網頁式之網路管理資訊系統

李思宏　　蔡錫鈞
資訊管理學系
國立暨南國際大學
南投縣埔里鎮大學路 1 號
leeh@im.ncnu.edu.tw　　tsai@im.ncnu.edu.tw

摘　　要

本論文介紹筆者以 JAVA 物件導向程式語言所研發的一套網路管理資訊系統。在我們所設計的網管系統，使用者只需要使用瀏覽器就能操作本系統，不需擔憂所使用的作業平台，因為 JAVA 具有跨平台的特性，不論是使用者層 (user level) 或是管理層 (management level)，都沒有此類的問題。本系統可加入資料挖礦(Data Mining)的技術，可以針對所收集到的資料，進一步地粹取出有用的訊息，以作為網路管理政策的規劃及訂定所用，並且可預防及偵測電腦犯罪及網路犯罪等事件的發生，杜絕色情網站及非法軟體網站的設立。本系統架構是建立在模組化的精神上，使得它極具彈性，隨時可以增減各項功能。每個功能同時也是一個物件，使得程式得以重覆使用。它們之間的溝通，是採用 SNMP (Simple Network Management Protocol)協定，這協定已成為網路界的基本標準，所有的網路產品均有支援。最重要的一點是本系統已在本校之網路管理系統試用。

關鍵字：網頁式、SNMP、RMI、網路管理、資料挖礦

壹、前言

伴隨著各種新型服務及網路資源陸續地出現，人們對網路的依賴也相對地提昇。網路頻寬的高度需求及使用品質的保證更是當前的熱門話題，而網路管理人員自然地要負起這些使命。為了達到上述的目的，網路管理人員必須分析網路上的流量，統計出錯誤的封包所佔的比率、各個通訊協定的使用情形、找尋網路的瓶頸及估計未來的使用量。經由使用 SNMP 及相關的通訊協定，就可以處理相關的事宜。

目前市場上的網管軟體皆為針對個別的作業平台所開發，而且往往價格偏高，以這種情形來看，就不是一般企業或學校所能負擔的了，若再加上教育訓練的成本，那就更可觀了。而目前這些軟體又沒有模組化的相關軟體可供使用者選購，使用者被迫連同購買了很可能用不到的部份。另外，不同的企業可能購買不同的的網管軟體，相關人員在工作變遷後，就得重新學習另一套網管軟體才能達到其工作需求。基於上述理由，我們開發了一套跨平台和網路系統的網管軟體。

1995 年，S. Chutani 和 H. J. Nussbaumer 提出了“網路管理與系統層次的偵錯”，同年 J. Li 和 B. J. Leon 對 SNMP 網路管理系統提出一套正規化的方法，另外 R. Konopka 和 M. Trommer 發表了多層次式的 SNMP 網管架構，而 M. R.Siegl 和 G. Trausmuth 則對 SNMPv2 提出階層式網管的原理及雛形，W. Stallings 於 1998 年下季針對 SNMPv3 的安全

性發表了一篇研究論文。Liebert 企業於 1996 年發表了一篇白皮書，指出經由 SNMP 可以做到更好的網路管理。在 1997 年，F. Stamatellopoulos, G.. Koutepas 及 B. Maglaris 提出了一個網路安全的架構， 建立了一套系統安全 MIB (System Security MIB)[7]。J. S. Han, S. J. Ahn 及 J. W. Chung 於 1998 年針對 Web Server 提出了一個網頁式的效能管理系統，而同年 M. Sharott, G. Hall, S. Fukui, W. Shibata 及 A. Enjou 共同發表了一篇有關多位管理者可以同時使用的網管系統。1999 年國內逢甲大學資工所 J. Y. Chen, D. L. Yang 及 A. C. Liu 提出了“以 MODEL (Managed Object DEfinition Language)為基礎的物件導向化拓撲規格。上述學者所做的研究，以提出理論架構為主，大部份並無真正實作系統。我們首開國內之先例以 JAVA 開發網路管理資訊系統，而這系統是以 SNMP 協定為網管核心。

