

# 多條供應鏈的資料傳輸系統設計與實作

研究生：張淙亮

指導教授：梁高榮 博士

國立交通大學工業工程與管理學系

## 摘要

本文完成花卉產業多條供應鏈間資料傳輸系統的設計與實作。在設計階段裡，資料的流通可分為兩種傳遞型態：單一或多數的資料接收端。以單一接收端的層面而言，使用檔案轉移協定的方式不但可使資料有效率的傳輸處理且其更具低建置成本的特性，此外，各地花卉公司的交易資料轉移至中央的倉儲系統已為一常見的案例。在多數的資料接收端層面而言，依其爪哇訊息服務技術易於管理使用者資訊的特色，在小行情、大行情、供應人匯款清單，及承銷人結帳清單等交易資料皆已使用於此技術傳送。在實作階段上，爪哇語言的技術皆充分支援於此兩種系統的建置上，本文並以目前花卉供應鏈中每日的資訊傳送過程作為其系統實作成功的案例說明。

關鍵字

供應鏈

檔案轉移服務

XML 文件

爪哇訊息服務

交易資訊處理系統

# Design and Implementation of Data Delivery System among Multiple Supply Chains

Student : Chung-Liang Chang

Advisor : Dr.Gau-Rong Liang

Department of Institute of Industrial Engineering & Management  
National Chiao Tung University

## Abstract

A data delivery system has been designed and implemented among multiple supply chains in flower industry. In the design phase, the receivers of the data delivery system are classified into two types: unique receiver and group receivers. For the unique receiver, the data are delivered using File Transfer Protocol (FTP) from the viewpoint of efficient processing and low software cost. A typical example is the delivery of transactional data from the distributed wholesale markets to the central data warehouse. For the group receivers, the data are processed by Java Message Service (JMS) techniques from the viewpoint of easy management. For example, Individual Statement (IS), Global Statement (GS), Supplier's Account Statement (SAS), and Buyer's Account Statement (BAS) are delivered in such a way. In the implementation phase, Java programming technology is applied to make the data delivery system work for both receivers. The successful experience in the daily data delivery among flower supply chains shows the feasibility of this approach.

Keywords : Supply Chain 、 File Transfer Service 、 XML Document 、 Java Message Service 、 Trading Information Processing System

## 誌謝

在本篇論文的研究過程中，首先要感謝恩師梁高榮教授的指導，除了學科上知識的教導，做事細心嚴謹的態度更讓我獲益良多，使我在面對未來種種挑戰中有所指引，謹此致上由衷的謝意與感激。此外在技術上也獲得漢卿學長與師翰學弟的建議與協助，使系統得以順利完成建置。實驗室的同窗好友詩涵、英泰、楓凱、瀚棠、士凱在研究所修課與撰寫論文的過程中得以互相勉勵與支持。而在論文實作期間，承蒙彰化花卉公司與台北花卉產銷股份有限公司的配合與協助，使合作計畫進行順利，特此致謝。最後，感謝家人與所有陪我度過這段求學歲月的老師與朋友們，希望大家都能過得順遂美好。

本人在研究期間，曾參加農委會漁業署-衛星網際網路資源及資料倉儲技術在漁業上之加值應用研究-91AS-1.9.1-FA-F2、農委會-整合性運銷職能技術研-91AS-1.5.2-FS-#1 和台北花卉產銷股份有限公司，為執行農委會之[台北花市電子交易市集的發展與建置]計畫(計畫編號:91 農科-6.1.3-輔-#1)，與交大進行合作，合作主題為「台北花市資料庫三階正規化、花卉資料倉儲設計及 XML 規格連線系統建置」(交大計畫編號:91B405)，獲得許多資源上的幫助，特此致謝。



# 目錄

摘要.....	I
誌謝.....	III
圖目錄.....	VII
表目錄.....	IX
專有名詞中英文對照.....	X
第一章 緒論.....	1
1.1 研究動機.....	1
1.4 研究方法.....	4
1.5 論文架構.....	5
第二章 文獻回顧.....	6
2.1 資訊電子化問題探討.....	6
2.1.1 時間去偶合性.....	6
2.1.2 空間去偶合性.....	7
2.1.3 同步去偶合性.....	7
2.2 常見資訊傳遞系統架構探討.....	8
2.2.1 訊息傳遞.....	8
2.2.2 遠端程序呼叫.....	8
2.2.3 通知機制.....	9
2.2.4 共享記憶體空間.....	10
2.2.5 訊息佇列.....	10
2.2.6 出版者/訂閱者.....	11
2.2.7 架構比較.....	11
2.3 爪哇訊息服務原理.....	12
2.3.1 爪哇訊息服務與中介軟體.....	12
2.3.2 爪哇訊息服務構成元件.....	13
2.4 爪哇訊息服務程式設計.....	13
2.4.1 連線工廠.....	14
2.4.2 議期.....	15
2.4.3 目的地.....	16
2.4.4 訊息生產者,消費者.....	16
2.4.5 訊息、訊息選擇器.....	17
2.4.6 訊息出版、訊息傾聽者.....	17
2.5 花卉公司供應鏈資訊流.....	19
2.5.1 檔案轉移服務系統.....	19
2.5.2 爪哇訊息服務.....	19
2.5.3 交易資訊處理系統.....	20
2.6 XML技術.....	21

2.6.1 可延伸性加註語言.....	21
2.6.2 可延伸性加註語言基本語法.....	21
2.6.3 XML文件排版樣本.....	22
<b>第三章 交易表單與其XML規格分析.....</b>	<b>24</b>
3.1 整合系統介紹.....	24
3.2 花卉交易資訊傳遞問題探討與架構.....	25
3.2.1 花卉交易資訊傳遞問題.....	25
3.2.2 資訊傳遞架構.....	25
3.3 交易表單規格整合.....	28
3.3.1 表單規格蒐集.....	28
3.3.2 表單分析整理.....	29
3.3.3 表單XML規格制定.....	30
3.4 花卉交易資訊XML文件使用.....	31
3.4.1 XML標準格式文件.....	31
3.4.2 XSL轉換.....	32
<b>第四章 爪哇訊息服務理念與設計.....</b>	<b>36</b>
4.1 爪哇訊息服務傳訊樣式.....	36
4.1.1 傳送/接收傳訊樣式.....	36
4.1.2 出版/訂閱傳訊樣式.....	36
4.2 爪哇訊息服務樣式與花卉公司文件整合.....	38
4.3 文件整合系統開發設計.....	40
4.3.1 整合系統開發流程.....	40
4.3.2 系統下載.....	40
<b>第五章 爪哇訊息服務應用操作.....</b>	<b>42</b>
5.1 爪哇訊息服務應用.....	42
5.2 使用者資料庫設定.....	43
5.3 花卉交易資訊處理系統操作說明—供應人.....	46
5.3.1 參數設定.....	46
5.3.2 啟動接收.....	46
5.3.3 停止接收.....	47
5.3.4 轉入資料庫.....	48
5.3.5 儲存XML文件.....	48
5.3.6 檢視XML文件內容.....	48
5.3.7 加值應用功能.....	49
5.4 花卉交易資訊處理系統操作說明—承銷人.....	53
5.4.1 參數設定.....	53
5.4.2 啟動接收.....	53
5.4.3 停止接收.....	54

5.4.4 轉入資料庫.....	55
5.4.5 儲存XML文件.....	55
5.4.6 檢視XML文件內容.....	55
5.4.7 加值應用功能.....	56
<b>第六章 檔案轉移服務系統設計.....</b>	<b>60</b>
6.1 檔案轉移服務模式探討.....	60
6.1.1 FTP架構.....	60
6.1.2 FTP 程式開發架構.....	61
6.2 Java語言與檔案轉移服務系統.....	64
6.2.1 Java檔案轉移協定API說明.....	64
6.2.2 自動化處理機制.....	66
6.2.3 Java檔案轉移協定系統程式開發.....	67
6.3 自動化檔案轉移服務系統應用.....	70
6.3.1 系統應用領域.....	70
6.3.2 系統功能與操作說明.....	71
<b>第七章 結論.....</b>	<b>74</b>
<b>參考文獻.....</b>	<b>76</b>
<b>附錄一：花卉公司交易表單.....</b>	<b>78</b>
<b>附錄二：出版者軟體設計程式碼.....</b>	<b>82</b>
<b>附錄三：供應商軟體設計程式碼.....</b>	<b>90</b>
<b>附錄四：承銷人軟體設計程式碼.....</b>	<b>102</b>
<b>附錄五：參數設定設計程式碼.....</b>	<b>114</b>
<b>附錄六：自動化檔案轉移服務程式碼.....</b>	<b>117</b>

## 圖目錄

圖 1-1 研究方法流程架構.....	4
圖 2-1 時間去耦合性.....	6
圖 2-2 空間去耦合性.....	7
圖 2-3 同步去耦合性.....	7
圖 2-4 訊息傳遞.....	8
圖 2-5 遠端程序呼叫.....	8
圖 2-6 同步去耦合性.....	9
圖 2-7 同步去耦合性質進階遠端呼叫.....	9
圖 2-8 通知機制.....	10
圖 2-9 共享記憶體空間.....	10
圖 2-10 訊息佇列.....	11
圖 2-11 出版/訂閱樣式.....	11
圖 2-12 訊息服務示意圖.....	12
圖 2-13 JAVA應用程式與MOM產品關係圖.....	12
圖 2-14 JMS管理物件.....	13
圖 2-15 JMS程式架構.....	14
圖 2-16 單一連線與多重連線架構.....	15
圖 2-17 花卉批發資訊分享熱線.....	19
圖 2-18 電子式文件交易模式.....	20
圖 2-19 XML與HTML關係圖.....	21
圖 2-20 表單與其WELL-FORMED XML格式對應.....	22
圖 2-21 XSL轉換架構圖.....	22
圖 3-1 花卉當日交易資訊傳遞過程.....	25
圖 3-2 台北花卉資訊傳遞方式.....	26
圖 3-3 彰化花卉交易資訊處理細節.....	27
圖 3-4 台北花卉交易資訊處理細節.....	27
圖 3-5 XML規格制定流程.....	28
圖 3-6 彰化供應人小行情與XML標準格式文件對照.....	31
圖 3-7 台北供應人小行情與XML標準格式文件對照.....	31
圖 3-8 花卉拍賣資料供應人小行情XSL轉換後畫面.....	35
圖 4-1 傳送/接收傳訊樣式.....	36
圖 4-2 出版/訂閱傳訊樣式.....	37
圖 4-3 花卉公司個人性與公開性資訊流示意圖.....	38
圖 4-4 文件整合系統開發流程圖.....	40
圖 4-5 文件整合軟體下載網頁.....	41
圖 5-1 系統操作流程圖.....	42
圖 5-2 使用者資料庫設定流程.....	43
圖 5-3 ODBC路徑視窗.....	43



圖 5-4 新增系統資料來源.....	44
圖 5-5 選取驅動程式來源.....	44
圖 5-6 ODBC參數設定 .....	45
圖 5-7 使用者資料庫設定完成.....	45
圖 5-8 供應人參數設定畫面.....	46
圖 5-9 與伺服器電腦連線成功畫面.....	47
圖 5-10 供應人小行情的花卉拍賣資料接收畫面.....	47
圖 5-11 轉進資料庫成功畫面.....	48
圖 5-12 檢視XML文件畫面.....	49
圖 5-13 加值功能應用畫面.....	50
圖 5-14 查詢結果以表格顯示.....	50
圖 5-15 查詢結果以長條圖顯示結果.....	51
圖 5-16 小行情花卉品名平均值、變異數查詢結果.....	51
圖 5-17 小行情資料庫查詢結果.....	52
圖 5-18 承銷人參數設定畫面.....	53
圖 5-19 與伺服器電腦連線成功畫面.....	54
圖 5-20 承銷人結帳清單的花卉拍賣資料接收畫面.....	54
圖 5-21 轉進資料庫成功畫面.....	55
圖 5-22 檢視XML文件畫面.....	56
圖 5-23 加值功能應用畫面.....	57
圖 5-24 查詢結果以表格顯示.....	57
圖 5-25 查詢結果以長條圖顯示結果.....	58
圖 5-26 結帳清單花卉品名平均值、變異數查詢結果.....	58
圖 5-27 結帳清單資料庫查詢結果.....	59
圖 6-1 主從式架構.....	61
圖 6-2 FTP通訊協定架構.....	61
圖 6-3 FTP 程式開發架構.....	62
圖 6-4 FTP檔案轉移流程.....	63
圖 6-5 自動化檔案轉移服務於花卉線上分析處理系統位置.....	70
圖 6-6 檔案轉移服務主畫面.....	71
圖 6-7 與選定之IP位址建立連線畫面.....	72
圖 6-8 上傳中的PROGRESS BAR 狀態.....	72
圖 6-9 成功上傳畫面.....	72
圖 6-10 設定自動化畫面.....	73
圖 6-11 確認自動化處理畫面.....	73
圖 7-1 自動化檔案轉移服務跨漁業領域的應用.....	75
附圖 1 FM_SUBSCRIBER.JAVA GUI畫面.....	90
附圖 2 FM_SUBSCRIBER_BUYER.JAVA主要畫面.....	102
附圖 3 PARAMETER_SETUP.JAVA程式畫面.....	114
附圖 4 MAINFRAME.JAVA主畫面與自動化設定畫面.....	117



## 表目錄

表 2-1 架構與去偶合性比較表 .....	11
表 2-2 連線工廠設定 .....	14
表 2-3 連線設定 .....	15
表 2-4 議期設定 .....	16
表 2-5 目的地設定 .....	16
表 2-6 訊息生產者設定 .....	16
表 2-7 訊息消費者建立方式 .....	17
表 2-8 XML文件訊息設定 .....	17
表 2-9 訊息選擇器設定 .....	17
表 2-10 訊息出版設定 .....	18
表 2-11 訊息傾聽者設定 .....	18
表 3-1 彰化花卉公司表單提供方式 .....	29
表 3-2 台北花卉公司表單提供方式 .....	29
表 3-3 個人性表單欄別分析 .....	29
表 3-5 交易表單XML標籤元素組成一覽表 .....	30
表 3-6 花卉拍賣資料供應人小行情的XSL轉換 .....	32
表 6-1 FTPCLIENT建構子 .....	65
表 6-2 FTPCLIENT物件方法 .....	65
表 6-3 TELNETINPUTSTREAM物件方法 .....	65
表 6-4 TELNETOUTPUTSTREAM物件方法 .....	66
表 6-5 「連線」與「斷線」工作項目的主端程式碼 .....	67
表 6-6 遠端列表 .....	67
表 6-7 「上傳」工作項目的主端程式碼 .....	68
表 6-8 遠端資料夾內容在上傳完成後列表 .....	68
表 6-9 檔案存在的判斷 .....	68

## 專有名詞中英文對照

Application Interface, API	應用程式介面
Business to Customer, B2C	企業對客戶
Buyer's Account Statement, BAS	承銷人結帳清單
Cascading Style Sheet, CSS	串接式樣式表
Client-Server	主從式架構
Data Transfer Protocol, DTP	資料轉移通訊協定
Data Transform Service, DTS	資料轉換服務
Document Type Definition, DTD	文件格式定義
Enterprise Resource Planning, ERP	企業資源規劃
eXtensible Markup Language, XML	可延伸性加註語言
eXtensible Stylesheet Language, XSL	可延伸性排版樣式語言
Flower_Trading Information Processing System, Flower_TIPS	花卉交易資訊處理系統
Fish_Trading Information Processing System, Fish_TIPS	漁業交易資訊處理系統
Fishery Information Sharing Hotline, FISH	漁業資訊分享熱線
File Transfer Protocol, FTP	檔案轉移通訊協定
File Transfer Service, FTS	檔案轉移服務
Global Statement, GS	大行情
Graphic User Interface, GUI	圖形化使用者介面
Hyper Text Transfer Protocol, HTTP	超文字標記語言
Individual Statement, IS	小行情
Java Message Service, JMS	爪哇訊息服務
Java Naming and Directory Interface, JNDI	爪哇命名與目錄介面
JMS Provider	爪哇訊息服務提供者
Message Passing	訊息傳遞
Message Queuing	訊息佇列
Message-Oriented Middleware, MOM	訊息導向中介軟體
Multi-Threading	多執行緒
Native Client	非爪哇語言撰寫軟體
Network News Transfer Protocol, NNTP	網路新聞傳輸協定
On Line Analytical Processing, OLAP	線上分析處理
Open Database Connectivity, ODBC	開放式資料連結
Platform Independence	跨平台
Point-To-Point, PTP	點對點網路拓樸
Protocol Interpreter, PI	通訊解譯模組
Publish/Subscribe, Pub/Sub	出版/訂閱樣式
Remote Method Invocation, RMI	遠端方法呼叫
Remote Invocations	遠端呼叫
Remote Procedure Call	遠端程序呼叫



Sender/Receiver	傳送/接收樣式
Shared Spaces	共享記憶體空間
Simple Mail Transfer Protocol, SMTP	簡易信件傳送協定
Supplier's Account Statement, SAS	供應商匯款清單
Trading Information Processing System, TIPS	交易資訊處理系統
Tuple Space	欄空間
Undocumented Package	無文件定義套件
User Interface, UI	使用者介面
Vessel Information Propagating System, VIPS	漁船資訊傳送系統
Well-Formed XML	制式可延伸性文字加註語言格式
World Wide Web Consortium, W3C	全球資訊網國際公會
XML Schema	可延伸性文字加註語言綱要



# 第一章 緒論

本章的內容可以分為五部份，分別為 1.1 節「研究動機」，1.2 節「問題界定」，1.3 節「研究目的」，1.4 節「研究方法」，1.5 節「論文架構」。

## 1.1 研究動機

由於電子化技術的快速發展及網際網路的使用便利，人們在資訊電子化的需求愈來愈迫切，並且在傳統一貫的交易模式下，愈來愈趨向快速而有效率的自動化方法上。隨著科技技術蓬勃的發展，在企業的規模逐漸成長下，在供應鏈資訊傳遞的方式上，必須講求正確與效率，相同產業之間供應鏈更由許多家公司所組成的，而在各公司企業與其顧客資訊上的傳遞系統中，除了資訊內容的私密性外，對於每一顧客而言，單據格式的多樣化的不只是在了解交易內容上較不友善，在單據的獲得上更由於每家公司有不同的方式而難以處理；如在花卉產業中，台灣的花卉批發公司如台北、彰化、台中、…，與高雄等花卉公司，各批發公司皆具其等性質表單，如供應人小行情表單、承銷人結帳清單、…，與大行情表單等，其等性質的表單中卻因所屬花卉公司的不同而格式有所差異，此外在單據取得上，各花卉批發公司更因多種不同的單據取得方式而使各花卉供應商與承銷商在資訊的處理上有所不便，如郵寄、傳真、…，或電子郵件等，在單據取得的方式不一下，資訊整合與在利用的問題即造成各供應或承銷商的困擾，除了表單的蒐集與轉檔外，蒐集的成堆表單內亦不具任何資訊價值。本研究即以彰化花卉公司與台北花卉公司為例，分別取得兩家公司的交易單據，各花卉公司於拍賣日當天將送到該公司的花卉拍賣後，為了將各拍賣資料正確有效率的傳送至每個供應人與承銷人手中，構建出更完善的花卉交易資訊整合傳送系統，讓整個花卉供應鏈資訊流能更簡單有效率，因此在花卉拍賣資訊處理系統 Flower\_TIPS(Flower\_Trading Information Processing System)上整合的功能即成為本研究的重點之一[9]，讓兩家花卉公司能在如此的系統規劃下更完善的處理該供應鏈資訊流的部份。

此外在花卉公司的資料倉儲系統中由於倉儲系統即時性的需求，必須每日的上傳當日大量的花卉交易紀錄，而此上傳交易紀錄資料的行為亦為固定重複性的工作，除了避免上傳檔案操作人員的人為疏失外(如，當日未上傳檔案)，如何以立即性的傳送最新資料以使花卉供應鏈資訊流更為流暢也為該資料傳輸系統的建構理念，以彰化花卉公司的檔案轉移服務(File Transfer Service, FTS)為例，建構以自動化及跨平台的資料傳輸系統，將彰化花卉公司當日的交易資料傳送至遠端交通大學的資料倉儲，實作花卉供應鏈前端資訊流的部分，使彰化花卉公司在轉移交易資料的方式更簡便友善，另外此系統的可重複使用性也是一大特色。

## 1.2 問題界定

本研究對於問題的界定，主要方向針對於企業間供應鏈資訊流系統的整合建置。在該花卉公司資訊流處理的過程中，早期所使用的傳遞方式除了傳統的郵寄、電話、傳真、電子郵件等，但是隨著企業的規模發展，這些方法不單耗費了大量的支出且造成使用上的不便與繁瑣，於是使用新的技術並加以整合各花卉公司資訊流，以取代傳統所使用的方法，並在檔案轉移系統上採以自動化處理，以提高供應鏈資訊流的流暢與效率[12]。

本研究主要針對兩個案例：一是彰化花卉與台北花卉公司花卉當日交易資料的整合系統設計，另一則為彰化花卉公司倉儲系統前端的自動化檔案轉移服務兩個部份。

### 一、資訊整合系統：

資訊整合系統的設計要點在於其速度性、低成本性、資訊的價值性、與整合性，透過系統架構的深入考量，使企業規模在發展過程中不受系統所侷限。網際網路的發展使資訊的傳遞更加快速，此外傳統的交易表單寄送方式在寄送的成本上更為網路傳遞所趨之若鶩，逐漸為網路電子式交易所淘汰，在資訊的整合與利用上電子化的交易表單更易於整理與分析及使用，經由網際網路的電子式表單資料，將資料儲存於各供應人與承銷人資料庫中，而資料庫內的資料可供各使用者進行分析的處理，以供使用者做交易決策之用。

### 二、自動化檔案轉移服務：

在檔案轉移服務的部分，設計目標在於其自動化的處理，科技技術的成熟促成所有人性的慾望的實現，在資訊流一貫的作業過程中減少不必要重複性的操作，將每日的花卉交易檔案即時性的傳送於遠端的資料倉儲中，以供花卉產業的線上分析處理之用，而此自動化即時上傳交易檔案的機制，決定了該系統的規劃目標。

綜合上述兩種資料傳輸系統設計，由於資料傳輸的型態與方式有所差異，本論文透過各傳輸的性質做一架構的先行考量，如在資訊整合系統中資訊的發送端為各花卉公司，而資訊的接收端則為各花卉公司所屬承銷人與供應人，其供應與承銷人數量龐大，若以一般普遍的E-mail寄送或網頁存取，不單伺服器必須處理各使用者的註冊資訊，其規模的可觀更造成伺服器負擔過重或形成日後系統維護與發展的瓶頸所在。而在自動化檔案轉移服務的部份，由於資訊的發送與接收端皆屬少數的終端使用者，資訊的發送端為花卉批發公司，而資訊的接收端則為交通大學FTP伺服器，其主要資訊傳送在每日性的寄送各交易資料檔，每日重複性的寄送檔案若因人為的疏忽而忘記傳送交易資料檔將造成花卉線上分析處理資訊流的不流暢，失去其即時性的資訊查詢結果，因而自動化的檔案上傳處理機制則該系統所重要功能，此外花卉公司的陸續成立，在系統的跨平台特性亦為系統建置前的考量要素。



### 1.3 研究目的

對於供應鏈資訊流中，當資訊流從訊息產生端傳送到訊息接收端的時候，這些資訊往往是利用紙張或是電子郵件的方式呈現出來，所以當接收端接收到這些資訊的時候，往往只能獲取這些資料表單的表面文字敘述，而在傳統的郵件寄送或語音傳真方式耗費了公司大量的成本，為提昇公司的收益與企業經營成本考量，資訊傳送的成本過高，對於企業的負擔也是相當大，所以當務之急即透過目前最盛行的網際網路技術架構新的供應鏈資訊傳送系統，以節省資訊傳送的大量成本。而在眾多的網路架構中，並非每種網路技術皆適於建置每每不同資訊傳送的屬性，因此在研究的過程中還需考量到每種資訊傳送系統的目的與性質，以供系統正確的採行建置。

在多條供應鏈的產業中，具同性質的資料表單更因交易地點的不同而有所差異。因此，蒐集的表單資訊不單無法進一步的分析與利用，在表單的獲取上更為繁雜與不便，所以在資訊價值的獲取上則無形受到了限制。有鑑於此，將接收端所接收的資訊加以整合利用即為本研究所追求的目的之一。另外在目前頗為風行的線上分析處理系統，大量使用的資料倉儲技術，其前端的資料來源與企業每日交易的資料產生與發送緊密結合著，唯有流暢的資訊流方能產生即時性的線上分析結果，以供決策者立即了解市場上快速、急遽的波動趨勢，因此為達成資訊流的順暢，在每日交易資料的產生到送達倉儲系統的過程中，以自動化即時的將交易檔案上傳至檔案伺服器為另一研究目標。

企業與企業間的多條供應鏈建置，不單著重於電子交易伺服器的穩定，其異質資料庫的資訊整合與供應鏈資訊流的流暢也都為企業增加競爭力的一項要素。在前小節問題的界定下，本研究目的即在實作花卉產業供應鏈的資訊流整合處理，透過倉儲系統建立的過程中，將其中個人的交易資訊以電子化方式傳送予各供應人與承銷人，而透過爪哇訊息服務與 XML 等的技術，將多條的花卉公司的供應鏈系統加以整合並與以傳送，使各花卉公司供應人與承銷人能以統一的訊息格式收取並儲存，以重複利用其資訊的價值性。另外在花卉公司的每日交易資料傳遞於遠端倉儲系統時，資訊傳遞的過程期望能以自動化的處理系統將遠端的交易資訊更有效率的傳遞，使其資訊流能更為順暢並且不受人為疏失的影響。在其檔案轉移系統內，由於花卉公司倉儲系統的陸續建置(如台中、台南與高雄等)，轉移系統的重複使用勢必透過模組化的設計與跨平台的特性以供多元化的使用。本論文以細部的探討各案例的資料傳送性質以實作其系統，以期在產業的多條供應鏈中實現其最佳的系統整合建置。

## 1.4 研究方法

在本論文研究的過程中，主要以下列步驟進行：

### 一、文獻回顧

在系統的建置過程中，首先將所需用到的技術領域加以整理分析，主要如 XML、爪哇技術應用與相關網路協定等，在各領域的探究除了一般課程的修習所得，系統在建置過程中尚牽涉到其它的知識領域，如資料庫系統等，所以在這些領域技術的求知則必須同時兼顧，以期能使整個系統實作與研究目標更為完善與順利。

### 二、交易表單與 XML 規格分析

在本論文的主要目標即為探討多條供應鏈系統的整合與建置，在此前提下，其供應鏈的作業流程必須透過相當詳細的廠商訪談與交易表單的分析，經由各表單的比較與整理，規劃出與該表單相對應的 XML 規格。

利用 IDEF0 及 Visual Object 等軟體，描繪出整個廠商資訊流的作業流程，藉由資訊流的作業流程藍圖分析，讓本研究在系統實作過程中，更容易規劃出系統實作步驟內所要考慮的方面及重要性程度分析；同時更明確劃分析每步驟所要完成的進度。

### 三、爪哇訊息服務實作

透過 XML 規格的制定與 IDEF0 技術呈現出花卉表單資訊流的藍圖，並加以細部規劃，依各文件屬性訂定其傳訊樣式而後進行爪哇訊息服務基本的功能撰寫，而後再強化其加值應用的功能部份。

### 四、測試與操作

透過細部規劃與實作後，系統仍會有不可預期的錯誤發生，為本研究較不好處理的部份，在數次的本地系統測試與實際的系統上線試行，重複的針對系統錯誤加以修正及效能提升。在系統完成建置後，除了錯誤的修正，同時對系統功能的強化並透過廠商的加以建議對系統更進一步的加值處理。

以下為本研究所進行的流程圖。

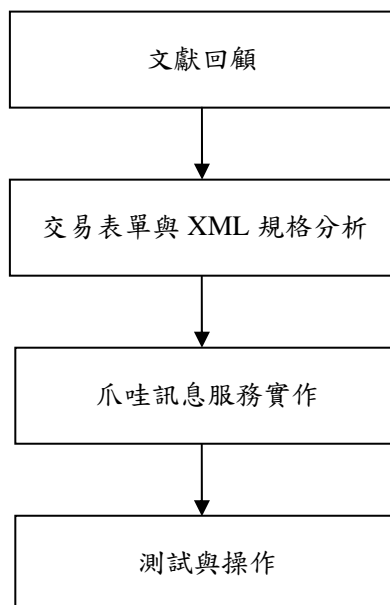


圖 1-1 研究方法流程架構



## 1.5 論文架構

本論文的研究架構如下：

### 第一章：緒論

說明本論文之研究動機與研究主題、研究方法與背景

### 第二章：文獻回顧

探討本研究案例中所需參考的傳送架構，與其供應鏈資訊流的傳遞方式做探討。

### 第三章：交易表單與其 XML 規格分析

對於彰化花卉與台北花卉公司在資訊流整合過程中做一詳細說明，並由其交易表單的分析與整理規劃出各交易表單整合後的 XML 規格，並藉由 XML 技術將文件視覺化的呈現於用戶端瀏覽器。

### 第四章：爪哇訊息服務的理念與設計

介紹爪哇訊息服務所提供的傳訊樣式，並根據各交易表單的性質規劃出該文件傳遞時所合適的樣式與細部的整合建置。

### 第五章：爪哇訊息服務應用操作

說明該系統在建制完成後的軟體操作方式，與資訊在儲存於本地端後可進行資訊價值性分析的部分。

### 第六章：檔案轉移服務系統的設計

以彰化花卉公司目前所使用的檔案轉移服務系統做一探討，說明其設計過程與自動化、跨平台與低成本等特性。

### 第七章：結論



## 第二章 文獻回顧

本章目的主要在探討花卉公司供應鏈整合作業流程、近代資訊電子化的常見架構、及了解使用訊息佇列(Message Queue)與出版者/訂閱者(Publish/subscribe)樣式的原因。第 2.1 節「資訊電子化問題探討」在說明高度網路化的現實生活中，常見電子交易問題的解決之道，第 2.2 節「常見資訊傳遞系統架構探討」比較多種不同於訊息佇列與出版者/訂閱者樣式的數種常見架構，第 2.3 節「爪哇訊息服務原理」則探討爪哇訊息服務的元件及結構組成，第 2.4 節「爪哇訊息服務程式設計」在說明其爪哇訊息服務的物件及方法使用，第 2.5 節「花卉公司供應鏈資訊流」在說明花卉公司資訊流的前後端來源輸出入，第 2.6 節「XML 技術」在說明當紅的 XML 技術與其對應於電子化表單文件的使用方式。

### 2.1 資訊電子化問題探討

在高度網路化的世界，網路的普及使得各種企業型態的資訊能透過網際網路，以各種不同的方式、轉移協定來達到資訊的交流，如常見的 Remote Method Invocation(RMI)的技術，為點對點中最常見的資訊傳輸技巧，但在傳輸的過程中，透過使用者的輸入指令後還須與伺服器建立持續性的協定，屬於緊密的偶合性(Tightly Coupled)通訊技術，但在用戶與伺服器端間分散式的系統架構上，隨著企業規模的發展與用戶端的增加，往往成為軟體在發產時的瓶頸，於是中介軟體(Middleware)的架構便隨之產生，目的就在減輕交易資訊系統在主從式(Client-Server)架構傳輸過程中伺服器的負荷。因此為使交易資訊系統更有效率、更具擴充性的運作考量下，本小節以時間性(Time)、空間性(Space)與同步化(Synchronization)三項的偶合(Coupling)觀點，來探討軟體在分散式的資訊交易模式裡常見的架構模式，並以此三項去偶合(Decoupling)的性質來強化企業軟體系統的效率[22]。以下彰化或台北花卉公司軟體負責每日交易後的資訊產出與發送稱為訊息發送端(Publisher)，而供應人或各市場承銷人為訊息接收者，則稱為訊息接收端(Subscriber)。

