

An Efficient Web-BBS System

黃年時

逢甲大學資訊工程學系

台中市文華路 100 號

TEL: (04)4517250 - 3128

EMAIL:magic@hvlw.iecs.fcu.edu.tw

竇其仁

逢甲大學資訊工程學系

台中市文華路 100 號

TEL: (04)4517250 - 3719

EMAIL:crdow@iecs.fcu.edu.tw

徐弘洋

逢甲大學資訊工程學系

台中市文華路 100 號

TEL: (04)4517250 - 3705

EMAIL:hyhsu@iecs.fcu.edu.tw

摘要

這篇文章介紹一個有效的 Web-BBS 系統，這個系統利用資料庫做為後端(backend)，以達到有效的資料管理與快速的資料搜尋為目標。這個系統將包涵六個基本的 BBS 功能(如 read posts、create posts、send message 等)，並將此六個功能轉換為相對映的 SQL 的指令。我們把這個系統實作在 Win95/NT 上，並採用 Microsoft Visual Fox Pro 3.0 為後端的資料庫。

1. 緒論

最近在短短的幾年中，TANet 的使用量有很大的增加，尤其在全球資訊網(WWW)，與 BBS 上更是方興未艾[1,6]。在 TANet 上的 BBS 都已將處理中文的能力視為基本的功能但所使用的介面仍是以文字介面為主。雖然在文字介面下可以使用 ANSI 字元做到稍具彈性的使用者介面，但在 Graphical User Interface(GUI) 流行的今天看來未免有些單調，所以在使用者介面的演化上有兩種主要的 approaches，一種以 client server 的方式使用專屬的 client 負責 UI 的部分[2]，另一種則是透過 WWW 的方式使用 WWW BROWSER 來當作 UI，在這裡我們簡稱此類的系統為 Web-BBS[3][4]。採用 WWW BROWSER 當作 BBS 使用者介面的方式讓使用者比較不會感到陌生。

隨著 WWW 的流行，使用者慢慢習慣於超連結(Hyperlink)以滑鼠點取所需的資料，另外使用者對多媒體資料的需求量也日益增大，除了文字的資料外，使用者還希望看到圖片，聲音或其他的多媒體資料和原有文字資料的結合，Web-BBS 除了 UI 上比傳統的 BBS 更具優勢外，透過 WWW BROWSER 對多媒體的支援 Web-BBS 可以輕易的結合多媒體與 BBS 的優點，提供使用者更多元化的資訊種類[3]。

Web-BBS 透過 WWW 的方式來表現 BBS 的 UI 主要工作即是依照使用者的動作把所對映的資料動態地表現出來，在這種動態的表現上 Web-BBS 大多使用 PERL 或 C 來做為編寫其 CGI 的語言，而所使用資料的存取方式也和傳統的 BBS 一樣以檔案的存取方式來實作[3]，在這種方式下每一封 BBS 上的 POST 被存成一個檔案而每個 BBS 上的 GROUP 被設定成一個目錄。這一類的實作方式雖然可以很快速直覺地把系統實作出來，但是當 POST 個數一多時太多的檔案存取與搜尋勢必造成系統的一大負擔，除此之外這類 BBS SYSTEM 將大量的消耗 FILE SYSTEM 中有限的 INODE 數目，這些就是現有 BBS system 與 Web-BBS 所共有的缺點。

這篇文章主要在介紹一個有效的 Web-BBS 系統，這個系統利用 database 當作 BBS 的 backend system，其優點為擁有強大的搜尋引擎以應付複雜的搜尋，能有效的管理資料與資料間的關係，以減少 BBS

系統所使用的fs24 INODE 數目來減輕對檔案系統所造成的負擔。在第二節我們將介紹一些相關的研究。在第三節我們提出一種與資料庫結合的 BBS 系統架構，在第四節我們介紹系統的實作部份及功能模組，第五節我們將展示這個雛型系統，最後是結論以及討論我們未來的工作方向。