本篇論文共分成五章節來說明：前言、動機、網路管理資訊系統、應用及未來工作。本文首先以前言來敘述整個網管界的情形及產品定位，在研究動機的章節裡，再點出各個網管軟體的缺失及不足的地方，說明我們研究的起因。在網路管理資訊系統的章節裡，共分成五個小節來敘述本系統，分別為系統架構、核心技術、已完成的系統功能、系統研發及實驗環境和本系統之功能定位及其優劣勢。最後一章則是討論本系統可能應用的領域及未來的發展。

貳、動機

在 JAVA 問世以前，其他程式語言皆有作業平台的限制，使得市面上的軟體常需限制其使用平台。此外，這些商業軟價位都很高，基於經費上的考量常無法滿足需求，加上顧慮到版權的問題，買了幾套的版權才能裝幾台的電腦，往往造成使用上的不方便。此外，這些軟體都很“專業”，內行人都得模索好一陣子才懂得操作，更何況是外行人。網管人員變換工作單位，則意味著一切需重新學習，因為不同的單位也許購買不同的網管軟體。因此對網管人員來說，一套低價位、易學易用的網管軟體是迫不及待的。

由於資訊產品的生命週期很短，有些軟體公司為了趕著把產品推出市場，這些產品往往還有很多“臭蟲”(bug)沒有捉完，就匆匆上市。因此提供了使用者錯誤的資訊，造成決策上的失誤。倘若自行開發相關的軟體，伴隨著運作的透明化，錯誤率相對地減少，同時又能在最短時間內完成系統的修正。我們自己研發的軟體，就能公開讓很多人一起研究，發現有問題的地方，更可以馬上做測試及修正，根據該程式語言的特性，達到最佳化的效果。JAVA 程式語言的種種特性，正可以達到我們想要的目的。物件化的設計精神，使得程式碼可以重複使用，繼承的觀念，使得程式寫得更輕鬆，系統在模組化的架構下更具彈性。

許多軟體在資料儲存方面，都自訂格式，造成資料的使用上失去彈性及無法多方面應用，只有該軟體本身才能比較正確讀取先前的資料。我們企圖把網管系統與市場上一般化的資料庫做結合，使用者可以自己選擇所熟悉的資料系統，而所儲存的資料就能輕易的在不同資料庫間做轉換，這是為了方便將來整合資料挖礦和專家系統的技術。國內曾有學者以一般文字檔的方式來儲存所收集到的網路資料，這種方式既不能及時的分析最新的資料，而且限於一人使用而無法多工。

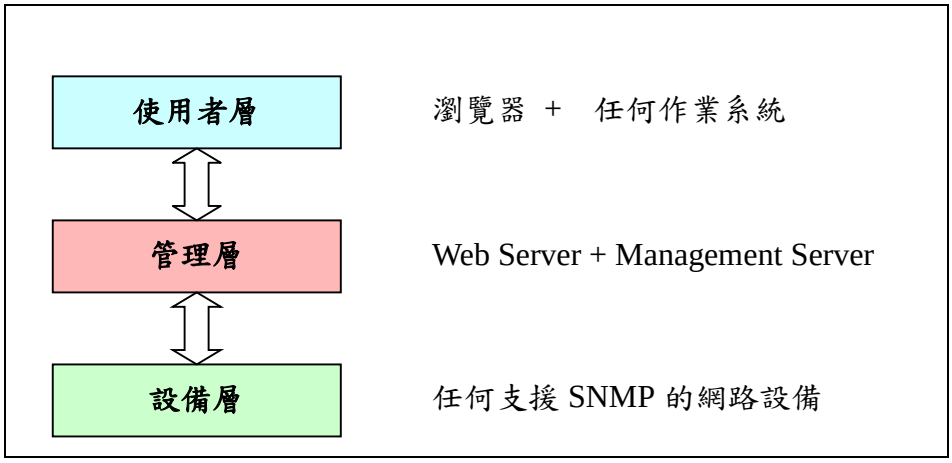
「網頁式的資訊系統」是近年來的熱門話題。使用者經由瀏覽器，就可以使用到相關的資訊系統。圖形化及統一化的操作界面，讓使用者能在很短的時間內就能