#### 2.1.1 時間去偶合性

在時間的觀點上，由於交易資訊不需即時的回報，發送端與接收端不需同時的運作，如圖 2-1，在發送端產生交易資訊後暫存於圖中的中介軟體(Event Service)，而接收端在該筆交易訊息存放後，可在行與訊息伺服器建立連結，如此發送端與接收端不需同時處於連線狀態，亦能獨立運作的交易模式則可稱為時間的去偶合性[22]。

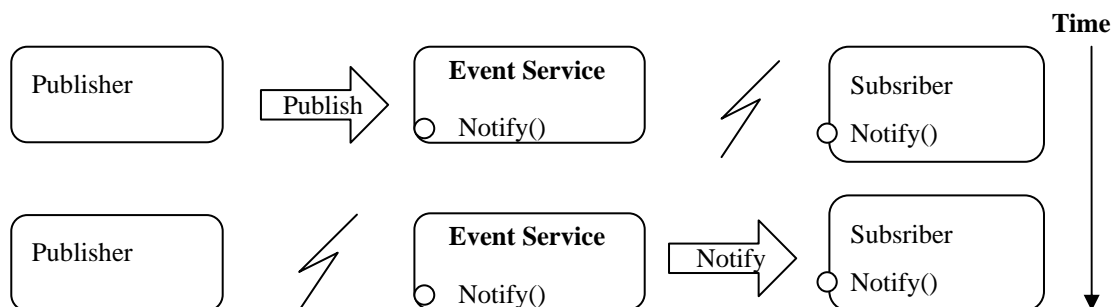


圖 2-1 時間去偶合性[22]

### 2.1.2 空間去耦合性

以空間的考量上，由於傳統的主從式(Client-Server)架構，發送端必須處理註冊在某一訊息事件上接收端的帳號資訊，例如花卉公司的大行情資料，若以主從式架構，則主端(Client)則必須詳知所有訂閱大行情的供應人與承銷人帳號資訊等，因而造成系統效率的減低。由此認知，中介軟體的架構便負責該筆交易資訊的來源與目的地的送達，發送端不需知道有多少接收端接收該筆訊息，接收端也不需與所有訊息來源地建立連線，主端與從端(Server)相互獨立運作，不但減少用戶端的程式的複雜度，也提高了可靠度與效率性。如圖 2-2，中介軟體負責訊息的所有來源，並負責將所有訊息送達各接收端[22]。

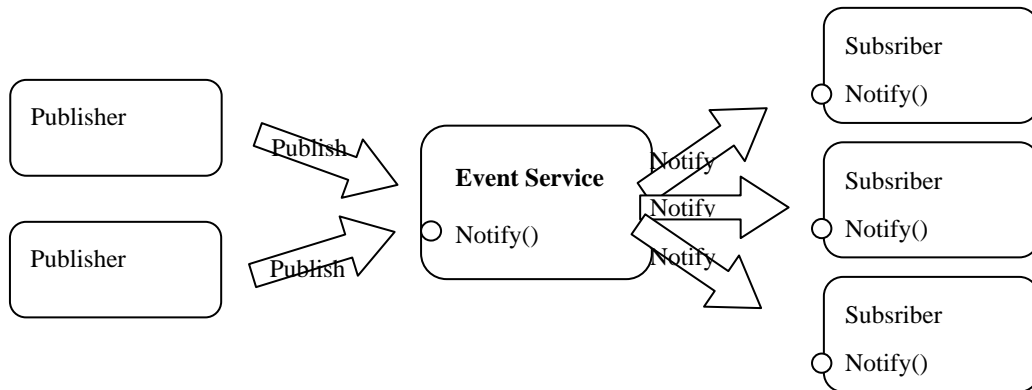


圖 2-2 空間去耦合性[22]

### 2.1.3 同步去耦合性

不同於時間的去耦合性觀點，時間的去耦合性在以概觀的方式說明交易資訊在產生時，接收端不需與傳送端同時運作；而非同步化耦合的性質在探討訊息在產生與接收時，中介器所應負責的細部工作，使得發送者與接收者依然能同時共作處理，並在共作處理中協調該訊息資料流，使收送兩端不需維持在通信的狀態，亦能保持其資料傳輸時的可靠性。以中介器的觀點而言，交易資訊在發送至中介器時，發送端若佔住中介器的資源，則接收端將無法立即得到已存放至中介器的訊息；相同地，若接收端在接收訊息時，若鎖住該訊息資料流時，則發送端將不可發送新的交易訊息。於是為避免主端與從端相互等待該資源，中介器並採以非同步通知的方式。如圖 2-3，在訊息接收端接收資訊時，若是此時發送端正產生訊息至中介器，中介器並不與接收端持續地連線，直到訊息發送端送完該筆訊息後，中介器便通知接收端接收該筆交易訊息[22]。

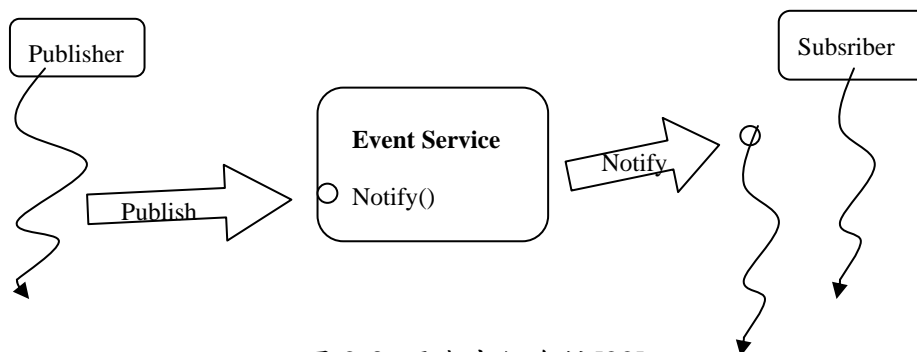


圖 2-3 同步去耦合性[22]

## 2.2 常見資訊傳遞系統架構探討

網際網路的應用，在資訊的交流中，促成了多樣的訊息交易模式與架構，如訊息傳送(Message passing)、遠端呼叫(Remote invocations)、共享記憶體空間(Shared spaces)與訊息佇列(Message Queuing)等，但在企業資訊的交流上，除了重視效率外，往往還需要軟體高度的正確、可靠性與錯誤還原機制，在移除上述三項偶合特性後，透過下列常見架構的比較，以時間(Time)、空間(Space)、與同步化(Synchronization)三項偶合特性的不同，來探討使用出版者/訂閱者(Publish/Subscribe)模式的效率與可靠度。第 2.2.1 節為訊息傳送，第 2.2.2 節為遠端呼叫，第 2.2.3 節為通知機制，第 2.2.4 節為共享記憶體空間，第 2.2.5 節為訊息佇列，第 2.2.6 節為出版者/訂閱者樣式，第 2.2.7 節為上述六小節的架構比較。其中交易資訊產生端稱為訊息的生產者(Producer)，訊息接受端稱為該訊息的消費者(Consumer)。

### 2.2.1 訊息傳遞

在分散式交易中，訊息傳送可視為最低階、最原始的交易模式，訊息的交流建立在最基礎的傳送與接收技術上。如圖 2-4，生產者(Producer)在消費者(Consumer)請求交易時，生產者以非同步的方式直接傳達訊息給消費者端，但消費者則必須持續的監聽通訊介質，以確保訊息傳送無誤，但此種方式生產者與消費者端皆須處於同個時序，且生產者與消費者皆必須知道所處來源或目的位址[22]。

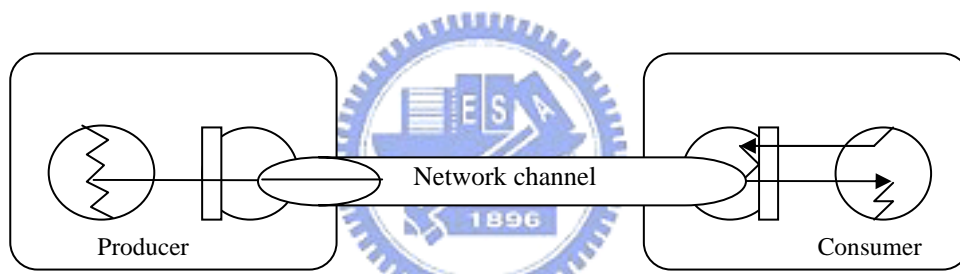


圖 2-4 訊息傳遞[22]

### 2.2.2 遠端程序呼叫

遠端程序呼叫(Remote Procedure Call)為目前最廣泛應用的一種分散式交易環境，又稱為遠端呼叫(Remote Invocation)，遠端呼叫強調在分散式系統裡，透過網際網路與緊密的參數呼叫，使得遠端的伺服器在處理完複雜的運算後，送回用戶端所請求的資訊，如同在本地端做處理一般。如圖 2-5，消費者端在提出接收請求後，經由網路介質與指令參數，要求生產者端即時傳回訊息，待生產者端經由持續性的通訊狀態處理完消費者指令要求，再行回報消費者端結果。此時生產者端處於同步化地聆聽網路介質，而消費者端則以非同步的方式接收訊息，且在時間性與空間性的偶合上，同訊息傳遞(Message Passing)，屬緊密耦合(Tightly Coupled)的性質[22]。

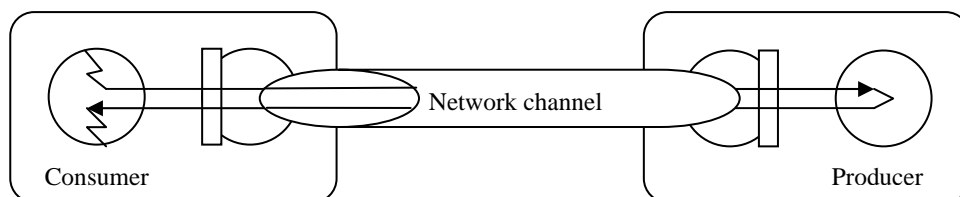


圖 2-5 遠端程序呼叫[22]



由於遠端呼叫在分散式系統內透過網路遠端的計算如在本地端處理一般，在處理龐大運算的作業相當具能力，並透過簡潔的架構使軟體的設計相當明朗化。但不同於出版/訂閱(Pub/Sub)的鬆散式(Loosely Coupled)架構，在於遠端呼叫強烈導向於時間與空間性的耦合及消費者端的同步化，如此的架構不單使一對多的伺服器負荷過重且用戶端尚須等待遠端的處理時間。在取代消費者端的同步性質後另一種的遠端程序呼叫的架構如圖 2-6 所示，消費者端不須以同步的方式接收訊息，僅需依靠相同時間性的協定，由生產者端直接發送訊息至消費者端，此種方式雖解決了生產者端同步的問題，但不可避免的生產者端在不需接收任何成功或失敗的回應下，消費者端並無錯誤回報的機制，可靠度較低，如此的交易方式又稱為 fire-and-forget[22]。

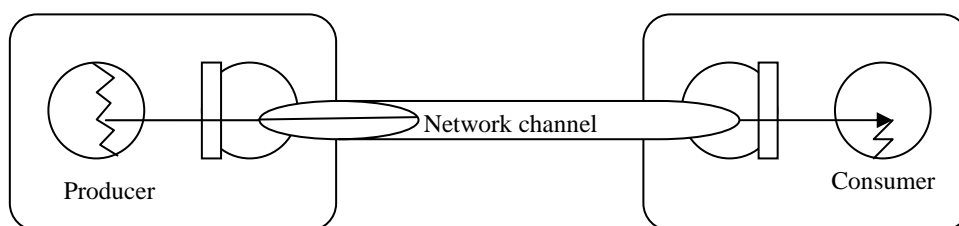


圖 2-6 同步去耦合性[22]

在生產者端單向傳送訊息予消費者端較不可靠的協定中，另一種強化遠端程序呼叫的交易機制如圖 2-7，經由消費者端非同步的提出請求後，生產者在接收到訊號後，處理完該指令計算在以非同步的方式回傳結果，消費者在間隔一段時間後可由其交易存放的位置得到該筆交易資訊[22]。



圖 2-7 同步去耦合性質進階遠端呼叫[22]

在遠端呼叫的方式中，非同步化的耦合性質與維持交易的可靠度及回報機制著實有相當完善的改進，但消費者與生產者端仍侷限於時間與空間性的耦合。

### 2.2.3 通知機制

遠端呼叫的模式裡，為達成生產與消費者端非同步化耦合的特性，實作了兩個非同步通知的方式：消費者端併隨著參數發出請求的訊號後，並要求生產者端給予回應，生產者端在收到訊號後，以數個非同步訊號給予回應，以驗證傳輸過程中資料無誤，但除了在時間與空間的耦合性外，在生產者端數個持續性的回應機制仍為系統在發展規模時通訊環境的累贅[22]。

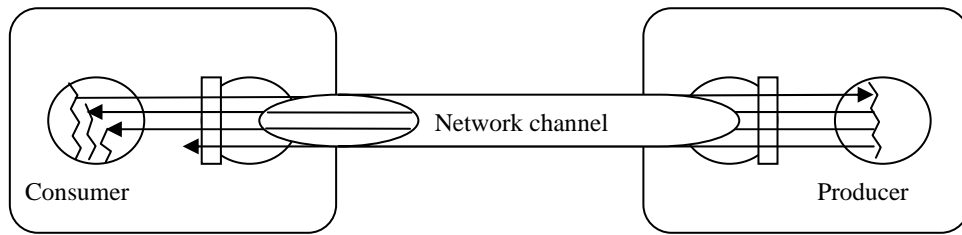


圖 2-8 通知機制[22]

### 2.2.4 共享記憶體空間

分散式系統經由共享記憶體空間提供了時間與空間性的去耦合性質。共享記憶體空間採分配位址空間存放資料的方式，達到共享一份相同的資料且不需佔有重複的記憶體空間，透過欄空間(Tuple Space)的觀念提供強大而簡易的方法存取目標位址空間。欄空間的觀念以插入/移除數個欄的方式蒐集所有具相同關係的集合，其主要操作有 out()、in()，與 read()等三種方法，out()將相關集合的欄輸出至欄空間存放，in()與 read()則為用來取出目標欄的方法，其中 in()方法並將該欄從欄空間中移除，read()則只進行取出的動作。在此種交易方式中生產者與消費者端軟體保留匿名的方式存取該欄空間資訊，不需額外耗費資源去處理帳號等相關的資訊；例如以 in()的方式可進行多中取其一(one-of-n)的匿名交易方式，以 read()的方式可進行一對多(one-to-n)的交易方式；由於 in()方法中，消費者在接收交易資訊後將該筆訊息移除，則接收到該筆訊息的只會有單一消費者，而 read()則以不移除欄的方式，可供數個消費者存取同一份匿名性的資料。

在常見的共享記憶體空間架構上，如圖 2-9，時間與空間型實現了去耦合的性質，但在消費者端，大多採以同步的方式以確保資料的可靠性，除了少數種產品在受限於規模發展的條件上，採以非同步呼叫消費者端的方式(如：JavaSpaces)[22]。

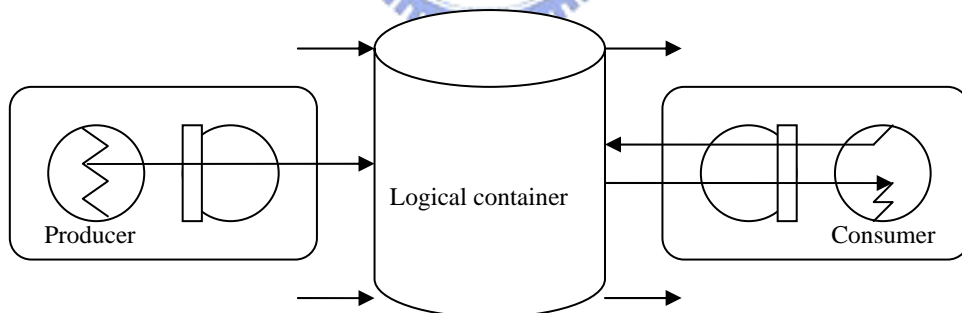


圖 2-9 共享記憶體空間[22]

### 2.2.5 訊息佇列

訊息佇列常與出版/訂閱(Publish/Subscribe)形式的交易整合在一起。訊息佇列可視為生產者將所有的資訊存放於訊息導向(Message-Oriented)的中介軟體，以功能上而言，訊息佇列有別於共享記憶體空間，主要在提供時序上的先後處理，但在欄空間上的觀念則非為必須。透過訊息佇列系統，可實作如同欄空間中 in()的方式多中取一(one-of-n)的交易處理，如此的方式一般稱為點對點(Point-To-Point, PTP)式的交易。

在耦合性的觀點而言，如圖 2-10，訊息佇列提供了與欄空間相同的時間與空間上去耦合的特性，但仍有少數產品亦支援消費者端的非同步處理機制[22]。

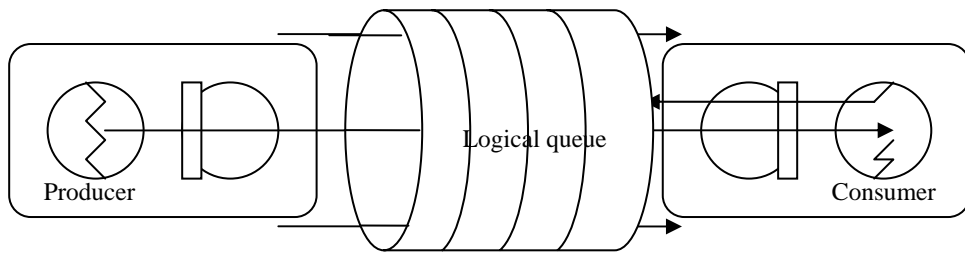


圖 2-10 訊息佇列[22]

### 2.2.6 出版者/訂閱者

出版者/訂閱者模式有別於傳統的交易模式達成了時間、空間與同步化反耦合的特性，如圖 2-11，生產者與消費者以非同步的方式經由網路存取該中介軟體，時間性上耦合的移除使得生產者與消費者不需侷限於同時間運作。空間耦合的移除更可減少生產者與消費者端在處理帳號資訊上所耗費的資源，在多對多(n-to-n)的交易模式中，消費者群僅需知道該中介軟體的所在，即可存取來自各方生產者所存入的訊息，而生產者群則在匿名性的資料上開放該訊息予消費者群，大幅提升系統的效率[22]。

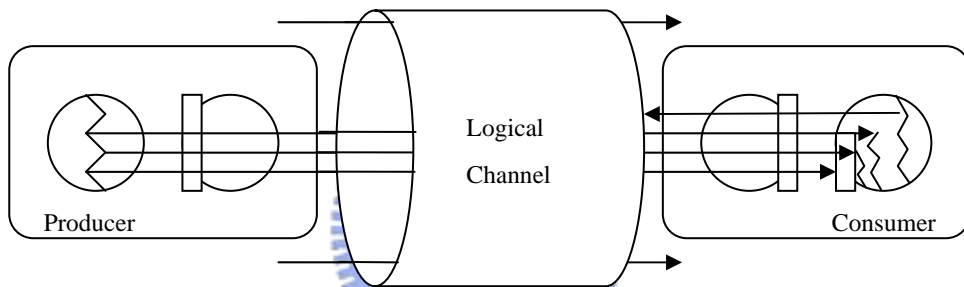


圖 2-11 出版/訂閱樣式[22]

### 2.2.7 架構比較

表 2-1 為上述八種常見訊息傳送架構以時間、空間與同步去耦合性的比較，其中在欄空間與訊息佇列中少數產品中亦支援消費者端同步的運作。

表 2-1 架構與去耦合性比較表[22]

Abstraction	Space Decoupling	Time Decoupling	Synchronization Decoupling
Message passing	No	No	Producer-side
RPC/RMI	No	No	Producer-side
Asynchronous RPC/RMI	No	No	Yes
Future RPC/RMI	No	No	Yes
Notifications	No	No	Yes
Tuple spaces	Yes	Yes	Producer-side
Message queuing(Pull)	Yes	Yes	Producer-side
Publish/subscribe	Yes	Yes	Yes



## 2.3 爪哇訊息服務原理

爪哇訊息服務(Java Message Service, JMS)為消費者端(Subscriber/Receiver)與生產者端(Publisher/Sender)非同步執行的訊息交易模式，消費者端接收來自於生產者端存放於訊息導向中介軟體(Message-Oriented Middleware, MOM)的交易訊息，透過中介軟體的支援與時間、同步、空間等去耦合性的機制，使得交易資訊的傳遞更有效率(見圖 2-12)。

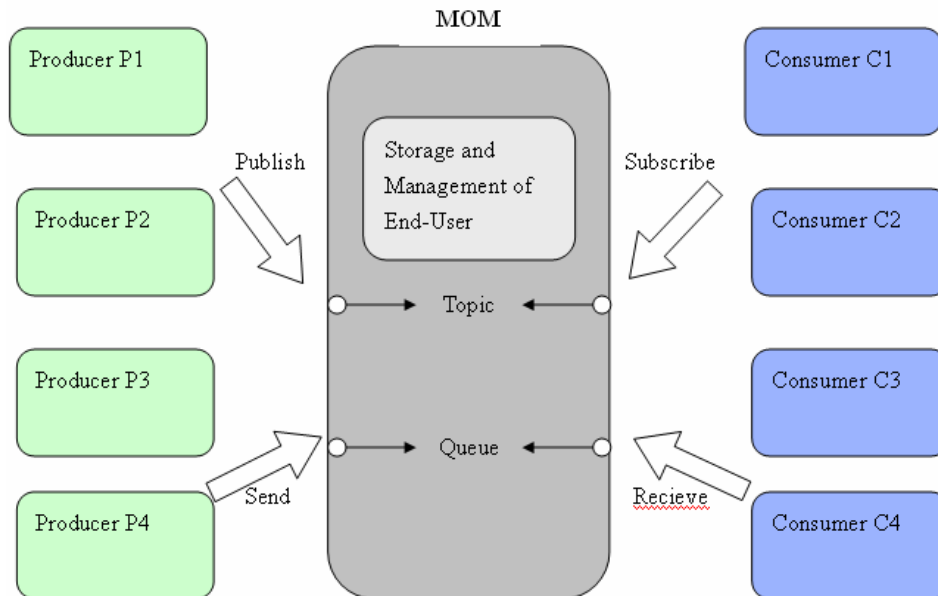


圖 2-12 訊息服務示意圖

然而訊息系統間並無其通用的程式介面，各不同產品的 MOM 會有不同的應用程式介面 (Application Interface, API)，在訊息系統間的交易便形成了訊息轉移的問題，於是，爪哇訊息服務在美國的昇陽公司(Sun Microsystems, Inc)於1998年與其主要訊息系統廠商提出爪哇訊息服務的標準，而目前發展的版本為 1.1 版，爪哇訊息服務是 Java 在支援 MOM 產品的一套爪哇 API，是一套提供程式開發的介面，各訊息產品廠商透過提供實作的方法，使 Java 程式可以一套標準的方式存取不同的訊息中介軟體。以下第 2.3.1 節「爪哇訊息服務與中介軟體」為說明爪哇訊息服務與中介軟體的溝通方式，第 2.3.2 節則介紹爪哇訊息服務的組成元件。

### 2.3.1 爪哇訊息服務與中介軟體

在支援訊息服務的 MOM 軟體與 Java 程式間的溝通方式如下圖所示：

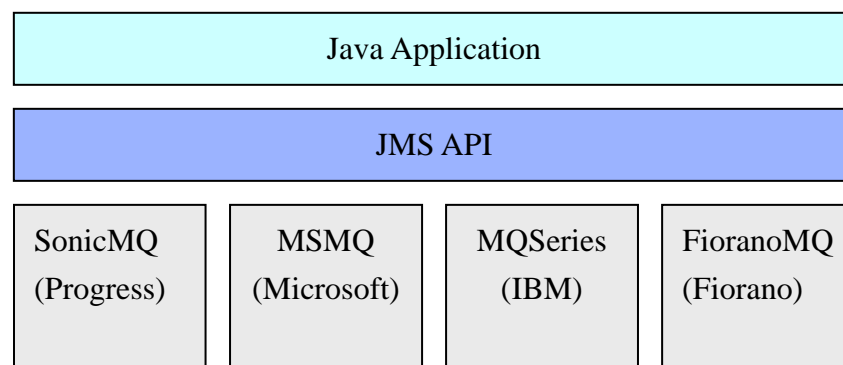


圖 2-13 Java 應用程式與 MOM 產品關係圖[2]

其中 Java 應用程式包括交易訊息的傳送端與接收端，在透過標準化的 JMS API，使用者可存取不同 JMS 提供者(Provider)的交易訊息。另外，此 JMS 提供者亦提供了負載平衡、安全性、錯誤回報、訊息導向等功能。

### 2.3.2 爪哇訊息服務構成元件

爪哇訊息服務的構成要件主要有五個：訊息、JMS 提供者、JMS 傳送端與接收端、JMS 管理物件、與非 Java 撰寫軟體(Native Client)等，分述如下：

- 一、訊息：訊息為企業在供應鏈上所進行的交易資訊，電子化的交易方式必須透過訊息的方式來達到資訊目地的交換。
- 二、JMS 提供者：在企業資訊的交換媒介中，JMS 技術在以透過某特殊的中介軟體儲存裝置，以存放企業成員所交流的訊息，可說是整個供應鏈資訊流中樞，此中介軟體則稱之。
- 三、JMS 傳送端與接收端：訊息儲存於中介軟體上，而訊息的來源產生與目的去處則分別稱之傳送端與接收端。
- 四、JMS 管理物件：JMS 技術的使用必須透過許多標準化物件的產生以協調各產品的規格，而並非所有物件皆由程式所產生，一般而言，除了各 JMS 提供者所制定的規格物件，最常見的方式則可經由 Java Naming and Directory Interface (JNDI)所提供的 API 以取得所需物件，此外 JMS 並提供標準的兩種管理物件，如 ConnectionFactory 與 Destination；ConnectionFactory 物件在協調其 JMS 傳送樣式的連線方法，而 Destination 物件則在使用戶端區別其訊息目的地所在。
- 五、非 Java 撰寫軟體(Native Client)：不同於 JMS 傳送與接收端，在於 JMS 傳送與接收端主要以 Java 語言規範所實作的傳送與接收方法，而 Native Client 在於各使用者應用軟體上不為其 Java 語言設計之外的應用程式皆稱之。

綜合以上 JMS 五項構成要素，其關係如圖 2-14 所示：

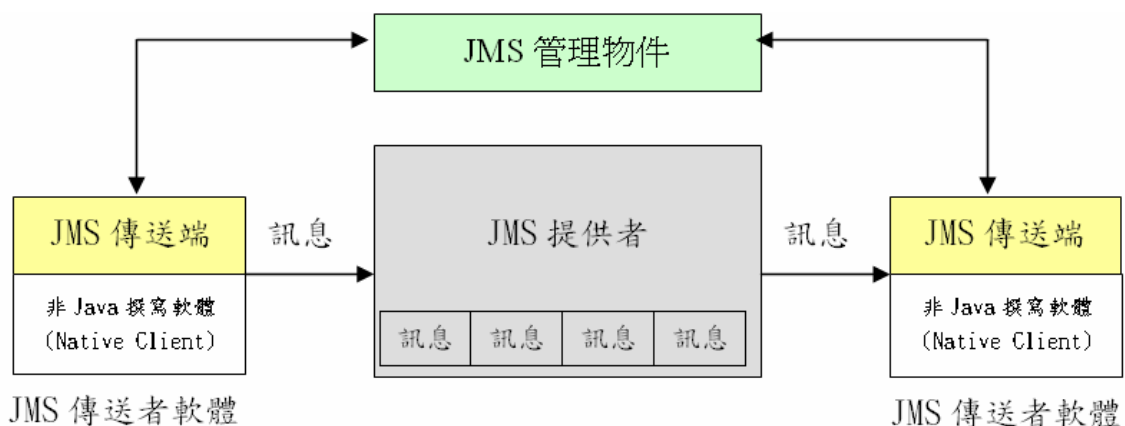


圖 2-14 JMS 管理物件[2]

### 2.4 爪哇訊息服務程式設計

爪哇訊息服務的傳訊樣式主要分為傳送/接收傳訊樣式及出版/訂閱傳訊樣式，對於使用的傳訊樣式不同，建爪哇訊息服務者建構與連線的方式也不相同[2]，但是兩種傳訊樣式間的差異性並不大，以下並同時說明兩種傳訊樣式的建構步驟。

在與爪哇訊息服務伺服器的連線過程中，依序以圖 2-15 所示建構出必須的物件，而圖左邊的

方框代表訊息生產者的建構步驟，右邊方框則為訊息消費者端的建構步驟[8]，本章節以 Progress 公司提供的 JMS 提供者(Sonic MQ)為例，將其物件方法建立過程分述如下，第 2.4.1 節為「連線工廠」，第 2.4.2 節為「連線」，第 2.4.3 節為「目的地」，第 2.4.4 節為「訊息生產者與消費者」，第 2.4.5 節為「訊息與訊息選擇器」，第 2.4.6 節為「訊息出版與訊息傾聽者」。

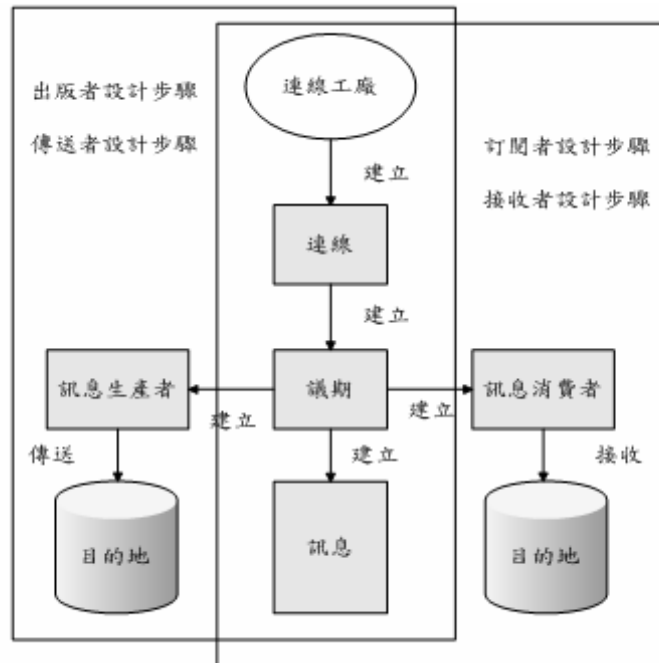


圖 2-15 JMS 程式架構[8]

## 2.4.1 連線工廠

連線工廠的產生即為用戶端以該 JMS 提供者所定義的參數與之建立連線，參數中包含了 JMS 提供者的 IP 位址與埠號，連線工廠的設定與訊息的屬性或行為無關，皆交由爪哇訊息服務者處理，對於出版/訂閱與點對點樣式則分別包含有兩種物件類別 TopicConnectionFactory 與 QueueConnectionFactory，其設定如下：

表 2-2 連線工廠設定

出版/訂閱連線工廠設定	<pre>javax.jms.TopicConnectionFactory factory = (new progress.message.jclient.TopicConnectionFactory(address));</pre>
點對點連線工廠設定	<pre>javax.jms.QueueConnectionFactory factory = (new progress.message.jclient.QueueConnectionFactory(address));</pre>

當連線工廠物件實體的建立時，即提供了傳訊樣式與爪哇訊息服務者間的 TCP/IP Socket 通訊協定，並可同時建立出多個連線，但為避免浪費許多系統資源，往往採以單一連線連建立多個議期，其單一連線與多重連線的架構如圖 2-16 所示。在出版/訂閱與點對點樣式的連線中為避免在建立連線時產生衝突，在與 JMS 提供者建立連線實體時，傳送了兩個字串參數 username 與 password，以做為連線識別，其建構方式如表 2-3 所示：

