

WWW 行政資料整合管理

Chiou-Feng Wang 王秋鳳

Tieshy Lin 林泰旭

{feng,ted}@iis.sinica.edu.tw

Institute of Information Science, Academia Sinica

中央研究院資訊所

摘要

在同一單位內，不同部門在不同的機器上可能都會存放著一份大同小異的資料如個人資料，除了必須各自建立檔案及維護屬於自己管理的資料外，其中資料若有異動還必須記得通知其他相關部門跟著修改，時間一久資料就容易產生不一致的狀況，因為有的部門改了，有的卻忘了更改，導致所要的資料在不同部門總有幾個欄位不是最新版本，徒增彼此之間資料對照的痛苦。基於此，我們希望有一套系統能夠整合處理這一類的資料，以避免同樣的問題一再重複發生；同時必須具備一套很好的使用介面以減少使用者的使用困擾。目前我們是利用 WWW 當使用介面，用 PERL 語言撰寫 CGI 程式把各個不同部門中所有的資料整合在一起，放置在 WWW SERVER 上，無論資料管理者使用的機器是個人電腦或是 UNIX 工作站，其作業系統是哪一種，只要利用 WWW 瀏覽器進入資料管理畫面（必須具有管理權力），他就可以輕易的執行資料建立、顯示、刪除及修改的動作。透過網路，不同部門分享著彼此之間的資料。

1. 簡介

原來的作業方式是各部門各自獨立建檔、維護及使用自己的資料管理系統，因為資料量不是很大，在 PC 上的是利用 DBASE 建立資料庫、在工作站上則是利用 shell scripts 寫簡單的資料管理，有的甚至只是儲存單純的文字檔，若有新資料要增加，則必須在各部門都填寫一份，如果只是某筆資料有異動，

那麼異動部份的管理者除了需作更改外，還必須通知資料相關部門也同時作更改動作，若忘了修改，資料就會不一致；而修改一筆資料牽扯到的地方可能是分散在各個不同機器上的不同檔案或者是資料庫。所以，重複填寫資料、資料的不一致性及需要人工一筆一筆去修改資料的問題是我們迫切想要解決之事，甚至我們希望各部門之管理者都可以輕易的看到需要的資料，但只需而且也只能更改屬於自己權力範圍內的部份。在不採購任何昂貴資料庫管理系統的前提下，我們提出的解決方法是利用公眾軟體庫再加上自己寫的一些小程序就可以解決所謂非有資料庫系統不可的情況。立即可見的成效是：

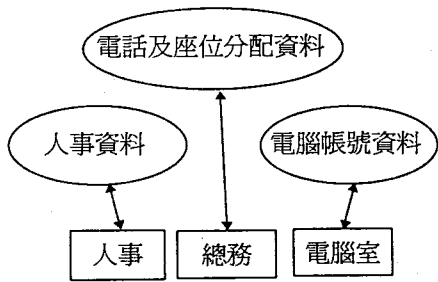
- 不用採購任何資料庫管理系統，
- 不必遷就現有使用的任何機器或作業系統，
- 可以立即解決資料整合所碰到的問題。

2. 原本作業方式

只要是彼此有相關的資料分由多個不同部門在管理的話，都可以利用上述方法解決之。現在我們就以其中一個例子來說明原來及整合後之整個作業處理方式。

2.1 資料架構

原來作業方式之資料架構如圖一，各部門各自增加、刪減、修改、維護屬於自己所管理的資料。



圖一 原本作業方式

人事部門管理項目包括：

姓名、照片、組別、隸屬老師、學歷、到職日期、離職日期 等等

總務部門管理項目包括：

姓名、組別、電話分機號碼、信箱號碼、房間號碼、鑰匙、乘車證、游泳證 等等

電腦室管理項目包括：

中英文姓名、組別、隸屬老師、帳號名稱、密碼、UID、GID、工作目錄 等等

圖書館管理項目包括：

中英文姓名、組別、隸屬老師、圖書證

2.2 缺點

- 資料無法互相分享，行政效率不佳
- 資料填寫麻煩
每一位新進人員在各部門都必須填寫一份大同小異的個人資料。
- 資料有任何異動，就必須通知所有相關部門更改，容易產生不一致情形，而且無法自動修改相關資料。
- 只能看到自己內部的資料，無法看到一筆較完整的相關資料