操作相關的系統。此類型的系統具有「中央集中管理的特性」及「主從式的運作架構」。「中央集中管理的特性」使得系統得以一致性的版本，使用者能獲得最新及同步的資訊，管理者能輕易地更新所有功能及建立資訊的索引化。「主從式的運作架構」讓工作負擔不是單獨落在伺服器或工作站的身上，而是以最佳調配法分派工作到伺服器及各個工作站。此架構很明顯地節約了電腦資源的使用，記憶空間的使用比起單機版系統來得有效。綜合以上種種因素，促成了我們研發一套低成本的網路管理資訊系統，而這套系統具有跨不同作業平台的特性、分散式的架構、模組化的功能、易學易用的優勢，以及前瞻性的潛力。

參、網路管理資訊系統

3．1　系統架構：

本系統整體架構嚴格上來說可分成三層：使用者層(user level)、管理層(management level)及設備層(appliances level)。使用者層為一台具有 Web 瀏覽器的電腦。此瀏覽器必須支援 JAVA 的執行環境，目前兩個最主要的瀏覽器：Netscape 及 Internet Exporer，在 4.0 以後的版本都可以支援 JAVA。管理層為一個 Web Server、不同功能的管理程式及資料庫系統。而設備端顧名思義，則是由許多的網路設備所組成，這些設備必須全部支援 SNMP 協定（請參閱圖一）。



圖一、系統三層架構簡圖

每個使用者在瀏覽器輸入網路管理站的網址之後，最先看到的是管理層的 Web Server。在眾多的選項中挑選了所想使用的系統後，經由 Web Server 的轉接，便由對應的網路管理程式接手，進行所規劃的工作。

3．2　核心技術

本系統同時涉及了「網路」和「資料庫」兩大領域。在「網路」方面，SNMP 為最主要的核心技術，MIB 則讓我們知道「要問甚麼問題？」及「答案是甚麼意思？」，而 RMON 更提供了高階的資訊，補足了 MIB-2 的不足之處。RMI 和 JDBC 兩者同為 JAVA 的技術名詞縮寫，RMI 提供了遠端物件叫用的服務，克服了 Applet 安全限制的問題；而 JDBC 則為系統連接資料庫的大功臣，經由 JDBC 的仲介服

務，系統能很方便且彈性的使用資料庫。以下章節，我們逐一地介紹這五大技術。

３．２．１　SNMP

SNMP 為 Simple Network Management Protocol 的縮寫，中名譯名為簡易網路管理協定。SNMP 是建立在管理端(Manager)／代理端(Agent)的兩層的關係架構上。它之所以被稱為“簡易”，是因為在代理端只需要很小的程式，而大部份的運作都是在管理端。SNMP 是利用 UDP 快速的特性，做資訊及時的反應，而不需把資源浪費在建立連線上。

３．２．２　MIB

有了 SNMP，網路管理者可以從「管理端」來管理「代理端」。但是管理端必須知道代理端提供了那些資料才能進行資料的蒐集、解釋及分析。這些資料都存在一個稱為 MIB(Management Information Base)的資料庫中，除了標準的 MIB 之外，不同的廠商可能會有自己的 MIB，以加強自身的產品優勢。

３．２．３　RMON

MIB-II 提供了管理個別設備的能力，但卻對於網路整體的狀況顯得無能為力。RMON(Remote Network Monitor)則彌補了這個缺點，它提供了我們更上層及更直接的訊息，例如網路上有哪些網路協定、哪一些協定吃掉了大部份的頻寬、那幾台電腦用最多資源等。

３．２．４　RMI

RMI 的全名為 Remote Method Invoking(遠端物件召喚)，是物件版的遠端函式叫用(RPC)。它讓 JAVA 物件在不同的機器上互相溝通，使用端引用這物件所提供的方法，感覺就如使用自身的方法一樣。例如使用端只需要求資料庫物件把資料欄做加總的動作，之後傳回該值，其運作效率相對於使用端傳回所有的資料後，才做加總的動作，是更有效的。