2. 相關研究

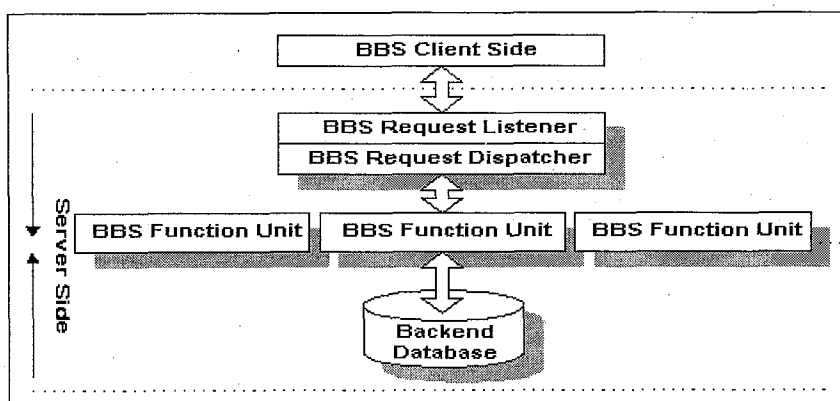
我們這一個系統是一個在 Web 上以 database 為 backend 的 BBS 系統，有許多的研究領域與我們的系統有關，而主要幾個比較重要的相關研究與技術包括：現有系統在 WWW 上的應用、WWW 與 database 的整合、各種 Web-BBS 系統與技術，以下分別對這幾個相關的研究與技術做一簡單的介紹：

越來越多的系統實作在 WWW 上，如 groupware[10]，網管軟體[21]，這些系統使用 WWW 的介面使得 client 端的設計得以簡化，並且可讓 client 在不同的平台上運作，另外更可以縮短使用者熟悉系統所需要的訓練時間，不需要使用者再去熟悉其他的系統 UI，所以將 BBS 系統加入 WWW 對使用者還說是有利的。於已存在的系統外加入一介面程式用來結合 WWW，這樣可以達到軟體再次使用(reuse)的目的，以降低開發成本、時間與風險[20]。

WWW 並沒有提供專屬的資料庫來支援資料存取，如此一來光靠檔案系統並不容易組織並管理一些大型的異質性資料，所以整合 WWW 與 database 是必要的。在整合 WWW 與 database 中資料庫通道(database gateway)接收來自 client 端的要求並加以處理，最後將結果傳回給使用者[18]。Pyung-Chul Kim[16] 將資料庫通道分為 server-side extension 與 client-side extension 兩大類，而 server-side extension 就是我們常用的 cgi 與 server API 方式。另外在 WWW 與分散式的資料庫整合上也有使用 HIRONA Object Request Broker(HORB)、X/Open compliant common sublayer 等的方式提出[14,17]。

Web-BBS 從最早的 HYPER BBS[4]，到國內最近中文化的版本[3]，都是以 cgi 與檔案存取的方式來實作，cgi 的方式在實作上有較大的彈性、快速、簡單等優點，但是在執行時對系統資源的消耗量較大[7-9,14]。雖然如此，cgi 用在系統雛型的建構上仍然是個很好的選擇。若使用 cgi 程式自動產生器或是由 cgi 程式呼叫其他程式產生器所產生的程式，這種作法可縮短整個系統的發展時間[11-13,15,19]。使用檔案存取的方式來管理資料是比較沒有效率的方式，這也是現有 BBS 的缺點。

3. 系統架構



圖一、BDB 架構

在這一節裡我們將提出一種與 Database 相結合的架構我們稱之為 Backend Database BBS (BDB)。如圖一所示我們將 BBS 中的各種功能畫分為各個不同的 BBS Function Unit，各個 BBS Function Unit 由 BBS Request dispatcher 根據使用者的動作來分配工作，其中 BBS 系統所使用到的資料全部放在 Backend Database 中，而每一個 BBS Function Unit 就是實際對 Database 動作的一部分。