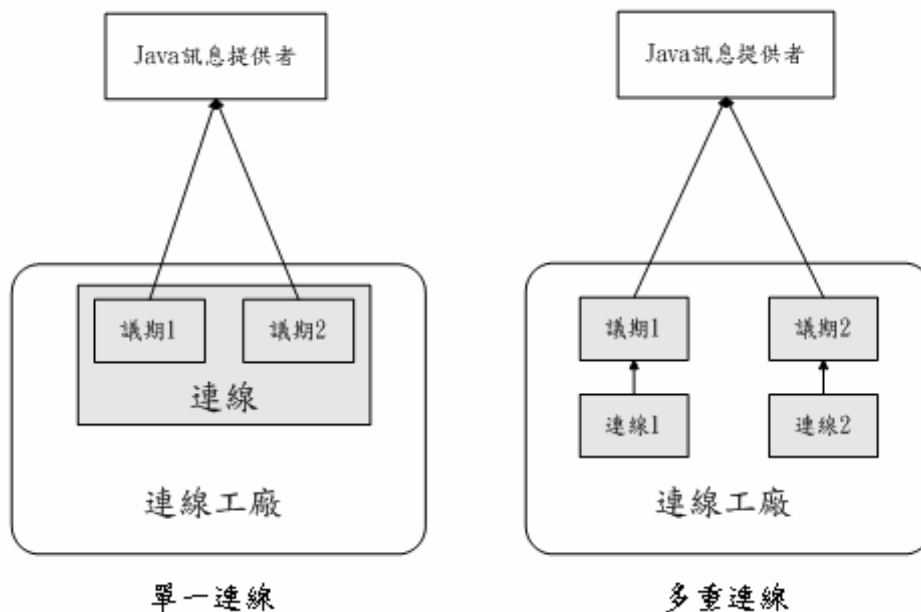


圖 2-16 單一連線與多重連線架構[8]

表 2-3 連線設定

出版/訂閱連線設定	<code>javax.jms.TopicConnection connect = factory.createTopicConnection(username,password);</code>
點對點連線設定	<code>javax.jms.QueueConnection connect = factory.createQueueConnection(username,password);</code>

在完成與 JMS 提供者連線的設定後，此時的 Connect 實體為停止的狀態，必須以 start() 指令啟動連線，而在連線中若以 close() 指令可將連線所產生的資源予以關閉並釋放(如：Session)，如需暫停資源的傳送則可下達 stop() 指令。

## 2.4.2 議期

議期的產生可建立起訊息的傳送與接收端，JMS 的標準制定下允許單一 Session 同時擁有多個傳送與接收端，但不可同時進行，只可以利用一個連線建立多個議期方能達成多執行緒目的地。此外 JMS 的議期規範可確保所接收到的訊息順序是相同的。在出版/訂閱與點對點樣式中 JMS 分別提供了 TopicSession 與 QueueSession 類別物件以建立議期實體，並須設定兩個參數，一為布林參數，另一則為整數參數。

- 一、第一個參數代表此議期的交易控管方式，參數設定為 true 或 false 分別指定訊息在接收過程中採以批次或單筆的方式給予接收回應。
- 二、第二個參數則用來決定訊息接收的回應確認方式，分別為自動致謝 (Session.AUTO\_ACKNOWLEDGE)、顧客致謝 (Session.CLIENT\_ACKNOWLEDGE) 及多重致謝 (Session.DUPS\_OK\_ACKNOWLEDGE)，底下則分別介紹這三種回應確認方式。

### 1. 自動致謝：

自動致謝，為預設值，當用戶端發送或收到訊息後會自動回報 JMS 提供者，以表示此訊息傳送或接收成功。

## 2. 顧客致謝：

訊息回應通知是由訊息接收者程式所控制，在 `onMessage()` 方法內正確執行完後，系統即自動呼叫 `acknowledge()` 方法，以確認訊息接收無誤，若在批次的處理方式中，可確保此次的成批訊息為完全的接收，只要有一個訊息傳送錯誤，則視為傳送失敗。

## 3. 多重致謝：

多重致謝的作用在允許重複收到相同的訊息，若當爪哇訊息服務者失效時，有可能會收到重複的訊息。

議期由連線所構建，其建立的過程分別如表 2-4 所示，對於回應確認的模式本文採以單筆的傳送回應與自動致謝模式，以保障每筆訊息能成功被傳送與接收。

表 2-4 議期設定

出版/訂閱議期設定	<code>javax.jms.TopicSession session = connect.createTopicSession(false, javax.jms.Session.AUTO_ACKNOWLEDGE);</code>
點對點議期設定	<code>javax.jms.QueueSession session = connect.createQueueSession(false, javax.jms.Session.AUTO_ACKNOWLEDGE);</code>

### 2.4.3 目的地

在爪哇訊息服務的兩種主要樣式而言，出版/訂閱與點對點訊息傳送的目的地分別為 Topic 與 Queue，且利用議期所提供的方法 `createTopic()` 及 `createQueue()` 來個別建立此兩種傳訊樣式的目的地實體，並傳入一個目的地名稱的字串參數，以使訂閱者或接收者識別其訊息接收的目的地，建構如表 2-5 所示：

表 2-5 目的地設定

出版/訂閱目的地設定	<code>javax.jms.Topic topic = session.createTopic("SupplyChain");</code>
點對點目的地設定	<code>javax.jms.Queue queue = session.createQueue("SampleQ2");</code>

### 2.4.4 訊息生產者,消費者

對訊息生產者而言，在確認訊息目的地後，才決定訊息發送者。出版/訂閱與點對點分別透過議期所提供的方法 `createPublisher()` 與 `createSender()`，並傳入目的地參數，以建立其個別的訊息生產者 `TopicPublish` 與 `QueueSender` 類別物件，如表 2-6 所示：

表 2-6 訊息生產者設定

出版/訂閱訊息生產者設定	<code>javax.jms.TopicPublisher publisher = session.createPublisher(topic);</code>
點對點訊息生產者設定	<code>javax.jms.QueueSender sender = session.createSender(queue);</code>

對訊息消費者的設定而言，可透過議期與兩個參數來建立，如表 2-7 所示：

- 一、第一個參數變數"topic"是用來設定其訊息來源為主題或佇列。
- 二、第二個字串參數"nctu"則是用來判斷主題或佇列使用者之用。



- 三、第三個參數代表此訊息所屬的類別與代號，用來判定是否為該位使用者所擁有。
- 四、第四個參數則說明此訊息是否能再被傳送一次。

表 2-7 訊息消費者建立方式

出版/訂閱訊息消費者設定	<pre> javax.jms.TopicSubscriber subscriber = session.createDurableSubscriber(topic,"nctu" PROPERTY_NAME+"="+"\'+selection+"\",false); </pre>
點對點訊息消費者設定	<pre> javax.jms.QueueReceiver qReceiver = session.createReceiver(queue"nctu", PROPERTY_NAME + " =" + selection + "\",false); </pre>

### 2.4.5 訊息、訊息選擇器

對訊息而言，這是指傳遞的資料內容，爪哇訊息服務提供了五種資料型態：文字訊息 (TextMessage)、物件訊息 (ObjectMessage)、位元訊息 (BytesMessage)、串流訊息 (StreamMessage)、與對應訊息 (MapMessage) 等 [19]，在本論文中的供應鏈資訊流格式則主要採以 XML 格式的文字訊息格式。在兩種傳訊樣式中，XML 訊息皆透過議期所提供 createXMLMessage() 方法以產生訊息實體，如表 2-8 所示。在產生訊息實體後便將所要傳送的訊息內容填入訊息物件內。其中所要傳送的物件為表 2-8 中的 msg，所傳送的訊息內容則為 passmessage 字串變數。

表 2-8 XML 文件訊息設定

<pre> progress.message.jclient.XMLMessage msg = ((progress.message.jclient.Session) sendSession).createXMLMessage(); msg.setText(passmessage); </pre>
---

在訊息的觀點上，在出版者或是傳送者設計的過程中，可指定發送的訊息為那一位使用者所擁有，而如此的設定方式則必須透過訊息選擇器，並傳入兩個字串參數。其設定方式可參考表 2-9 所示，其第一個字串參數 PROPERTY\_NAME 代表此份訊息在訊息選擇器內所指定的類別名稱，而第二個字串參數 selection 則代表在該類別名稱內的指定代號，依此這兩個字串參數則可指定此訊息為那一位使用者所有。

表 2-9 訊息選擇器設定

<pre> msg.setStringProperty(PROPERTY_NAME,selection) </pre>
---

### 2.4.6 訊息出版、訊息傾聽者

JMS 的一向特點在於其提供可靠的訊息傳送機制，除了傳送錯誤時的訊息回報外，在傳送失敗時更可採以訊息重送的方式將失敗的訊息資料重新送出。在完成前面許多設定後，最後將所需傳送的交易訊息包裹於訊息生產者內，共加入四個參數送出，即完成訊息的傳送過程，其中並可決定訊息重送及優先等級設定，如表 2-10 所示，其涵義如下。

- 一、第一個參數為訊息生產者(Publisher/Sender)所需傳送至 JMS 提供者的訊息變數，其內容文件本文以 XML 格式的文字訊息作為資訊流的傳遞。
- 二、第二個參數則可設定訊息傳遞的模式，爪哇訊息服務並提供永續傳遞(PERSISTENT)與非永續傳遞(NON\_PERSISTENT)兩種訊息傳遞模式。永續傳遞模式即在訊息傳遞前先行儲存於持續的儲存體上，若爪哇訊息提供者發生錯誤情況，則可以避免訊息的遺失，為預設的交易模式。非永續傳遞模式在傳送訊息前則不需先將訊息儲存持續性的設備上，可避免訊息在儲存時所耗用的資源而降低效能，但在爪哇訊息服務發生錯誤時，則會造成交易訊息的遺失。
- 三、第三個參數則代表訊息傳送時該筆訊息的優先權，爪哇訊息服務將訊息的重要性分成 0 至 9 十個優先權等級，值愈大優先順序愈高，而預設的優先權等級為 4。
- 四、第四個參數則代表訊息的有效時間，訊息的存活時間以毫秒(1/1000sec)為計算單位。假若訊息以 30 分鐘存活，其變數值為 1800000，其預設值為 0，表示訊息為永久存活，無過期的情況出現。

表 2-10 訊息出版設定

出版/訂閱訊息出版設定	<code>publisher.publish(msg,javax.jms.DeliveryMode.PERSISTENT, javax.jms.Message.DEFAULT_PRIORITY,Message_lifespan);</code>
點對點訊息出版設定	<code>sender.send(msg,javax.jms.DeliveryMode.PERSISTENT, javax.jms.Message.DEFAULT_PRIORITY,Message_lifespan);</code>

在出版/訂閱與點對點傳訊樣式中，爪哇訊息服務皆支援非同步的訊息傳送與接收端，訊息傾聽者主要是以監聽方式聆聽所訂閱的主題或佇列內是否有新的訊息到達，以採取接收的動作。訊息傾聽者的行為設計在以實作 `MessageListener` 介面，並覆寫 `MessageListener` 介面內的 `onMessage(javax.jms.Message amessage)` 方法，並將訂閱者程式或是接收者程式類別物件註冊於此 `MessageListener` 事件上，而註冊此訊息傾聽者介面的方式如表 2-11：

表 2-11 訊息傾聽者設定

覆寫 <code>MessageListener</code> 介面	<code>void onMessage(Message msg)</code>
註冊 <code>MessageListener</code> 事件	<code>subscriber.setMessageListener(this)</code>

在完成訊息傾聽者設定之後，可將相對應的訊息接收動作鑲於 `onMessage()` 方法內，在接收端所訂閱的主題或佇列內有新的訊息到達時，則自動呼叫 `onMessage()` 方法，並對此新接收的訊息採以相對應的行為[19]。



## 2.5 花卉公司供應鏈資訊流

科技技術的發展，使得人們在資訊的取得非常容易，而產業規模的龐大，隱含在每日交易資訊裡的意義與趨勢更為企業決策者所關注的細節，此外 Internet 技術的普遍使用，在客戶端於資訊的取得與利用如能透過網路的方式，不單可節省可觀的成本資源，客戶端對其交易資訊的儲存能更為完善以供日後的在利用。當下在資料庫系統與資料倉儲之應用已非常普遍，除了針對不同的產業建置外，穩定的倉儲技術與倉儲資訊的重要性，在相同產業的規模發展上使得企業對倉儲系統的需求更為迫切。本節即以說明目前花卉產業供應鏈的主要業務情報來源---農產運銷資訊網[16]，其下的花卉批發資訊分享熱線在建置過程中，其資訊流前端的資訊來源取得，並透過其倉儲系統的在利用性，將每日傳送給客戶端的交易單據以電子化的方式加以整合。如圖 2-17 所示，其中在供應鏈資訊流的前端可透過點對點檔案轉移服務系統的每日傳送交易資料，而在交易的單據電子化中，可利用爪哇訊息服務的 Pub/Sub 與 Message Queue 樣式作為交易文件的資訊傳送，以下並個別探討其檔案轉移服務系統(左虛線方框)與爪哇訊息服務(上虛線方框)的利用與其架構探討。第 2.5.1 與 2.5.2 節在說明「檔案轉移服務系統」與「爪哇訊息服務」的建置位置與其功能所在，第 2.5.3 節「交易資訊處理系統」在回顧目前台北花卉公司的爪哇訊息服務系統。

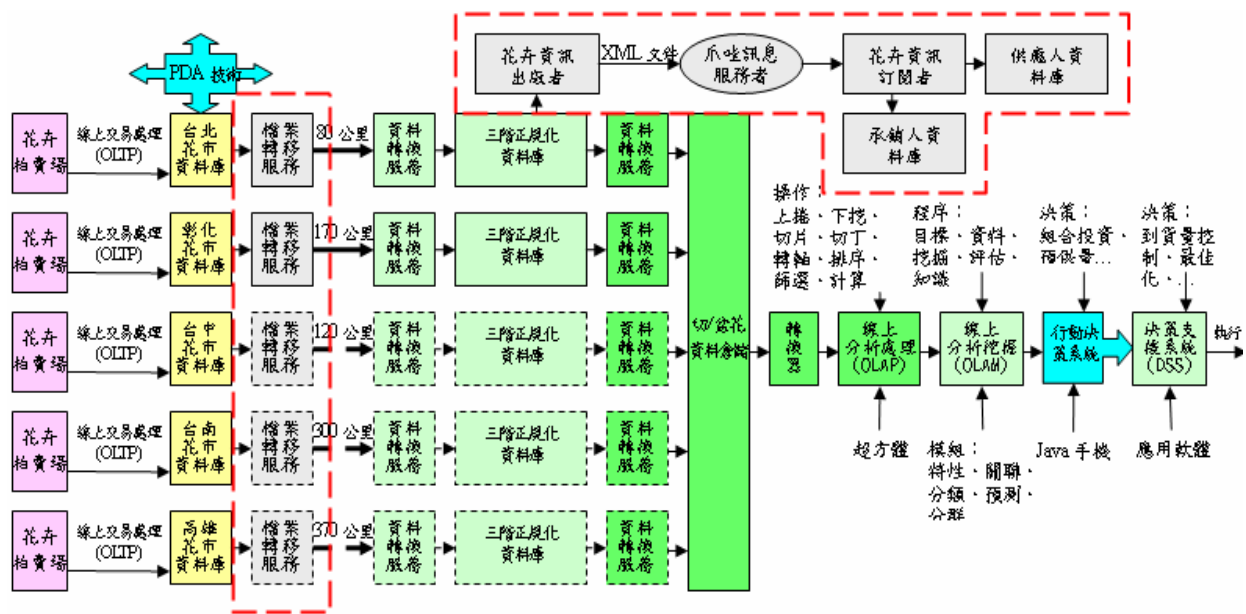


圖 2-17 花卉批發資訊分享熱線[16]

### 2.5.1 檔案轉移服務系統

檔案轉移服務系統目的在透過花卉公司的檔案系統轉檔後，將每日的檔案傳送至遠端的交通大學伺服器，以提供花卉公司切/盆花資料倉儲的資料來源。而檔案的轉移為各花卉公司一對一傳送架構，由於檔案的傳送為每日的重複性操作且對資訊的取得在講求其即時性，在本論文中即以自動化的處理機制使其資訊流更為流暢也避免人為的疏忽(如操作人員當日未傳檔)。

### 2.5.2 爪哇訊息服務

如圖 2-18 所示，目前花卉公司的交易文件主要以小行情、匯款清單、大行情、與結帳清

單等四份交易文件，本論文在利用爪哇訊息服務，以電子化的方式實作其花卉供應鏈在客戶端的資訊流部份，並將兩花卉公司文件予以整合。在此四份文件中，兩花卉公司目前皆開放小行情、大行情、與結帳清單等三份電子交易文件，在此三份交易文件中，包含私人性的資料與統一公開性的資料單據，此外在提供電子式的文件交易伺服器屬於一對多的網路拓樸。

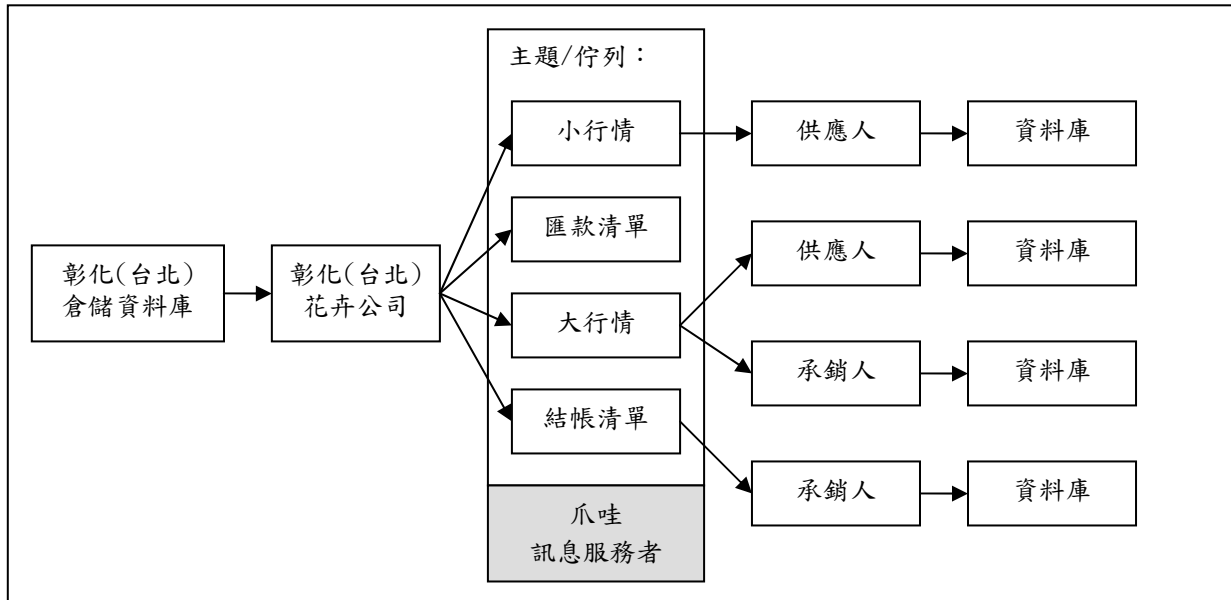


圖 2-18 電子式文件交易模式

### 2.5.3 交易資訊處理系統

交易資訊處理系統(Trading Information Processing System, TIPS)為果菜供應鏈裡透過爪哇訊息服務與 XML 文件將各農產交易資訊穿梭於公司資料庫間，目前透過此交易資訊處理系統已成功實作漁業交易資訊處理系統(Fish\_Trading Information Processing System, Fish\_TIPS)與台北花卉公司的花卉交易資訊處理系統(Flower\_TIPS)。在目前台北花卉公司的交易資訊處理系統內，已透過花卉公司各供應人與承銷人交易表單的欄位名稱，制定出合法的 XML 規格，並以爪哇訊息服務公開的兩種樣式：訊息佇列(Message Queing)與出版/訂閱(Publish/Subscribe)，建構出該公司的交易資訊處理系統[8]。

在爪哇訊息服務的規格裡，透過中介軟體的方式將台北花卉公司的當日交易資訊儲存其內並管理各交易訊息所屬的接收者帳號，而此中介軟體目前以 Progress 公司所發展的 Sonic MQ 為其訊息存放中心，並提供時間、空間與同步去耦合性質的訊息佇列與出版/訂閱樣式，提高使用者在交易資訊的傳送與接收過程中電子化機能的高品質性。

在台北花卉公司的交易資訊處理系統中，開放了電子交易表單個人性的有供應人小行情、承銷人結帳清單，而公開性的文件則有大行情的統計資料[9]；在個人性的交易資料由於該份文件專屬於單獨個人所有，以訊息佇列的方式存放，共使用了兩個訊息佇列分別存放供應人小行情與承銷人結帳清單資料；而公開性的大行情統計資料由於該份文件屬公共的使用者所有，採以出版/訂閱樣式存放，接收者須先註冊於該訊息主題上方能接收該份文件訊息[8]。

花卉公司的陸續建置，透過台北花卉公司 Flower\_TIPS 的成功案例，本論文在實作彰化花卉公司的交易資訊處理系統過程中，一併將台北花卉公司的交易資訊整合於該處理系統中，目的在實現一套處理系統可接收同企業性質的交易資訊。

## 2.6 XML 技術

企業間的文件格式多半不一，而在企業間的資料庫內容的交流則必須透過相同規格的制定，以使彼此熟悉相互的文件內容，而在電腦系統的辨識中，除了文件的數據傳輸外，電腦系統尚需了解其數據所代表的涵義與內容，而當紅的 eXtensible Markup Language(XML)則為一個相當具資料描述能力的語言，透過該 XML 技術使企業間的文件能更具交流性的使用於各電子商務交易行為上。以下並說明論文在使用 XML 技術上所秉持的理念與 XML 技術的規範及應用。第 2.6.1 節「可延伸性加註語言」在說明 XML 與 HTML 的同異處，第 2.6.2 節「可延伸性加註語言基本語法」在說明 XML 的標準語法與格式，第 2.6.3 節「XML 文件排版樣本」則說明排版樣本的常見種類與其功能。

### 2.6.1 可延伸性加註語言

可延伸性加註語言(eXtensible Markup Language, XML)[6]為 1996 年由 W3C(World Wide Web Consortium)所制定出來的標準[5]，如同 HTML 一般，透過標籤與屬性的方式使得電腦系統得以辨識其意義，但不同於 HTML 的地方在於 XML 標籤可透過使用者自訂的樣式作為資訊在交流時的定義標籤，並依照其標籤屬性與標籤內容值來發展合適的使用者應用程式。

在 XML 自訂的標籤樣式上，除了適合作為描述文件的內容意義外，藉由驗證規則(如 DTD 或 XML Schema)來做為 XML 文件標籤屬性與結構的規範，以提高 XML 技術的穩定與可靠性。此外，在 XML 技術於 Web 的使用上並可透過一份排版樣本(Style Sheet)，使得該 XML 結構化的文件得以呈現於網頁瀏覽器上，而不僅限於標籤格式的內容呈現。圖 2-19 為 XML 與 HTML 的關係圖：



圖 2-19 XML 與 HTML 關係圖

### 2.6.2 可延伸性加註語言基本語法

在 XML 的基本語法中，依據 XML 1.0 規格所訂定的規範下，必須遵循下列規則[6]：

- 一、XML 宣告需在文件第一列設定 version 屬性並以小寫「xml」宣告。
- 二、須有一個根標籤節點(Root)。
- 三、就邏輯結構而言，XML 標籤須以樹狀結構排列，其樹狀各節點皆可稱為元素，而元素間仍具階層性的關係。
- 四、除非標籤內容值為空值，標籤須以開始及結束成對的標籤呈現。
- 五、空值標籤結尾需加以「/」作為結束識別符號。
- 六、標籤名稱大小寫視為不同標籤。
- 七、屬性值前後需以「”」或「'」符號作為識別
- 八、特殊字元需依規定撰寫。

在透過上述八個條件的 XML 文件規範，即可稱此文件為 Well-Formed XML 文件。



在花卉產業的交易文件中，如下面供應人小行情表單為例，其相對應的 Well-Formed XML 格式如圖 2-2 所示：

供應號	序號	品名	等	單價	把數	箱數	供應號	序號	品名	等	單價	把數	箱數
KH001	26400-01	FR108	A+	79	9	1	KH001	26437-01	FR412	A1	30	5	1
26401-01	A1	82	A1	10	1	26438-01	A2	26	7	1			
26402-01	A3	29	9	1	26439-01	A3	18	8	1				
26403-01	FR601	A1	64	8	1	26439-02	A3	18	8	1			
26404-01	FR108	A2	62	10	1	26440-00	A4	殘貨	8	1			
26405-01	A2	59	8	1	26441-01	FR108	A1	102	10	1			
26406-01	A3	28	10	1	26442-01	A2	72	11	1				
26407-01	FR408	A1	53	7	1	26443-01	A3	29	13	1			
26408-01	A2	38	10	1	26445-01	FR408	A2	31	7	1			
26408-02	A2	38	10	1	26446-01	A3	18	10	1				
26409-01	A2	31	9	1	26446-02	A3	18	10	1				
26409-02	A2	31	9	1	26447-01	A3	20	8	1				
26410-01	A3	17	10	1	26447-02	A3	20	8	2				
26410-02	A2	17	10	1	26447-03	A2	20	0	1				

圖 2-20 表單與其 Well-Formed XML 格式對應

在 Well-Formed XML 格式中，各標籤組成皆為其 XML 元素，XML 格式文件並以階層式的方式呈現出其邏輯結構，如<交易筆>的子元素則有<供應號>、<序號>、<品名>、<等>、<單價>、<把數>，與<箱數>等七個主要組成，而<供應人小行情>標籤則有每個<交易筆>所組成，此外每組非空值的標籤皆予以成對出現。

### 2.6.3 XML 文件排版樣本

使用 XML 文件來描述資料為一個非常好的選擇，但在呈現視覺化的資料時，卻不比 HTML 更適於顯示資料，例如使用瀏覽器來讀取 XML 文件資料，顯示的結果為上圖 2-20 右方的樹狀階層呈現 XML 文件資料，如此的呈現方式在使用者的理解上較不直觀。於是在 XML 文件的顯示上，便以排版樣本的技术使得 XML 格式資料可呈現出 HTML 網頁般易於閱讀的效果，並可透過排版樣本進行 XML 文件內容的資料篩選及排序等功能。

常見的排版樣本可分為兩種，Cascading Style Sheet(CSS)與 eXtensible Stylesheet Language(XSL)排版樣本，CSS 排版樣本為 HTML 所採用的輔助顯示樣板，在 XML 文件的排版設計上由於能力發揮上的許多限制，XML 技術便制定了另一套 XSL 排版樣本，目前已為 W3C 的標準，透過特殊的轉換機制已達成了比 CSS 更強大的功能，此外 XSL 排版樣本也為一個 Well-Formed XML 文件，因此，XSL 文件內的標籤元素為標準化特定涵義的標籤，用來處理 XML 文件內的標籤顯示，以便依照 XSL 排版樣本輸出規定將 XML 文件顯示轉換出來。因此在 XSL 處理器解讀 XML 文件之後，便可以產生相對的輸出，其 XSL 的架構如圖 2-21 所示：

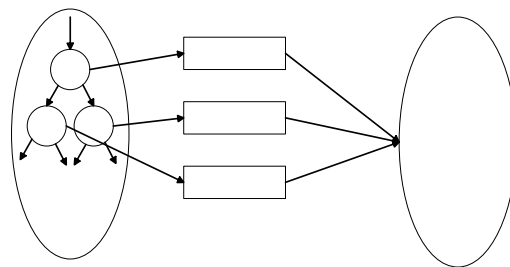


圖 2-21 XSL 轉換架構圖

而在 XSL 的轉換方法而言，其常見的排版樣本法則如<xsl:stylesheet>、<xsl:template>、

<xsl:for-each>、<xsl:apply-templates>，及<xsl:value-of>等處理 XML 文件的樣板元素，在下一章節表 3-6 及圖 3-8 並有 XSL 轉換樣板及轉換後顯示於瀏覽器說明。



## 第三章 交易表單與其 XML 規格分析

在花卉交易資訊處理系統的整合設計與實作過程中，在本章首先說明了目前在花卉供應鏈體系中花卉交易資料傳遞的架構，其次建立一個新的整合交易資訊處理架構，接著將花卉公司所傳送的花卉交易資訊樣張統一成相對應的 XML 文件，並利用該格式文件搭配爪哇訊息服務技術，建構出整合後的花卉交易資訊傳遞系統。第 3.1 節「整合系統介紹」在概述此整合系統如何處理不同身分其相對接收的交易表單，第 3.2 節「花卉交易資訊傳遞問題探討與架構」在詳細說明各花卉公司的交易資訊流以供不同屬性的交易表單作其傳送目的地的分析，第 3.3 節「交易表單規格整合」在說明交易表單在整合時所進行的步驟與其規範比較，第 3.4 節「花卉交易資訊 XML 文件使用」在以供應人小行情為例說明 XML 標準規格的文件格式並使用 XML 的子技術 XSL 轉換視覺化的呈現兩家公司整合後的文件內容。

### 3.1 整合系統介紹

隨著企業電子化的普及與企業規模的成長，企業的整合問題便隨之而來，且為了解決企業所面臨到的問題，也帶動了科技的進步。本章將從技術架構的角度，來說明花卉交易資訊處理系統的整合設計與實作過程，而在花卉交易資訊處理過程中，以彰化花卉公司與台北花卉公司為例來說明，主要是針對彰化在建制交易資訊處理系統時，與台北花卉公司所面臨之系統整合所遇到的問題做探討。

目前彰化花卉公司與台北花卉公司傳送花卉交易資訊主要採電腦語音傳真方式傳送至各市場的供應人與承銷人。以花卉公司的觀點來看，採用此方式對供應人而言，由於各供應人代號在整個花卉供應鏈內為一對一的配與各供應人，亦即不論該供應人在台北或彰化花卉公司，其供應代號是唯一的，所以在接收當日交易文件的供應人對於該交易表單，可分為彰化與台北的兩份當日交易資料；對承銷人而言，承銷人代號在不同的花卉公司即有不同的代號，於是在處理該日交易的表單時即有彰化與台北花卉公司兩份相同承銷人代號的不同文件。於是除了傳送文件所必須的郵寄成本與資料的可再利用性外，最重要的就是文件的統合。供應人代號既是唯一的，即可以供應代號為主鍵，將兩花卉公司同質性的交易資料更友善的整合送出；而承銷人代號根據所在花卉公司的不同而有所差異，應以不同的文件形式呈現於各花卉公司的承銷人，於是在本章為改進彰化花卉公司與台北花卉公司對於花卉交易資訊傳遞整合所遇到的問題，採用 XML 技術與爪哇訊息服務技術，將目前花卉公司的資訊整合，並讓交易資訊更加便時便利的呈現於供應人與承銷人的電腦內，對於交易資訊內容更進一步的再利用[9]。

### 3.2 花卉交易資訊傳遞問題探討與架構

本節在針對花卉產業在規模擴充時的整合問題探討，並在軟體系統建置前作一細部的規範考量。第 3.2.1 節「花卉交易資訊傳遞問題」在說明目前花卉交易的傳送方式與問題所在，第 3.2.2 節「資訊傳遞架構」說明花卉的多條供應鏈在資訊流部份的處理細節。