３．２．５　JDBC

JDBC 全名為 Java Database Connectivity，透過 JDBC，將可輕而易舉地在任何虛擬關聯式資料庫系統執行 SQL 指令。換言之，不再需要為 Sybase、Oracle、Informix、MS-SQL server 等各種資料庫撰寫不同的程式。

３．３　系統模組

目前已實作的部份可分為五大部份來說明：1.資料結構(Data Structure) 2.程式庫(API) 3.管理伺服器(RMI Server) 4.網頁應用程式(Applet) 5.應用程式(Non-Applet)。一般使用者只有真正的感受到第四項(Applet)的存在，但也都間接地使用到另外四項。這五大部份又可歸類成『共通性模組』或『區域性模組』。『共通性模組』會在本節裡分成三個小節來介紹，這部分可以直接套用到任何的地方直接使用。『區域性模組』目前是以暨大為應用的範例，將在第四章詳細介紹。

３．３．１　資料結構：

(1) MibTreeDS

它有五個欄位，其中 Parent、 Sibling 及 Child 為指標(pointer)，內容為指向自己同型態的點或空值(Null)； No 為整數值，記錄 MIB 物件的 OID 編號；Name 則為該 MIB 物件的名稱。

(2) routerInfo

為了讓系統能自動地向路由器(router)索取資料，同時又兼顧模組化的設計，經由資料

庫作為中介者，就能儲存每個路由器的相關資訊。負責索取資料的程式，由資料庫取得上述資訊後，為每個路由器建立個別的點，互相地串起來，每個點就可以交由一個 Thread 去處理。這資料結構由五個部份組成， IP 欄位記錄該路由器的其中一個 IP 位址，Community 記錄使該路由器的合法帳號， NumIf 為該路由器的介面卡（網路卡）數或是特定某片網路卡。Packet 為指向 SNMP 封包的指標，把要問的問題等資訊編碼成 SNMP 的格式，用位元組陣列(byte array)儲存；Next 的形態為指標，指向下一個 routerInfo 的點。

(3) SnmpPacket

SnmpPacket 是兩個資料結構的統稱，顧名思義是 SNMP 的封包資訊。VarBind 為四個欄位所組成，為一個 MIB 物件最基本單位，包含了一個物件及其相關值。Oid 記錄該物件的 OID 編號。value 為 String 型態，記錄該物件的值。不論其原始型態為何，一律以 String 的方式來儲存，而 type 則記錄該原始值的資料形態，next 連接了下一個 VarBind 點。ans_struc 除了擁有本身六個欄位的資訊外，更連接了一個以上的 VarBind 點，Version 記錄了目前所使用的 SNMP 協定版本。Community 記錄該路由器使用的帳號，RequestID 記錄此 SNMP 封包的編號，Err_status 記錄處理此封包時的錯誤訊息。PDU 為 SNMP 的操作指令，如 get-Request 等。NumObject 記錄了後面連接了多少個 MIB 物件(VarBind)。

(4) ImageRectangle

這個資料結構繼承自 java.awt.Rectangle 的屬性，同時外加上兩個欄位 loc 及 stat。Rectangle 是個四方形物件，提供了許多方法，其中 contains()可以判斷其個座標(x,y)是否被它所包含。我們利用了這特性來作圖形界面的選擇器，當在滑鼠按了一下，滑鼠事件監聽器(Mouse Listener)會傳回該點的座標值，我們再將此值傳給 Rectangle 的物件來判斷是否在它的範圍內。外加的 loc 是用來記錄 Rectangle 的位置，把每個預定的位置用 Vector 的結構集合起來，每個在 Vector 中的元素就是 ImageRectangle 實作。而 stat 則是 status 的縮寫，為記錄這四方區域是否被選，其資料型態為布林乙函數值(boolean)，「是」代表被選，「否」則相反。

３．３．２　API：

(1) SNMP 編碼器

我們將四個參數傳給 SNMP 編碼器，它就會負責把這些參數分解並組合成標準的 SNMP 封包傳回給我們。此封包交給傳輸層的 UDP 介面並往下加上每層適當的標頭就可以送出去。