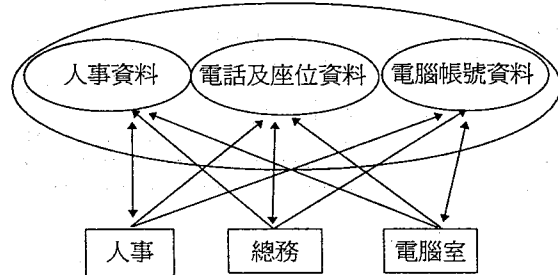
例如：希望知道一筆資料項目包括：姓名、電話、房間號碼、電子郵件地址、組別，那麼依照現有架構則無法辦到。

3. 資料整合後之作業方式

3.1 資料架構

整合後的資料架構如圖二

把各部門所有相關資料整合在一起後，各部門管理者可以看到其他部門所建立的相關資料(如圖二中之單箭頭部份)但只能對屬於自己權力範圍內的資料作增加、刪減或修改動作(如圖二中之雙箭頭部份)。至此，人員從進用到離職，資料完全電腦化。



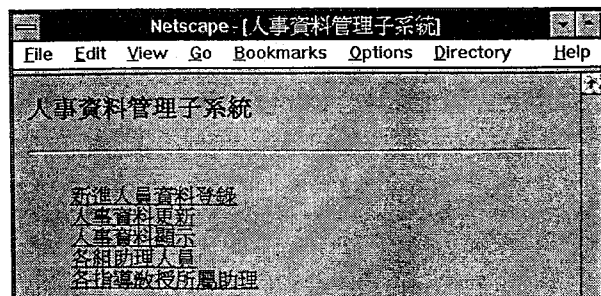
圖二 整合後作業方式

3.2 系統使用工具

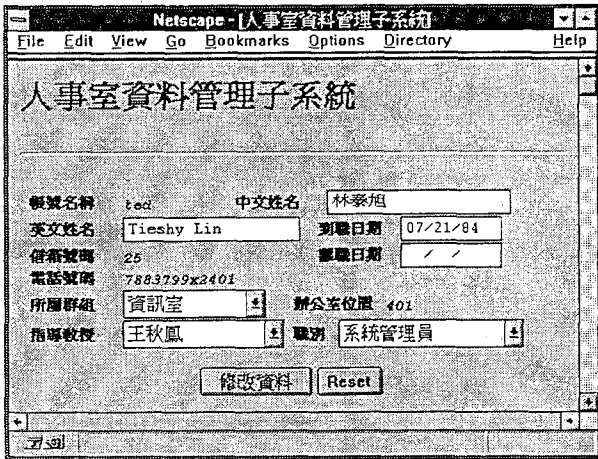
- 公眾軟體 WWW SERVER (httpd1.4版)
- 公眾軟體 PERL5.001m 版語言編譯器
- 自己用 PERL 語言撰寫的 CGI 程式
- UNIX 作業系統本身所提供之 DBM 程式庫

3.3 資料處理方式

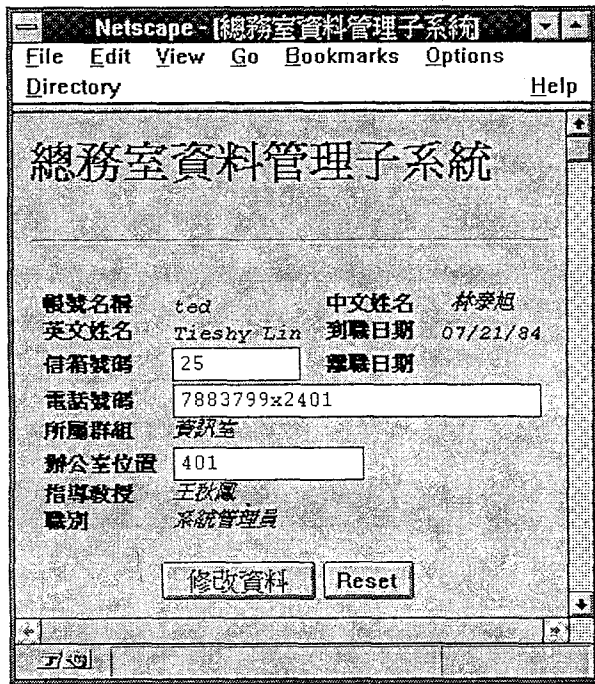
各部門所有相關資料整合成一份存放在 WWW 伺服器主機上，而各部門管理者都一樣只需利用 WWW 瀏覽器(無論是個人電腦或 UNIX 工作站皆可)進入「資料管理畫面」即可進行資料輸入、刪除及修改動作甚至作資料查詢，如圖三。各部門之管理者也都擁有各自不同的處理畫面，如圖四、五、六。各管理畫面中可輸入資料的地方表示是管理者有權更改的項目，而另外顯示的資料則是來自其他部門所建立。



