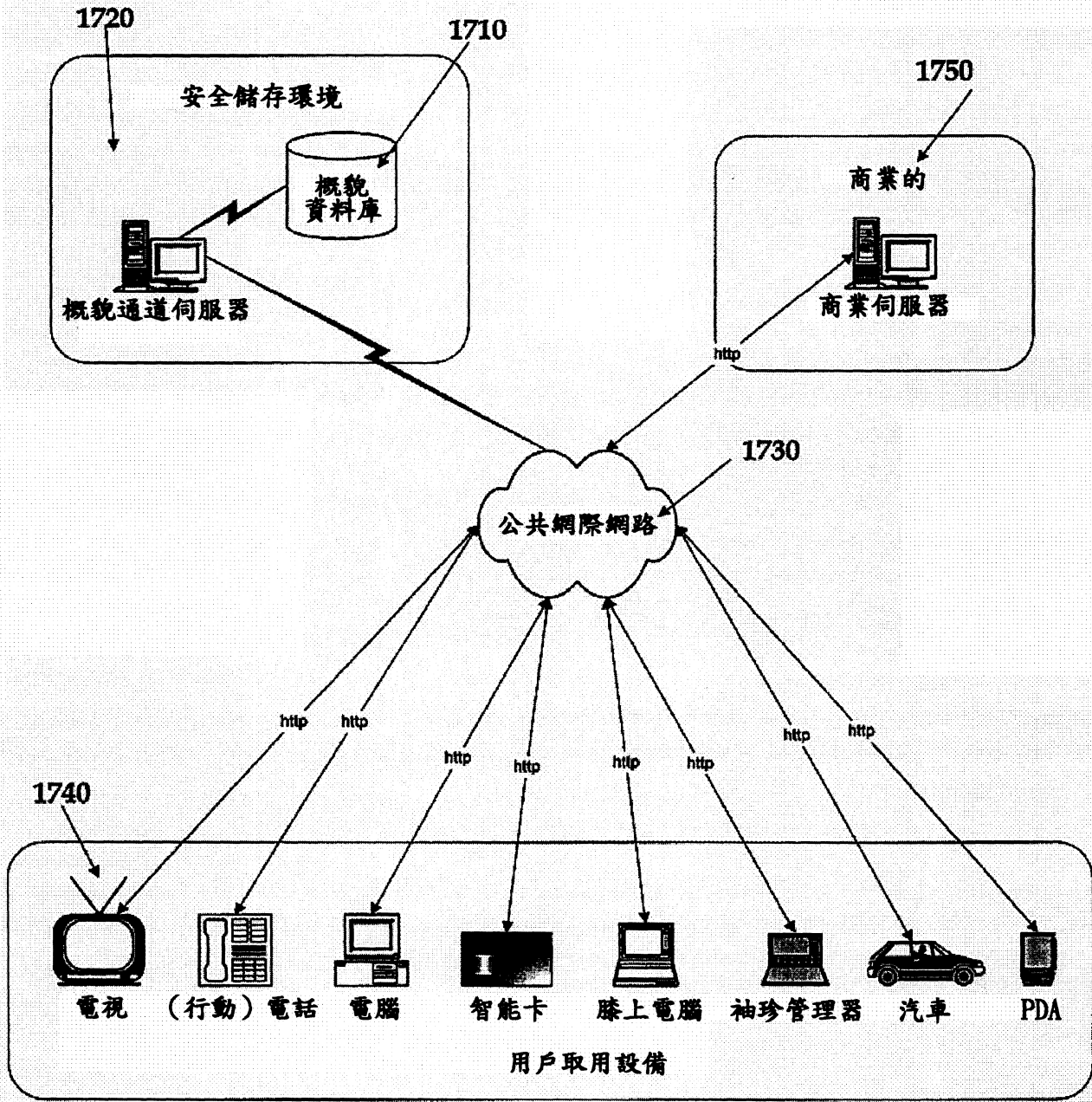


圖 17

圖式

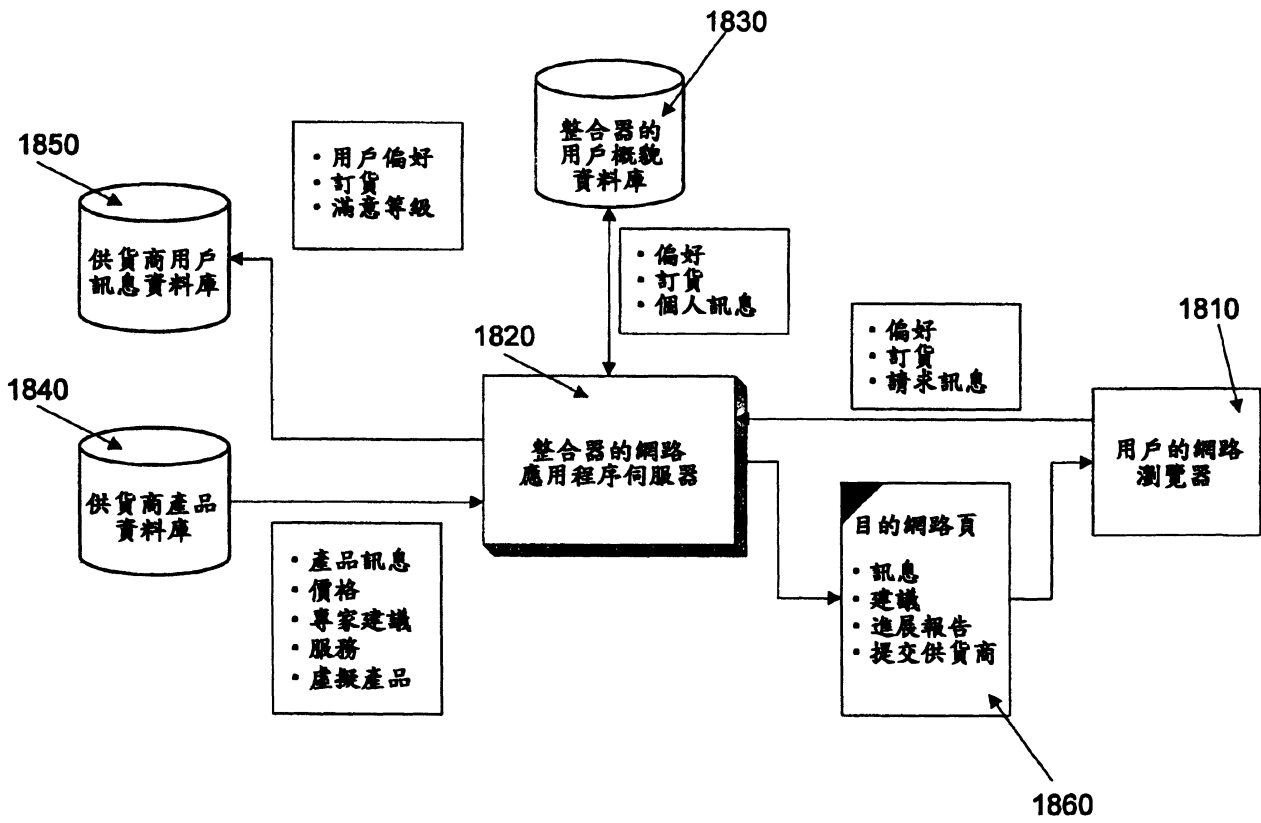


圖 18

圖式

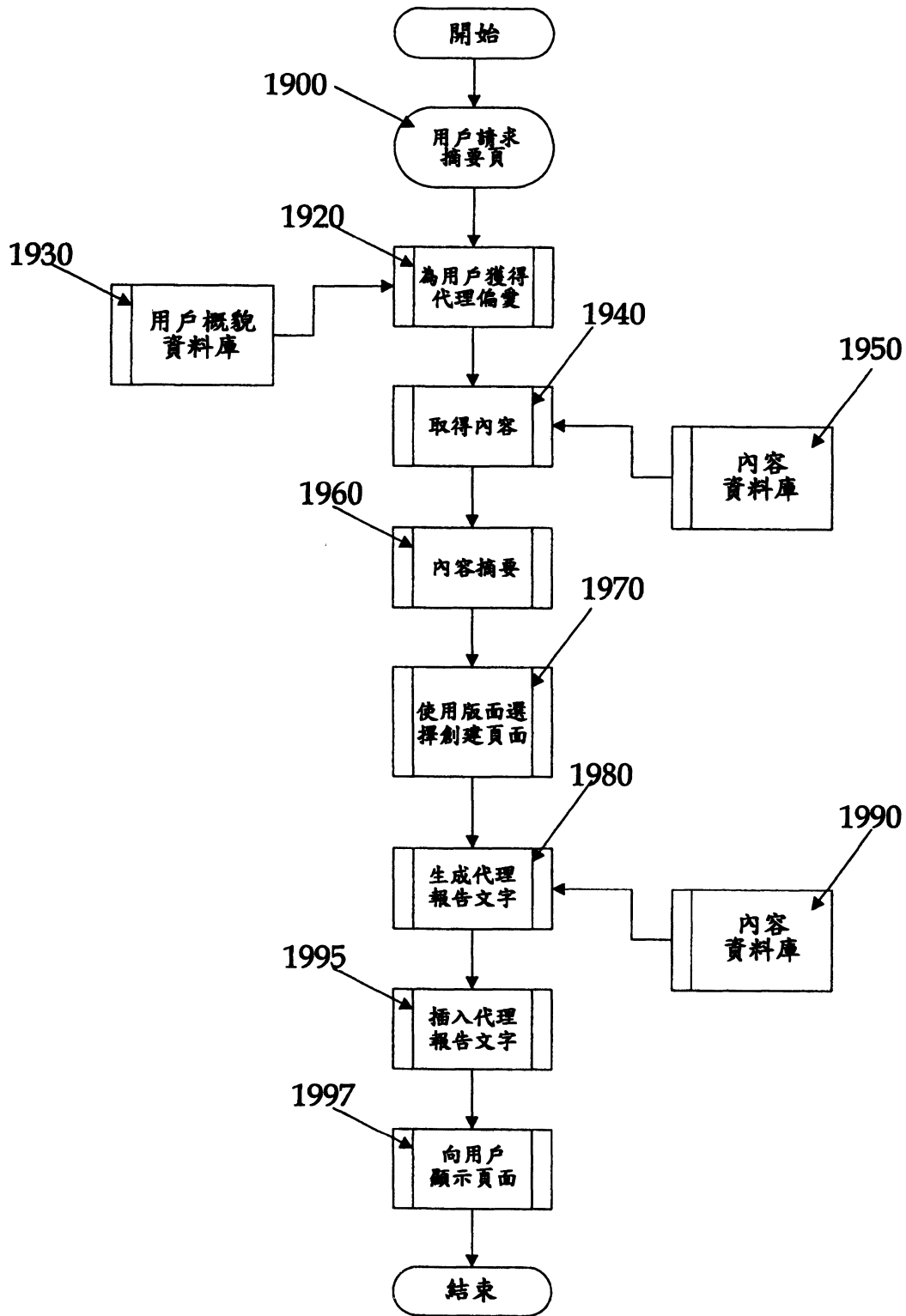


圖 19

圖式

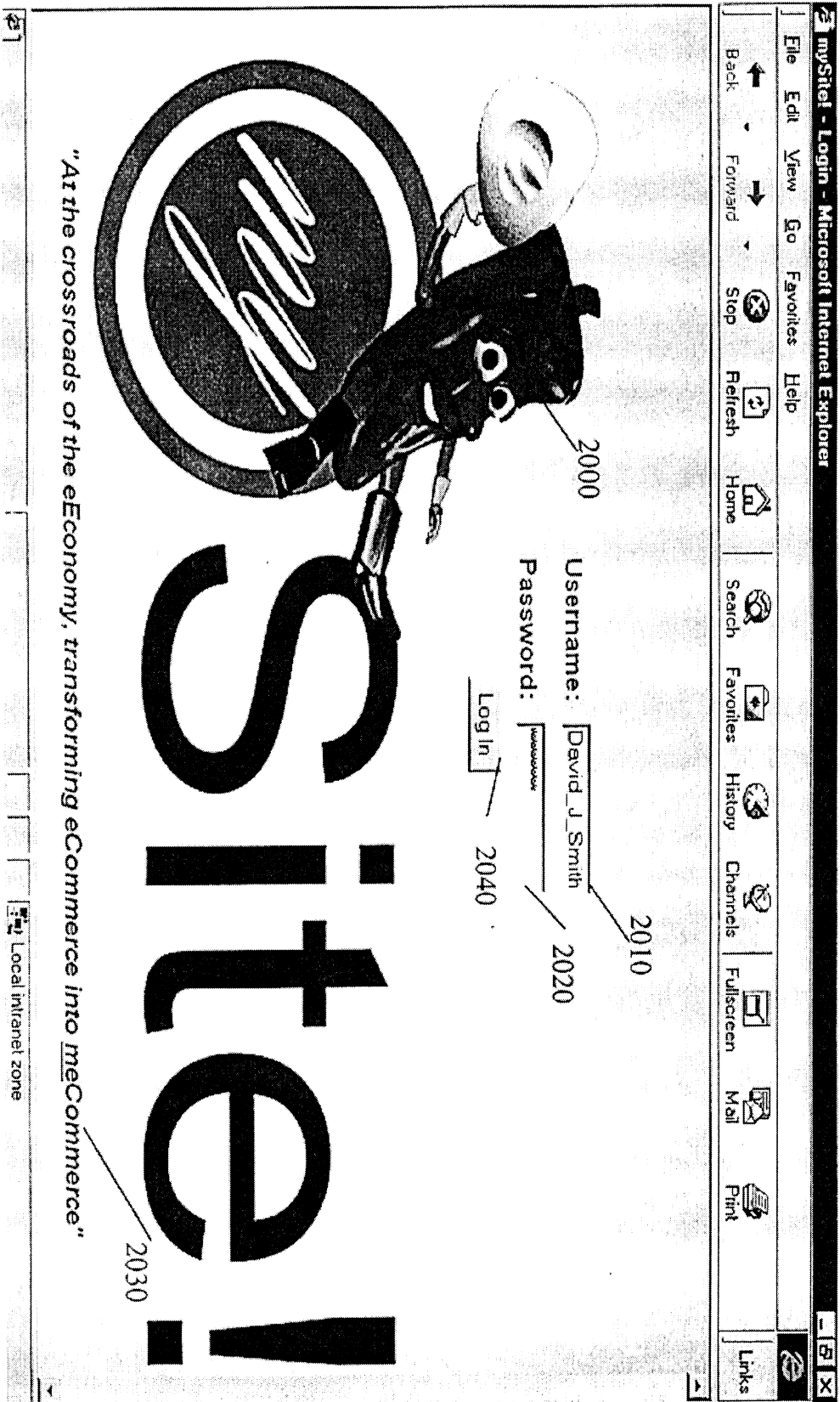


圖 20



圖式

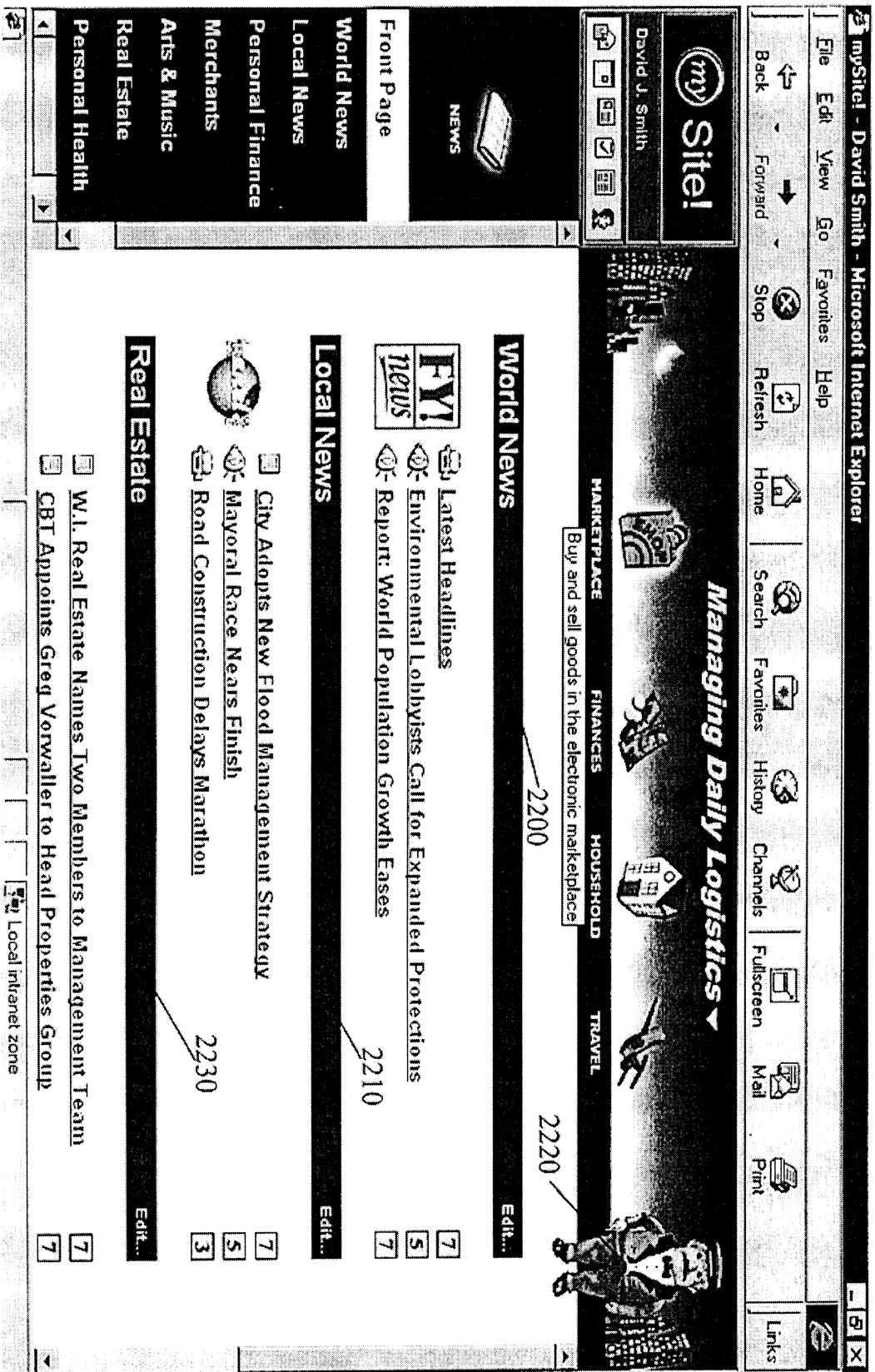


圖 22

myStiel - David Smith - Microsoft Internet Explorer

File Edit View Go Favorites Help

Back Forward Stop Refresh Home Search Favorites History Channels Fullscreen

http://www.mystiel.com

my Site!

David J. Smith  
FINANCIAL PLANNER

FINANCIAL PLANNER

CONQUESTOR INSURANCE

**Your Life Insurance at a Glance**

Policy Details

Policy Type: Whole Life  
 Policy Number: LP 79851-2  
 Monthly Premium: \$45  
 Total Benefit: \$750K

Coverage Calculation

Long Term Household Needs	\$890,000
Household Assets	\$750,000
Coverage Requirement	No coverage needed. (\$60,000)

Life Insurance Needs

Dollars (in thousands)

Age	Current Coverage Level	Recommended Coverage Level
27	~100	~100
31	~100	~100
35	~100	~100
39	~100	~100
43	~100	~100
47	~100	~100

Change Policy

Overview Auto Home Health Life

Local intranet zone

Life insurance needs have changed. The chart

2300

2310

2360

2395

Tips for You

You are over-insured!  
 With recent life events your life insurance needs have changed. Your household now has enough assets to manage should there be an unexpected death in the family. Press the "Change Policy" button to make adjustments or cancel your policy.