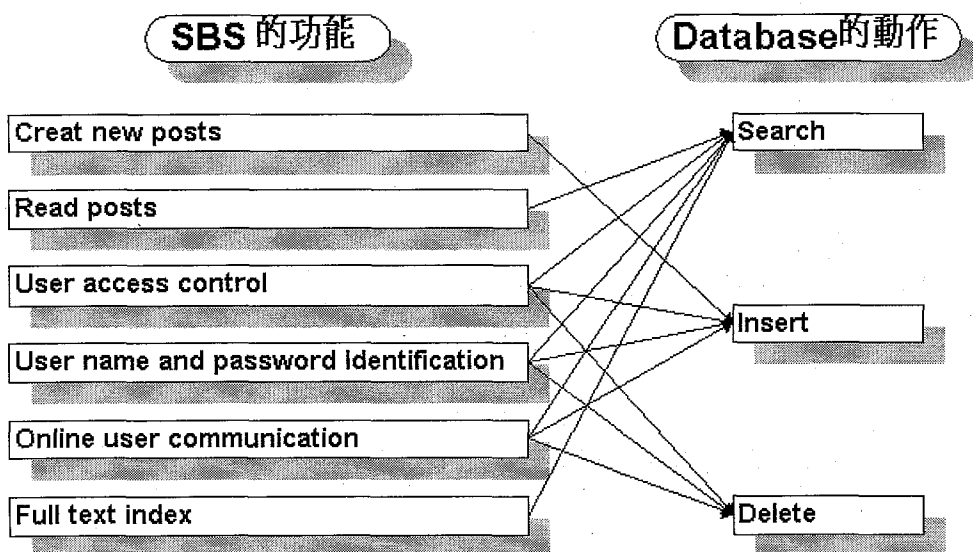
為了驗證 BDB 架構是否可行，在這裡我們依據 BDB 架構設計一個簡單的 BBS 模型，我們把它稱之為 Simple BBS System (SBS)。首先我們要知道一個 BBS system 的功能有哪些，經過分析我們得知一個 BBS 的基本功能有下列幾項：

1. Create new posts
2. Read posts
3. User access control
4. User name and password identification
5. Online user communication
6. Full text index

也就是說 SBS 必須要能在 BDB 架構下提供以上六項功能，SBS 中的各項功能都是由 BBS Function Unit 所提供，因為 BBS Function Unit 是對 Backend Database 動作所以我們就要確保 SBS 中的每項功能都能由 Database 的動作完成。Database 的動作有下列幾項：

1. SEARCH (相當於 SQL 中 Select 指令)
2. INSERT (相當於 SQL 中 Insert 指令)
3. DELETE (相當於 SQL 中 Delete 指令)

根據我們的分析我們將 SBS 中的各項功能轉換為 Database 中的各種動作，轉換的結果如圖二所示：



圖二、SBS 中的各項功能與 Database 動作間的轉換

- Create new posts:

在 SBS 裡我們把每個 news group 的信件儲存在一個 Database 的 Table 中，當使用者要 create new posts 的時候 BBS Function Unit 以對 Database 做 Insert 的動作來實作此項功能。

- Read posts:

當使用者要 Read posts 的時候 BBS Function Unit 以對 Database 做 Search 的動作，來實作此項功能。

- User access control:

為了實作 Online user communication 與系統安全的理由 SBS 在 Database 中有一個 online user table 用來記錄已在線上的使用者和使用者目前的狀況，所以當使用者對 SBS 提出 request 時我們不只要 search 使用者的權限加以比對外還要對 online user table 做 update，在這裡我們用 insert 來做 update 的動作，除此之外當使用者離線時 SBS 還要將使用者從 online user table 中刪除。

- User name and password identify:

當一個使用者嘗試著 login 到 SBS 時，相對映的 BBS Function Unit 就要對儲存有 user name 和 user password 的 table 作 search 的動作，驗證使用者的 user name 和 password 是否合法，在合法的 login 後還要對 online user table 作 insert 的動作，表示使用者已處於上線狀態。除此之外，若使用者帳號有新增或刪除的變動時則要對 database 作 insert 和 delete 的動作。

- Online user communication:

當一個由使用者所產生的訊息嘗試著要傳送到另一個已上線的使用者時，首先相對映的 BBS Function Unit 先要去 search online user database 查看接收此訊息的使用者是否上線中，除此之外 SBS 還保留一定量的訊息於 database 的系統中以供使用者查詢，所以相對映的 BBS Function Unit 要對 database 作 insert 的動作，最後當使用者離開 SBS 或保留的訊息超過 SBS 允許範圍時，相對映的 BBS Function Unit 就要對 database 作 delete 的動作。

- Full text index:

在 SBS 中 BBS Function Unit 以 search 儲存 posts 的所有 table 來實作 full text index 的動作。

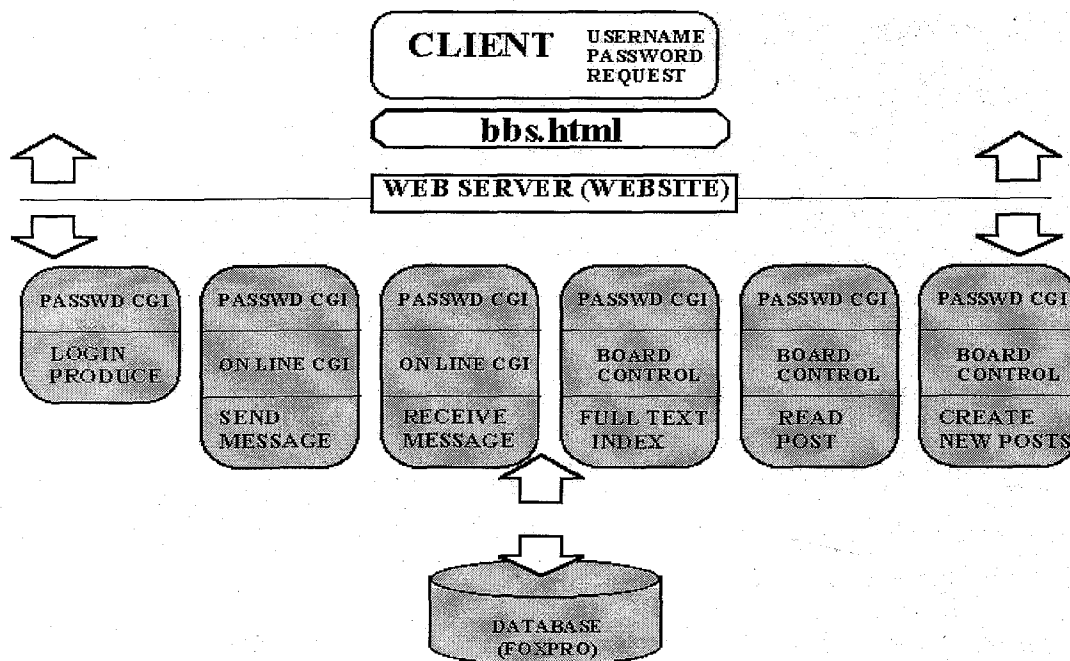
以上就是 Backend Database BBS (BDB) 架構與根據 BDB 而設計之 SBS 模型的介紹與可行性分析。

4. 系統實作

在實作 SBS 上我們選擇了以將 SBS 實作成 Web-BBS 的型式，整個系統的功能模組如圖三所示。我們採用了 cgi 的方式結合 WWW 與 database 快速的建立雛型系統。根據 SBS 的功能我們將 SBS 分為六個 BBS Function Unit，而每個 BBS Function Unit 由數個 cgi 副程式所組成，而每個 BBS Function Unit 的最底層 cgi 副程式就是用來完成此 BBS Function Unit 的主要動作，而第一層副程式的作用主要是實作有關 security 的部分。