(2) SNMP 解碼器

SNMP 解碼器會幫我們把 SNMP 的封包，從一個位元組陣列(byte array)解讀成我們所能理解的形式。參考３．３．３節所提到的資料結構 ans_struc 及 varBind，所解碼出來的資訊全以這種方式儲存。使用時，我們只要傳入一個位元組陣列，就可以得到 ans_struc 的傳回值。其資訊是以 TLV(Type, Length, Value)的方式儲存，T 是型態（那一類的資料），L 代表長度（接在後面的資料有多長），V 是值（真正資料的部份）。

３．３．３　RMI SERVER：

(1) DbBase

資訊就是一種無形的資產，它具有無窮盡的潛力。如何在一堆資料中取得有用的資訊，這就是資料挖礦的精神所在。要有資訊，就得先要有一堆的資料，而這些資料就必須用某些管道取得後儲存起來。資料庫在這方面就

扮演了極重要的角色。由於 Applet 有安全性的限制，無法和 Web Server 之外的第三者溝通，因此我們藉助了 DbBase 來達成此項任務。它作為系統和資料庫的仲介，負責資料存取、插入、刪除及更新的任務。

(2) MibBase

MIB 在３．２．２節裏有詳細的介紹，它呈現樹狀，一棵龐大的樹。為了克服 Applet 的安全限制，我們創造了 MibBase。MibBase 是一個負責處理 MIB 相關事務的 RMI Server。它在啟動時，會自動讀入一個名為 MibFile.txt 的檔案。該檔案記載了 MIB 的所有資料，MibBase 會依此資料建立實際的 MIB 樹狀結構，所使用到的資料結構是３．３．１節所提到的 MibTreeDS。

(3) SnmpBase

網路管理就表示必然會管理到其他台電腦與設備，而 Applet 的安全限制卻使我們難以和其他台電腦溝通。因此我們創造了 SnmpBase 來傳送 SNMP 封包給任何一個網路點。它總共提供了兩個方法，其差別在於一個要傳入“逾時之預定時間”，另一則不需傳入逾時參數而採用預設值（三十秒）。

(4) ToolBase

使用網路常需要使用到一些小工具，如 Ping、Traceroute、nslookup、Port Checker 等，來測試網路的狀況或查詢網域名稱等。我們把這些應用類的小工具全集中在 ToolBase 內實作。

肆、應用

本系統可應用的層面很廣，只要擁有支援 SNMP 協定的網路設備，本系統就能和它做資訊的交流。近年來，國家資訊基礎建設的列車已開到中、小學，使得我國上網人數以倍數增加，因此各校都必須有專門人員來管理網路相關的設備。因此本系統正好可提供給欠缺經費與人力的學校的一個很好的工具。

往下是到基層網路的管理，往上則是決策的參考，各個區域網路中心都可以應用本系統來得知每個子網路的流量、協定的使用情形；找出網路流量的瓶頸，從而改變該段網路的拓撲設計，或是考慮改用更適當的網路儀器。另外，當發現某個子網路異常過度使用某個協定（例如 DNS），這也許意味著該網路可能存在錯誤的設定，造成很多電腦一直往外查詢網址。這當然也可能是該區網 DNS Server 出了問題，因此使用本系統就能協助網管人員及早發現這類問題。

本章節之後續部份，我們將以實際的例子來介紹本系統的應用。目前本系統以國立暨南國際大學為示範地點，把區域性模組結合第三章所提到的通用性模組，形成實用性的網路管理系統。區域性模組分為 Applet 及 NonApplet 兩種。Applet 為使用者常接觸到的介面，在網頁中的呈現互動性的效果。NonApplet 則為幕後英雄，它們負責收集資料等工作，讓前端的 Applet 可以發揮特定的功效。

４．１　Applet

(1)Ping

本系統可用來測知某台機器是否斷線。使用本系統可以讓使用者不必進入指令模式，就可以使用上述功能，因為很多使用者不太熟悉指令模式，應用本系統（圖形化介面）可以讓他們分擔部份網路管理者的工作，經由他們的報告，網管人員能更快掌握實驗的狀況。