#### 3.2.1 花卉交易資訊傳遞問題

在彰化花卉公司的交易資訊傳遞方式中，目前主要以語音及傳真的方式傳送給各供應人及承銷人，而在台北花卉公司的傳送方式中，除了語音及傳真的方式外，尚有郵寄、網際網路、及電子信箱等方式傳遞交易資訊，在眾多的傳遞方式中，即使傳送方式的多樣化，但除了郵寄及語音傳真費用的浪費外，以電子信箱、網際網路的方式仍不能充分地利用個人在完成交易後的數據資訊，且在取得了該交易文件後，各花卉公司單據的不一，往往在整理文件時造成不必要的困擾。所以為解決各花卉公司在交達客戶的交易資訊時，除了每年高額成本的浪費及資訊無法再利用的功能外，並以整合各公司單據為標的，將花卉交易資訊傳遞的方式，採用 XML 文件格式的方式使電腦能簡易迅速的識別該訊息所內含的珍貴資訊，並加以區別客戶在所屬花卉公司進行的各項交易紀錄。

#### 3.2.2 資訊傳遞架構

在目前花卉交易資訊傳遞系統中，主要的系統成員由彰化花卉公司、台北花卉公司、各花卉供應人與承銷人所組成，其交易資訊流分別由各花卉公司傳送交易資訊到當日所交易的供應人與承銷人，圖 3-1 以 IDEF0 技術描述出目前花卉當日交易資訊的傳遞情形，該交易資訊傳遞的過程主要是當日拍賣結束後，每筆拍賣的交易內容為其輸入，花卉交易資訊接收端則為供應人與承銷人，其中承銷人又細分為彰化花卉公司承銷人與台北花卉公司承銷人。

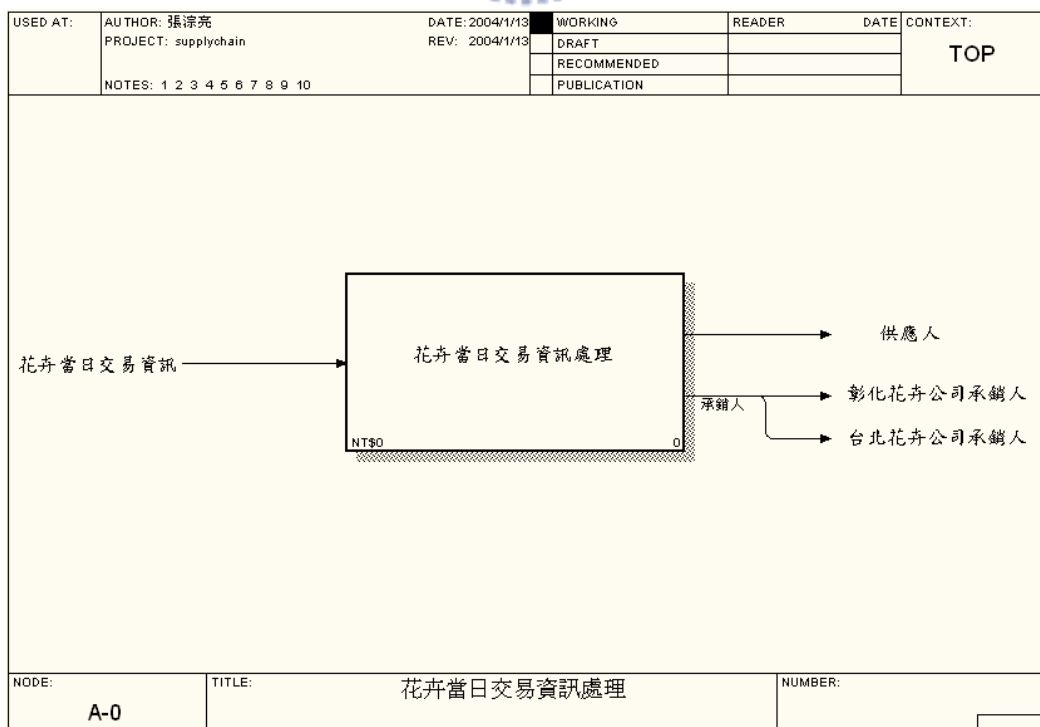


圖 3-1 花卉當日交易資訊傳遞過程



圖 3-1 再細分為圖 3-2 彰化花卉當日交易資訊處理與台北花卉當日交易資訊處理兩個部份。以接收端的點來看，對供應人而言，由於供應人代號唯一，所以在交易訊息接收時是來自彰化花卉公司與台北花卉公司的兩份交易文件；對承銷人而言，彰化花卉與台北花卉公司有個別的承銷人代號，因此彰化花卉公司的承銷人固然不可取得台北花卉公司同代號的私人交易資料，反之台北花卉公司的承銷人亦然。在圖 3-2 中，實線部份為個人性私密的交易資訊，而虛線部份則表示公開性的資料，如彰化花卉當日交易大行情，為當日交易的統計資料，屬公開性資料，於是彰化花卉公司在交易資訊處理的過程中，可另行開放其統計資料給台北花卉公司的承銷人所有，相同地，彰化花卉公司承銷人亦可得到台北花卉公司的大行情統計資料。圖 3-3 表示圖 3-2 中彰化花卉公司當日交易資訊處理中的細節部分；在圖 3-3 中，由於供應人小行情與承銷人結帳為私人性的交易資料，屬於各供應人與承銷人所有，而大行情的部分則公開予彰化花卉公司與台北花卉公司的供應人與承銷人。在小行情的部分，小行情為供應人在當日所進行的交易資料，每一處理過程如拍賣成交、殘貨或著是訂貨均有所詳細記錄，根據這些紀錄提供各地區地方的供應人作為交易的憑據，在承銷人匯款清單的部分，為該市場承銷人所購得花種、數量及購得花價的紀錄，而開放性的資料類別，大行情則為每天交易結束後，根據拍賣花卉種類在拍賣價格做數據上的統計。圖 3-4 則為台北花卉公司中當日交易資料處理的細節部分。

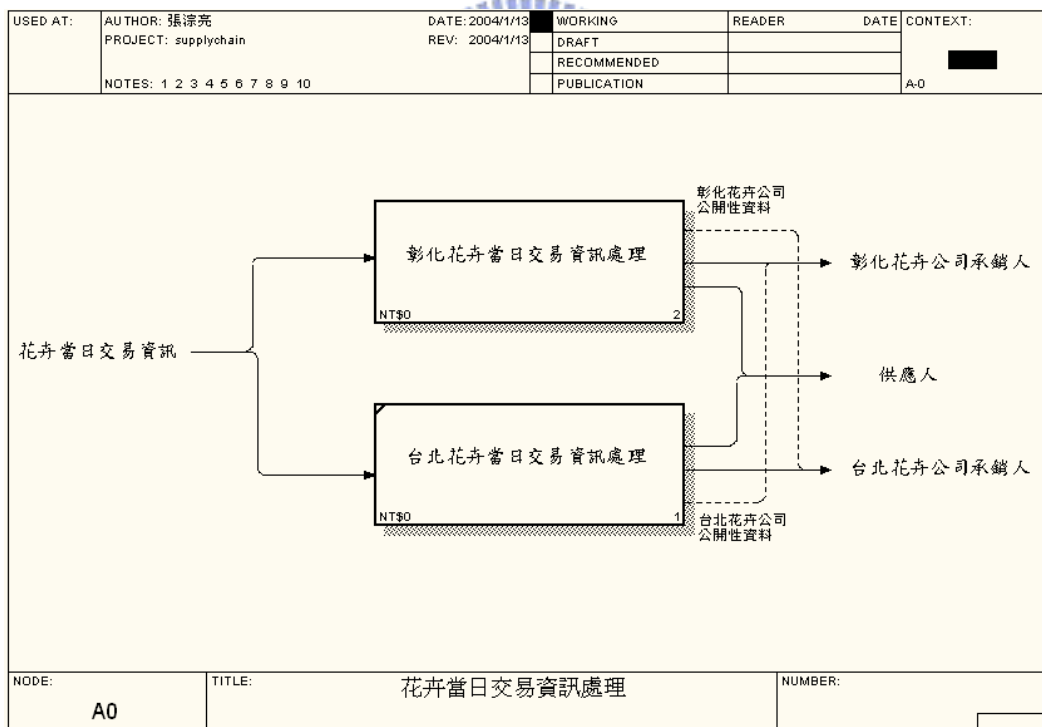


圖 3-2 台北花卉資訊傳遞方式

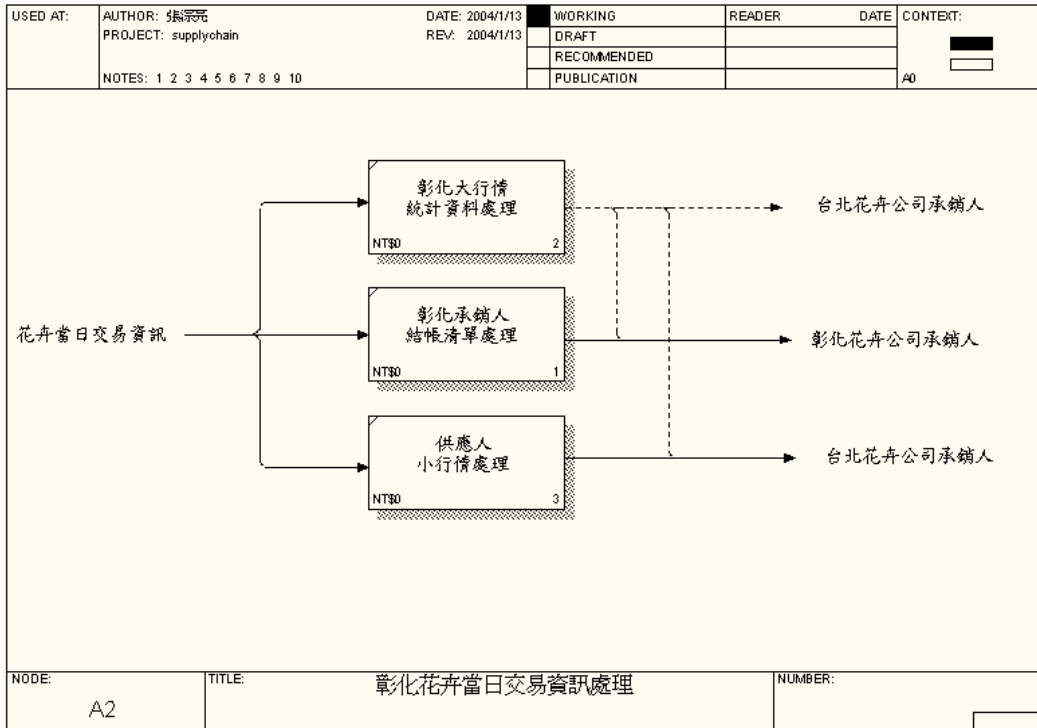


圖 3-3 彰化花卉交易資訊處理細節

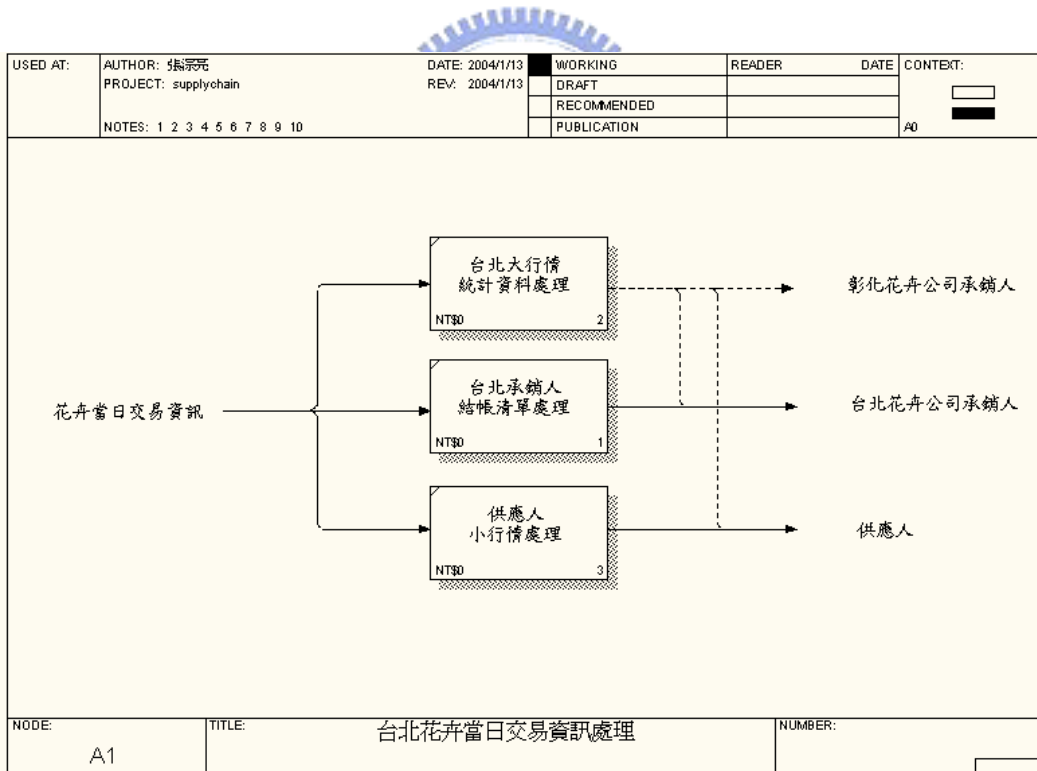


圖 3-4 台北花卉交易資訊處理細節

### 3.3 交易表單規格整合

由於 WWW 的普及，Html 成為網路交流上最通行的語言，但 Html 標籤只適用於將資料呈現於使用者瀏覽器上，並不具任何內容的代表性，更不能對這些標籤內容值做更進一步的處理，於是為使這些標籤能透過電腦的辨別使用，XML 即為當前最被看好的網路資訊整合工具。而在花卉公司表單的整合過程中，首先必須蒐集相關的交易文件，並透過與花卉公司的規格協調將各表單欄位予以整合，再以整合後的欄位別更進一步的制定 XML 規格文件，本論文即以下面流程圖作為 XML 文件在整合步驟時的規劃。第 3.3.1 節「表單規格蒐集」在比較各表單的取得方式，第 3.3.2 節「表單分析整理」在比較各表單的欄位異同，第 3.3.3 節「表單 XML 規格制定」在將各表單規格整合後的 XML 標籤元素作一規範整合。

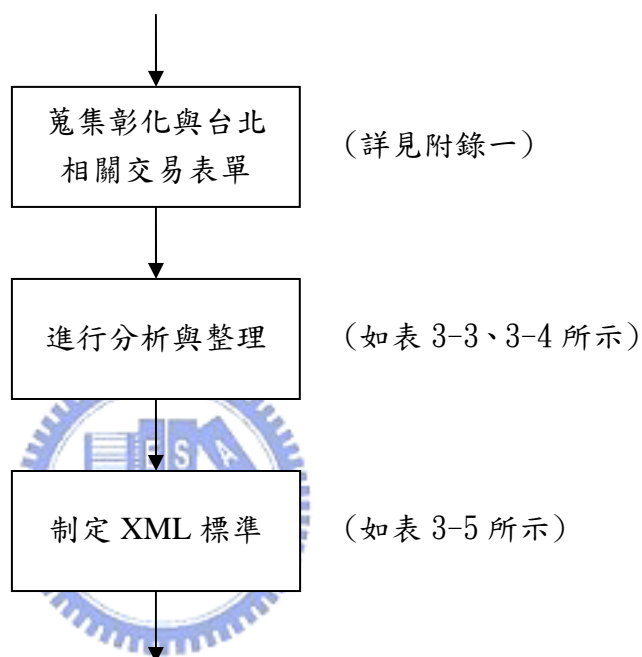


圖 3-5 XML 規格制定流程

#### 3.3.1 表單規格蒐集

在目前各花卉公司所提供的交易表單種類與其交易傳遞方式分別可以下列表格所示，表 3-1 為彰化花卉公司所提供的表單傳遞方式與種類，表 3-2 則為台北花卉公司所傳送表單的方式種類。其中信箱投遞指各花卉公司將交易單據放入該地零批場承銷人的信箱內，而信件郵寄則委託郵局以寄送方式傳遞交易資訊，在蒐集了相關的交易表單後，表單資料可參閱附錄一所示。

表 3-1 彰化花卉公司表單提供方式

資訊接收者	小行情	大行情	匯款清單	結帳清單
信箱投遞				承銷人
信件郵寄			供應人	
電腦語音	供應人	供應人/承銷人		
電腦傳真	供應人	供應人/承銷人		承銷人
網頁公佈		供應人/承銷人		

表 3-2 台北花卉公司表單提供方式[9]

資訊接收者	小行情	大行情	匯款清單	結帳清單
信箱投遞				承銷人
信件郵寄			供應人	
電腦語音	供應人	供應人		
電腦傳真	供應人	供應人		
網頁公佈	供應人	供應人/承銷人		

### 3.3.2 表單分析整理

在各花卉公司的表單文件中，個人性的文件有供應人小行情、供應人匯款清單與承銷人結帳清單三種，公開性的交易文件則有大行情統計資料。在經由各文件欄位別的統一整理後以表 3-3 及 3-4 所示，以作為 XML 規格統一的文件依據。其中表內括號部份代表相同欄位意義但以不同名稱表示於其表單內。

表 3-3 個人性表單欄別分析

表單	供應人小行情		供應人匯款清單		承銷人結帳清單	
	彰化	台北	彰化	台北	彰化	台北
市場						
(交易)日期		V	V	V		
供應人代號	V(供應號)	V(供應商)			V(供應號)	V(供應人代號)
供應人						
承銷人			V(承銷商)	V(承銷商)		
交易序號	V	V	V(序號)	V(序號)	V(序號)	V(拍賣序號)
交易別		V				
品名代號	V	V	V(品號)	V(品名)		
品名		V	V	V	V	V
等級	V	V	V	V	V	V
單價	V	V	V	V	V	V
把數	V	V	V	V	V	V
箱(件)數	V(箱數)	V(件數)	V	V	V	V
總金額			V	V	V	V
合計欄數	7	10	10	10	8	7

表 3-4 公開性表單欄別分析

表單	大行情統計資料	
	彰化	台北
市場		
交易日期		
品名代號	V(代號)	V(品名)
品名	V(名稱)	V(名稱)
進貨量	V	V
拍賣量	V	V
訂貨量	V	V
行口量		V
殘貨量	V	V
訂貨價	V(訂貨價)	V(定高價)
上價	V	V
中價	V	V
下價	V	V
最高價	V	V
最低價	V	V
合計欄數	12	13

### 3.3.3 表單 XML 規格制定

在交易表單的整合過程中，除了藉由上述的表單整理，並透過花卉公司的 XML 規格協調，在相同意義卻不同名稱的欄別賦予相同的欄位代號(如：供應人小行情表單的供應商與供應號欄別名稱)，並透過 XML 彈性的標籤辨識，在每一張交易的單據上加上”市場”的欄位，以作為各供應人與承銷人在接收 XML 文件時辨別其當日在何處所進行的交易，其各表單規格整合後的 XML 標籤元素分別組成如下表 3-5 所示，透過 XML 規格的整合，在彰化與台北花卉公司的交易資訊便可以統一的格式傳送至遠端的各供應人與承銷人資料庫，使資料更具透通性的往來花卉業務的資訊流中。

表 3-5 交易表單 XML 標籤元素組成一覽表

表單	供應人小行情	供應人匯款清單	承銷人結帳清單	大行情
XML 元 素 組 成	<市場>	<市場>	<市場>	<市場>
	<日期>	<供應人代號>	<日期>	<交易日期>
	<供應人代號>	<日期>	<交易序號>	<品名代號>
	<交易序號>	<交易序號>	<承銷人>	<品名>
	<交易別>	<品名代號>	<供應人代號>	<進貨量>
	<品名代號>	<品名>	<供應商名稱>	<拍賣量>
	<品名>	<等級>	<品名代號>	<訂貨量>
	<等級>	<單價>	<品名>	<行口量>
	<單價>	<把數>	<等級>	<殘貨量>
	<把數>	<箱數>	<單價>	<訂貨價>
	<箱數>		<把數>	<上價>
	<總金額>		<箱數>	<中價>
			<總金額>	<下價>
				<最高價>
			<最低價>	
合計	12	10	13	15



### 3.4 花卉交易資訊 XML 文件使用

本節主要在以供應人小行情表單作為範例，第 3.4.1 節「花卉交易資訊 XML 文件使用」在該表單作一 XML 規範文件的整合示範，第 3.4.2 節「XSL 轉換」在以 XSL 技術將整合後的 XML 文件視覺化的呈現於使用者瀏覽器。

#### 3.4.1 XML 標準格式文件

在整合花卉交易資訊 XML 文件來源中，分別使用了彰化及台北花卉公司的供應人小行情、供應人匯款清單、大行情與承銷人結帳清單等四份花卉交易資料，設計屬於各文件的 XML 文件以作為傳遞花卉交易資訊時，搭配爪哇訊服務所應用的傳輸內容，本論文以供應人小行情為例，如圖 3-6 與圖 3-7 所示，圖 3-6 左邊為供應人小行情單據，右邊則為本資訊處理系統整合後的 XML 標準格式文件 [13]。在圖 3-6 中彰化花卉公司的小行情單據，主要是由日期、供應號、序號、品名、等級、單價、把數、箱數所組成，而圖 3-7 中台北花卉公司的小行情單據，主要組成則為日期、供應商、交易序號、交易別、品名、名稱、等級、單價、把數、件數等。

在彰化和台北花卉公司在與供應人異質資料庫進行資訊傳遞的文件中，由於少數個欄位名稱的不一，但卻擁有相同的涵義(如:箱數與件數)，及數個欄位名稱的有無(如:花卉品名所屬的花類名稱)，在整合上透過與兩家花卉公司的協調，制定應有的欄位與欄位名稱，並加以花卉公司的市場別欄位，做為該筆交易所進行場所的識別區分。當該花卉公司所傳送的 XML 文件訊息符合供應人資料庫時，則供應人資料庫承認此花卉公司所傳送的 XML 文件為有效的(Valid)，並允許所傳送的花卉交易資訊 XML 文件訊息存入供應人資料庫，以進行供應人更進一步的分析。

供應號	序號	品名	等	單價	把數	箱數	供應號	序號	品名	等	單價	把數	箱數
KH001	26400-01	FR108	A+	79	9	1	KH001	26437-01	FR412	A1	30	5	1
26401-01	A1	82					26438-01	A2	26	7	1		
26402-01	A3	29					26439-01	A3	18	8	1		
26403-01	FR601	A1	64	8	1		26439-02	A3	18	8	1		
26404-01	FR108	A2	62	10	1		26440-00	A4	殘貨	8	1		
26405-01	A2	59					26441-01	FR108	A1	102	10	1	
26406-01	A3	28					26442-01	A2	72	11	1		
26407-01	FR408	A1	53	7	1		26443-01	A3	29	13	1		
26408-01	A2	38					26445-01	FR408	A2	31	7	1	
26408-02	A2	38					26446-01	A3	18	10	1		
26409-01	A2	31					26446-02	A3	18	10	1		
26409-02	A2	31					26447-01	A3	20	8	1		
26410-01	A3	17					26447-02	A3	20	8	2		
26410-02	A2	17					26447-02	A2	20	8	1		

```

<?xml version="1.0" encoding="BIG5"?>
<供應人小行情>
  <交易筆>
    <市場>彰化</市場>
    <日期>91.7.2</日期>
    <供應號>KH001</供應號>
    <交易序號>26400-01</交易序號>
    <交易別>拍賣成交</交易別>
    <品名代號>FR108</品名代號>
    <品名>玫瑰香檳粉</品名>
    <等級>A+</等級>
    <單價>79</單價>
    <把數>9</把數>
    <箱數>1</箱數>
    <總金額>711</總金額>
  </交易筆>
</供應人小行情>
  
```

規格整合後的 XML 標籤元素

圖 3-6 彰化供應人小行情與 XML 標準格式文件對照

日期	供應號	交易序號	交易別	品名	等級	單價	把數	件數
9910702	KH001	1781601	成交	玫瑰愛麗莎	A4	32	9	1
9910702	KH001	1781801	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1781901	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782001	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782101	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782201	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782301	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782401	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782501	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782601	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782701	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782801	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1782901	成交	玫瑰愛麗莎	A1	28	10	1
9910702	KH001	1783001	成交	玫瑰愛麗莎	A1	28	10	1

```

<?xml version="1.0" encoding="BIG5"?>
<供應人小行情>
  <交易筆>
    <市場>台北</市場>
    <日期>91.7.2</日期>
    <供應號>KH001</供應號>
    <交易序號>17816-01</交易序號>
    <交易別>拍賣成交</交易別>
    <品名代號>FR425</品名代號>
    <品名>玫瑰愛麗莎</品名>
    <等級>A+</等級>
    <單價>32</單價>
    <把數>9</把數>
    <箱數>1</箱數>
    <總金額>288</總金額>
  </交易筆>
</供應人小行情>
  
```

圖 3-7 台北供應人小行情與 XML 標準格式文件對照

### 3.4.2 XSL 轉換

XML 技術的成熟與多方面的應用，透過 XML 文件格式的交易傳輸後，供應人與承銷人在本地端儲存該筆交易資料，除了要求交易資料的正確性外，在本地端瀏覽該交易資料更是 XML 技術的範疇，XML 文件在經由 XSL 檔案的轉換(表 3-6)後，可以瀏覽器呈現出如各花卉公司報表的效果[13]，如圖 3-8，其中更以顏色識別區分各筆交易在何處所進行，以更友善、明確的表達該供應人在當日所有進行的交易資訊。

表 3-6 花卉拍賣資料供應人小行情的 XSL 轉換[6]

```
<? xml version="1.0" encoding="Big5"?>
<xsl:stylesheet version="1.0" xmlns:xsl=http://www.w3.org/1999/XSL/Transform
  xmlns:fo="http://www.w3.org/1999/XSL/Format">
<xsl:template match="/供應人小行情">
<html>
<head>
<title>花卉訊息--供應人<xsl:value-of select="交易筆/供應商"/>小行情訊息資料
</title>
</head>
</html>
<body bgcolor="#ccccff">
<p align="center">花卉訊息訂閱者</p>
<p align="center">供應人<font size="+2" color="#FF0000">小行情</font>資料</p>
<table width="98%" border="1" bordercolor="#909090" cellspacing="0" cellpadding="2" align="center">
  <tr bgcolor="#9999FF">
    <td>
      <div align="center">市場別</div>
    </td>
    <td>
      <div align="center">日期</div>
    </td>
    <td>
      <div align="center">供應號</div>
    </td>
    <td>
      <div align="center">交易序號</div>
    </td>
    <td>
      <div align="center">交易別</div>
    </td>
    <td>
      <div align="center">品名代號</div>
    </td>
    <td>
      <div align="center">品名</div>
    </td>
    <td>
      <div align="center">等級</div>
    </td>
    <td>
      <div align="center">單價</div>
    </td>
    <td>
      <div align="center">把數</div>
    </td>
    <td>
      <div align="center">箱數</div>
    </td>
    <td>
      <div align="center">總金額</div>
    </td>
  </tr>
```



```

<xsl:for-each select="交易筆">
<xsl:variable name="Market" select="市場"/>
<xsl:choose>
<xsl:when test="$Market[.='彰化']">
<tr bgcolor="#bfe7e7">
<td>
<div align="center"><xsl:value-of select="市場"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="日期"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="供應號"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="交易序號"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="交易別"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="品名代號"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="品名"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="等級"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="單價"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="把數"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="箱數"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="總金額"/></div>
</td>
</tr>
</xsl:when>
<xsl:when test="$Market[.='台北']">
<tr bgcolor="#FCA5FB">
<td >
<div align="center"><xsl:value-of select="市場"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="日期"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="供應號"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="交易序號"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="交易別"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="品名代號"/></div>
</td>
<td>
<div align="center"><xsl:value-of select="品名"/></div>

```





花卉訊息-供應人MBA05小行情訊息資料

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(T) 說明(H)

地址: C:\Program File\Flower TIPS(供應人)\02\_12\_17供應人小行情.xml

花卉訊息訂閱者  
供應人小行情資料

市場別	日期	供應號	交易序號	交易別	品名代號	品名	等級	單價	把數	箱數	總金額
彰化	2003/12/10	MBA05	05669-01	拍賣成交	FG412	劍蘭粉巨人	A	77	10	1	770
彰化	2003/12/10	MBA05	05669-02	拍賣成交	FG412	劍蘭粉巨人	A	77	10	1	770
彰化	2003/12/10	MBA05	05669-03	拍賣成交	FG412	劍蘭粉巨人	A	77	10	1	770
(彰化)	2003/12/10	MBA05	總交易筆數: 3	-	-	-	-	-	-	3	2310 (結束資料)
台北	2003/12/10	MBA05	03621-02	拍賣成交	FG412	劍蘭粉巨人	A	50	10	1	500
台北	2003/12/10	MBA05	03620-01	拍賣成交	FG406	劍蘭青殼粉	B	20	10	1	200
台北	2003/12/10	MBA05	03621-01	拍賣成交	FG412	劍蘭粉巨人	A	50	10	2	1000
台北	2003/12/10	MBA05	03621-04	拍賣成交	FG412	劍蘭粉巨人	A	50	10	1	500
台北	2003/12/10	MBA05	03621-03	拍賣成交	FG412	劍蘭粉巨人	A	50	10	2	1000
(台北)	2003/12/10	MBA05	總交易筆數: 5	-	-	-	-	-	-	7	3200 (結束資料)

完成 我的電腦

圖 3-8 花卉拍賣資料供應人小行情 XSL 轉換後畫面





## 第四章 爪哇訊息服務理念與設計

在第二章提到網路多樣的架構中，本章節將說明在產業電子化中，如何以爪哇訊息服務實作其中訊息佇列與出版者/訂閱者(Publisher/Subscriber, Pub/Sub)等樣式，並在爪哇訊息服務技術中以花卉公司供應鏈整合體系作一細部探討。第 4.1 節「爪哇訊息服務傳訊樣式」解釋爪哇訊息服務的主要兩種傳訊樣式與其特性說明，4.2 節「爪哇訊息服務與整合花卉公司」即以爪哇訊息服務技術將花卉公司的整合過程作一關鍵性的探究與說明，4.3 節「文件整合系統開發設計」在以說明整合系統的開發至完成上線的流程經過。

### 4.1 爪哇訊息服務傳訊樣式

在爪哇訊息服務所提供的樣式中依據文件性質的不同，爪哇訊息服務並提供兩種不同功能的傳訊樣式，分別為傳送/接收樣式(第 4.1.1 節)與出版/訂閱(第 4.1.2 節)兩種樣式，其分述如下：

#### 4.1.1 傳送/接收傳訊樣式

傳送/接收(Sender/Receiver)傳訊樣式又稱為點對點(Point-To-Point, PTP)，其組成元件主要由傳送者(Sender)、接收者(Receiver)及訊息佇列(Message Queue)三者所組成。傳送者為交易訊息的生產端，主要負責將訊息傳送至目的訊息佇列中，在傳送的過程中必須指定此訊息的接收者為誰；訊息佇列的功用則為暫存與維護訊息，除了控制訊息的來源與去處外，訊息並在指定的期限內予以自動刪除，另外接收者與出版者無時間性的前後問題，不需先行註冊於該佇列上，而佇列中若有其接收者資料且無過期則可送出其相關訊息予此接收者。其三者關係如圖 4-1 所示，接收者必須先向訊息佇列提出身份確認之後方能接收所屬的訊息，在接收該筆訊息後並回報確認接收，將此訊息從佇列中移除，所以傳送/接收傳訊樣式的特色在於單一訊息只可由單一消費者所擁有[1]。