2320

2330

2340

2350

HEALTH WATCH

RETIREMENT GOALS

ESTATE PLANNING

LIFESTYLE

2370

2380

2390

圖 23

圖式

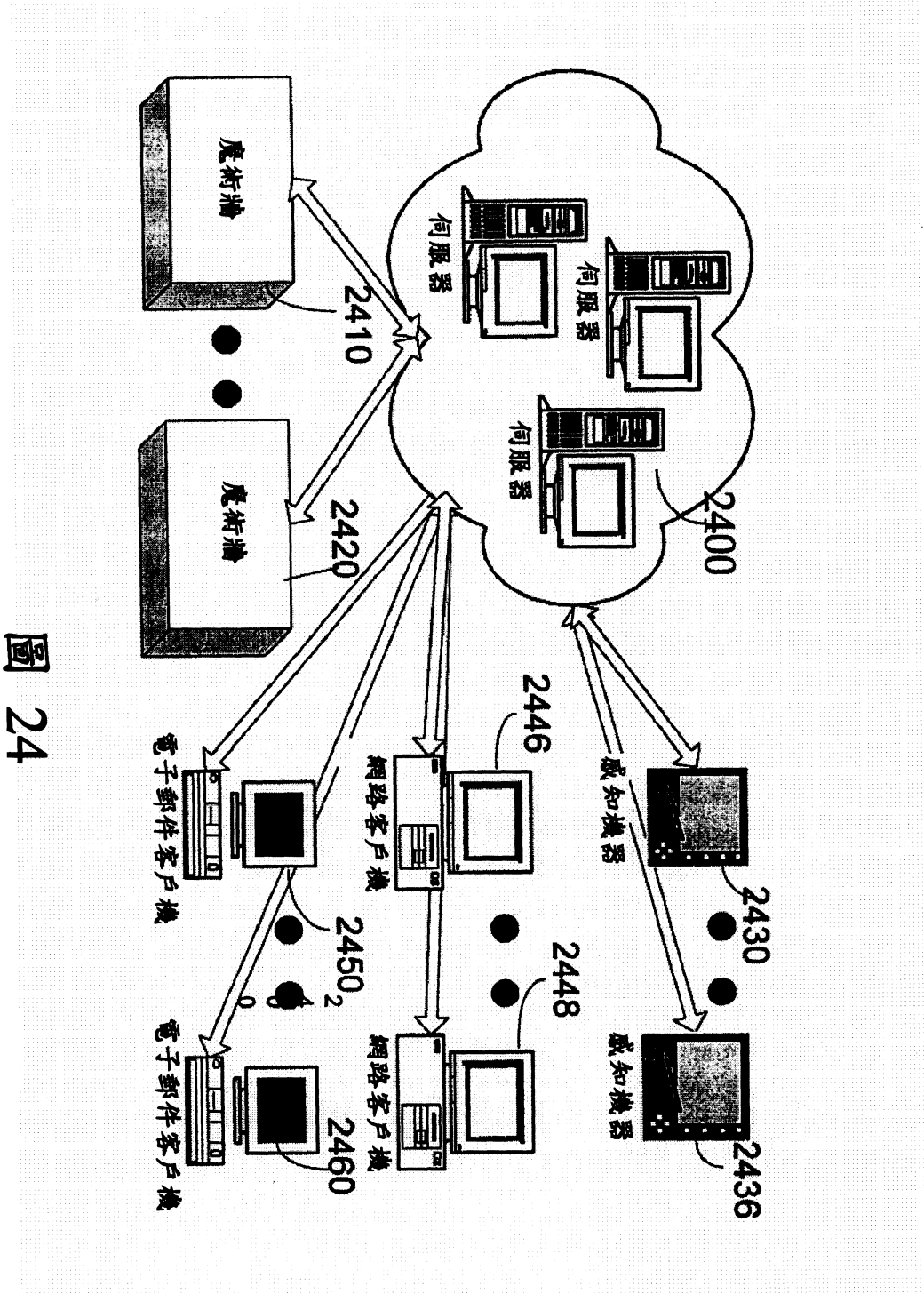


圖 24

圖式