(2) Traceroute

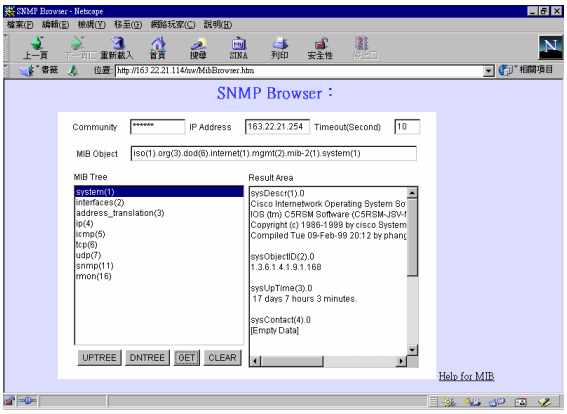
這是一個查詢網路行經路徑的系統，每次所查詢到的結果不一定都一樣，這是決定於每顆路由器的遞送演算法(routing algorithm)。

(3) Port Checker

本系統可讓管理者查詢任何一台電腦的埠(port)，是否正被使用。當管理者接到反應說 FTP server、WWW server 或其他的 server 有問題時，管理者可先使用本系統查知相關程式是否已經啟動，之後才做進一步的處理。

(4) Nslookup

本系統為一網路位址正／反查詢系統，當你輸入 IP 位址(IP Address)時，系統會告訴你其相對應網域名稱(Domain Name)。相反地，你若要查 IP 位址，則只要輸入網域名稱就可以了。事實上，網路世界裡，電腦與電腦之間的溝通，其實是用數字在溝通，所以您的網路主機其實不認得這個 domain name，它一定要有 IP 才知道要把訊息送到哪裡。這就是 DNS 的作用了，它能夠幫您把 domain name 轉換成 IP，使您能夠順利找到您要去的地方。因此 ToolBase 管理程式所執行的那台電腦必須設定好 DNS 等相關設定，本系統才能正常運作。



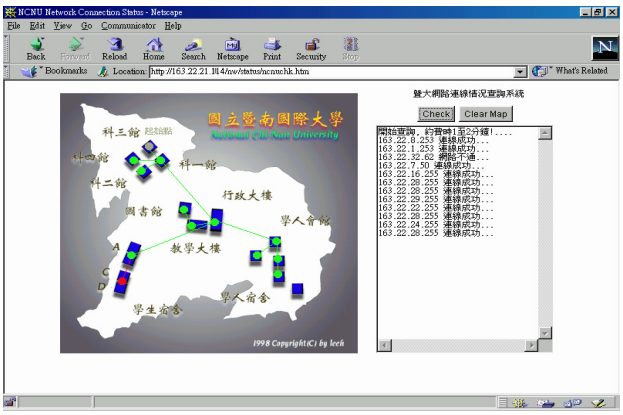
圖二、SNMP 瀏覽器

(5) SNMP 瀏覽器

本系統屬於比較低階的管理工具。研發的目的是為了讓有經驗的管理人員，針對個別的設備做低階且較彈性的查詢。使用者必需具備 MIB 樹狀結構的背景知識，才能知道要查詢的物件是位於哪一個目錄之下。另外，我們利用了『網頁式』的特性，把大部份 MIB 物件的詳細資訊，用網頁來表示，使用者遇有不明白的地方就能及時地線上查詢。

(6) 網路連線情形查詢系統

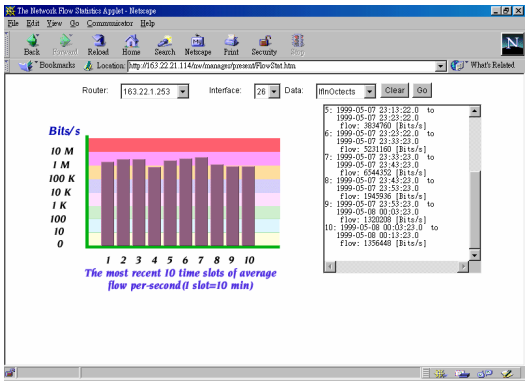
本系統目的在於方便網管人員能在最短的時間內，一目瞭然地檢測網路之連線情形，這屬於及時性的處理，若沒有本系統，管理者必須在 dos 指令模式下各別針對不同的機器測試其連線情形。開始使用時，管理者只要在網頁上設定每個網路點的資料，以後就不需再設定，只要輕按一下<Check>的按鈕即可獲得所需的資訊。