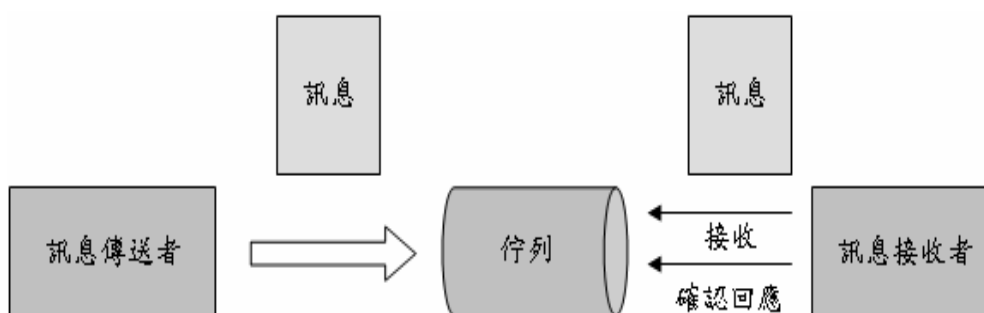


圖 4-1 傳送/接收傳訊樣式[8]

#### 4.1.2 出版/訂閱傳訊樣式

出版/訂閱(Publisher/Subscriber)傳訊樣式的組成主要為訊息出版者(Publisher)、訊息訂閱者(Subscriber)、與主題(Topic)三者。訊息出版者為訊息的生產端，主要將訊息傳送至爪哇訊息服務者的主題內；由於該訊息文件為共同性公開的資料，傳送的过程僅需指定該筆主題即可，不需指定訊息的接收者；主題主要暫存及維護訊息，並判斷訊息訂閱者是否為合法的用

戶，由於訊息的出版與訂閱具時間性的前後關係，訊息的訂閱者必先向該主題訂閱，而後主題依其訂閱名單確認其為合法用戶方送出其主題內容。其三者關係如圖 4-2 所示，訊息訂閱者從主題內接收訊息，但主題的內容並不會在單一接收者在接收完資料後予以刪除，主題的運作主要在一方接收端接收訊息後，以刪除其該筆主題文件的訂閱者名單為主，所以主題的特色主要為同一份訊息資料以複製給多位訂閱者，而不須重複的出版相同的訊息，使單一訊息可由多個消費者所有。

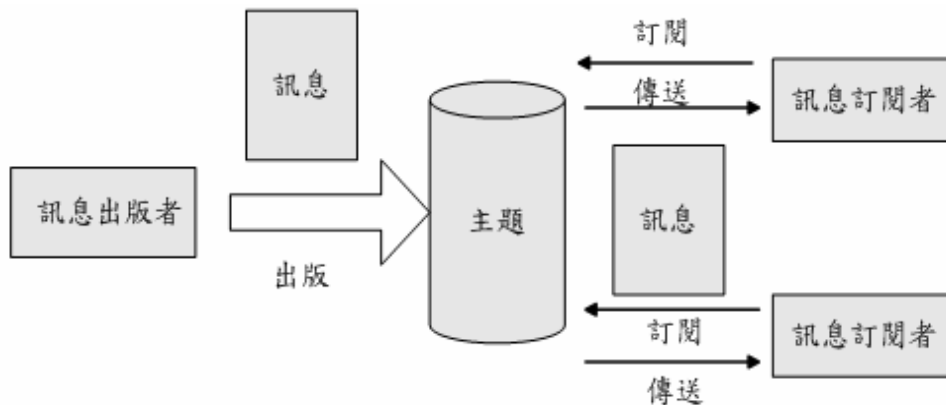


圖 4-2 出版/訂閱傳訊樣式[8]



## 4.2 爪哇訊息服務樣式與花卉公司文件整合

爪哇訊息服務技術的成熟，不單提供了訊息傳送的可靠品質，在其出版/訂閱與點對點兩種傳訊樣式的提供，並可用於企業交易資訊的整合傳遞，以花卉公司為例，目前爪哇訊息服務已實作了台北花卉與彰化花卉公司的兩家交易資訊處理系統，在兩者公司的交易文件整合上，除了以制定相同欄位名稱的 XML 文件作為傳送格式外，在公開性的大行情統計文件與私密性的個人文件裡，在本節以利用出版/訂閱與點對點兩種傳訊樣式，實作其資訊流在整合過程的處理機制。

花卉公司在交易文件上，如第三章所言，兩公司目前在供應人方面皆開放了大行情統計資料與小行情個人交易資料，而在承銷人方面則開放了大行情統計資料與個人結帳清單，而在各花卉公司的交易資訊流可以下圖表示，以下則分別以相同公開性的資料與個人的私密性資料作一探究。

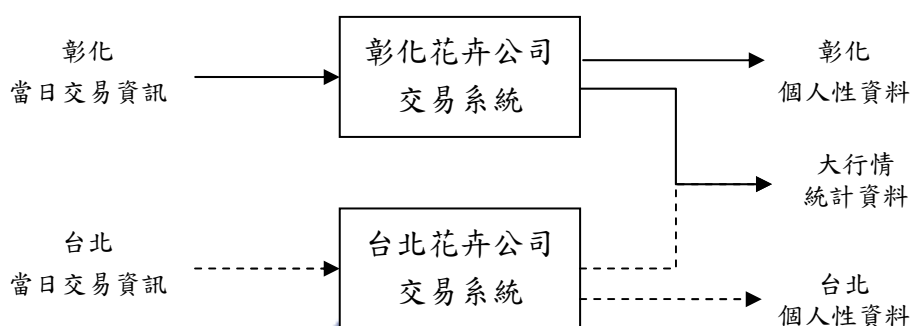


圖 4-3 花卉公司個人性與公開性資訊流示意圖

在大行情資料上由於兩花卉公司的個別統計資料皆開放給所有供應人與承銷人，所以使用了出版/訂閱樣式將個別的資料複製予所有訂閱在此主題上的訂閱者。在花卉公司的大行情資料區分上，除了彰化的客戶端與台北的客戶端皆可接收兩花卉公司的資料外，同時在客戶端的資料儲存以再利用上，考量到歷史資料的完整性，在此採以相同的主題名稱以供兩客戶端同時接收兩花卉公司的統計資料，並不可單獨接收個別的統計資料。但透過 XML 相同標籤(市場別)而不同內容值的方式，在客戶端接收並儲存資料後，仍可利用其加值功能查看個別花卉公司的統計資料。

此外，在供應人與承銷人的代號方面，由於彰化與台北供應人代號為統一發放的方式，供應人代號唯一，而承銷人代號則為個別的發放方式，承銷人代號不唯一，出版/訂閱機制為訂閱端接收完資料則將訂閱該筆訊息的訂閱清單予以刪除，亦即若相同代號的彰化花卉公司供應人與台北花卉公司的兩方供應人只可有一方接收到該筆資料，則此機制便不可行，故在此以訂閱者端採不同的供應代號識別碼加以區分，在其供應代號前加一個訂閱識別碼，如圖所示，則在彰化公司的供應人訂閱端即與台北花卉公司的供應人訂閱端有所區別，皆接收相同的一份統計文件。綜合其上，共使用了一個大行情主題與三方的訂閱端，一為供應人訊息訂閱端、二為彰化承銷人訂閱端，第三則為台北承銷人訂閱端。

在供應人小行情與承銷人結帳清單的文件上，由於兩份文件屬於個人性私密的資料，且在接收訊息正常的情況下，只可單一的收取該文件訊息，於是在此採以佇列的方式傳送個別交易訊息，而兩花卉公司共有四種交易文件，分別為彰化供應人小行情、彰化承銷人結帳清單、彰化供應人小行情、與台北承銷人結帳清單等。

在供應人方面，由於供應代號唯一，一個供應代號在彰化花卉公司與台北花卉公司為同一人，所以在文件的整合上除了以 XML 文件標籤區別該筆為何處的交易訊息外，在彰化供應人小行情與台北供應人小行情可由該供應人接收端統一接收，亦即將彰化供應人小行情與台北供應人小行情資料存放於同一佇列中，以由單一供應代號可同時收取兩公司文件內容。而在承銷人代號方面，不同花卉公司的承銷人會有不同的代號，所以依花卉公司交易資料個別存放的方式，彰化承銷人代號只可收取該代號在彰化所進行的交易資料，台北承銷人亦相同，但若該位承銷人在當日分別在彰化與台北花卉公司進行交易，則也可以兩公司個別給予的承銷人代號分別予以接收。綜合其上，共需使用三種佇列分別予以三種不同接收端，以供交易資訊的整合處理，分別為供應人小行情訊息佇列、彰化承銷人結帳清單訊息佇列、與台北承銷人結帳清單訊息佇列等。



### 4.3 文件整合系統開發設計

本節在說明各交易表單的整合系統建制流程，由文件的分析至系統的上線過程，4.3.1 節說明整合系統的開發流程，4.3.2 節在說明系統上線後的軟體取得。

#### 4.3.1 整合系統開發流程

在花卉公司的表單整合系統建制過程中，以供應人與承銷人作為整合軟體的分界，分為供應人軟體與承銷人軟體，在供應人軟體方面，共可接收小行情、匯款清單與大行情等三份文件，而承銷人軟體則可接收結帳清單與大行情等兩份交易文件。在各軟體的發展過程中首先進行各表單文件的整合分析，並制定 XML 標準規格(參照表)與爪哇訊息服務所合適的樣式規範(如 4.2 節所述)。在系統的實作上，採用 Borland 公司所開發的 JBuilder 軟體，以其雙向、視覺化的設計功能高效率的開發此爪哇訊息服務應用程式與圖形化使用者介面(Graphic User Interface, GUI)介面，而後在系統實作完成便可以網頁的方式提供各供應人與承銷人上網下載該整合系統軟體。以下則為此花卉文件整合系統開發的流程圖：

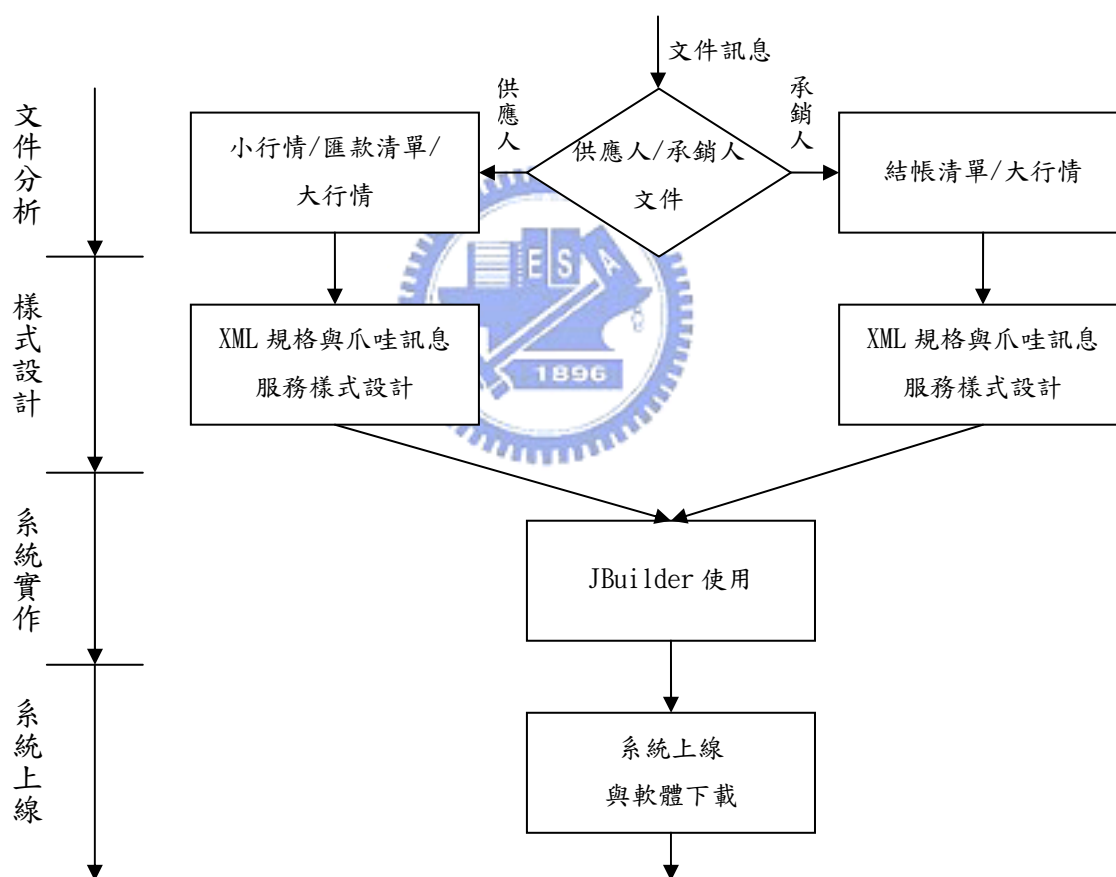


圖 4-4 文件整合系統開發流程圖

#### 4.3.2 系統下載

在整合系統建置完成後，花卉公司各供應人與承銷人可透過網頁的方式下載所需的應用軟體，其下載網址為 <http://eat.nctu.edu.tw/fdownload/index.html>，圖 4-5 為下載網頁畫面，在軟體下載並安裝完成後，使用者便可透過該整合系統收取每日在彰化或台北花卉公司所進行的交易資訊。





圖 4-5 文件整合軟體下載網頁

## 第五章 爪哇訊息服務應用操作

藉由前面章節所探討的 XML 與爪哇訊息服務技術，目前實作了其爪哇訊息服務的應用---花卉交易資訊處理系統(Flower\_Trading Information Processing System, Flower\_TIPS)，此花卉交易資訊處理系統為目前花卉產業的資訊處理系統，主要在將各花卉公司供應人以及承銷人當日在其批發市場所進行的交易資料，以電子式的方式將其資料傳送至各使用者端，並透過加值運用的方式將儲存於使用端電腦內的歷史資料加以利用及分析[9]。並透過詳細的花卉多條供應鏈資訊流架構探討，目前此花卉交易資訊處理系統已成功整合了彰化與台北花卉公司的客戶端交易文件，並利用其附屬的加值功能，使用者可對其每日接收的交易訊息作進一步的分析與決策，以下則分別以供應人與承銷人層面作一系統操作說明。第 5.1 節「爪哇訊息服務應用」介紹爪哇訊息服務的應用流程。5.2 節「使用者資料庫設定」在說明系統安裝後的初次設定方式以使每日接收的資料可存於使用者當地電腦，5.3 節「花卉交易資訊處理系統操作說明—供應人」在提供花卉交易資訊處理系統供應人軟體的使用說明，5.4 節「花卉交易資訊處理系統操作說明—承銷人」則說明花卉交易資訊處理系統的承銷人使用說明。

### 5.1 爪哇訊息服務應用

在各供應人與承銷人於網頁上下載軟體並安裝後，其軟體的使用方式可藉由下圖流程進行該系統的操作與運用。使用者在初次使用時需進行使用者本地資料庫的設定，在資料庫設定完成後，而後訊息的接收便不需重複進行資料庫的設定。在資料庫的正確設定後，使用者必須輸入該使用者帳號與密碼認證的參數，方能接收該帳號在該日所擁有的交易資訊，在訊息接收完成後，使用者便可對本地資料庫的資料進行加值應用的進一步分析。

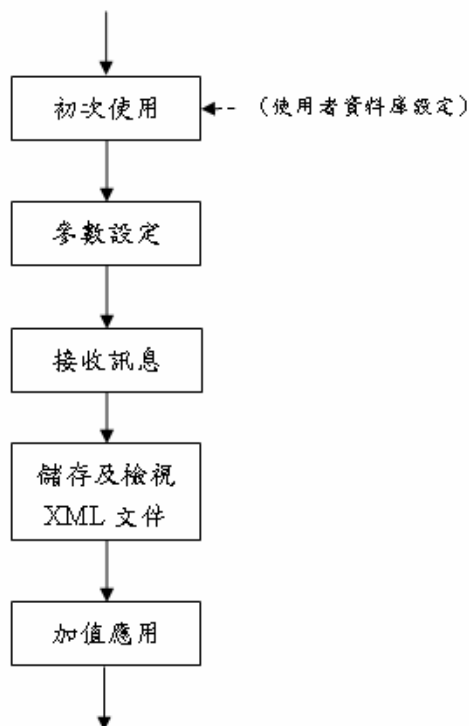


圖 5-1 系統操作流程圖

## 5.2 使用者資料庫設定

在各供應商與承銷人軟體使用時必須先進行使用者資料庫的設定，以將每日接收的資料存入使用者的資料庫內，方能進行而後的加值運用分析，其設定流程如下：

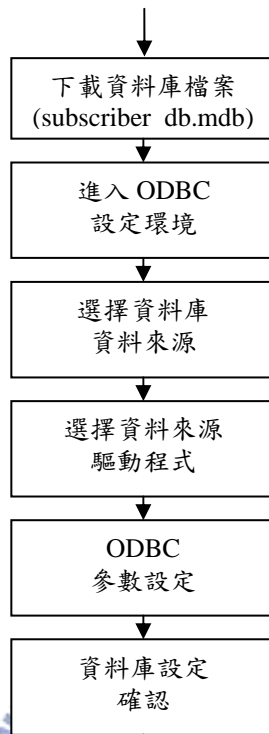


圖 5-2 使用者資料庫設定流程

在使用者資料庫設定的步驟詳述如下：

步驟一：

首先下載該網站所提供的 Access 資料庫 subscriber\_db.mdb，切勿使用自行設計 Access 資料庫，否則可能會造成資料存放時，產生錯誤情況出現。

步驟二：

從「開始/設定/控制台/系統管理工具」進入後，會出現如下的視窗，選擇「資料來源(ODBC)」的圖示，進入 ODBC 的設定環境。



圖 5-3 ODBC 路徑視窗

步驟三：

進入 ODBC 設定環境後選擇上方的「系統資料來源名稱」，如下所示，按下「新增」選擇一種資料來源型態做為驅動程式。

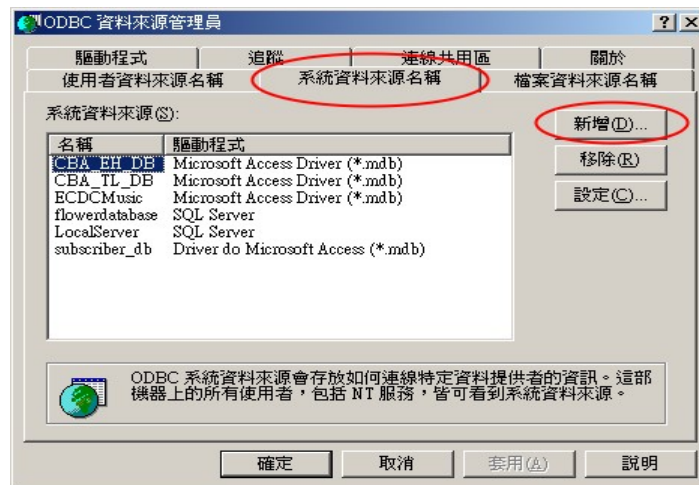


圖 5-4 新增系統資料來源

步驟四：

按下「新增」按鈕之後會要求使用者選擇一種適合的驅動程式來設定資料來源，由於使用者所使用的資料庫是 Access，所以需選擇「Microsoft Access(\*.mdb)」此驅動程式。



圖 5-5 選取驅動程式來源

選擇好之後，按下完成，以進行下一設定。

步驟五：

接著設定資料來源的相關設定，在「資料來源名稱」欄位為設定在連結資料庫時所使用的名稱，根據程式內部的寫法，在此請輸入『subscriber\_db』，「描述」這個欄位則是對此資料來源做說明，可以依照使用者喜愛輸入，在此假定為『花卉訂閱者資料庫』，填寫完畢之後，按下「選取」按鈕來設定存放資料的 Access 資料庫，在此必需使用從網站上下載回來的 Access 資料庫 subscriber\_db.mdb。由於所存放的 subscriber\_db.mdb 資料檔位置不同，使用者只需求要選取到下載回來的 subscriber\_db.mdb 資料檔即可，再按下「確定」就可以完成 ODBC 連線

設定。



圖 5-6 ODBC 參數設定

步驟六：

設定完成之後，會發現在系統資料來源中顯示出使用者之前所設定的 subscriber\_db，如此 ODBC 設定即算是完成使用者資料庫的設定。

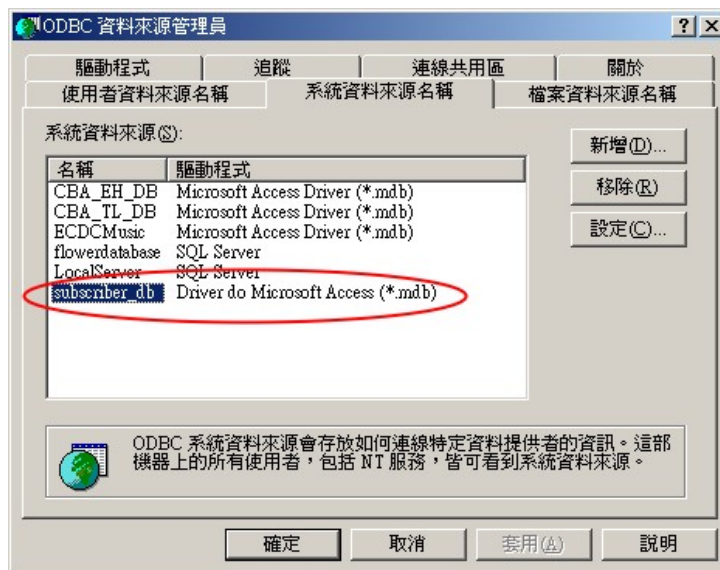


圖 5-7 使用者資料庫設定完成



### 5.3 花卉交易資訊處理系統操作說明—供應人

在供應人第一次使用此軟體時，必須先完成訂閱端 Access 資料庫與 ODBC 設定，完成 Access 資料庫與 ODBC 設定才能儲存拍賣花卉交易資料。在使用者資料庫設定完成後，以下七小節依序為該軟體操作步驟，由參數設定(第 5.3.1 節)>啟動接收(5.3.2)>停止接收(5.3.3)>轉入資料庫(5.3.4)>儲存 XML 文件(5.3.5)>檢視 XML 文件內容(5.3.6)>加值應用功能(5.3.7)：

#### 5.3.1 參數設定

在選單上選擇『設定(Alt+F)』，然後選取『參數設定(Ctrl+U)』，或是在功能列上直接按下



，在出現的畫面如圖 5-8 輸入訂閱者帳號及訂閱者密碼，完成之後，接著選擇訂閱主題，而供應人可以選擇的訂閱的主題分別有『小行情』、『大行情』、等二項，完成所有的參數設定之後，最後按下『完成參數設定』按鈕即可。『重設參數』可以清除之前所輸入的訂閱者帳號與訂閱者密碼。



圖 5-8 供應人參數設定畫面

#### 5.3.2 啟動接收

在選單上選擇『設定(Alt+F)』，然後選取『啟動接收(Ctrl+C)』，或是在功能列上直接按下



，便可以根據之前所設定的參數與系統連線，假如驗證訂閱者帳號與訂閱者密碼正確無誤，便可以從伺服器上開始接收花卉訊息資料，要是訂閱者帳號與訂閱者密碼有錯誤或是伺服器目前有問題，可以直接從下面狀態列得知問題發生之處，在圖 5-9 則是顯示連接伺服器電腦成功時的畫面，而圖 5-10 則是顯示接收到供應人小行情的花卉拍賣資料畫面，在圖 5-3 中出現「結束資料」則是代表接收供應人小行情資料結束，可以停止與伺服器之間連線。

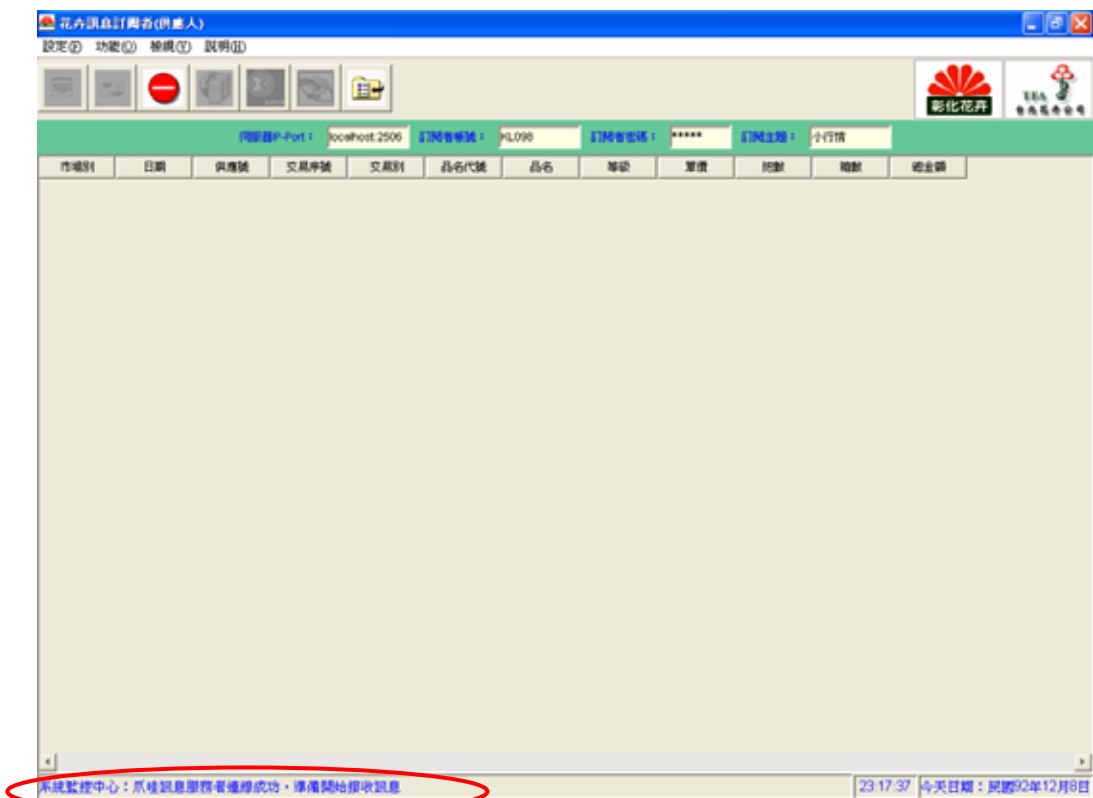


圖 5-9 與伺服器電腦連線成功畫面

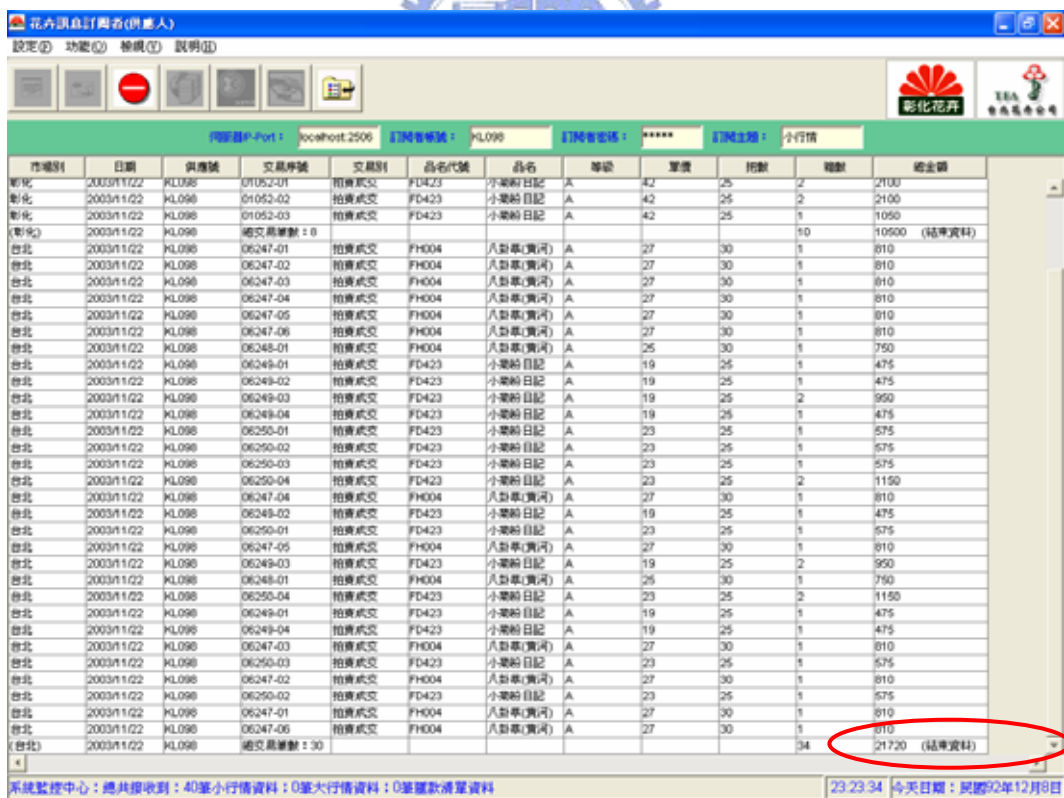




圖 5-10 供應商小行情的花卉拍賣資料接收畫面

### 5.3.3 停止接收

當在接收的過程中，接收畫面出現「結束資料」如圖 5-10 所示，便可以停止接收花卉資

料，在選單上選擇『設定(Alt+F)』，然後選取『停止接收(Ctrl+D)』，或是在功能列上直接按下 ，便可以停止與伺服器之間的連線接收資料過程。

### 5.3.4 轉入資料庫

對於所於接收回來的花卉資料，須先轉入訂閱端的資料庫系統，在轉入資料庫之前，必須先完成訂閱端 Access 資料庫與 ODBC 設定(如前面所述)，在選單上選擇『功能(Alt+O)』，然後選取『轉入資料庫(Ctrl+T)』，或是在功能列上直接按下 ，就可以將所接收到的花卉資料轉入資料庫內，在狀態列則會出現轉進資料庫成功畫面，假如轉進資料庫失敗，狀態列會出現轉進資料庫失敗提示訊息，轉進資料庫成功的畫面如圖 5-11 所示。

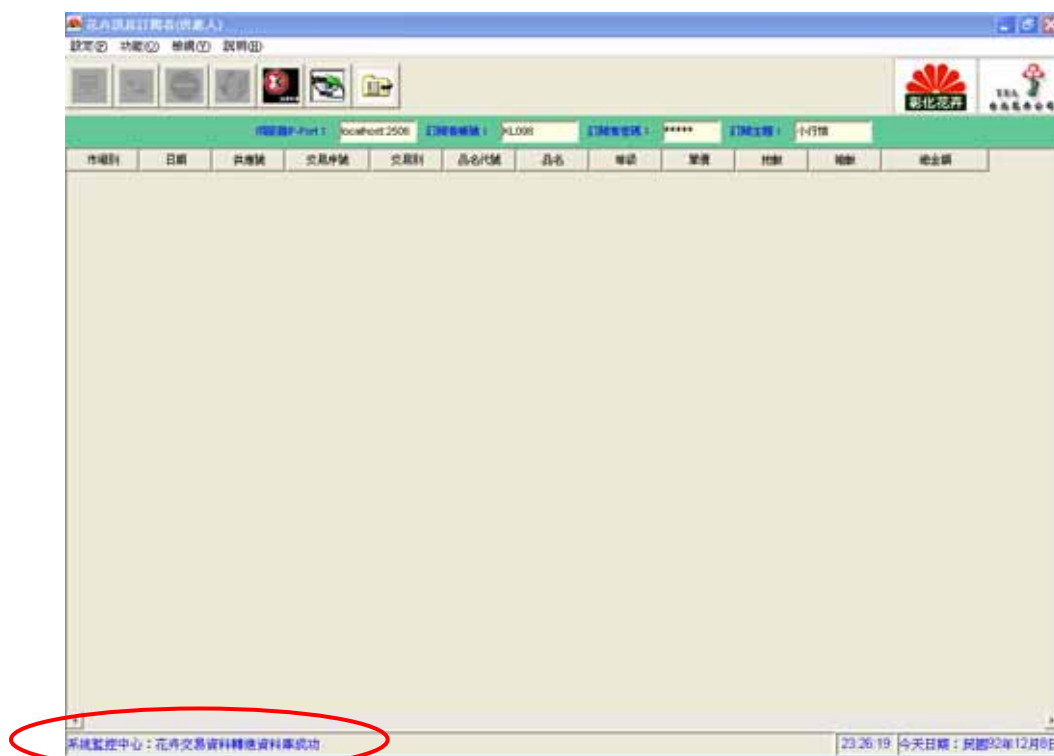



圖 5-11 轉進資料庫成功畫面

### 5.3.5 儲存 XML 文件

當完成轉入資料庫動作之後，才能儲存 XML 文件，在選單上選擇『功能(Alt+O)』，然後選取『轉入資料庫(Ctrl+S)』，或是在功能列上直接按下 ，就可以完成儲存 XML 文件的過程。