因為 SBS 為一個 Web-BBS 所以資料的傳輸上是以 stateless protocol 來進行，和傳統 BBS 不一樣的地方為使用 stateless protocol 時，SBS 和使用者間只有在有資料或訊息要交換時才有連線，所以當使用者成功的 login 到 SBS 後兩者間的連線已不存在，所以系統無法單靠使用者傳送的 request 來判斷使用者是否上線，或者確定使用者是否合法。在這裡我們使用一種變通的方式來解決此一問題，SBS 與使用

者間的界面 bbs.html 使用 javascript 在 client 端隱藏了使用者的 username 和 password，當使用者成功的 login 的同時，在 client 端的 javascript 將使用者的 username 和 password 記錄下來，此後每當 client 端對 SBS 送出 request 時，在 client 端的 javascript 會把已存在記錄中的 username 和 password 一起送到 SBS，SBS 在 BBS Function Unit 中處理 request 時先將送過來的 username，password 與 SBS 資料庫中的記錄作比較如果符合了才對此 request 進行服務，若不符則傳送出錯誤訊息。



圖三、SBS 的功能模組圖

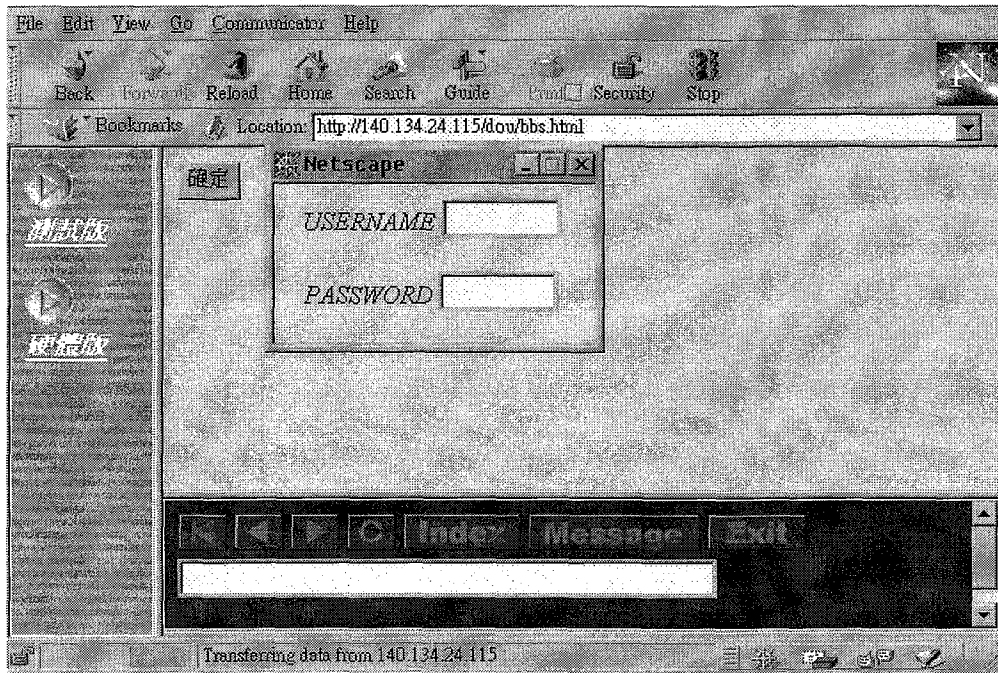
在與使用者的介面上我們採用 bbs.html 這一個 hyper link document，在 client 端控制方面我們採用 javascript 來做所有在 client 端的控制，也就是 javascript 根據使用者下達的動作將之轉換為相對的 request 送回 SBS。在 SBS 中除了必要的 hyper link document 外其他的 document 都是由 cgi 程式動態產生的，換句話說 bbs.html 的內容是動態的，連一些在 client 端負責控制的 javascript 或 form 也是由 cgi 程式動態的產生，這樣做的原因是為了增加 SBS 的安全性(security)，當使用者在系統的任意位置嘗試著 add bookmark 時，此動作只把 SBS 和使用者間的介面 bbs.html 之 URL address 加到 client 的 browser 中，而 bbs.html 的初始值就是要求使用者 login 到 SBS 的畫面。