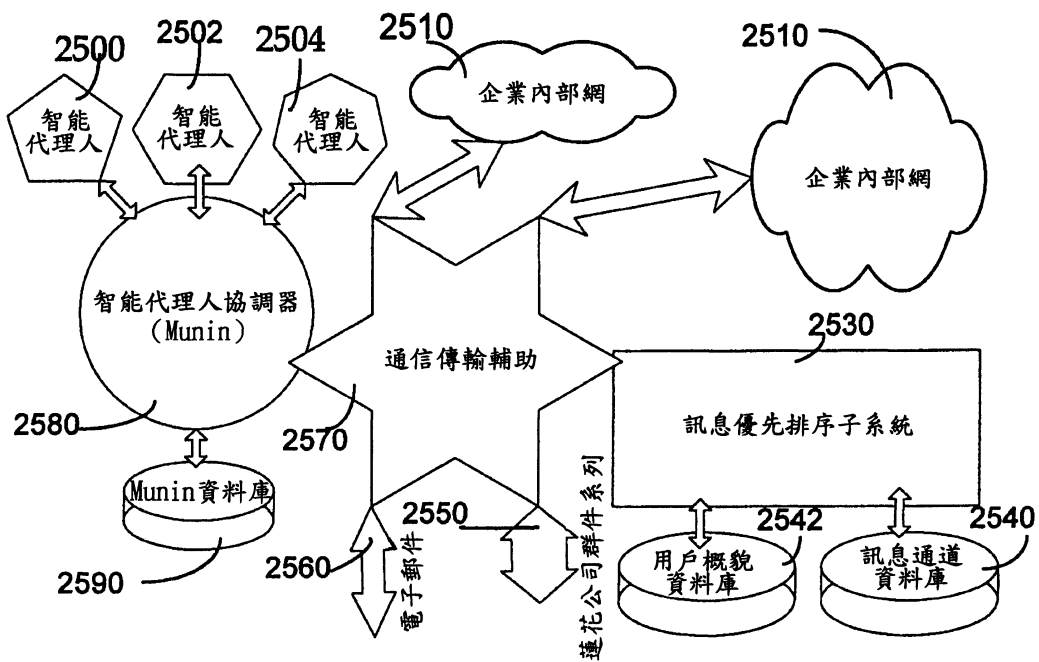


圖 25

圖式

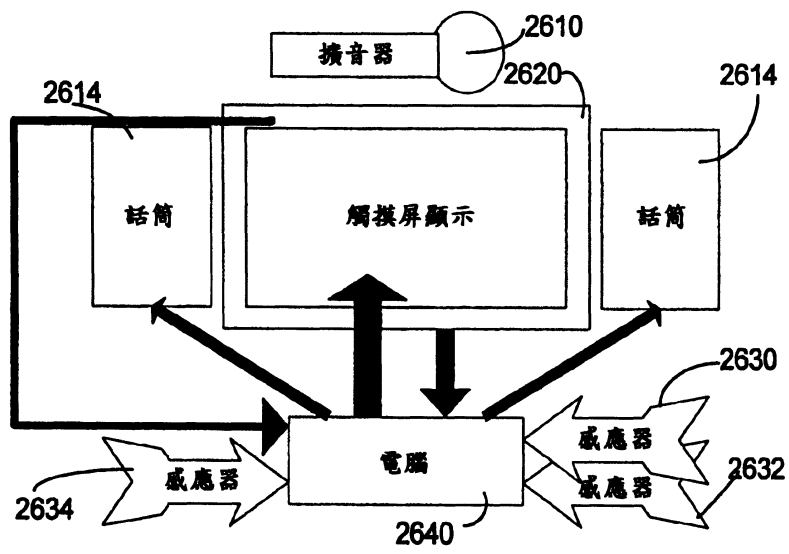


圖 26

圖式

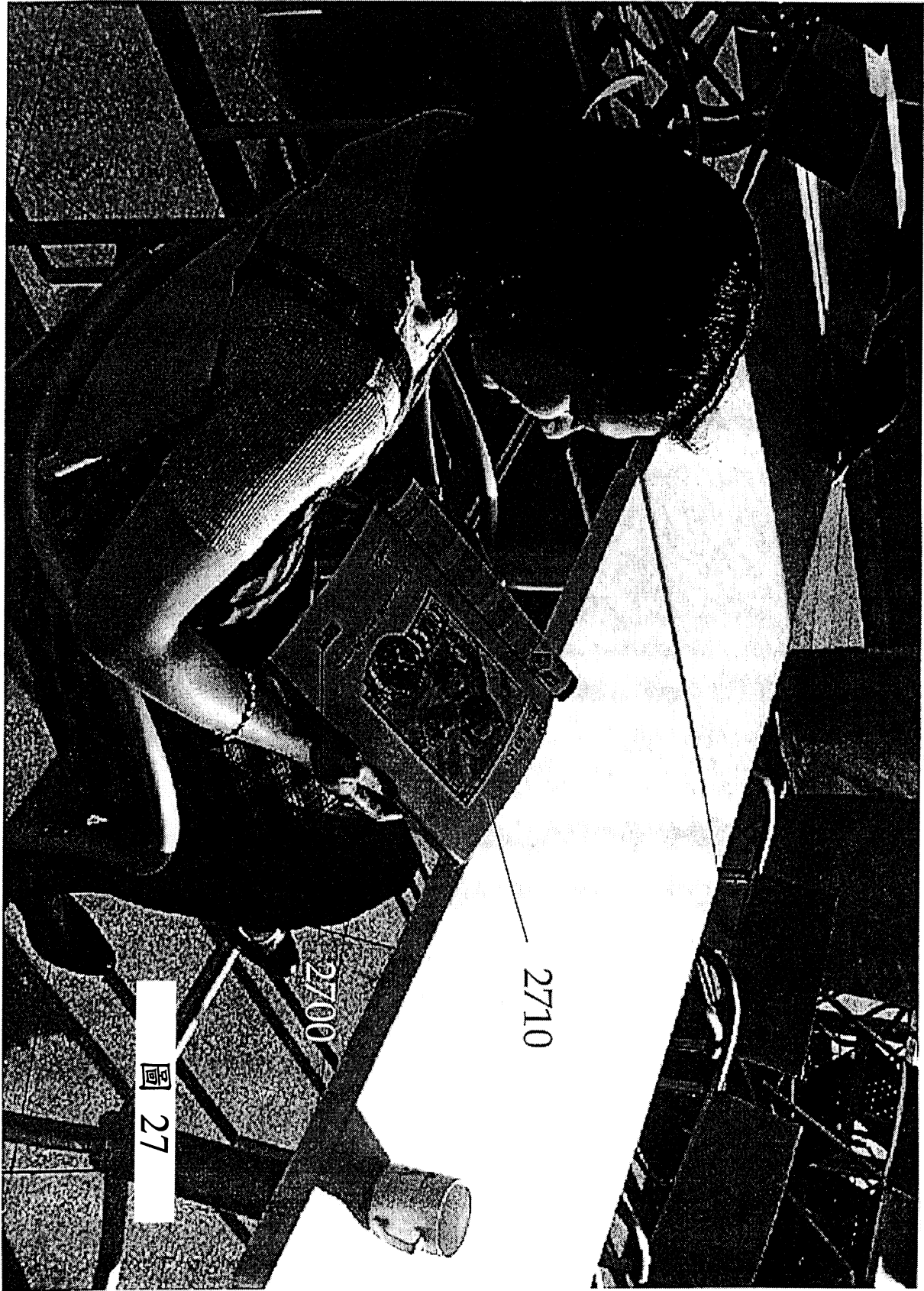
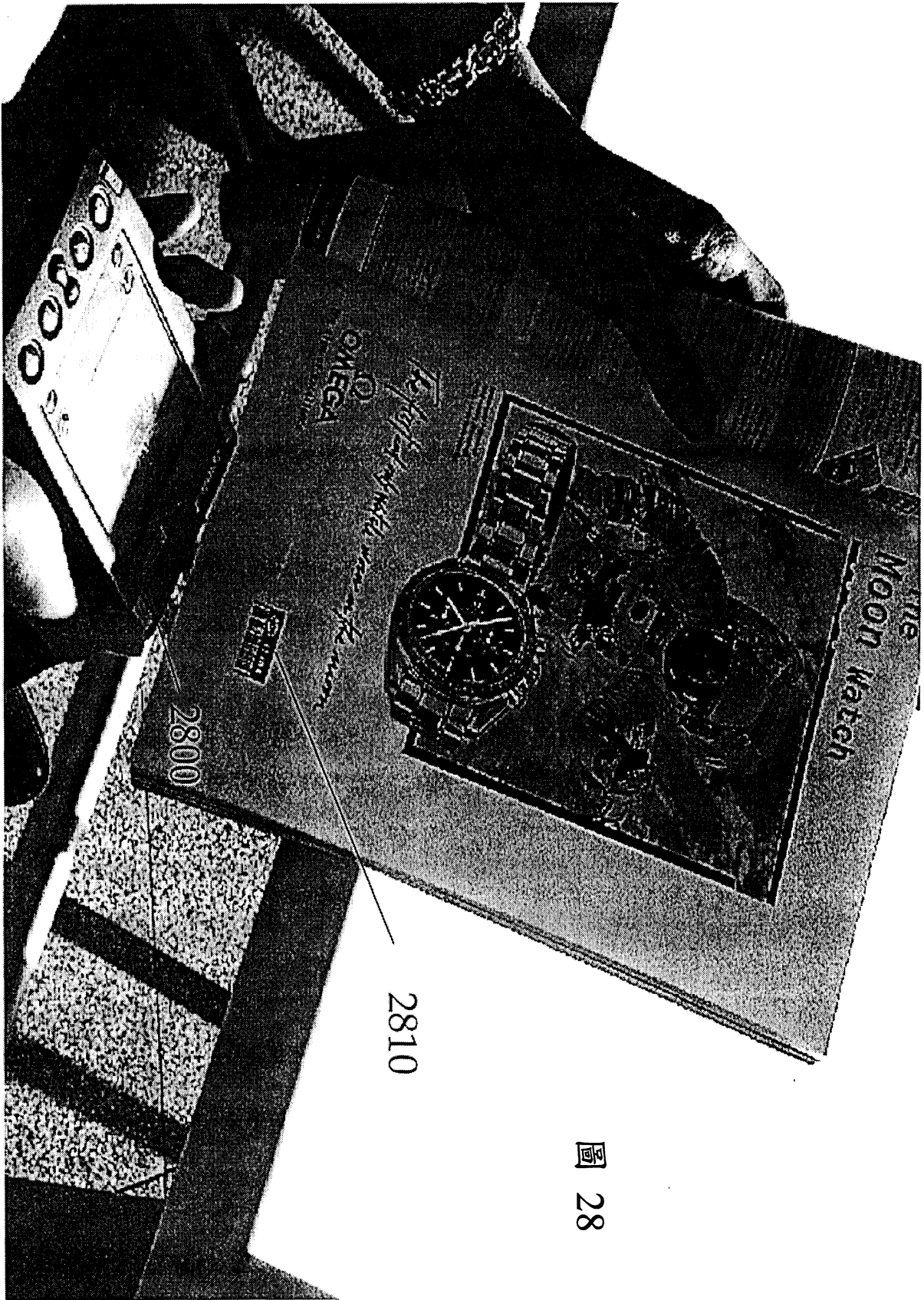