### 5.3.6 檢視 XML 文件內容


當只要完成接收花卉訊息資料之後，在選單上選擇『檢視(Alt+V)』，然後選取『XML 文件內容(Ctrl+S)』，或是在功能列上直接按下 ，圖 5-12 則是顯示出檢視 XML 文件的內容。



圖 5-12 檢視 XML 文件畫面

### 5.3.7 加值應用功能

目前花卉交易資訊處理系統所提供的功能有三種，一種是根據花卉品名種類的把數與件數的加總總和，一種為花卉品名單價平均值、變異數、標準差分析，第三種則為儲存於資料庫中的交易資料查詢。

在選單上選擇『功能(Alt+O)』，然後再選取『加值應用(Ctrl+A)』，則出現如圖 5-13 的加值應用功能畫面，選取「小行情計算把數、件數總和」功能，設定查詢資料的時間範圍，按下『開始查詢』按鈕，則可以查詢出某段時間內花卉品名種類的把數與件數總和，圖 5-14 是查詢結果以表格顯示，圖 5-15 則是查詢結果以長條圖方式顯示：

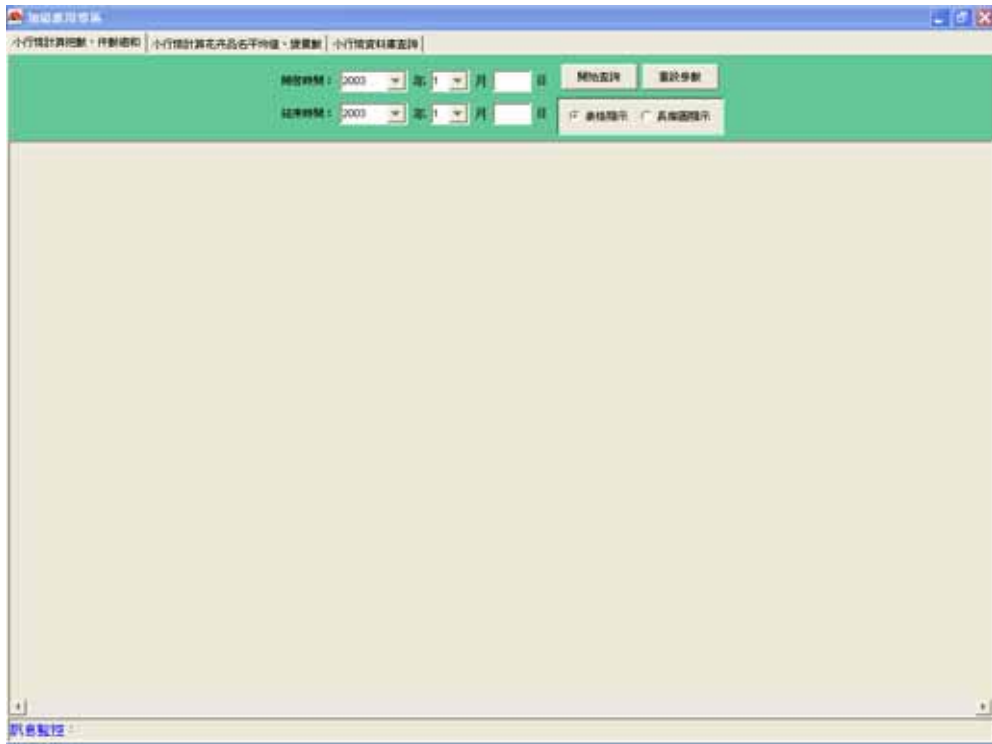


圖 5-13 加值功能應用畫面

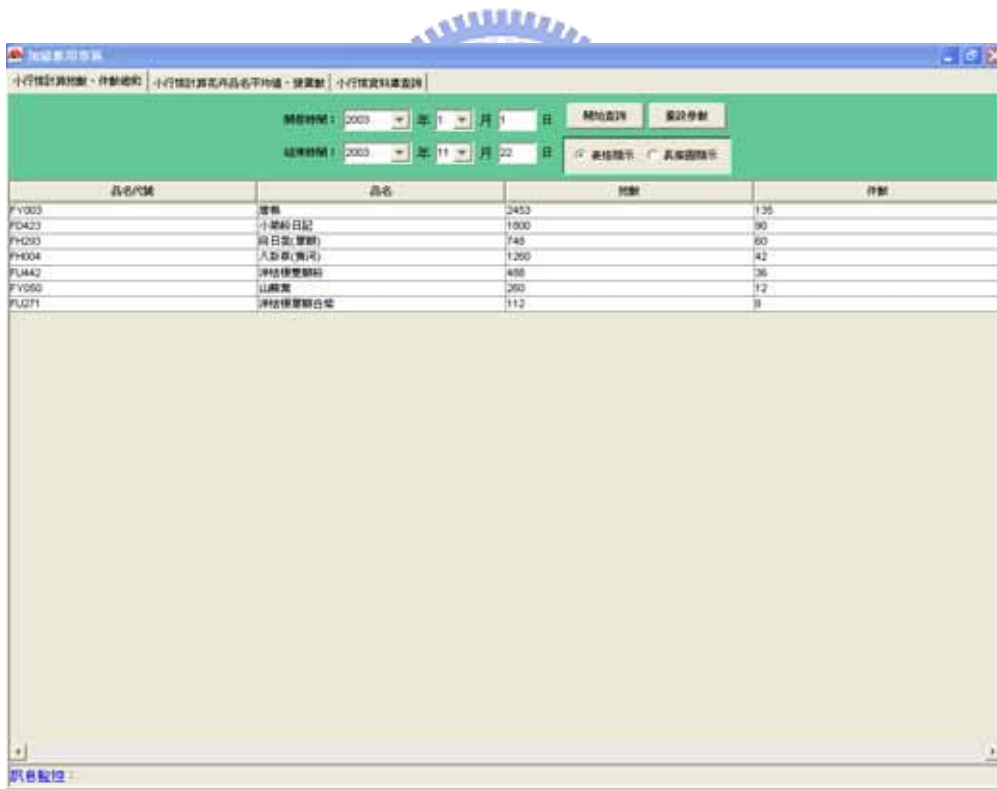


圖 5-14 查詢結果以表格顯示



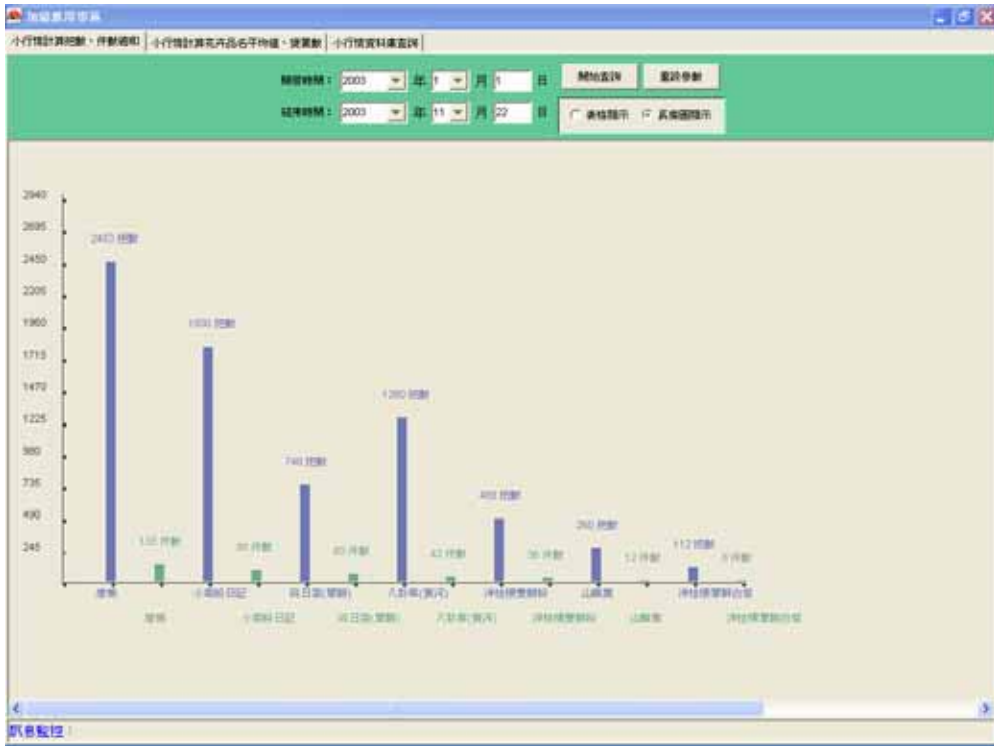


圖 5-15 查詢結果以長條圖顯示結果

在圖 5-13 選擇「小行情花卉品名平均值、變異數」選項，也是設定查詢資料時間範圍，設定好時間範圍後按下『開始查詢』按鈕，圖 5-14 則是小行情花卉品名平均值、變異數查詢結果以表格方式顯示出來。

名次	品名代號	品名	總量	單價-最大值	單價-最小值	單價-變異	單價-平均值	單價-標準差	單價-變異數
第1名	FD423	小蘭粉(日誌)	63000	42	19	72	29	10	100
第2名	FV003	唐華	53684	24	10	118	19	4	19
第3名	FU442	洋桔梗(雙瓣粉)	42104	503	73	36	86	11	138
第4名	FH604	八點草(黃/白)	39660	27	26	42	26	0	0
第5名	FH283	向日葵(單瓣)	29384	45	26	62	34	5	27
第6名	FUG71	洋桔梗(單瓣白藍)	12096	108	108	8	108	0	0
第7名	FV055	山繭菜	3900	15	15	12	15	0	0

圖 5-16 小行情花卉品名平均值、變異數查詢結果

在圖 5-13 選擇「小行情資料庫」選項，也是設定查詢資料時間範圍，設定好時間範圍、

市場別後按下『開始查詢』按鈕，圖 5-17 則是小行情舊資料查詢結果以表格方式顯示出來，並加以小計值。使用者若要查詢某當日交易資料，即輸入時間範圍為該日即可。

市場別	日期	委託號碼	交易序號	交易別	品名代號	品名	單位	數量	價格	總金額
期先	2003/1/22	NR400	03030-01	拍賣	FV003	債券	A	1.7	20	34
期先	2003/1/22	NR400	03030-03	拍賣	FV003	債券	A	1.6	20	32
期先	2003/1/22	NR400	03030-02	拍賣	FV003	債券	A	1.6	20	32
期先	2003/1/22	NR400	03030-01	拍賣	FV003	債券	A	1.6	20	32
期先	2003/1/22	NR400	03037-02	拍賣	FV003	債券	A	1.5	20	30
期先	2003/1/22	NR400	03030-03	拍賣	FV003	債券	A	1.6	20	32
期先	2003/1/22	HL098	01062-03	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01062-03	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01062-02	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01062-01	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-05	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-04	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01062-02	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01062-02	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01062-01	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-05	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-04	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-03	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	NR400	03030-01	拍賣	FV003	債券	A	1.7	20	34
期先	2003/1/22	HL098	01061-01	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	NR400	03036-01	拍賣	FV003	債券	A	1.2	21	252
期先	2003/1/22	HL098	01062-02	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01062-01	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-05	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-04	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-03	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-02	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	HL098	01061-01	拍賣	FD423	小票前日誌	A	42	25	1050
期先	2003/1/22	NR400	03041-02	拍賣	FV003	債券	A	1.0	22	220
期先	2003/1/22	HL098	01061-02	拍賣	FD423	小票前日誌	A	42	25	1050
期先	總計	總計	總計					89		4246.36

圖 5-17 小行情資料庫查詢結果



## 5.4 花卉交易資訊處理系統操作說明—承銷人

承銷人第一次使用此軟體，必須先完成訂閱端 Access 資料庫與 ODBC 設定，完成 Access 資料庫與 ODBC 設定才能儲存拍賣花卉交易資料。在使用者資料庫設定完成後，以下七小節依序為該軟體操作步驟，由參數設定(第 5.4.1 節)>啟動接收(5.4.2)>停止接收(5.4.3)>轉入資料庫(5.4.4)>儲存 XML 文件(5.4.5)>檢視 XML 文件內容(5.4.6)>加值應用功能(5.4.7)：

### 5.4.1 參數設定

在選單上選擇『設定(Alt+F)』，然後選取『參數設定(Ctrl+U)』，或是在功能列上直接按下



，在出現的畫面如圖 5-18 輸入訂閱者帳號、身份證字號及訂閱者密碼，完成之後，接著選擇訂閱主題，而承銷人可以選擇的訂閱的主題分別有『大行情』和『結帳清單』等兩項，完成所有的參數設定之後，最後按下『完成參數設定』按鈕即可完成參數設定。『重設參數』可以清除之前所輸入的訂閱者帳號、身份證字號、訂閱者密碼。

圖 5-18 承銷人參數設定畫面

### 5.4.2 啟動接收

在選單上選擇『設定(Alt+F)』，然後選取『啟動接收(Ctrl+C)』，或是在功能列上直接按下



，便可以根據之前所設定的參數與系統連線，假如承銷人身份驗證無錯誤，便可以從伺服器上開始接收花卉訊息資料，要是承銷人身份有錯誤或是伺服器目前有問題，可以直接從下面狀態列得知問題發生之處，在圖 5-19 則是顯示連接伺服器電腦成功時的畫面，而圖 5-20 則是顯示接收到承銷人結帳清單的花卉拍賣資料畫面，在圖 5-20 中出現「結束資料」則是代表接收承銷人結帳清單資料結束，可以停止與伺服器之間連線，。

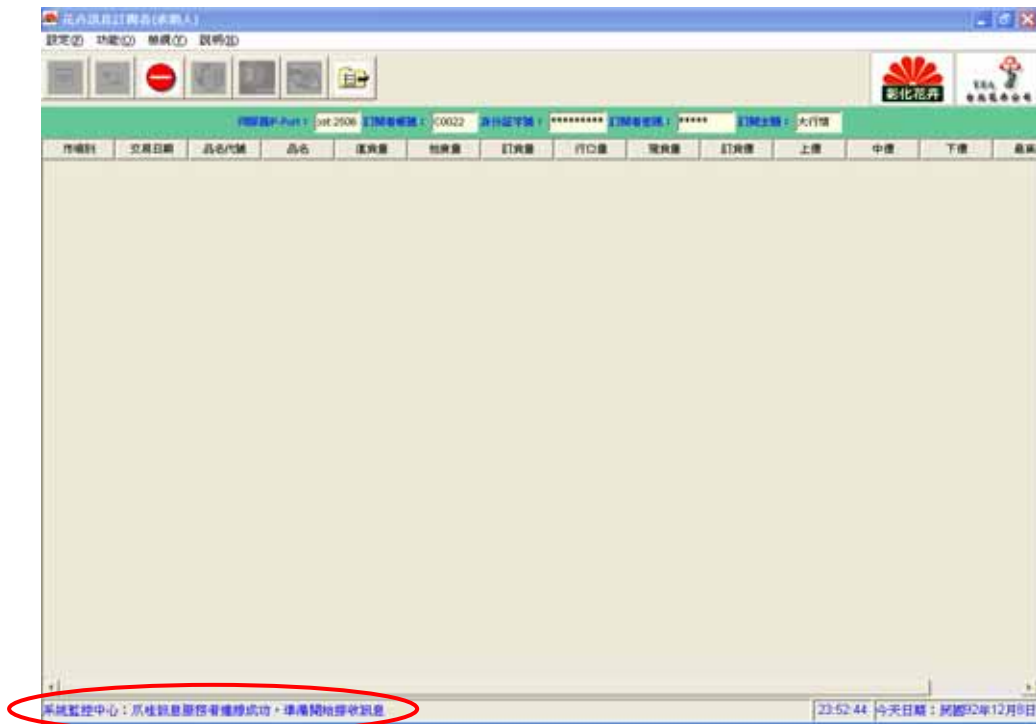


圖 5-19 與伺服器電腦連線成功畫面

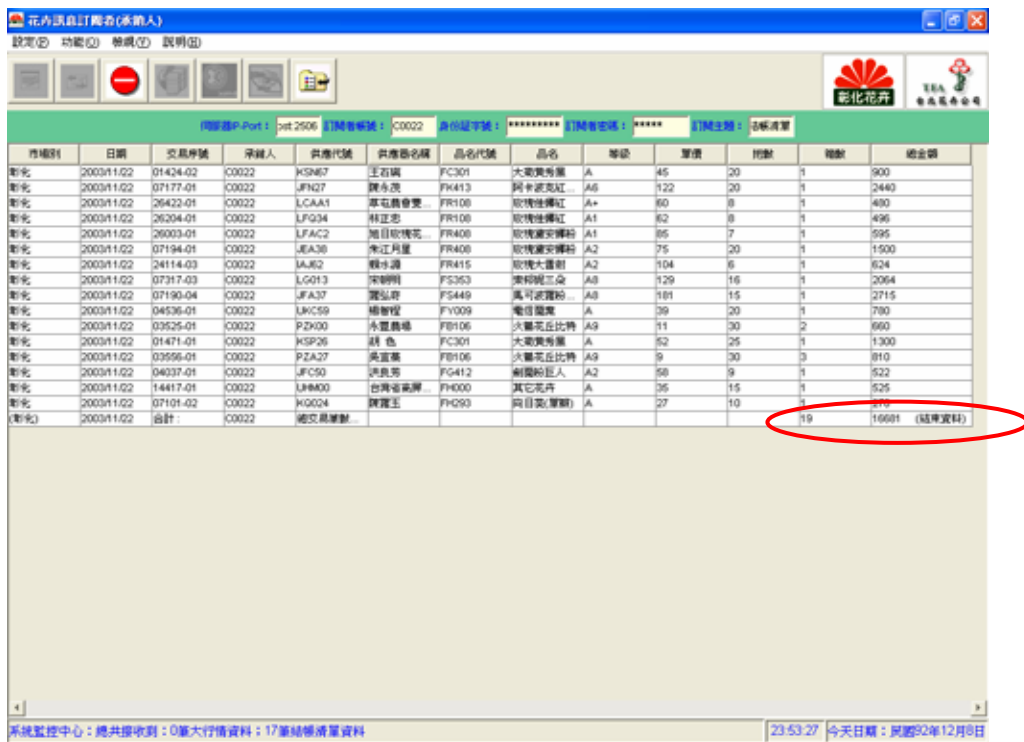


圖 5-20 承銷人結帳清單的花卉拍賣資料接收畫面


### 5.4.3 停止接收

當在接收的過程中，接收畫面出現「結束資料」如圖 5-20 所示，便可以停止接收花卉資料，在選單上選擇『設定(Alt+F)』，然後選取『停止接收(Ctrl+D)』，或是在功能列上直接按下



，便可以停止與伺服器之間的連線接收資料過程。

#### 5.4.4 轉入資料庫

對於所於接收回來的花卉資料，須先轉入承銷人的資料庫系統，在轉入資料庫之前，必須先完成訂閱端 Access 資料庫與 ODBC 設定(如前面所述)，在選單上選擇『功能(Alt+O)』，然後選取『轉入資料庫(Ctrl+T)』，或是在功能列上直接按下 ，就可以將所接收到的花卉資料轉入資料庫內，在狀態列則會出現轉進資料庫成功畫面，假如轉進資料庫失敗，狀態列會出現轉進資料庫失敗提示訊息，轉進資料庫成功的畫面如 5-21 所示。

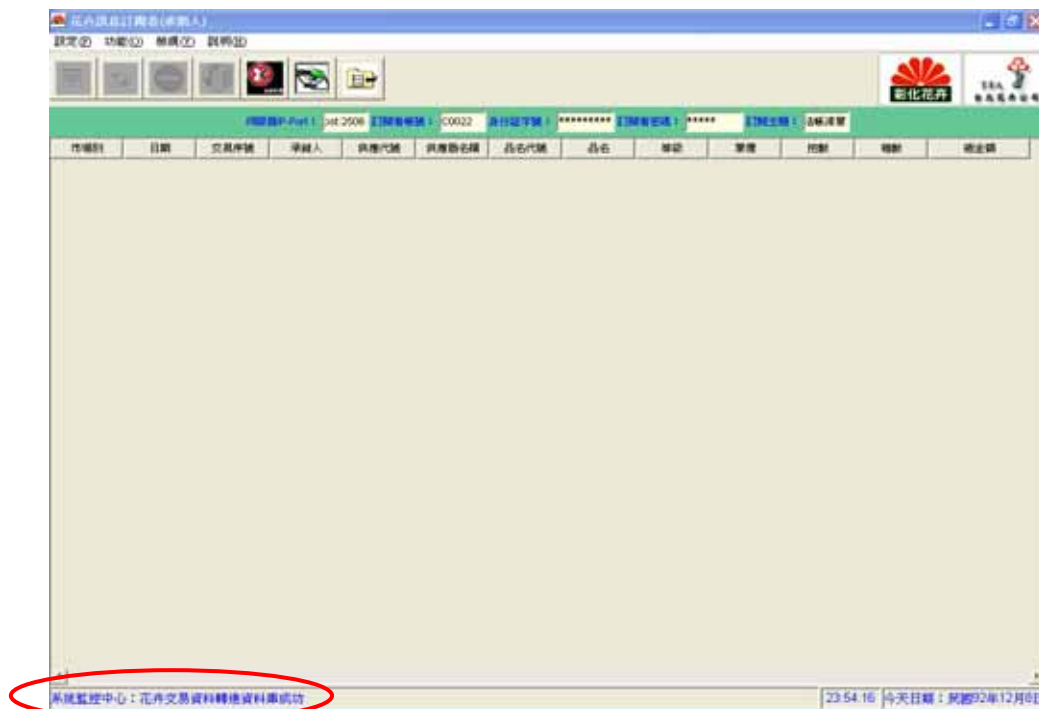



圖 5-21 轉進資料庫成功畫面

#### 5.4.5 儲存 XML 文件

當完成轉入資料庫動作之後，才能儲存 XML 文件，在選單上選擇『功能(Alt+O)』，然後選取『轉入資料庫(Ctrl+S)』，或是在功能列上直接按下 ，就可以完成儲存 XML 文件的過程。

#### 5.4.6 檢視 XML 文件內容

當只要完成接收花卉訊息資料之後，在選單上選擇『檢視(Alt+V)』，然後選取『XML 文件內容(Ctrl+S)』，或是在功能列上直接按下 ，圖 5-22 則是顯示出檢視 XML 文件的內容。





圖 5-22 檢視 XML 文件畫面

#### 5.4.7 加值應用功能

目前花卉交易資訊處理系統所提供的功能有兩種，一種是根據花卉品名種類的把數與件數的加總總和，另一種為花卉品名單價平均值、變異數、標準差分析。

在選單上選擇『功能(Alt+O)』，然後再選取『加值應用(Ctrl+A)』，則出現如圖 5-23 的加值應用功能畫面，選取「結帳清單把數、件數總和」功能，設定查詢資料的時間範圍，按下『開始查詢』按鈕，則可以查詢出某段時間內花卉品名種類的把數與件數總和，圖 5-24 是查詢結果以表格顯示，圖 5-25 則是查詢結果以長條圖方式顯示，

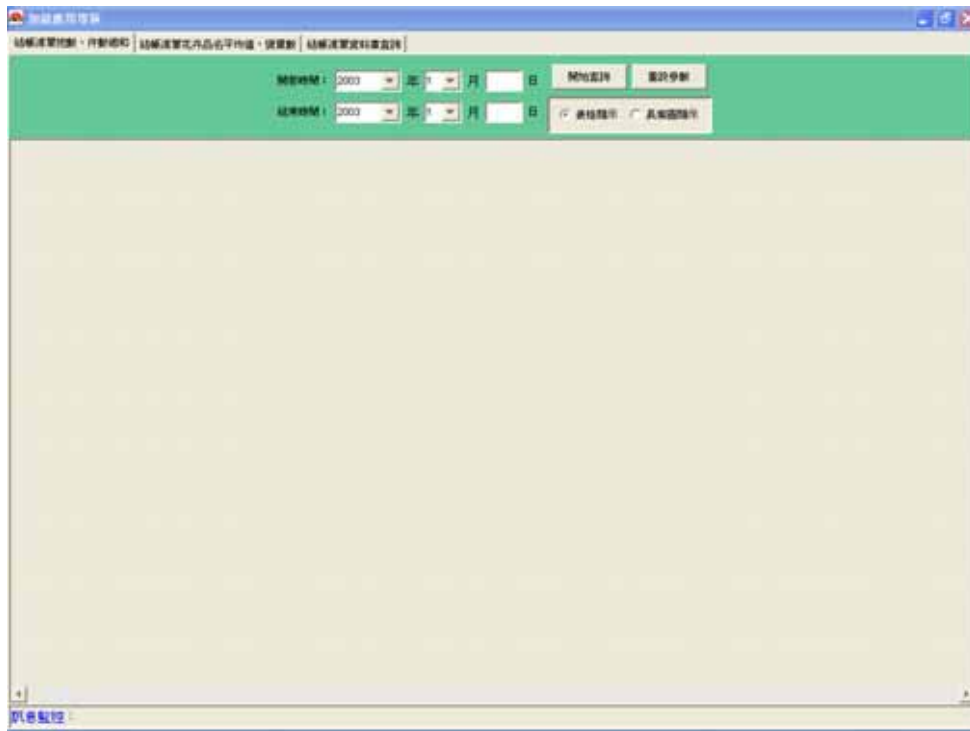


圖 5-23 加值功能應用畫面

品名代號	品名	數量	件數
FB106	次藥花丘批發	200	21
FO014	精鹽	220	2
FC301	大藥膏藥	135	6
FR408	玫瑰蓮花糖粉	113	6
FB407	次藥花學口	100	6
FR195	玫瑰(大船百合)	92	6
FO331	大文心糖	86	4
FE400	神油高粉	64	2
FV009	藥油龍精	60	3
FB408	次藥花珍佳人	60	2
FR413	神手冰花紅三朵	60	3
FA084	東方藥中藥藥紅	60	2
FE172	香水芭蕉紅藥染	50	2
FO440	次藥神冠花	50	2
FR108	玫瑰蓮花糖	48	6
FE353	神神紅三朵	48	3
FO000	其它花卉	45	3
FE449	風可波羅紅三朵	45	3
FE179	香水芭蕉紅藥染	40	2
FV220	藥牡丹	40	2
FE353	香水神神紅三朵	32	2
FL453	神神佛藥神	30	2
FO093	向日(藥神)	30	3
FE363	香水神神紅三朵	28	2
FO412	神神紅佳人	27	3
FL029	神神佛藥神色藥?	26	2
FO333	香水文心糖	20	4
FR415	玫瑰大蓮花	18	3
FR112	玫瑰紅衣武士	18	2
FO104	神神22罐	16	4
FR412	玫瑰新香糖粉	12	2

圖 5-24 查詢結果以表格顯示

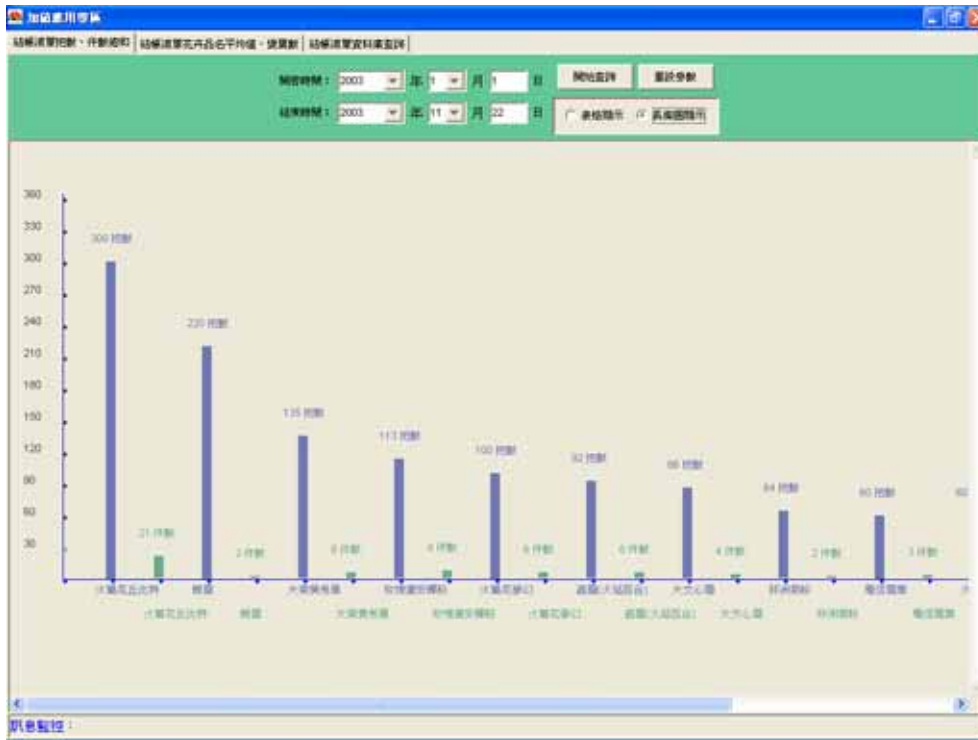


圖 5-25 查詢結果以長條圖顯示結果

在圖 5-23 選擇「結帳清單花卉品名平均值、變異數」選項，也是設定查詢資料時間範圍，設定好查詢時間範圍後按下『開始查詢』按鈕，圖 5-26 則是結帳清單花卉品名平均值、變異數查詢結果以表格方式顯示出來。

名次	品名代號	品名	總筆數	單筆最大值	單筆最小值	單筆筆數	單筆平均值	單筆標準差	單筆變異數
第1名	FR430	玫瑰葉文輝粉	8973	95	75	8	81	4	24
第2名	FS449	萬可波羅粉多朵	8145	181	181	3	181	0	0
第3名	FR413	阿卡波克紅三角	7300	122	122	3	122	0	0
第4名	FC301	大紫美秀麗	8600	52	45	8	48	3	14
第5名	FS363	奧利維三角	8192	129	129	3	129	0	0
第6名	FR106	大紫花庄比時	5940	11	7	10	9	1	2
第7名	FS363	香水奧利維三角	5068	181	181	2	181	0	0
第8名	FS172	香水奧利維三角	5050	101	101	2	101	0	0
第9名	FR185	高麗(大紅百合)	4870	55	45	8	50	4	20
第10名	FS353	香水奧利維三角	4864	152	152	2	152	0	0
第11名	FO331	大文心蘭	3652	82	32	4	47	17	300
第12名	FO014	糖龍	3080	14	14	2	14	0	0
第13名	FR108	玫瑰性輝紅	2528	82	60	8	81	1	1
第14名	FR453	洋桔梗雙瓣綠	2430	81	81	2	81	0	0
第15名	FO333	香水文心蘭	2400	80	80	2	80	0	0
第16名	FY009	雙喜龍索	2340	39	39	3	39	0	0
第17名	FU271	洋桔梗單瓣白瓣?	2132	82	2	2	82	0	0
第18名	FS171	香水奧利維三角	2040	81	81	2	81	0	0
第19名	FR040	大紅錦紅花	2000	40	40	2	40	0	0
第20名	FR415	玫瑰大蕾劇	1872	104	104	3	104	0	0
第21名	FV230	紫牡丹	1720	43	43	2	43	0	0
第22名	FR400	非洲紫羅蘭	1664	26	26	2	26	0	0
第23名	FG104	紅龍223號	1632	51	51	2	51	0	0
第24名	FR400	其它花卉	1575	35	35	3	35	0	0
第25名	FG412	紅龍超巨人	1568	58	58	3	58	0	0
第26名	FR412	玫瑰新香羅粉	1440	120	120	2	120	0	0
第27名	FR407	大紫花學以	1420	14	9	4	10	1	1
第28名	FR112	玫瑰紅武士	1296	72	72	2	72	0	0
第29名	FA084	東方學中賽博冠	1260	21	21	2	21	0	0
第30名	FR083	向日葵(單瓣)	810	27	27	3	27	0	0
第31名	FR408	大紫花科德人	300	5	5	2	5	0	0