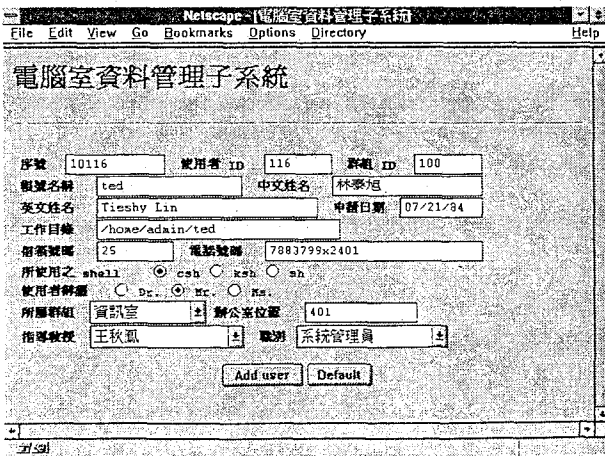
圖三 資料管理畫面



圖四 人事管理畫面



圖五 總務管理畫面



圖六 電腦室管理畫面

3.4 範例

當任一部門更動資料後立即執行產生最新的其他資料檔。

3.4.1 電話號碼更動

總務部門更改電話號碼後除了產生新的電話一覽表外，也會自動產生新的 whois server 資料檔、gopher server 中的人員聯絡資料一覽表及 NIS 伺服器主機內之 passwd 檔

3.4.2 人員離職

當人事部門處理完人員離職後，一樣會自動處理總務部門的資料和電腦室中的 whois server 資料、gopher server 資料及帳號管理資料。

3.5 資料表及資料庫資料產生

由整合資料可以產生所需的資料表及其他資料庫所需資料如：

- 所有聘僱人員資料表
 - 包括：姓名、組別、隸屬老師、學歷、到職日期、離職日期
- 電話、房間表
 - 包括：姓名、組別、電話分機號碼、信箱號碼、房間號碼
- 帳號管理資料
 - 自動產生電腦 NIS 伺服器主機中之 passwd 檔、group 檔、alias 檔，其項目包括：帳號名稱、密碼、UID、GID、中英文姓名、工作目錄、及所使用之 login shell
- whois server 所需資料
 - 自動產生 whois server 所需資料，包括：中文全名、英文全名、抬頭、組別、電話號碼、電子郵件地址、房間位置

- gopher server 所需資料

自動產生放置在 gopher server 中之所有人員聯絡一覽表的資料檔，包括：姓名、組別、電話分機號碼、電子郵件地址

統”，第一屆即時與媒體系統研討會，July 1995, 159-164。

[2] Larry Wall, Randal L. Schwartz, “Programming Perl”, Jan 1991.

[3] L. Stein, “CGI.pm - a Perl5 CGI Library”, Aug 1995.

4. 資料安全問題

為了顧及資料安全可分為三部份考慮

- 專屬的管理畫面

各部門管理者只能對自己權力範圍內的資料項目作輸入、刪除及修改動作

- 各部門管理者之 ID 及 PASSWORD

各部門管理者必須擁有自己的 ID 及 PASSWORD 才能進到自己專屬的管理畫面

- 鎖檔功能

為避免多個部門同時修改同一筆資料，作有簡單的鎖檔功能

5. 結論

一般在談論資料處理時，總是把它渲染的很大，需要購買資料庫系統方能解決問題；對於我們這種資料量不是很大，但很容易產生不一致而且資料也無法彼此分享種種問題，在一個單位內所造成的困擾並不小，所以在不採購資料庫管理系統的前提下，我們結合了一些網路上的公眾軟體(public domain software) WWW (它提供跨平台的使用介面，解決了各部門使用不同機器、不同作業系統的問題)，再加上自己撰寫的程式就很漂亮的解決了不同部門之間資料溝通及分享的問題。不過目前的作法還是必須一一的視應用狀況而編寫程式，若有不同需求，則需另外寫程式，這種處理方式顯然不夠好；我們的下一個目標是想結合 WWW 發展出另一種介面，屆時，使用者即可自行藉由此種介面訂定資料庫之欄位、處理資料畫面，進而自行建立資料庫系統及產生管理畫面，也就不需要再請求他人代寫程式了。

6. 參考資料

[1] 王秋鳳及林泰旭，“WWW 學術研討會籌辦系