圖 27

圖式



2810

圖 28

圖式

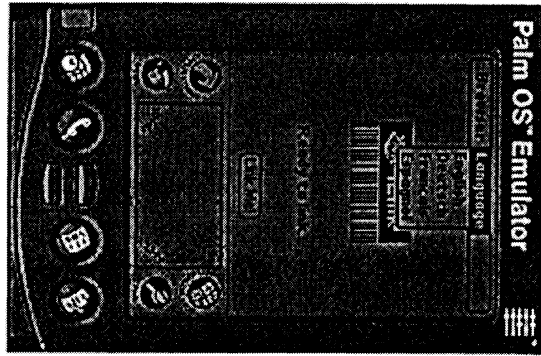


圖 29

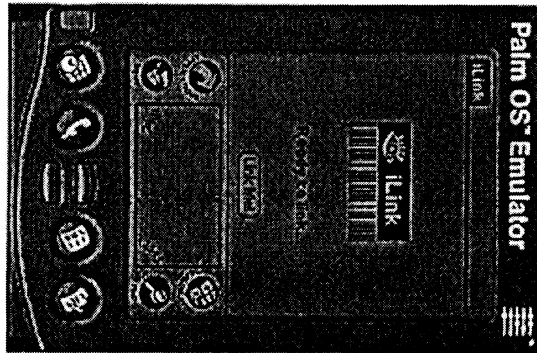


圖 30

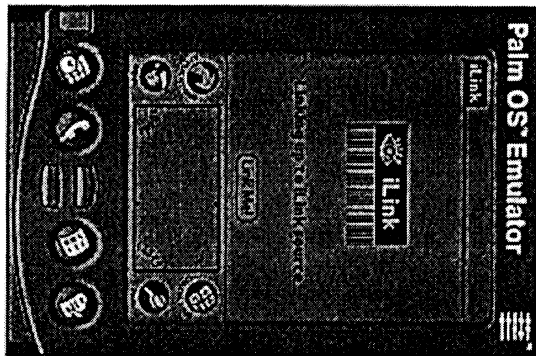


圖 31

圖式



圖 32

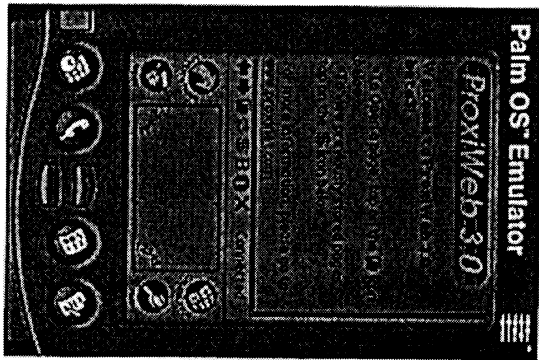


圖 33

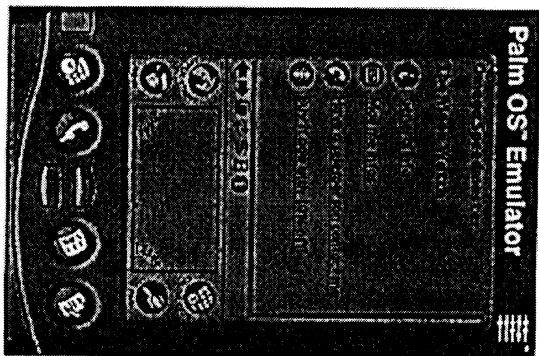


圖 34

圖式

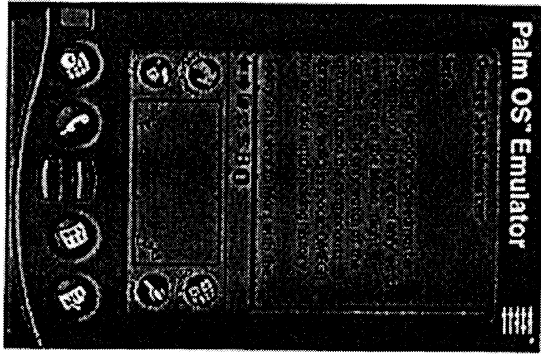


圖 35

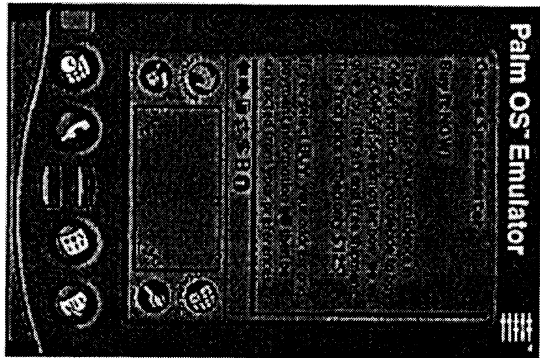


圖 36

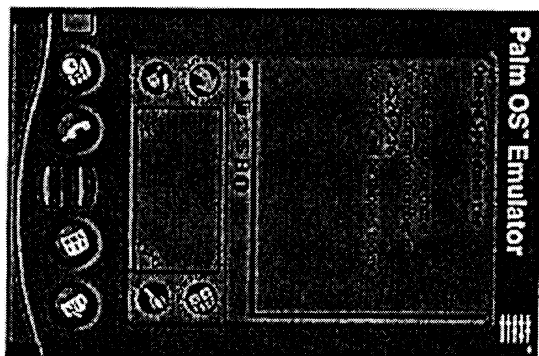


圖 37

圖式

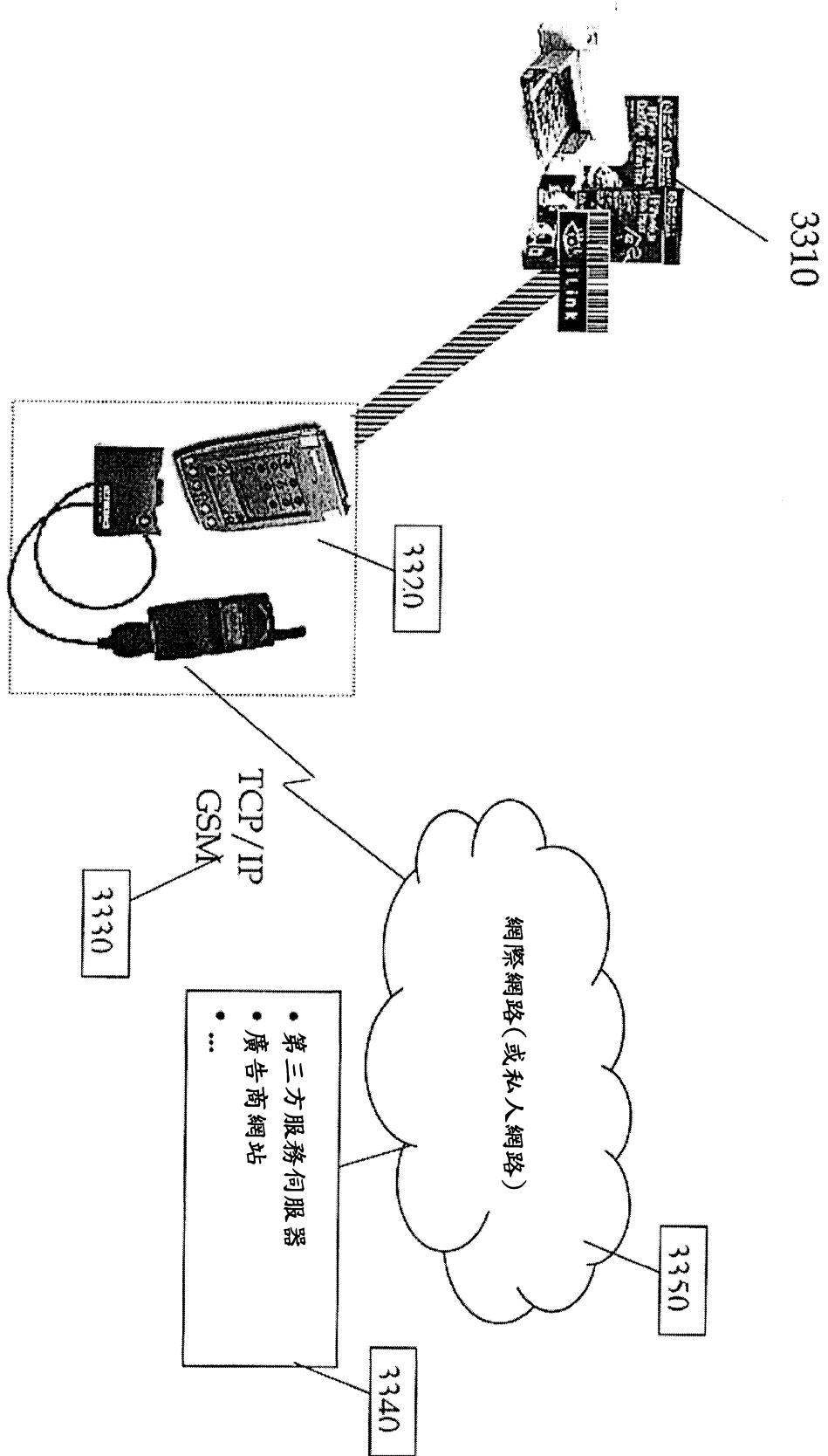


圖 38

# 公告表

91年1月29日 修正 補充

申請日期	89.2.29
案 號	89103629
類 別	G06F 17/30

A4  
C4

541471

(以上各欄由本局填註)

中文說明書修正頁(91年1月)

## 發 明 專 利 說 明 書

煩請  
修正本  
本右無  
變更費  
費者  
者  
91年  
1月  
29日  
所提  
之

一、發明 新 型 名 稱	中 文	利用網路技術進行高級資訊收集之系統、方法和物品
	英 文	A SYSTEM, METHOD AND ARTICLE OF MANUFACTURE FOR ADVANCED INFORMATION GATHERING UTILIZING WEB TECHNOLOGY
二、發明 創 作 人	姓 名	1. 理查德·內爾·卡姆龍 Richard Neill Cameron 2. 羅依克·猶恩·費因比爾 Loic Jouen Feinbier 3. 亞歷山大·皮特·萬·登·洪得 Alexander Peter Van den Hondel
	國 籍	均法國
三、申請人	住、居所	1. 法國索菲亞·安蒂波利斯市克里茲路449號 2. 法國利卡尼特市喬治·帕姆匹多街604號 3. 法國卡蒂紐夫·格拉斯市齊明科萊斯街1986號
	姓 名 (名稱)	美商艾森裘公司 ACCENTURE LLP.
	國 籍	美國
	住、居所 (事務所)	美國加州帕拉奧圖市皮吉米爾路1661號
	代 表 人 姓 名	韋恩·索邦 WAYNE P. SOBON

裝  
訂  
線

## 五、發明說明 ( 43 )

器根據對資料庫的需求，為特定的用戶取得佈局和偏愛內容1002。接著，伺服器為該頁面從內容資料庫1005中檢索所要求的內容1004。相關的用戶中心內容，如日曆、電子郵件、約會列表、和任務列表項並接著檢索1006（見圖11對這個過程更詳細的說明）。對資料庫的查詢使用作為用戶特徵儲存在用戶特徵資料庫1003中的用戶偏好來篩選返回的內容。返回的內容根據用戶特徵定義的佈局偏好被格式化放進網頁1007。接著將網頁返回客戶機並顯示給用戶1008。

圖11說明檢索加入網頁的用戶中心內容的過程。圖10B中對這一過程的說明1006是一個更詳細的例子。它假定伺服器已經獲得用戶特徵和現有的將要整合到網頁中的內容。伺服器1110分析經過篩選的內容，尋找事例、約會時間、和電子郵件的地址。如果找到其中任何一個，就將其標記並存於臨時儲存空間裡。接著，伺服器試圖從各個資料庫中發現用戶中心的內容1120。這包括將臨時儲存空間中的標記內容和日曆資料庫1140中的日曆項進行匹配1130；和電子郵件資料庫1114中的電子郵件項目1115進行匹配；交際資料庫1116中的交際項目1117進行匹配；任務列表資料庫1118中的任務列表項目1119進行匹配；新聞資料庫1120中的新聞項目1121進行匹配。經過檢索所有相關的用戶內容之後，將結果編輯在一起並返回1122。

**用戶角色**

系統允許用戶創建一批不同的角色，將其綜合為概貌

## 五、發明說明 ( 55 )

### 後備伺服器過程 ( 2400 )

圖 25 是較佳實施例的後備伺服器的方塊圖。後備 (圖 24 的 2400) 是一個電腦系統，它包括以下的軟體活動：智能代理人協調器 (Munin) 2580、資訊優先排序子系統 2530、一系列連續和斷續運作的資訊搜集和處理的智能代理人 2500、2502、與 2504、用戶概貌資料庫 2542 和支援軟體、資訊通道資料庫 2540 和支援軟體、通訊軟體 2550、資訊傳遞軟體 2560 和輔助軟體。

### 認知機器 (圖 24 的 2430 和 2436)

認知機器是硬體設備和軟體應用的綜合。硬體包括手提個人電腦和無線通訊設備。認知機器反映了一個透過無線資訊流不斷更新的"個人世界的狀態"。這一資訊，經過智能代理人元件的開發處理，包括電子郵件消息、滿足用戶個人偏好的新聞、更新的計劃、最新會議和事件的背景資訊、以及天氣和交通。認知機器包含於另一個專利申請。