圖 5-26 結帳清單花卉品名平均值、變異數查詢結果

在圖 5-23 選擇「結帳清單資料庫查詢」選項，也是設定查詢資料時間範圍，設定好時間範圍、市場別後按下『開始查詢』按鈕，圖 5-27 則是結帳清單舊資料查詢結果以表格方式顯示出來，並加以小計值。使用者若要查詢某當日交易資料，即輸入時間範圍為該日即可。

作場	日期	交易序號	承辦人	供應商代號	供應商名稱	品名代號	品名	等級	單位	數量	控制	總數	總金額
彰化	2003/11/22	04536-01	C0022	LMC58	楊智強	F7009	龍作龍葉	A	39	20	?	780	
彰化	2003/11/22	07190-04	C0022	JFA37	羅弘佑	F5448	萬可波羅粉	A8	181	15	?	2715	
彰化	2003/11/22	07317-03	C0022	L0013	李朝興	F5253	佛利龍三心	A8	129	16	?	2064	
彰化	2003/11/22	04114-03	C0022	JA62	賴小漢	FR415	玫瑰大薯粉	A2	104	8	?	824	
彰化	2003/11/22	07194-01	C0022	JA38	許江丹蓮	FR408	玫瑰蓮安補粉	A2	75	20	?	1500	
彰化	2003/11/22	26003-01	C0022	JFAC2	趙日欣樓花	FR408	玫瑰蓮安補粉	A1	85	7	?	595	
彰化	2003/11/22	26204-01	C0022	JFG34	林正忠	FR108	玫瑰樓樓花	A1	82	8	?	496	
彰化	2003/11/22	26422-01	C0022	LCAA1	廖文貴會豐	FR108	玫瑰樓樓花	A+	80	8	?	480	
彰化	2003/11/22	14417-01	C0022	JH400	台灣省武庫	PH000	其它花卉	A	35	15	?	525	
彰化	2003/11/22	26003-01	C0022	JFAC2	趙日欣樓花	FR408	玫瑰蓮安補粉	A1	85	7	?	595	
彰化	2003/11/22	07101-02	C0022	KG024	陳麗玉	PH023	向日葵(單瓣)	A	27	10	?	270	
彰化	2003/11/22	14417-01	C0022	JH400	台灣省武庫	PH000	其它花卉	A	35	15	?	525	
彰化	2003/11/22	04037-01	C0022	JFC50	洪良芳	FG412	刺繡紅巨人	A2	58	9	?	522	
彰化	2003/11/22	03856-01	C0022	FZA27	吳宜漢	FR106	大蘭花五比精	A9	9	30	?	810	
彰化	2003/11/22	01471-01	C0022	KSP26	顏 色	FC301	大蘭黃秀星	A	52	25	?	1300	
彰化	2003/11/22	03825-01	C0022	FZK00	林豐盛樓	FR106	大蘭花五比精	A9	11	30	?	860	
彰化	2003/11/22	04536-01	C0022	LMC58	楊智強	F7009	龍作龍葉	A	39	20	?	780	
彰化	2003/11/22	07190-04	C0022	JFA37	羅弘佑	F5448	萬可波羅粉	A8	181	15	?	2715	
彰化	2003/11/22	07317-03	C0022	L0013	李朝興	F5253	佛利龍三心	A8	129	16	?	2064	
彰化	2003/11/22	07317-03	C0022	L0013	李朝興	F5253	佛利龍三心	A8	129	16	?	2064	
彰化	2003/11/22	07194-01	C0022	JA38	許江丹蓮	FR408	玫瑰蓮安補粉	A2	75	20	?	1500	
彰化	2003/11/22	01424-02	C0022	KP467	王石園	FC301	大蘭黃秀星	A	45	20	?	900	
彰化	2003/11/22	26204-01	C0022	JFG34	林正忠	FR108	玫瑰樓樓花	A1	82	8	?	496	
彰化	2003/11/22	26422-01	C0022	LCAA1	廖文貴會豐	FR108	玫瑰樓樓花	A+	80	8	?	480	
彰化	2003/11/22	07177-01	C0022	JFN27	陳永洪	PK413	阿卡波克紅	A8	122	20	?	2440	
彰化	2003/11/22	01424-02	C0022	KP467	王石園	FC301	大蘭黃秀星	A	45	20	?	900	
彰化	2003/11/22	07101-02	C0022	KG024	陳麗玉	PH023	向日葵(單瓣)	A	27	10	?	270	
彰化	2003/11/22	14417-01	C0022	JH400	台灣省武庫	PH000	其它花卉	A	35	15	?	525	
彰化	2003/11/22	04037-01	C0022	JFC50	洪良芳	FG412	刺繡紅巨人	A2	58	9	?	522	
彰化	2003/11/22	03856-01	C0022	FZA27	吳宜漢	FR106	大蘭花五比精	A9	9	30	?	810	
彰化	2003/11/22	01471-01	C0022	KSP26	顏 色	FC301	大蘭黃秀星	A	52	25	?	1300	
彰化	2003/11/22	03825-01	C0022	FZK00	林豐盛樓	FR106	大蘭花五比精	A9	11	30	?	860	
彰化	2003/11/22	04114-03	C0022	JA62	賴小漢	FR415	玫瑰大薯粉	A2	104	8	?	824	
彰化	總交易金額	48							57			50043	

圖 5-27 結帳清單資料庫查詢結果

## 第六章 檔案轉移服務系統設計

本章在說明檔案轉移服務系統設計的理念與系統的特性，並以兩個已實作成功的案例來解釋其自動化運作的方式。第 6.1 節「檔案轉移服務模式探討」在探討數種常見的檔案傳送方式與檔案轉移協定的原理，第 6.2 節「Java 語言與檔案轉移服務系統」在以說明 Java 技術建置自動化轉移服務的方式與方法，第 6.3 節「自動化檔案轉移服務系統應用」在探討該檔案轉移服務的應用領域與其操作說明。

### 6.1 檔案轉移服務模式探討

對於網際網路的使用者而言，企業在交換檔案資訊時，最常見的傳送模式除了 Http 網頁的直接下載與 E-mail 的郵寄往來外，另一種耳熟能詳的技術就是檔案轉移協定(File Transfer Protocol, FTP)，透過 FTP 所提供對遠端伺服器的上傳與下載指令，可以方便快速的將異地的檔案分享與交流。在網頁上的操作，許多的使用者以利用網路的資料夾虛擬自己的磁碟機一般，將檔案上傳以提供各地檔案需求者的開放介面，而 E-mail 則是以夾帶檔案的方式作為小型檔案的傳遞。在上述成熟的技術中，在 Http 協定上所整合功能的便利遠大於 FTP 建制指令的複雜，但在簡易的 Client/Server 架構中，除了 Client 端繁複的流程操作，略顯出 Http 協定在連結操作上的繁瑣，並在機密性的文件上還需建立使用者認證的機制，而 E-mail 附加檔案的方式則侷限了上傳與下載檔案的大小[7]。種種條件下，採以 FTP 作為檔案轉移服務的模式建制，本節並更進一步以 FTP 作深入的探討。第 6.1.1 節說明 FTP 的基本架構，第 6.1.2 節說明 FTP 在開發程式時的 Server 與 Client 端架構與轉移流程。

#### 6.1.1 FTP 架構

對於網際網路的使用者而言，主機間最常見的點對點方式來傳遞小型檔案和資料是圖 6-1 的主從式(Client/Server)架構；從端(Server)聆聽來自所有主端(Client)的處理請求，再回報以所求的資訊。但對大型檔案的上傳與下載而言，使用者常以檔案轉移協定來進行。通常主從式架構會在主端與從端間建立一個包含交易連線與資料傳遞的插座(Socket)連結，但檔案轉移協定則使用兩條 TCP 連線。這是在主端與從端間分別進行通訊與資料傳遞的處理；其中一條用來建立持續的通訊監聽，而另一條連結則作為主端傳送指令與從端回應結果之用。在檔案轉移協定規格制定時，保留了 TCP 埠號(Port) 21 作為檔案轉移協定的管道。在主端與從端建立連線時，埠號 21 的通訊連結則一直保持連線並持續的進行控制指令的傳輸直到主端應用程式下達 QUIT 指令為止。在兩端在進行資料轉移時會臨時建立資料轉移的通道，此時便由主端先開啟另一資料轉移埠(預設埠號 20)並傾聽從端傳輸回應。此資料連線在轉移結束後便中斷，直至下一轉移指令下達時方再度建立連結[14, 30]。



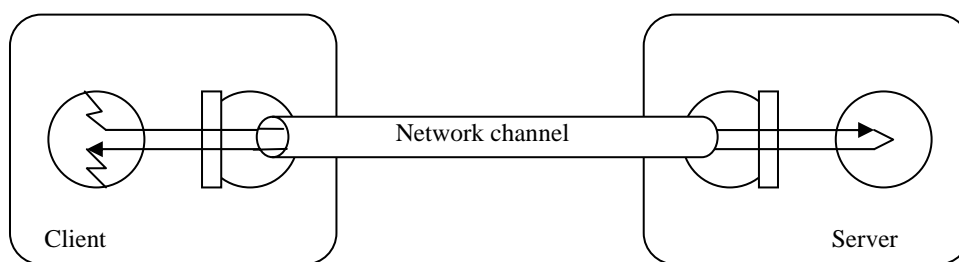


圖 6-1 主從式架構

檔案轉移協定的運作是由協定解譯器(Protocol Interpreter, PI)與資料轉移協定(Data Transfer Protocol, DTP)兩模組所構成如圖 6-2 所示。使用者透過使用者界面(User Interface, UI)命令列或圖形化介面進行檔案處理機制時，協定解譯器如同一般應用程式下達操作指令，會先判斷此指令在針對本地端或遠端伺服器的操作。若為本地端的操作，則會直接執行並將處理結果回報。若為遠端伺服器的連結操作，則會先轉譯成檔案轉移協定的指令，以通訊連線傳遞指令至從端，並由從端處理轉換後執行。若在該指令集中要求轉移檔案的處理，便交由資料轉移協定模組進行處理。在資料轉移協定模組進行資料轉移前會先識別彼此系統的差異，而後開始採行資料轉換與傳輸的工作。

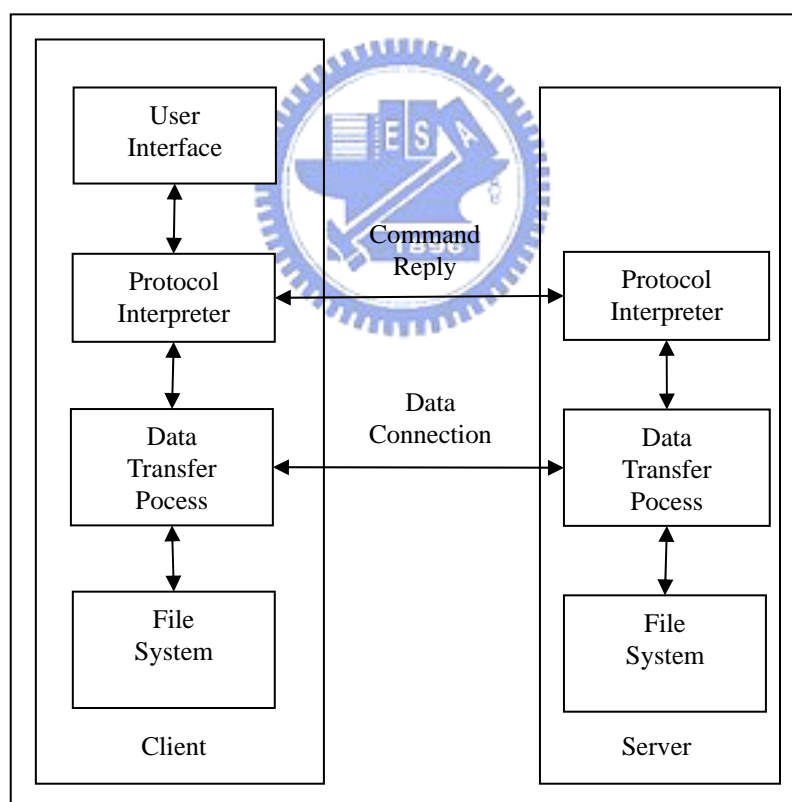


圖 6-2 FTP 通訊協定架構[14]

### 6.1.2 FTP 程式開發架構

透過 Internet 傳輸資料的方式中，Socket 的觀念讓網路連線可視為另一種資料流，在建立 Socket 連線後，將寫入 socket 的資料傳送至遠端，簡易的 Socket 技術即廣泛運用到網路的層面上。

在檔案轉移協定系統內，ServerSocket 類別為伺服器端所建立的 Socket，伺服器以 ServerSocket() 建構式在主機內所提供的特定通訊埠，建立一個 ServerSocket 物件，並以 ServerSocket 的 accept() 方法傾聽來自主端的連線，在主端達成與 ServerSocket 的連線後，才會建立聯繫兩端的 Socket 物件，並透過兩端 Socket 物件所提供的輸出入資料流，取得伺服器與主端之間的 I/O 溝通管道，即前小節所提及的 PI 模組，此 PI 模組內的兩端 Socket 物件(或通訊連結)，持續作用至主或從端提出 close() 方法即結束，其作用在於負責檔案轉移協定中連線的控制。在 FTP 伺服器開啟一個埠後，主端的 PI 向伺服器的預設埠進行連線動作，在連線正確完成時並依伺服器端的要求進行確認登入(login)的動作，主端順序輸入正確的 ID 與認證密碼後，通訊的連結才算建立成功，而後的指令則需經由此通訊連線得以轉移。

在通訊連線的狀態下，通訊的連結首先為從端開啟其預定的埠以聆聽來自主端的連線請求，但在資料轉移的連結中(資料轉移協定模組)，則為主端先開啟預定檔案轉移的埠，從端經接收到來自於主端的檔案處理指令，如 LIST(詳細列檔)、NLST(列檔)、RETR(接收)、與 STOR(上傳)等時，主端首先會以 PORT 指令傳送予從端目前所使用的正確位址與埠號；再告知從端需透過該 IP 位址與埠號，方可建立與主端資料轉移連結的資料轉移插座(Data Transfer Socket)如圖 6-3 所示，即前節的資料轉移協定模組。其中協定解譯器埠號若為 N，則資料轉移協定埠號則預設為(N-1)。主機間檔案的轉移透過資料轉移協定模組的插座與插座所提供的 I/O 資料流進行檔案的傳遞處理。在檔案轉移過程中，若遇到主端發出中斷的指令、連線埠號的異動、通訊連結的中斷或檔案結尾的提示(EOF)情況下，則資料轉移的連結將被終止。

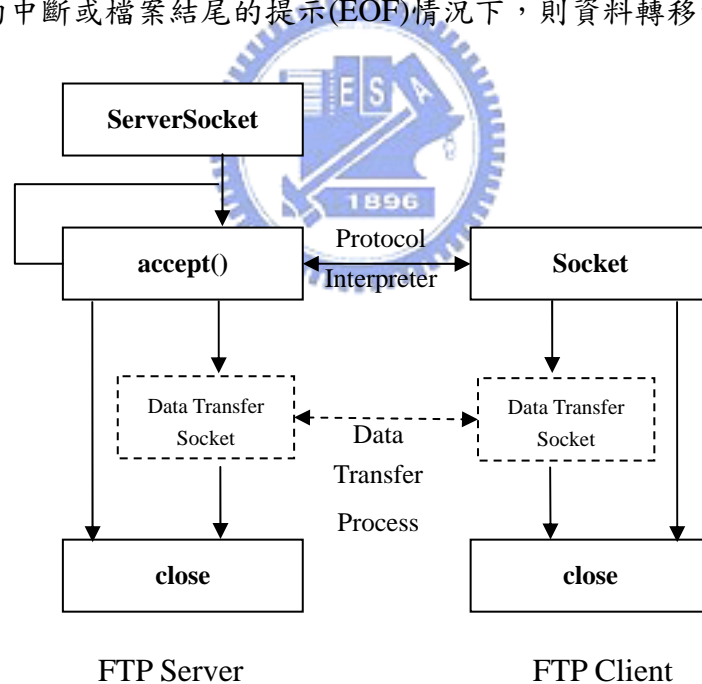


圖 6-3 FTP 程式開發架構

在上述以 FTP 為主體的架構下，FTP 的檔案轉移流程可簡述如圖 6-4。

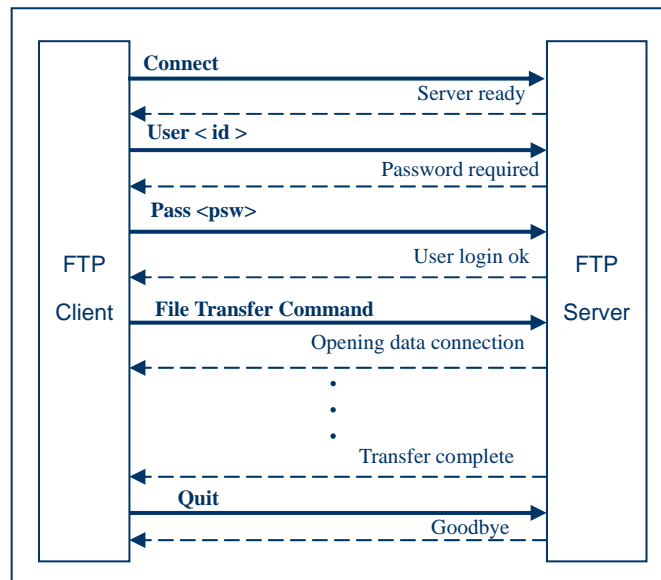


圖 6-4 FTP 檔案轉移流程[14]

上述的檔案轉移協定可簡述如圖 6-4 的流程，其中主端的轉移步驟為(1)與從端建立連線，(2)輸入使用者帳號指令，(3)傳送密碼驗證指令，(4)下達檔案轉移指令，與(5)從端中斷連線，而伺服器端也相對的給予主端適當回應。其中在步驟(4)在下達檔案轉移的請求指令時，主端會先透過通訊連結以 PORT 指令要求與從端建立資料轉移連結。



## 6.2 Java 語言與檔案轉移服務系統

網際網路的普及，使得網路用戶端在使用網路系統時，必須透過相同的協定才能存取相對應的資源，而程式語言的多樣化卻成為許多網路資源的限制。近年來 Java 技術的成熟已經普遍運用到網際網路上，Java 語言為一種多用途的物件導向語言，除了在動態網頁豐富的能力與套裝的應用軟體外，其內含的類別函式庫更支援多樣層次架構的網路溝通套件，透過其深入網路程式設計弱點的考量，提供了跨平台作業(Platform Independence)、安全性(Security)、與國際字元集(International Chracter Set)等特性，解決了許多網路的問題。

結合以上特性，網路的用戶端不單可安全的下載及執行程式，更可充分的利用網路上的資源，這些都是其他語言能力有所不足的地方，此外，Java 物件導向的觀念更使得網際網路的設計更為簡便，其中與網路協定相關的完整程式碼，甚至只佔完整程式碼最簡單最短的部分[7]。本小節即說明檔案轉移協定網路程式的部分。第 6.2.1 節說明以 Java 語言建置 FTP 的物件與方法，第 6.2.1 節「自動化處理機制」在說明自動化 FTP 的兩種自動化機制，第 6.2.3 節「Java 檔案轉移協定系統程式開發」，在以示範 Java 語言建置檔案轉移協定系統。

### 6.2.1 Java 檔案轉移協定 API 說明

在 JDK(Java Development Kit)所開放的 API 套件裡，除了簡易的 Socket 程式撰寫外，Java 也支援許多的網路通訊協定，FTP(File Transfer Protocol)、NNTP(Network News Transfer Protocol)、SMTP(Simple Mail Trasfer Protocol)、HTTP(Hyper Text Transfer Protocol)…諸如此類，其中在本論文中提及的支援檔案轉移協定的 API 套件為 sun.net 與 sun.net.ftp。

許多 Java 程式的開發工具皆附以其相關 API 套件的檔案，如 java.awt、java.io、java.net、javax.swing…等在 JDK 安裝目錄下的 rt.jar 檔中皆有其輔助說明文件，但 java 仍有一些未附說明的套件(Undocumented Package)，如支援檔案轉移協定的 sun.net 與 sun.net.ftp 套件，其中支援 FTP Client 端的物件類別為 sun.net.ftp.FtpClient，而檔案輸出入資料流則有 sun.net.TelnetInputStream 與 sun.net.TelnetOutputStream。以下則分別介紹與 FTP 相關的類別與方法[14]。

一、FtpClient：

FtpClient 主要為建立 FTP Client 端 Socket 的物件，為 FTP Client 與 Server 端建立連結之用，其主要繼承結構[14]為：

```
java.net.Socket
|
+--sun.net.NetworkClient
|
+--sun.net.TransferProtocolClient
|
+--sun.net.ftp.FtpClient
```

其建構子有三：

表 6-1 FtpClient 建構子

<pre>public FtpClient() public FtpClient(String host) throws IOException public FtpClient(String host,int port) throws IOException</pre>
--

其中參數代表意義：

1. host：代表遠端 Server 主機的 IP 位址，以字串型態表示。
2. port：代表 FTP Server 端協定使用的通訊埠。部分伺服器管理者為防止有陌生連線試圖闖入破壞主機系統，除了預設埠號為 21 外，伺服器管理者可主動更改此通訊連線埠號。

在 Client 端建立起 FtpClient 連線物件後，可以下列 FtpClient 物件所提供的方法進行與 Server 端的檔案協定處理：

表 6-2 FtpClient 物件方法

功能	方法
登錄	public void login(String id,String psw) throws IOException
列出遠端檔案明細	public TelnetInputStream list() throws IOException
更改目錄	public void cd(String path) throws IOException
Ascii 編碼型態	public void ascii() throws IOException
Binary 編碼型態	public void binary() throws IOException
上傳檔案	public TelnetOutputStream put(String filename) throws IOException
下載檔案	public TelnetInputStream get(String filename) throws IOException
中斷連結	public void closeServer() throws IOException

## 二、TelnetInputStream：

TelnetInputStream 主要用途為建立 FTP Client 與 Server 端資料轉移的資料流，為 FTP Client 端提出檔案下載或列檔指令時的回傳型態，其繼承關係[14]為：

```
java.io.FilterInputStream
|
+--sun.net.TelnetInputStream
```

其中並提供三種方式以讀取來自 FTP Server 端回傳的輸入資料流：

表 6-3 TelnetInputStream 物件方法

<pre>public int read() throws IOException public int read(byte b[]) throws IOException public int read(byte b[],int off,int len) throws IOException</pre>
---

在讀取回傳資料流的 int 型態後，經由本地端檔案的輸入資料流，可儲存為本地端的檔案型態。



### 三、TelnetOutputStream：

TelnetOutputStream 用途在建立 FTPClient 與 Server 端資料轉移的資料流，不同於 TelnetInputStream，作用在 FTP Client 端提出上傳檔案指令時，為輸出資料流，其繼承結構為：

```
java.io.FilterOutputStream
|
+--java.io.BufferedOutputStream
|
+---sun.net.TelnetOutputStream
```

TelnetOutputStream 提供兩種主要方式以提供 FTP Client 端將檔案寫入輸出資料流：

表 6-4 TelnetOutputStream 物件方法

<pre>public void write(int b) throws IOException public void write(byte b[],int off,int len) throws IOException</pre>
---

在本地端與 FTP Server 端建立起 TelnetOutputStream 後，Client 可將本地端的檔案資料流寫入至 TelnetOutputStream，以上傳檔案至 Server 端主機。

## 6.2.2 自動化處理機制

檔案文件透過電子化的傳輸，大幅提高了企業在作業流程中的效率，而企業除了在電子化層層技術的改良，透過自動化的設計更使得一貫的作業流程可免去重複性的操作，在時間與人因的考量上，頻繁的檔案傳送作業勢必趨向於自動化的處理機制，於是自動化加入了本研究檔案轉移服務的必要機制，透過 Java 語言的執行緒(Thread)觀念與檔案系統(File System)所提供的方法，實作該 Client 端自動化檔案轉移系統，以下建立兩種自動化機制：

### 一、即時上傳：

即時上傳在偵測本地端檔案是否建立，檔案系統內若在指定的資料夾裡有檔案的產生則自動化將此資料夾內檔案上傳，需透過 Java 檔案系統技術。由於持續性的監控該檔案系統將佔據系統大半的資源，因此輔以執行緒的觀念，以間隔的時間偵測本地端檔案系統，提高系統的效率，此外，由於檔案在上傳完成後，本地端檔案若無做刪除的動作，自動化機制即重複上傳該檔案，形成檔案的重複傳送，為避免此現象的發生，在該檔案正確上傳完成後予以自動刪除。

### 二、定時上傳：

定時上傳即以特定的時間將指定的本地端資料夾內檔案上傳，需透過 Java 多執行緒技術。以多執行緒的方式進行直接的時間比對，其中可選擇刪除本地端檔案與否，但在每日性的傳送檔案處理中，則建議以刪除檔案為原則，以避免檔案的重複傳送。

在自動化檔案轉移服務系統裡，自動化的設計往往需透過企業的行為模式來判斷，於是自動化的進行還須仰賴部分客製化的處理，為使自動化檔案轉移服務系統能跨企業領域重複使用，透過模組化的設計，使用者可透過外部的檔案直接更改連線的設定，使該系統更可充分的運用。[17]

### 6.2.3 Java 檔案轉移協定系統程式開發

對程式開發而言，這包括主端與從端的兩個爪哇程式。由於從端的爪哇程式可免費下載，這裡只說明主端的爪哇程式設計。主端的爪哇程式必須包括「連線」、「選檔」、「上傳」、「斷線」與「結束」五個工作項目；其中「選檔」與「結束」兩個工作項目屬於圖形人機界面(Graphic User Interface, GUI)的設計，而「連線」、「上傳」、「斷線」三個工作項目則和檔案轉移協定有關。這裡將針對檔案轉移協定有關的程式來說明，而圖形人機界面部份則忽略之。「連線」與「斷線」兩個工作項目的爪哇程式可整理如表 6-5 所示並加上註解來說明其設計原因，而「上傳」工作項目的爪哇程式可整理如表 6-7 所示。[17]

表 6-5 「連線」與「斷線」工作項目的主端程式碼

```

package ftpdemo;

import sun.net.ftp.*;
import sun.net.*;
import java.io.*;

public class FTPDemo {
    sun.net.ftp.FtpClient ftp;
    public FTPDemo(){ }
    void connect() throws IOException{
        ftp = new FtpClient("localhost", 21);           // 指定位址與埠號建立「連線」
    }
    void disconnect() throws IOException{
        ftp.closeServer();                             // 與伺服器的「斷線」
    }
    void login() throws IOException{
        ftp.login("userName", "password");             // 帳號與密碼的登錄
    }
    void listDir() throws IOException{
        sun.net.TelnetInputStream tis = ftp.list();
        java.io.InputStreamReader isr = new InputStreamReader(tis);
        int i;
        String fileList = "";
        while(true){
            i = isr.read(); if(i < 0) break; else fileList += (char)i;
        }
        System.out.println(fileList);                 // 列印表三資料
        tis.close(); isr.close();
    }
}

```



表 6-6 遠端列表

drwxr--r--	1 user	group	0 Apr 01 12:16 .
drwxr--r--	1 user	group	0 Feb 15 06:49 FTPServer1
drwxr--r--	1 user	group	0 Feb 15 07:07 FTPServer2

表 6-7 「上傳」工作項目的主端程式碼

```

void changeDir()throws IOException{
    ftp.cd("FTPServer1");
}
// 更改從端的路徑
void upload()throws IOException{
    ftp.binary(); // 「上傳」檔案
    sun.net.TelnetOutputStream tos = ftp.put("clientFile.bmp"); // 設定二進位編碼型態傳輸
    File f = new File("C:/FTPClient/clientFile.bmp"); // 建立遠端檔案資料流
    System.out.println("File size:"+f.length()); // 列印上傳前的檔案大小
    java.io.FileInputStream fis = new FileInputStream(f);
    int i;
    while(true){
        i = fis.read();
        if(i >= 0) tos.write(i); else break;
    }
    tos.close(); fis.close();
}

```

在執行上述主端程式碼與從端建立連線並列出遠端目錄後，遠端列表回傳如表 6-6 所示。該表中顯示從端提供兩個上傳路徑，即 FTPServer1 與 FTPServer2，而其目錄屬性為可選擇更改路徑至該資料夾底層。表 6-6 可看出目錄的屬性，例如第一個字元「d」就代表資料夾。

對「上傳」工作項目而言，這裡以主端檔案 clientFile.bmp 上傳至從端目錄 FTPServer1 為例來說明其爪哇程式設計。表 6-7 中顯示可更改工作目錄至其下層，而程式說明則如註解所示。在上傳檔案後並列出 FTPServer1 資料夾內容明細如表 6-8 所示，並可由其內容比對與原始檔案的大小以確定檔案上傳無誤。

表 6-8 遠端資料夾內容在上傳完成後列表

File size:1329					
drwxr--r--	1 user	group	0 Feb 15 06:49	.	
drwxr--r--	1 user	group	0 Feb 15 06:49	..	
-rwxr--r--	1 user	group	1449 Feb 09 17:49	Bu.txt	
-rwxr--r--	1 user	group	1329 Dec 13 23:51	flower.gif	
-rwxr--r--	1 user	group	17513 Feb 09 17:49	大行情.txt	
-rwxr--r--	1 user	group	53102 Feb 09 17:38	供應人匯款清單.TXT	
drwxr--r--	1 user	group	0 Feb 15 06:53	data1	
drwxr--r--	1 user	group	0 Feb 15 06:53	data2	
-rwxr--r--	1 user	group	1329 Feb 15 13:46	clientFile.bmp	

表 6-9 檔案存在的判斷

```

boolean haveAnyFile(){
    boolean flag = false;
    try{
        File dir = new File("localFile_path");
        String[] files = dir.list();
        if(files.length != 0){ //判定資料夾內是否有檔案
            flag = true;
        }
    } catch(Exception e){ }
    return flag;
}

```

對自動化機制的設計而言，這是建立(1)即時上傳與(2)定時上傳兩種機制。對即時上傳自

動化機制而言，這是隨時偵測主端的檔案是否已建立，其爪哇程式如表 6-9 所示。若已建立，則將這些檔案給予上傳。由於持續性的監控這些檔案是否已建立將耗費大量電腦的資源，因此用爪哇多執行緒(Multi-threading)的技術來實作。執行緒在上傳檔案完成後會進行刪除檔案的動作以避免重複的傳送檔案。

對定時上傳自動化機制而言，這是在特定的時間將主端所選定的檔案上傳，而其設計則是透過多執行緒技術來進行。當多執行緒進行時間的比對時，使用者可選擇與否要刪除已上傳的檔案。通常使用者會選擇刪除已上傳檔案來避免檔案的重複傳送。