SBS 所選用的資料庫系統為 Microsoft Visual Fox Pro 3.0，SBS Function Unit 以 Microsoft Visual Fox Pro 3.0 完成，SBS Function Unit 與 web server 結合的部分採用的是 Fox Web 公司的 Fox Web 1.2 這套工具，而 web server 我們採用的是由 O'Reilly 公司的 website 1.1，整個系統可以架構在 Microsoft 95/NT 下。

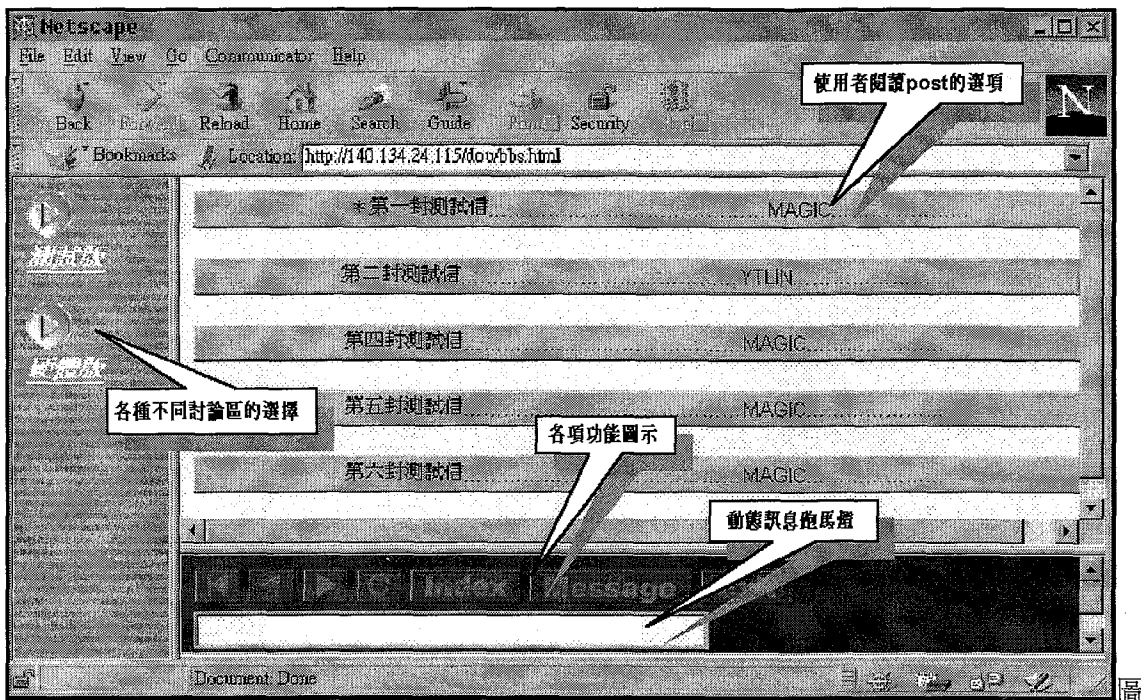
5. SBS 系統展示

SBS 的啓始畫面如圖四所示，SBS 系統要求使用者做 login 的動作，在使用者尚未正確的 login 之前系統的其他部分是不可動作的。在使用者正確的 login 到 SBS 後系統自動進入閱讀 post 的功能畫面，如圖五所示，此時在畫面左邊的 frame 是 news group 切換的功能，每個 news group 將 list 於左邊的 frame 上，而位於畫面下方的 frame 則是一些功能圖示擺設與系統動態資訊跑馬燈顯示的地方。在圖六我們展示了 SBS 在傳送訊息給 online user 的畫面，中間的 frame 在選擇新訊息的接收者和心訊息的輸入列，