圖 26 是一個依照較佳實施例的魔術牆的方塊圖。

### 魔術牆

魔術牆硬體包括：

- 與後備伺服器連接的電腦系統 2640
- 感應器組 2634、2630、和 2632 用來探測一個人的存在、位置、和標識
- 大的觸摸顯示屏 2620
- 聲音輸入 2610 / 輸出 2614 硬體

魔術牆軟體支援：

## 五、發明說明 ( 76a )

圖式元件符號說明

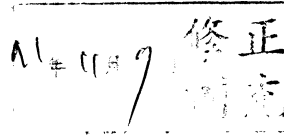
110	中央處理機	930	天線
112	系統匯流排	940	小型交易服務模組
114	隨機存取記憶體 (RAM)	950	網路供應商
116	唯讀記憶體 (ROM)	960	第三方網路供應商
118	輸入/輸出配接器	1001	要求
120	磁碟儲存單元	1002	偏愛內容
122	用戶介面配接器	1003	用戶特性資料庫
124	鍵盤	1004	內容
126	滑鼠	1005	內容資料庫
128	揚聲器	1006	檢索
132	擴音器	1007	網頁
134	通信配接器	1008	用戶
136	顯示配接器	1110	伺服器
138	顯示器	1114	電子郵件資料庫
200	主程式	1115	電子郵件項目
210	ProcessCommandLine	1116	交際資料庫
220	CreateStopListist	1117	交際項目
230	CreatePatterns	1118	任務列表資料庫
240	GoBackgroundFinder	1119	任務列表項目
900	書或其他產品	1120	用戶中心的內容
910	微型條碼閱讀器	1121	新聞項目
920	無線電話或類似的無線手 提設備	1122	返回
		1130	日曆項

## 五、發明說明 ( 76b )

1140	日曆資料庫	1450	名字
1200	用戶概貌	1460	數值
1220	工作的角色	1470	目的的步驟
1230	家庭的角色	1480	完成域
1240	小屋的家中的角色	1490	通用目的
1250	總體概貌	1510	用戶要求代理人統計資料 頁
1260	總體概貌	1520	檢索用戶的統計資料
1270	總體概貌	1530	用戶的概貌資料庫
1280	工作旅行概貌	1540	標準的統計資料
1290	家庭旅行概貌	1550	檢索公式
1310	用戶表格	1560	內容資料庫
1320	口令	1570	產生統計資料圖形
1330	角色	1580	創建統計資料頁
1340	概貌	1590	返回用戶
1350	概貌域	1610	用戶要求產品 X 的產品報 告
1360	概貌約束	1620	檢索用戶概貌
1370	名字	1630	用戶的概貌資料庫
1380	規則	1640	匹配概貌的預設門檻值
1390	許可	1650	內容資料庫
1410	用戶角色	1660	概貌匹配運算表
1411	用戶規則	1670	是否在設置中的用戶概貌 的距離 y 之內
1412	用戶規則	1680	門檻值變量被釋放
1420	有效的用戶目的		
1430	暱稱		
1440	資料域		

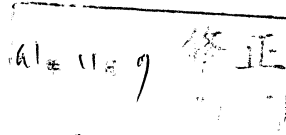
## 五、發明說明(76c)

- |      |  |      |            |
|------|--|------|------------|
| 1690 | 透過 n 個最近鄰居得到產<br>品的評估並將其用於決定<br>產品統計資料 | 1980 | 產生代理人講話的正文 |
| 1695 | 統計資料插進產品報告樣<br>板                       | 1990 | 內容資料庫      |
| 1697 | 產品報告返回用戶                               | 1995 | 講話正文插入網頁   |
| 1710 | 概貌資料庫                                  | 1997 | 網頁被返回用戶    |
| 1720 | 概貌通道伺服器                                | 2000 | 代理人        |
| 1730 | 公用網際網路                                 | 2010 | 用戶名字       |
| 1740 | 電視機                                    | 2020 | 口令         |
| 1750 | 店主                                     | 2030 | 標幟         |
| 1810 | 網路瀏覽器                                  | 2040 | 按鈕         |
| 1820 | 整合器的網路/應用伺服器                           | 2100 | 動畫代理人      |
| 1830 | 整合器的顧客概貌資料庫                            | 2110 | 旅行         |
| 1840 | 供貨商的產品資料庫                              | 2120 | 家務         |
| 1850 | 供貨商的顧客資訊資料庫                            | 2130 | 財務         |
| 1860 | 網頁                                     | 2140 | 逛商場活動      |
| 1900 | 用戶要求摘要頁                                | 2142 | 圖標         |
| 1920 | 給出用戶代理人的偏愛                             | 2146 | 直接鏈接       |
| 1930 | 用戶概貌資料庫                                | 2150 | 許可         |
| 1940 | 取得內容                                   | 2170 | 添加         |
| 1950 | 內容資料庫                                  | 2180 | 內容         |
| 1960 | 摘要內容                                   | 2190 | 個人資訊       |
| 1970 | 根據樣板格式化進網頁                             | 2200 | 世界         |
|      |  | 2210 | 本地新聞       |
|      |  | 2220 | 代理人        |
|      |  | 2230 | 不動產        |



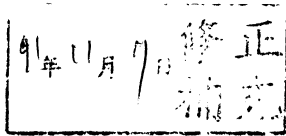
## 五、發明說明 ( 76d )

- |      |                      |      |          |
|------|----------------------|------|----------|
| 2300 | 交流資訊                 | 2560 | 資訊傳遞軟體   |
| 2310 | 代理人                  | 2580 | 智能代理人協調器 |
| 2370 | 圖表                   |      | (Munin)  |
| 2380 | 按鈕                   | 2610 | 聲音輸入     |
| 2390 | 按鈕                   | 2614 | 聲音輸出     |
| 2395 | 特別暗示                 | 2620 | 大的觸摸顯示屏  |
| 2400 | 後備                   | 2630 | 感應器組     |
| 2410 | 公共客戶機 (魔術牆)          | 2632 | 感應器組     |
| 2420 | 公共客戶機 (魔術牆)          | 2634 | 感應器組     |
| 2430 | 個人行動無線客戶機 (認<br>知機器) | 2640 | 電腦系統     |
| 2436 | 個人行動無線客戶機 (認<br>知機器) | 2700 | iLink 登入 |
| 2446 | 網路客戶機                | 2710 | 雜誌廣告     |
| 2448 | 網路客戶機                | 2800 | 手持電腦     |
| 2450 | e-mail 客戶機           | 2810 | iLink 條碼 |
| 2460 | 客戶機                  | 3310 | 產品       |
| 2500 | 智能代理人                | 3320 | 電腦設備     |
| 2502 | 智能代理人                | 3330 | 通信協議     |
| 2504 | 智能代理人                | 3340 | 第三方網站    |
| 2530 | 資訊優先排序子系統            | 3350 | 網際網路     |
| 2540 | 資訊通道資料庫              |      |          |
| 2542 | 用戶概貌資料庫              |      |          |
| 2550 | 通訊軟體                 |      |          |



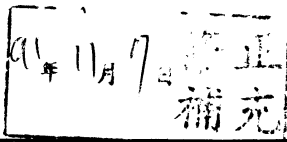
## 六、申請專利範圍

1. 一種創建資訊摘要的方法，包含以下步驟：
  - (a) 檢索目標項目的電子標記；
  - (b) 創建基於電子標記的查詢；
  - (c) 應用該查詢來查詢資訊網絡；和
  - (d) 向用戶提供與目標項目相關的資訊。
2. 如申請專利範圍第 1 項所述之創建資訊摘要的方法，包括根據事先定義的標準分析電子標記的步驟。
3. 如申請專利範圍第 1 項所述之創建資訊摘要的方法，包括將目標項目的電子標記用於用戶的概貌資訊，以獲得附加資訊的步驟。
4. 如申請專利範圍第 3 項所述之創建資訊摘要的方法，其中概貌資訊包括一種選擇語言。
5. 如申請專利範圍第 1 項所述之創建資訊摘要的方法，其中標記中包括用於裁剪對用戶的回應的內容資訊。
6. 如申請專利範圍第 1 項所述之創建資訊摘要的方法，包括將資訊郵寄用戶的步驟。
7. 如申請專利範圍第 1 項所述之創建資訊摘要的方法，其中查詢包含一個網際網路通信協議的地址。
8. 如申請專利範圍第 1 項所述之創建資訊摘要的方法，其中的標記是一個條碼。
9. 如申請專利範圍第 1 項所述之創建資訊摘要的方法，包括使用電子標記購買一種產品的步驟。
10. 一種創建資訊摘要的裝置，包括：
  - (a) 一個處理器；



## 六、申請專利範圍

- (b) 一個在處理器控制下儲存資訊的記憶體；
  - (c) 檢索目標項目的電子標記的邏輯；
  - (d) 基於該電子標記創建查詢的邏輯；
  - (e) 使用該查詢查詢資訊網絡的邏輯；和
  - (f) 向用戶提供與目標項目相關的資訊的邏輯。
11. 一種含有記錄在電腦可讀的媒體上用來創建資訊摘要的電腦程式之記錄媒體，包括：
- (a) 一個代碼段用來檢索目標項目的電子標記；
  - (b) 一個代碼段用來創建基於電子標記的查詢；
  - (c) 一個代碼段用來使用該查詢查詢資訊網絡；和
  - (d) 一個代碼段用來向用戶提供與目標項目相關的資訊。
12. 如申請專利範圍第 11 項之記錄媒體，包括根據事先定義用來創建查詢的標準來分析電子標記的邏輯。
13. 如申請專利範圍第 11 項之記錄媒體，包括將目標項目的電子標記用於用戶的概貌資訊，以獲得附加資訊的邏輯。
14. 如申請專利範圍第 13 項之記錄媒體，其中概貌資訊包含一種選擇語言。
15. 如申請專利範圍第 11 項之記錄媒體，其中查詢包含一個網際網路通信協議地址。
16. 如申請專利範圍第 11 項之記錄媒體，其中標記中包括用於裁剪對用戶的回應的內容資訊。
17. 如申請專利範圍第 11 項之記錄媒體，包括將資訊郵寄