圖三、暨大網路連線情況查詢系統

(7) 專業網路流量查詢系統

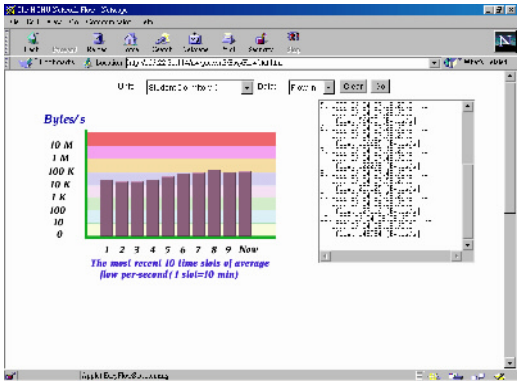
本系統屬於模組化的架構，嚴格上來說是屬流量查詢的部份，前半部為一個永生的資料收集程式，每隔十分鐘就向路由器(router)送出 get-request 的要求，等路由器回覆訊息後，就進行解碼(decode)的動作，並把資料存入資料庫中。而後半部就像下圖所示，當使用者按下<go>按鈕時，再到資料庫查詢相關的資料，因有 Java applet 的安全限制，故本系統則使 DbBase 管理程式來當中介者，進行資料庫的存取。



圖四、專業網路流量查詢系統

(8) 簡易網路流量查詢系統

本系統大致上和『專業網路流量查詢系統』的運作模式是一樣的，存取同一個的資料庫的資料。但為了便於決策者或一般人使用選單上是以單位為主，並非主要單位上使用的介面卡，在選單上就不列出以免造成使用者的困惱。



圖五、簡易網路流量查詢系統

圖六、台灣學術網路資料設定系統

(9) 台灣學術網路 TANet 資料設定系統

若只有一所學校的範圍，我們是可以收集全部路由器上的流量資料。但是當把範圍擴大到全台灣時，就會造成額外嚴重的網路流量，反而違反了系統的宗旨。基於這個理由，我們設計了這個設定系統，經由圖形化的介面，使用者可以直接在地圖上選擇所要查看的點;儲存後，再到流量查詢系統查看。此時另有一個“永生”的程式每隔三十分鐘就會檢查此設定值，負責向所有被選的路由器要求流量資料。

４．２　NonApplet：

(1) getOne

此程式屬於文字模式的系統，必須在命令模式下執行，它能連續取得某一 MIB 樹的整個目錄資料。使用時必須傳入路由器的 IP 位址、帳號及物件識別號，這程式就會依此物件識別號為基準，往下查詢所有相關資訊。

(2) getData

這程式為“永生”的長駐程式，平時都

在睡覺，每十分鐘醒來產生多個 thread 來執行其任務，完成後就繼續睡覺。其任務就是到資料庫取得路由器的相關訊息，並向各個路由器索取每片介面卡的流量資料，再把資料及時間存到資料庫內以供其他應用。

(3) 應用程式：getTanet

這程式的設計精神和 getData 很像，但資料索取頻率較低，也就是說它睡的比較久，目前設定為三十分鐘。另外，配合『台灣學術網路資料設定系統』的使用，程式只要求某片特定介面片的資料而已，因此可以降低整體的流量，避免對其他區網中心的影響。

伍、結論與未來發展

和市場上一般的網管軟體相比，本系統的成本很明顯地降低非常多，而且不會隨著作業系統起舞。若以 OSI 的功能模組來看的話，本系統將橫跨設定(Configuration & Device diagnostic)、錯誤(fault)及效能(performance)這三方面的領域，另外若以最新的技術來評比的話，本系統可結合資料挖礦及資料倉儲等領域。其競爭優勢是很顯然的。