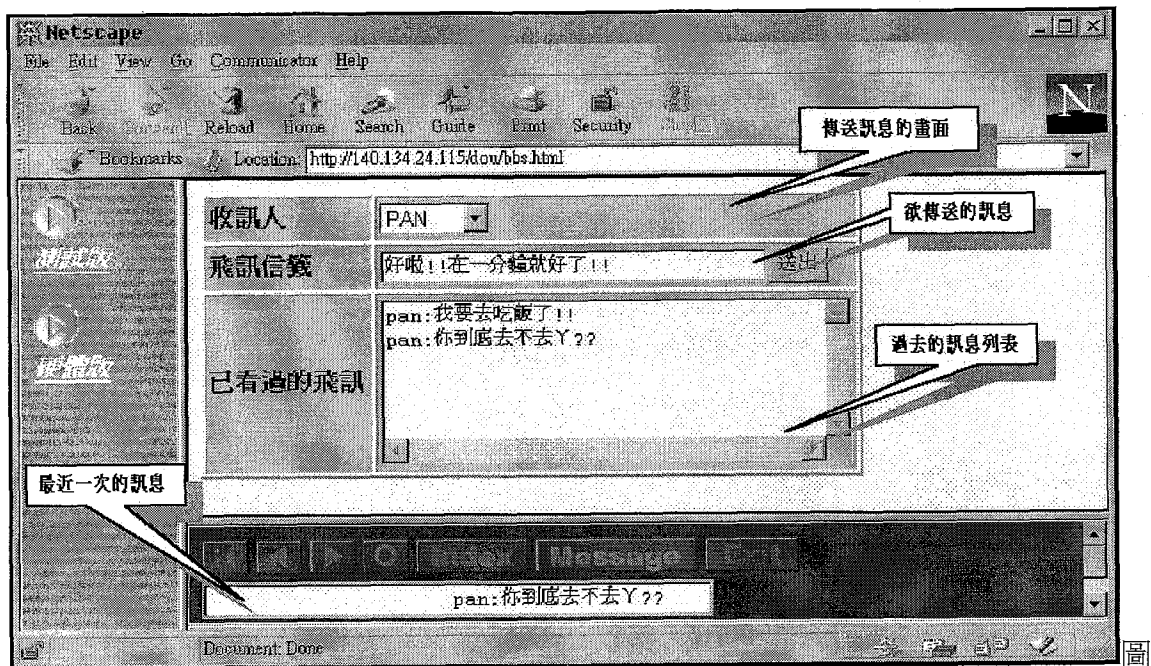
另外還有使用者本身已收到的訊息列表。若有新的訊息傳來，在下方 frame 的跑馬燈將顯示最新的訊息，原來的系統動態資訊將被覆蓋消失。



圖四、要求使用者 login 的畫面



五、使用者閱讀 post 的畫面



六、使用者傳送訊息的畫面

6 結論與未來工作

在這篇文章中我們提出 BDB 這種新的 BBS 系統架構用以解決傳統 BBS 系統所遭遇的問題，並依照 BDB 實作出 SBS(Simple BBS system)來驗證 BDB 架構的正確性。根據統計 WWW 在 TANet 上的使用量以超越 TELNET 並且還在持續的增加中[5]，所以我們選擇了將 SBS 實作成 Web-BBS 的方式，並且已得到初步的成果，SBS 的所有功能都能可以正常的運作。