## 六、申請專利範圍

用戶的邏輯。

18. 如申請專利範圍第 11 項之記錄媒體，其中標記是條碼。
19. 如申請專利範圍第 11 項之記錄媒體，包括透過電話傳遞資訊的邏輯。

91年11月7日 修正  
補充

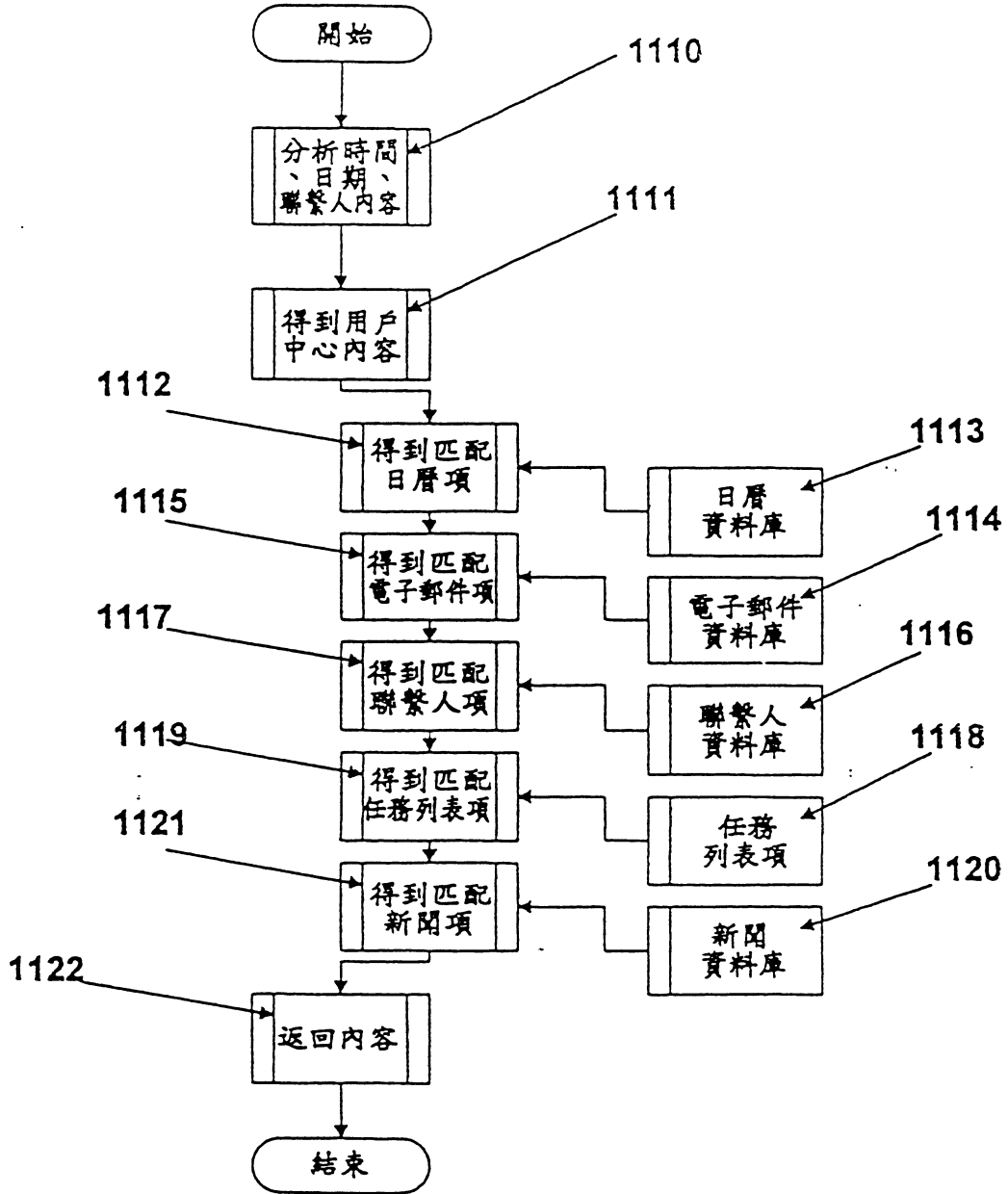


圖 11

91年11月7日 修正補充

第 089103629 號專利申請案  
中文圖式修正頁(91年11月)

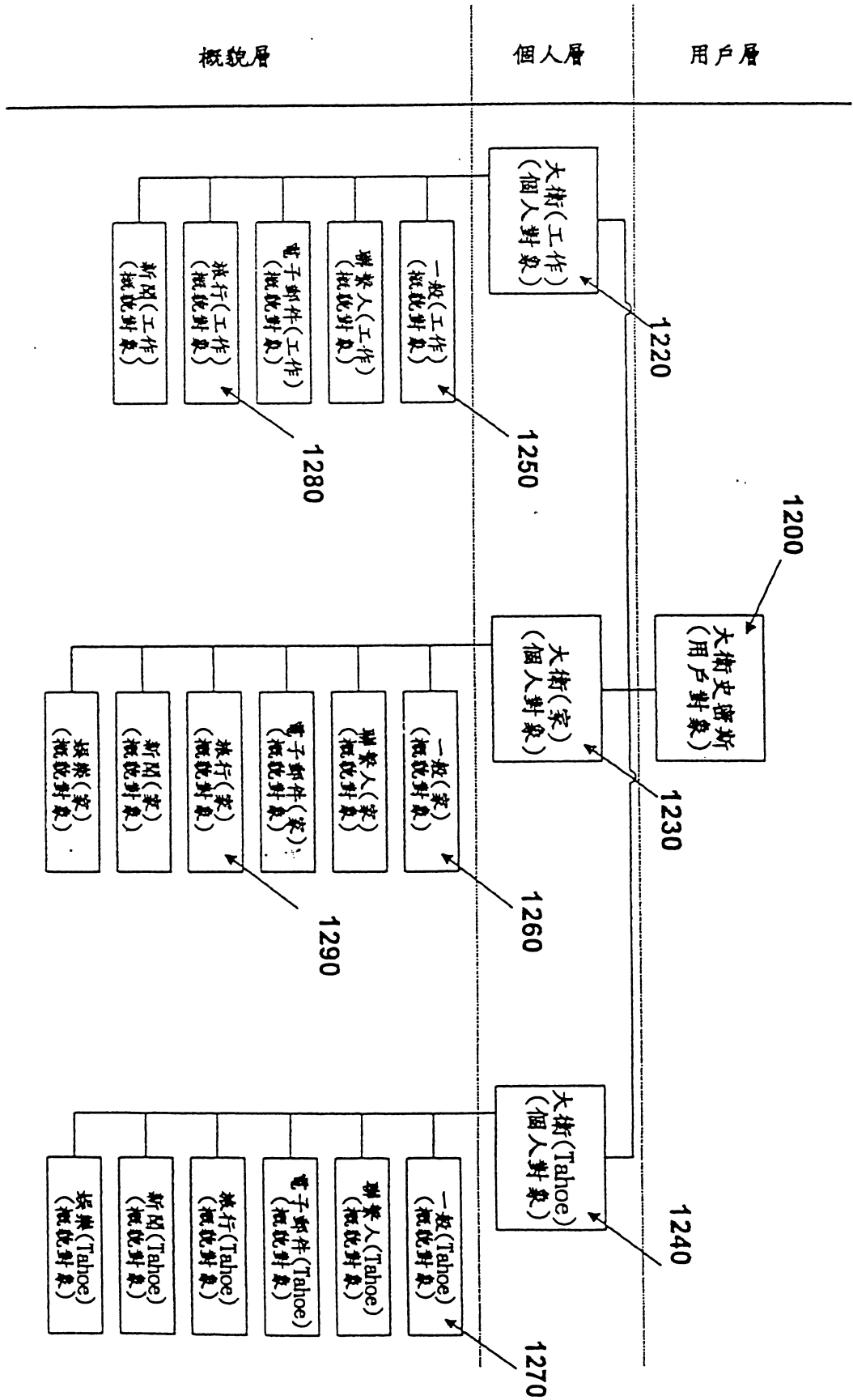


圖 12

第 089103629 號專利申請案  
中文圖式修正頁(91 年 11 月)

第 089103629 號專利申請案  
中文圖式修正頁(91 年 11 月)