目前 IP 位址已快不敷使用，故 IPv6 快取代 IPv4，這很可能又是一次資訊危機。而 JAVA 很早就考慮到這問題，只要把 InetAddress 的類別(class)替換就可以解決這個問題了，因此本系統比起用其他語言所開發的軟體來得強固多了。

本系統是以網頁式的設計精神為主，因此自然承續了網頁式集中管理的優點，只要更新一台機器的系統，其他電腦馬上就能同步使用到最新的服務。其他的 SNMP 網管軟體安裝時，必須同時也安裝 MIB 資料庫才能使用，造成資源的重複浪費。而本系統的 MibBase 在啟動時會把 MIB 讀入，其他電腦使用時只要用所需要的部份即可。

本系統的缺點在於，當作為使用者和系統之間橋樑的 web server 停擺時，或使用者到 web server 之間的網路中斷時，整個系統就不能正常的運作。不過這也是所有 Web-based 系統所有共同面臨的問題。

我們是希望短期內能把各個協定分析及統計的功能加入，以強化整個系統的功能。雖然說微軟(Microsoft)產品的普及率很高，但為了讓使用者在資料庫方面不受限特定的平台，我們計劃也同時會採用 unix 作業系統上的資料庫系統來一同實作，使整個系統更具彈性。中期目標方面，我們會加入針對整個 Tanet 網路使用情形的查詢系統。對於網路使用者而言，長時間不停收集到的資料，若不能充分的利用是很可惜的。因此我們的長期目標，是計劃把本系統與資料挖礦的技術整合起來，針對各個資料做不同的分析，以供策略考量時的參考之用。

美商昇陽(Sun)公司最近發表了 jini 的技術。這技術讓家電也進入網路化的世界，許多生產家電器材的大公司紛紛加入，讓家裡的電腦能控管所有的家電。而 jini 本身就是用 java 所設計的系統，燒錄在晶片上。其核心技術和 RMI 的原理是一樣的。因此，本系統未來將會實驗如何和 jini 系統結合及溝通。

參考文獻

[1] F.Stamatellopoulos, G..Koutepas and B.Maglaris, “System Security Management via SNMP”, *HP OpenView University Association Workshop, Marid, April 1997*, p1-11

[2] Jeng-Yueng Chen, Don-Lin Yang, and An-Chi Liu, "A Model-Based Object-Oriented Topology Specification For Network Management", *1998 Intenational Computer Symposium, Workshop on Computer Networks, Internet, and Mutimedia*, p164-169

[3] Jeong Soo Han, Seong Jin Ahn and Jin Wook Chung, "Web-based Performance Manager System for a Web Server", *0-7803-4351-4/98 $10.00 1998 IEEE*, p1-10

[4] Jianxin Li and Benjamin J.Leon, "A Formal Approach to Model SNMP Network Management Systems", *0-8186-7180-7/95$04.00 1995 IEEE*, p284-287

[5] Manfred R.Siegl and Georg Trausmuth, "Hierarchical Network Management : A Concept and its Prototype in SNMPv2", *Proceedings JENC6*, p122-1

[6] Michael Sharott, Greg Hall, Shingo Fukui, Wakako Shibata and Atsushi Enjou, "Multi-Operator Access to a Network Management System", *0-7803-4351-4/98/$10.00 1998 IEEE*, p334

[7] R. Konopka and M. Trommer, "A Multilayer-Architecture for SNMP-Based, Distributed and Hierarchical Management of Local Area Networks", *0-8186-7180-7/95$04.00 1995 IEEE*, p272-279

[8] Sailesh Chutani and Henri J.Nussbaumer, "Network Management and System-Level Diagnosis", *0-8186-7180-7/95$04.00 1995 IEEE*, p280-283

[9] "The Simple Times", http://wwwsnmp.cs.utwente.nl/

[10] Tomas Isakowitz, Michael Bieber and Fabio Vitali, "Web Information System", *Communictions of the ACM, July 1998/Vol.41*, No7, p78-80

[11] William Stallings, "SNMPv3: A Security Enhancement For SNMP", *IEEE Communications Surveys, Fourth Quarter 1998, Vol. 1, No1*, p1-17