現在 SBS 採用 cgi 的方式來存取 database，使用 cgi 的方式在 server 端系統資源的消耗與執行效率上的有負面的影響[14]，所以在未來我們希望結合 java 以 three tiers 的方式來實作 SBS[14]，在 server 端執行一個 java program server 當作 stand alone 的 server，並且在使用者的 WWW browser 上使用 java applet 當作 client 端，除此外更希望朝向分散式的方面來發展整個 SBS 的系統。

參考文獻

1. 竇其仁(1995), "台灣學術網路上 BBS 使用情形之探討", TANet'95 (pp.102 -108), 國立中央大學, 1995.
2. 林昱仁、沈嘉峻(1996), "Client-Server BBS 之 Protocol", TANet'96, 行政院國科會國家高速電腦中心, 1996.
3. 曾元顯(1996), "一個製作在全球資訊網路上的電子佈告欄", TANet'96, 行政院國科會國家高速電腦中心, 1996.
4. A.Damas A.Almeida, L. Roque A. D. Figueiredo(1995), "HyperBBS A World Wide Web Bulletin Board System", <http://lila.dei.uc.pt/~hbbs/hbbspaper.html> .

5. 謝淵傑、宋明弘、張雲景(1995), "我國全球資訊網(WWW)使用現況與未來發展方向", TANet'95 (S42-45), 國立中央大學, 1995.
6. 徐黎鎮、趙國仁、陳年興、何扭今(1995), "TANet BBS的發展經驗與管理策略", TANet'95(pp80-85), 國立中央大學, 1995.
7. 葉炳倉, "CGI 程式設計", 元智資訊講座, <http://www.yzit.edu.tw/>.
8. 火昌華, "CGI 程式設計進階", 元智資訊講座, <http://www.yzit.edu.tw/>.
9. 余善真, "WWW SERVER 端程式設計, CGI 程式設計", 元智資訊講座, <http://www.yzit.edu.tw/>.
10. 鄭鳳生, "Implementing Groupware using WWW Technologies", 元智資訊講座, <http://www.yzit.edu.tw/>.
11. 葉良偉 (1997), "Viewdb:一個可應用於 WWW 與資料庫整合之視覺化系統", 逢甲大學資訊工程研究所碩士論文.
12. VPE, Inc., "WebBUILDER", <http://www.webbuilder.com/>.
13. Webgenie, Inc., "CGI*Star", <http://www.webgenie.com/Software/Cgistar/>.
14. Nick N. Duan, Bell Atlantic (1996), "Distributed Database Access in a Corporate Environment Using Java", Fifth International World Wide Web Conference, Paris, France, May 6-10, 1996.
15. P.Freeman (1987), "A Conceptual Analysis of the Draco Approach to Constructing Software System", IEEE Trans. Software Eng.(pp.830-844), July 1987.
16. Pyung-Chul Kim (1996), "A Taxonomy on the Architecture of Database Gateways for the Web", <http://grigg.chungman.ac.kr/~uniweb/documents/taxonomy/text.html>.
17. Stathes P. Hadjiefthymiades, Drakoulis I. Martakos (1996), "A generic framework for the deployment of structured database on the World Wide Web", Fifth International World Wide Web Conference, Paris, France, May 6-10, 1996.
18. Stathes P. Hadjiefthymiades, Drakoulis I. Martakos (1997), "Improving the performance of CGI compliant database gateways", Sixth International World Wide Web Conference, Santa Clara, California U.S.A, 1997.
19. December John and Ginsburg M (1995), "HTML & CGI Unleashed", Sams.net Publishing.
20. Constantinos Phanouriou, Marc Abrams (1997), "Transforming Command-Line Driven Systems to Web Applications", Sixth International World Wide Web Conference, Santa Clara, California USA, 1997.
21. Advent Network Management, Inc. (1997), "JAVA SNMP Package Tutorial", Advent Network Management, Inc. Press.