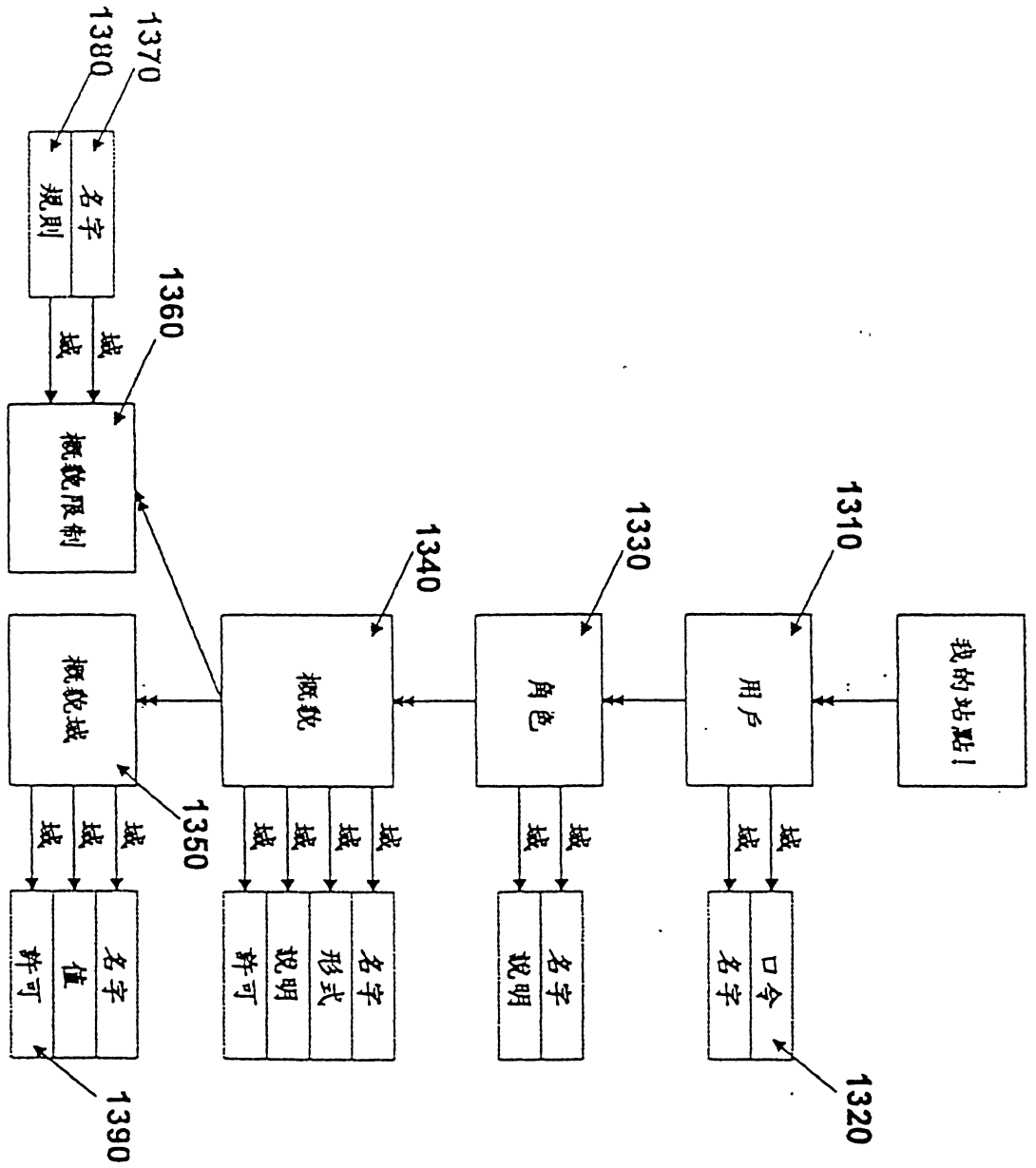


圖 13

9

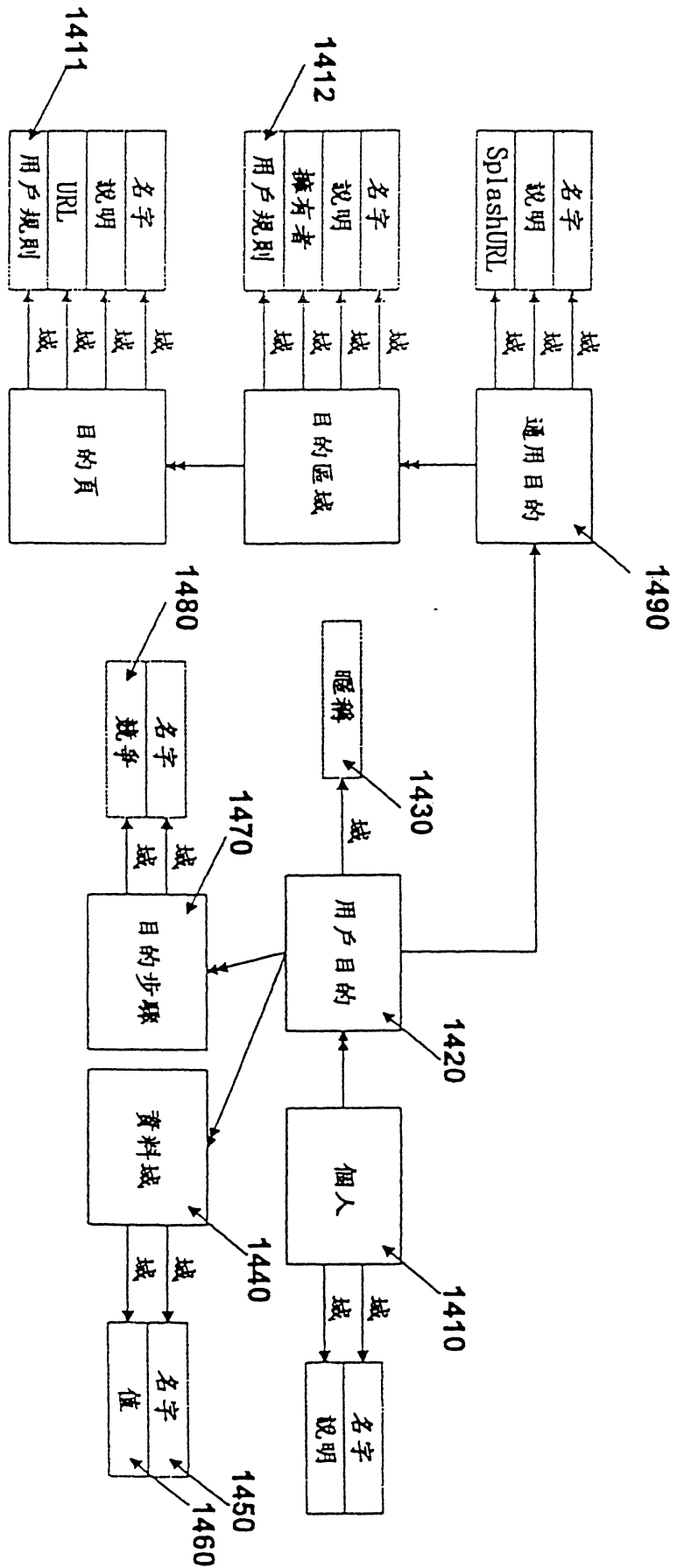


圖 14

91.11.7

第 089103629 號專利申請案  
中文圖式修正頁(91 年 11 月)

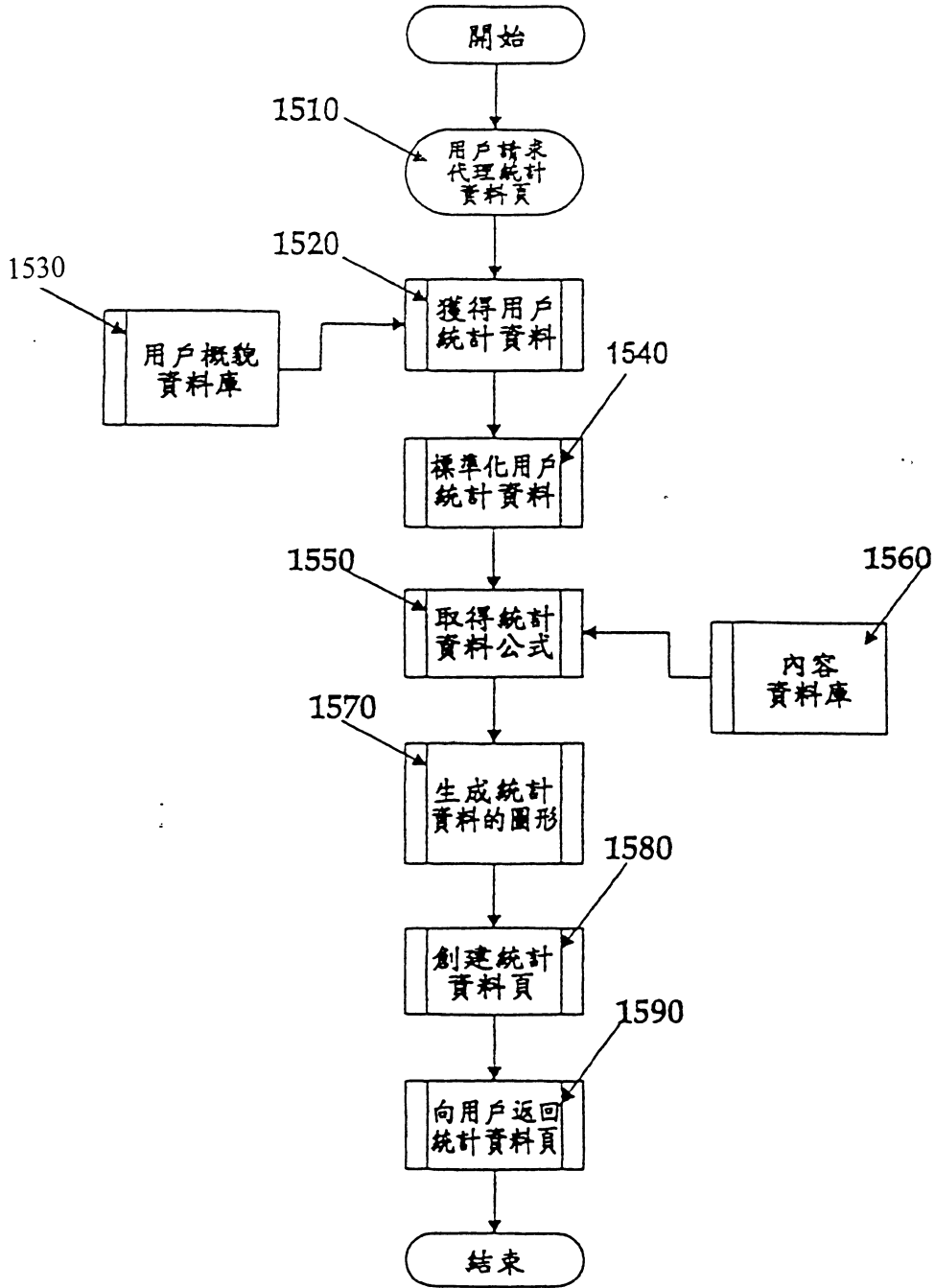


圖 15