### 6.3 自動化檔案轉移服務系統應用

網路多層次的架構，最常見的點對點傳輸透過檔案轉移協定，將檔案有效率的轉移至遠端主機，透過模組化設計的自動化檔案轉移服務系統，可以動態變更上傳主機位址的方式，使此系統可跨領域的重複使用，以下將自動化檔案轉移服務系統目前於花卉產業的研究發展上，依其所建置位置的層面做一說明(第 6.3.1 節)。第 6.3.2 結則為該系統的操作與功能說明。

#### 6.3.1 系統應用領域

知識經濟的當前，業務情報扮演非常重要的角色，電子化技術的成熟促使業務資訊的取得容易且可靠，目前花卉供應鏈的業務情報來源主要為農產運銷資訊網，其中的子系統---花卉批業務情報網(<http://flower.nctu.edu.tw>)，已由一連串的資訊技術整合而成[15]，其中在前端資訊流的部分需透過自動化檔案轉移服務系統，將每日進行批發交易的資訊傳送至交大 FTP 伺服器以供後端的線上分析處理(On Line Analytical Processing, OLAP)之用。如圖 6-5 所示。

花卉供應鏈中的決策者主要有三：花卉供應人、花卉承銷人、與各花卉公司(彰化、台中、與台南花卉公司)；對於花卉的供應人與承銷人而言，每日的花卉拍賣價格與每年每季的花卉價格波動都會影響各供應人與承銷人的決策，經由檔案轉移系統每日的上傳，與倉儲系統每日資料的更新，供應人與承銷人可以最新、最準確的資訊，採取最利於當前的行動，譬如供應人可以選擇將所栽種的花種送往何處的花卉批發公司，以謀取更高的利潤，甚至在了解花卉價格的歷年走勢後可選擇在下一年度應栽種的花類、花種等；而承銷人則可了解何處的花卉批發公司可以最低的購買成本選購所需的花種...等，諸如此類；在各花卉公司的決策者部份，以往各批發市場對於供應人所提供的花卉，採以照單全收的收貨方式，不單造成拍賣後殘貨處理上成本的增加，批發市場的虧損及農產品價格不穩定更使成本無形的浪費。因此，各批發公司的決策者所要面臨的問題，就是如何以即時性的資料決定各供應人的供貨量範圍以及批發市場最佳的到貨量。

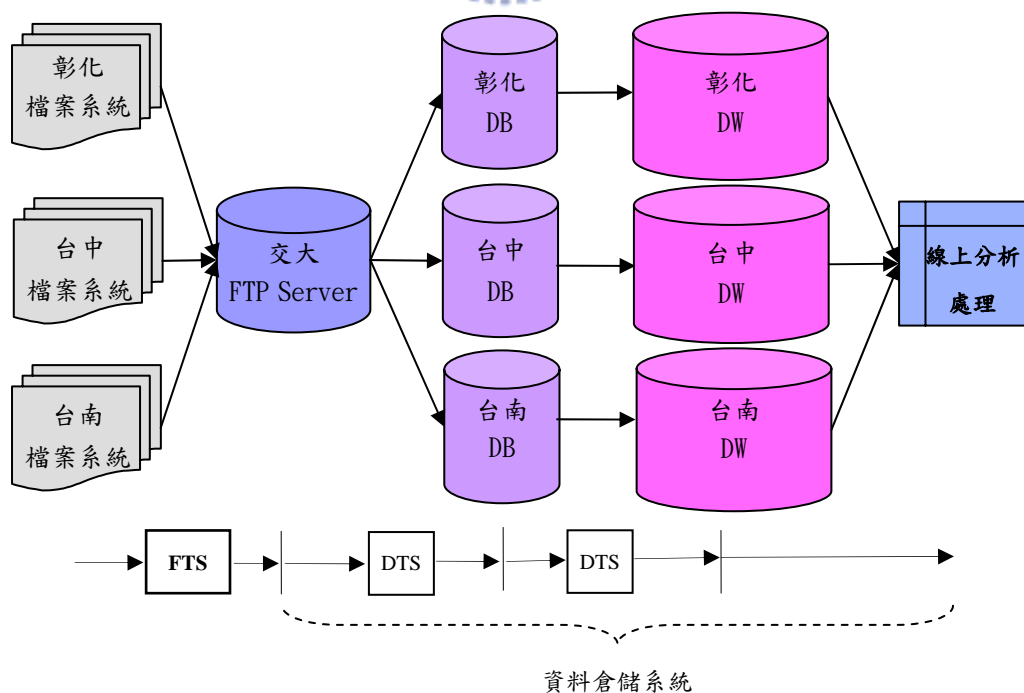


圖 6-5 自動化檔案轉移服務於花卉線上分析處理系統位置



各花卉批發公司以自動化檔案轉移系統，將當日交易檔以每日即時傳送的方式上傳至 FTP 伺服器，而後在資料庫與倉儲處理的部分，可依各倉儲系統資料轉換服務(Data Transform Service, DTS)取出所指定之交易檔作為處理，在倉儲的建置完成後，各決策者可透過網際網路的方式做線上的統計與分析處理，作為即時性或趨勢性的最佳指導方針。

目前此自動化轉移服務系統已於彰化花卉公司成功的運作，而台中與台南花卉公司的倉儲系統在陸續建置的過程中，規劃以該自動化檔案轉移系統上傳每日的花卉交易檔案。另外，在本篇論文曾提及的台北花卉公司，由於該公司的檔案轉移系統目前採以另一套非自動化操作方式的軟體運作，以致檔案操作人員必須頻繁的上傳當日的交易檔案，而後端的倉儲系統亦須等待該日的檔案送達後方可順利的執行資料轉換的工作，為達成花卉產業各條供應鏈資訊流的流暢以使線上分析處理系統更具即時性的資料分析，已擬將本自動化轉移系統取代該套系統。

### 6.3.2 系統功能與操作說明

在此自動化檔案轉移軟體設計上，主要目的是用來將資料上傳至遠端伺服器，並不需伺服器間的檔案相互交流，為保有資料的正確性與安全性，限制了二點功能：

- 一、允許相同檔名覆蓋舊有資料，這是為了檔案上傳失敗時，可以重新上傳檔案。
- 二、資料倉儲伺服器端只允許檔案上傳，不允許檔案的下載及修改，此限制是為了防止其他不明使用者更動與竊取上傳的資料，造成資料倉儲在轉換過程中失敗或資料的外流。



圖 6-6 檔案轉移服務主畫面

本節以彰化花卉目前使用的檔案轉移服務系統為例，其自動化檔案轉移系統的開機畫面與基本操作流程如圖 6-6 所示，使用者在設定好連線位址與帳號資訊後，按下連線按鈕，即可建立與該位址的連線，結果如圖 6-7 所示。其中 IP 位址為下拉式選單，並由外部的.txt 檔案動態匯入 IP 位址，使此軟體更具彈性的改變位址。



圖 6-7 與選定之 IP 位址建立連線畫面

在建立連線後，可由訊息視窗得知最近成功上傳的紀錄，並可進行下一步選擇檔案的動作。在選取檔案後，按下上傳按鈕，結果如圖 6-8 上傳中的狀態所示。訊息視窗上方的 progress bar 顯示目前上傳中的進度，且輔以下方的訊息中心，得知上傳中的檔案名稱。

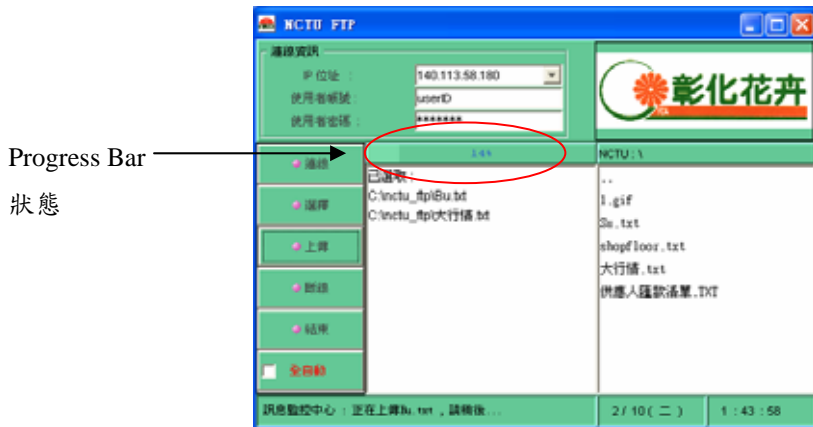


圖 6-8 上傳中的 progress bar 狀態

在上傳動作完成時，訊息視窗即顯示出上傳成功的檔案畫面，同時右方的檔案列表立即更新，將遠端的檔案列出。確認完成後即可按下斷線按鈕，與遠端伺服器結束連線。



圖 6-9 成功上傳畫面

在每日固定的傳送檔案而言，此 FTP 軟體透過自動化的設計，將花卉公司每日的交易資

料檔自動轉移到遠端的資料倉儲系統。按下主畫面『彰化花卉』的圖示後，可進行自動化的設定，如圖 6-10。遠端路徑即為連線至資料倉儲伺服器所開放的目錄下，設定遠端路徑在於加快整理遠端的目錄，使用者可不必重複動作，選取所該上傳的目錄。本地路徑的設定，在於讓使用者確認該自動化機制的來源檔位置，避免誤傳錯誤的資料來源檔。

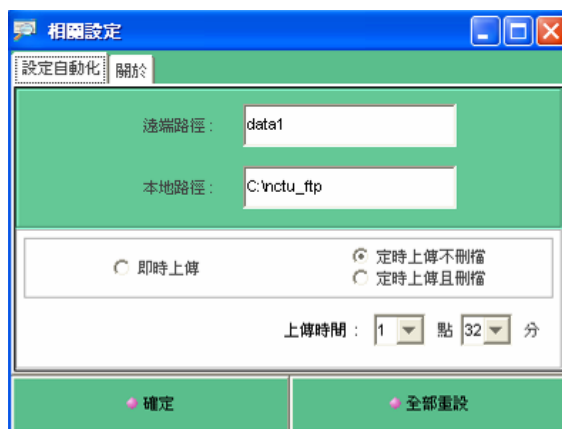


圖 6-10 設定自動化畫面

在設定好當地與遠端路徑後，選擇自動上傳的方式，分為(1)即時上傳與(2)定時上傳兩種自動化處理方式。(1)即時上傳即以固定的時間間隔，偵測本地路徑是否有新的檔案準備上傳，並將檔案直接上傳，為避免自動化機制重複上傳該檔案，在系統確認該檔案上傳成功後即自動刪除檔案；(2)定時上傳又細分為刪檔的與否，使用者在確認了上傳的時間後，系統即以該時間進行自動化的處理。



圖 6-11 確認自動化處理畫面

## 第七章 結論

Internet 技術的蓬勃發展，使企業文件的管理導向於企業的電子化技術，電子化的交易除了節省成本與簡便的資訊取得外，電子資訊系統的在利用性更使客戶端簡易得到新一代知識經濟的利潤獲取。此外企業與企業間的競爭已不再是以製造的角度，而是以企業整體供應鏈管理績效；企業的經營強調即時、彈性、與速度，網際網路上多樣的伺服器架構，本論文即探討了各樣的伺服器模組，以考量企業在導入 Enterprise Resource Planning(ERP)軟體時所需重視的擴充性與維護性，並透過公開的 Java 技術與 XML 成熟的技術使用，使花卉產業內部的資訊系統更具整合性。

XML 的使用除了提供彈性的識別標籤，在異質系統或平台上的整合更為容易，在實作了花卉多條供應鏈的整合系統後，上下游的客戶端更擁有一套完善的資訊服務系統。以目前的花卉交易資訊處理系統中，對表單資訊接收者而言，將彰化與台北花卉公司的各供應商與承銷人表單(如，供應商小行情、承銷人結帳清單、...，與大行情等)，等性質的表單透過整合的機制使各供應商與承銷人的交易資訊得以整理及再利用，透過儲存於使用者資料庫內的資料加以分析其供應(或承銷)花種的價值趨勢，為各供應商或承銷人帶來更可觀的利潤；對資訊發送端而言，各花卉公司透過該交易資訊處理系統，得以節省大量不必要的表單寄送成本與表單寄送的人力資源。花卉產業的供應鏈資訊流屬於大型的 Business to Customer(B2C)架構，除了著重伺服器的穩定性外，考量到了軟體的應用架構以提升其整體供應鏈資訊流規模的擴充性，唯有完整的技術與組織結構方能使資訊整合系統發揮至極致。

在花卉供應鏈資訊流的處理上，透過檔案轉移服務的自動化機制使其資訊流更為流暢，繁瑣的交易檔案上傳行為，不單造成檔案操作人員的不便，對於即時性的線上分析處理系統更為即時性的瓶頸點所在，以自動化檔案轉移服務的實作，每日交易資料的即時上傳減少了交易資訊上傳人員的工作負擔，線上分析處理系統作業流程的即時性更加快了企業反應的速度，以提高企業競爭力。此外，由於漁業署漁船軌跡資料倉儲系統與花卉公司倉儲系統檔案轉移的性質極為相似，並透過此檔案轉移系統模組化的特性，可將此自動化系統建置於該漁船軌跡倉儲系統的前端(如圖 7-1 所示)，其中與漁船軌跡資料倉儲相關的系統有「漁業資訊分享熱線」(Fishery Information Sharing Hotline, FISH)與「漁船資訊傳送系統」(Vessel Information Propagating System, VIPS)兩大系統。FISH 系統以網頁的方式將倉儲中的資料予以整合並呈現，而 VIPS 系統則是將 FISH 裡資料倉儲的漁船位址資訊以 Java 視覺化的技術呈現於電子海圖中。模組化的特色不單使系統實現了跨領域的操作，在每日自動化的處理機制中，花卉與漁業產業更可以最快速、簡便方式得到的最新的資訊。

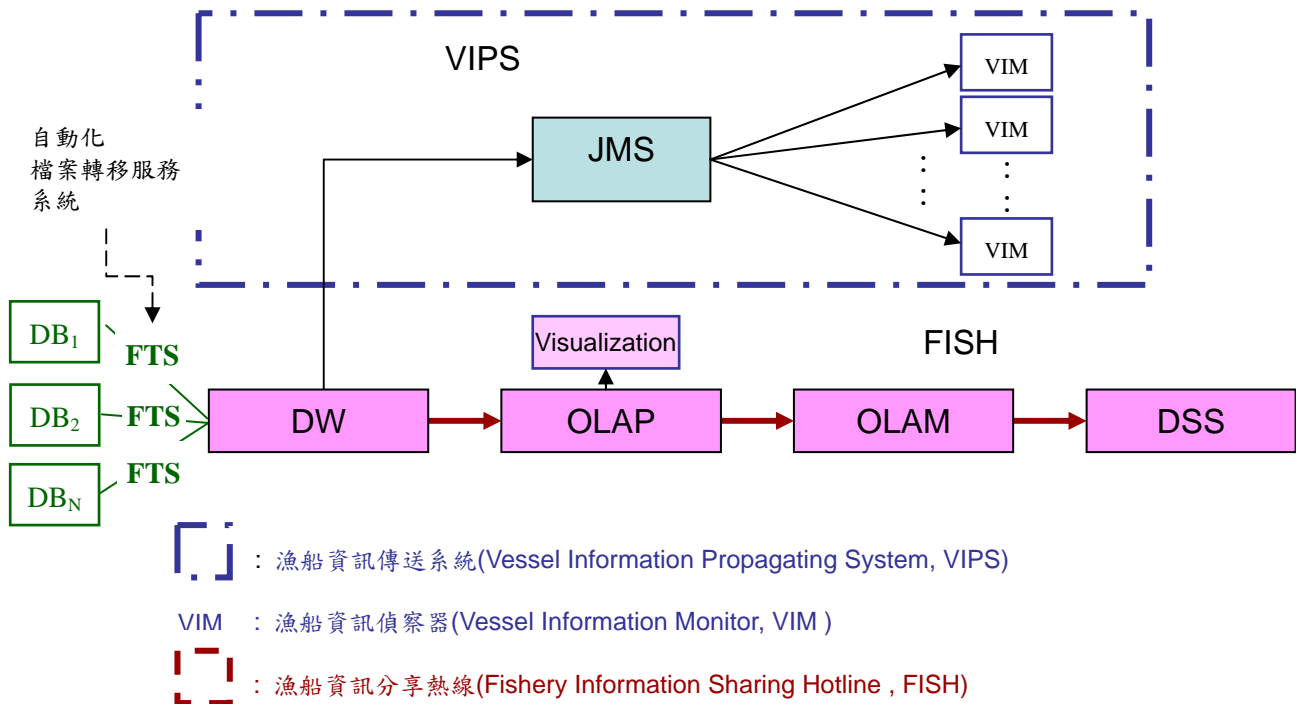


圖 7-1 自動化檔案轉移服務跨漁業領域的應用

在未來的研究上，由於目前的花卉資訊整合系統僅提供了彰化與台北花卉公司的整合實作，但花卉公司倉儲的陸續建置(如：台中、台南，與高雄花卉公司)，將使得此交易資訊處理系統的使用更為迫切，單靠客製化軟體的方式只會增加 MIS 技術人員工作的繁重，而如何使此整合系統能更具模組化的使用，便為未來的一項挑戰。此外在資訊接收端(即各供應人與承銷人)的資訊儲存後，還需探討如何以更多樣及更有資訊價值的附加功能分析，讓資訊接收端更能有效了解其資訊趨勢以供其買賣行為做為決策，為該系統帶來更高階的資訊再利用價值。



## 參考文獻

1. 台北花卉產銷股份有限公司，台北花卉批發市場產銷年報，1993 年。
2. 周政宏，Java 訊息傳遞，文魁資訊股份有限公司，2002 年。
3. 施威銘研究室，SQL Server2000 設計實務，旗標出版股份有限公司，2000 年。
4. 施威銘研究室，SQL Server2000 管理實務，旗標出版股份有限公司，2000 年。
5. 施重豪，Java 與 XML 技術導向之交易資訊處理系統，工業工程與管理學系碩士論文，國立交通大學工業工程與管理研究所碩士論文，2002 年。
6. 陳錦輝、王景皓，XML 與 JAVA 程式設計大全，金禾資訊股份有限公司，2001 年。
7. 陳冠銘、姜凱文，超廣角網際網路原理與應用，金禾資訊股份有限公司，2002 年。
8. 曹漢清，爪哇訊息服務技術在供應鏈之設計與實作，工業工程與管理學系碩士論文，國立交通大學工業工程與管理研究所碩士論文，2003 年。
9. 曹漢清、梁高榮、黃綉蓉，「花卉交易處理系統的設計與實作」，機械工業，十月，228-239 頁，2002 年。
10. 梁高榮，農產品交易工程學，國立交通大學出版社，1999 年。
11. 梁高榮，「企業對企業電子商務執行系統的設計與實作」，機械工業，十月，295-310 頁，2000 年。
12. 梁高榮，「農產品供應鏈體系電子商務」，機械工業，十一月，222-233 頁，2001 年。
13. 黃中杰、洪菁懌著，Java 與 XML 手冊，基峰出版股份有限公司，2002 年。
14. 黃嘉輝，Java 網際網路程式設計，文魁資訊股份有限公司，2003 年。
15. 郭軒豪、賴俊翰、梁高榮、江東陽，「網路型三階正規化切花/盆花資料庫的建立」，機械工業，十二月，216-224 頁，2002 年。
16. 郭孟涵、梁高榮，「利用線上分析處理技術建構花卉的線上產銷年報」，機械工業，六月，216-224 頁，2002 年。
17. 張淙亮、梁高榮，「利用爪哇技術建立檔案轉移服務」，機械工業，五月，215-223 頁，2004 年。
18. 戴天時、林長毅譯/ Elliotte Rusty Harold 著，Java 網路程式設計，美商歐萊禮股份有限公司，2000 年。
19. 蔡學鏞譯/Richard Monson-Haefel & Chappell 著，Java 訊息服務，美商歐萊禮股份有限公司，2001 年。
20. 農產運銷資訊系統網站，<http://www.amis.gov.tw>。
21. COGNOS Web Site, <http://www.fast.com.tw>。
22. Eugster, P. T., Felber, P. A., Guerraoui, R., and Kermarrec, A., "The Many Faces of Publish/Subscribe," ACM Computing Surveys, Vol. 35, No. 2, pp. 114-131, June 2003.
23. Haase, K., Java™ Message Service API Tutorial, Sun Microsystems, 2001.
24. Huang, Y. and Garcia-Molina, . Publish/subscribe in a Mobile Environment. In Proceedings of MobiDE. 27-34, 2001.
25. Java Message Service Specification: Version 1.1, Sun Microsystems, April 2002.
26. Murata, T., "Petri Nets: Properties, Analysis and Applications," Proceedings of the IEEE, Vol. 77, No. 4, pp. 541-580, 1989.

27. SUN Web Site, <http://www.sun.com> ◦
28. SUN Web Site, <http://java.sun.com/j2ee/1.4/docs/api/> ◦
29. SUN Web Site, <http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html> ◦
30. W3C Web Site, <http://www.w3.org/Protocols/rfc959/Overview.html> ◦



附錄一：花卉公司交易表單

以下為花卉公司四種表單：A1 為供應人小行情，A2 為供應人匯款清單，A3 為承銷人結帳清單，A4 為大行情統計資料表單。各表單又分為(a)彰化花卉公司表單與(b)台北花卉公司表單，A1.供應人小行情為花卉供應人每日在當地花卉公司所進行的交易紀錄表單，A2.供應人匯款清單為花卉公司每隔十日所寄送的匯款紀錄，A3.承銷人結帳清單則為各花卉公司承銷人每日在當地花卉公司所購取花種(類)的交易紀錄，A4.大行情統計資料則為各花卉公司每日對其花種(類)所統計出的統計資料，如每一花種的最高或最低價等。各表單的格式如下。

附圖 A1(a)：彰化花卉公司供應人小行情表單

彰化縣花卉批發市場						交易資料											
市場電話：04-8834062			市場傳真：04-8832846			日期：91.07.02			單位：KH001			Fax:0492572005			Tel:		
系統研發：柏美電腦公司 02-29337801																	
供應號	序號	品名	等	單價	把數	箱數	供應號	序號	品名	等	單價	把數	箱數				
KH001	26400-01	FR108	A+	79	9	1	KH001	26437-01	FR412	A1	30	5	1				
	26401-01		A1	82	10	1		26438-01		A2	26	7	1				
	26402-01		A3	29	9	1		26439-01		A3	18	8	1				
	26403-01	FR601	A1	64	8	1		26439-02		A3	18	8	1				
	26404-01	FR108	A2	62	10	1		26440-00		A4	殘貨	8	1				
	26405-01		A2	59	8	1		26441-01	FR108	A1	102	10	1				
	26406-01		A3	28	10	1		26442-01		A2	72	11	1				
	26407-01	FR408	A1	53	7	1		26443-01		A3	29	13	1				
	26408-01		A2	38	10	1		26445-01	FR408	A2	31	7	1				
	26408-02		A2	38	10	1		26446-01		A3	18	10	1				
	26409-01		A2	31	9	1		26446-02		A3	18	10	1				
	26409-02		A2	31	9	1		26447-01		A3	20	8	1				
	26410-01		A3	17	10	1		26447-02		A3	20	8	2				
	26410-02		A3	17	10	3		26447-03		A3	20	8	1				
	26411-00		A4	殘貨	10	2		26448-01	FR108	A1	59	9	1				
	26412-01	FR412	A1	72	9	1		26449-01		A2	54	8	1				

附圖 A1(b)：台北花卉公司供應人小行情表單

台北花市花卉電子交易市集 - 行情下載 - 切花小行情										第 1 頁, 共 1 頁	
FLOWER 台北花市花卉電子交易市集											
行情下載 - 切花小行情											
查詢成交日期 91   7   2											
日期	供應商	交易序號	交易別	品名	名稱	等級	單價	把數	件數		
0910702	KH001	1781601	成交	FR425	玫瑰愛麗莎	A+	32	9	1		
0910702	KH001	1781701	成交	FR425	玫瑰愛麗莎	A+	36	8	1		
0910702	KH001	1781801	成交	FR425	玫瑰愛麗莎	A1	26	12	1		
0910702	KH001	1781901	成交	FR425	玫瑰愛麗莎	A1	28	10	1		
0910702	KH001	1781902	成交	FR425	玫瑰愛麗莎	A1	28	10	1		
0910702	KH001	1782001	成交	FR425	玫瑰愛麗莎	A2	20	12	1		
0910702	KH001	1782101	殘貨	FR425	玫瑰愛麗莎	A2	0	10	1		
0910702	KH001	1782201	成交	FR420	玫瑰蘇菲亞	A+	25	10	1		
0910702	KH001	1782301	殘貨	FR420	玫瑰蘇菲亞	A2	0	12	1		
0910702	KH001	1929901	成交	FR910	玫瑰迷你紅	A4	51	9	1		
0910702	KH001	2002301	訂貨	FR910	玫瑰迷你紅	A2	200	8	1		
0910702	KH001	2002401	訂貨	FR910	玫瑰迷你紅	A3	110	9	1		

附圖 A2(a)：彰化花卉公司匯款清單

FROM：彰化縣田尾鄉饒平村光復路二段583號  
彰化縣花卉批發市場 寄

~1X0W2; KSA01 楊智華 啓~W1E;  
~1;  
~1; TO：522彰化縣田尾鄉新厝村光明巷31號

※ 台端帳號如有變動或農會系統與信合社被銀行合併者，須事先知會本拍賣市場已便利予作業。

台中區中小企業銀行社71201033341 楊智華

---

報表日期：092.09.30 結帳日期：092/09/11至：092/09/20 頁次：0001

日期	序號	承銷商	品名	等級	單價	把/箱	箱數	金額
920913	08716-01	C0107	FT661 雷山1號單朵	A	40	25	1	1,000
920913	08717-01	C0155	FT661 雷山1號單朵	A	39	30	1	1,170
920913	08718-01	C0106	FT662 雷山1號雙朵	A	51	25	1	1,275
920913	08719-01	C0078	FT663 雷山1號三朵	A	68	12	1	816
頁計	00000				46	92	4	4,261
合計	00000				46	92	4	4,261

---

合計 00000 46 92 4 4,261

管理費：83 增值稅： 本月傳真費： 整理搬運： 43  
 容器費用： 事故金額： 應付金額： 4,135

附圖 A2(b)：台北花卉公司匯款清單

TO: 264  
宜蘭縣員山鄉隘界路138號

AEW06  
江招益 啓

01110310040986 江招益

---

報表日期：91.08.01 結帳日期：91.07.21 至 91.07.31 頁次：1

日期	序號	承銷商	品名	名稱	等級	單價	把/箱	箱數	金額
910724	80670.01	P0309	N1102	大豬籠草	15	100	12	3	3,600
910724	80671.01	P0309	N1102	大豬籠草	09	35	24	3	2,520
910724	80672.01	P0309	P3702	果樹	18	60	10	2	1,200
910724	80673.01	P0309	P1920	西瓜皮椒草	15	45	12	1	540
910724	80674.01	P0161	G1312	小孿天	09	20	24	3	1,440
910725	80148.02	P0312	C6902	袖珍椰子	09	15	24	3	1,080
910725	80676.01	P0309	C4902	白紋草	09	15	24	3	1,080
910726	80676.02	P0309	C4902	白紋草	09	15	24	3	1,080
910729	80675.01	P0199	C6902	袖珍椰子	09	15	24	3	1,080
910730	80675.02	P0429	C6902	袖珍椰子	09	15	24	3	1,080
910731	81231.01	P0167	C4902	白紋草	09	15	24	3	1,080
910731	81233.01	P0161	G1312	小孿天	09	20	24	3	1,440
910731	81234.01	P0161	N0502	袋鼠花	09	20	24	3	1,440
總計						24	788	36	18,660

---

管理費：345 增值稅： 次月傳真費： 0 整理搬運： 188  
 平準基金： 應付金額： 18,127



附圖 A3(a)：彰化花卉公司承銷人結帳清單

彰化縣花卉批發市場 承銷商結帳清單列印						
~I;	C0088	陳東坡	承別:0	092-08-03	0001	
序號	供應號	品名	級	單價	把箱數	總價
* 1425-01	KS023	大菊黃精競	A	75 20	2	3,000
* 1427-01	KMA52	小菊吉祥金	A	56 25	1	1,400
* 3201-01	KMA02	小菊緋紅	A	54 26	1	1,404
* 3203-01	KSP57	小菊金	A	63 26	1	1,638
* 3204-01	KSP57	小菊吉祥黃	A	64 24	1	1,536
* 4032-01	JFC29	劍蘭 2 3 5 號	A1	103 8	2	1,648
* 7125-01	KQ006	秀線	A0	17 30	1	510
* 8668-01	LFUA8	非洲菊混合色	A5	17 30	3	1,530
*11280-01	KSKA8	小菊丁字菊	A	61 25	1	1,525
*12186-01	KSA01	小菊吉祥小粉	A	59 25	1	1,475
*12187-01	KSA01	小菊吉祥黃	A	66 21	1	1,386
*22520-02	NPA00	洋桔梗雙瓣粉	A7	54 15	1	810
*22701-01	KSM03	卡斯比亞<無葉>	A	60 20	1	1,200
立式容器				60	2	120
13				50	17	19,062
前結餘:	-60,460	今繳款:	承銷:	19,062	管理費:	372
營業稅:	19	退款:	容器:	120	餘額:	-80,033

附圖 A3(b)：台北花卉公司承銷人結帳清單

台北花卉產銷股份有限公司 承銷人結帳清單						
承銷代號:	C0129	郭正和	日期:	91.07.30	任務號:	109
				頁次:	1	
拍賣序號	供應代號	品名代號 / 品名	等級	單價	把數/總箱數	總價/總金額
01084-01	NDAA2	阿卡波克紅二茶	A9	179	14 1	2,506
01529-01	OHM00	野薔花白	A	26	10 1	260
03610-01	RAX00	嘉蘭<火焰百合>	A6	29	25 1	725
05034-01	RQR16	麒麟草	A	25	25 2	1,250
05608-21	GXX67	石斛蘭深紅	A6	77	10 1	770
05611-05	GXX67	石斛蘭白西施白	A7	91	10 1	910
05908-01	QMA08	LA百合橙多朵	A7	59	12 1	708
07023-09	LGZ09	香水姬羅沙多朵	A9	171	5 2	1,710
07026-02	LGZ09	寶貝粉多朵	A9	182	5 1	910
07082-07	L GK21	香水百合多朵	A0	170	8 2	2,720
07173-04	LFU63	馬可波羅粉多朵	A9	148	18 1	2,664
07302-02	LG Y00	伯利恆之櫻	A9	101	20 1	2,020
07318-03	LQU04	星辰花	A	81	20 1	1,620
08306-05	LFB86	火鶴花	A1	10	24 1	240
09218-06	BGY01	水仙百合粉	A7	72	20 1	1,440
09335-04	MDA22	馬可波羅粉多朵	A9	191	16 1	3,056
09338-01	LGA23	香水辛普隆多朵	A9	145	10 1	1,450
09342-01	LGA23	香水辛普隆雙朵	A9	125	11 1	1,375
09365-04	LGA24	香水百合白多朵	A9	181	6 1	1,086
09527-03	LFP08	星辰花粉	A8	74	25 1	1,850
20					23	29,270
前日餘額		本日繳款		本日承銷		管理費
營業稅		容器扣款		容器退款		其他扣項
容器數量		香板扣款		香板退款		本日餘額



附圖 A4(a)：彰化花卉公司大行情傳真表單

■IX10; 彰化縣花卉批發市場 大行情 報表日期： 093.03.02  
系統設計：柏美電腦公司 02-29337801

代號	品名	進貨量	拍賣量	訂貨量	殘貨量	訂貨價	上價	中價	下價	最高價	最低價
FA104	康乃馨櫻紅	248	248				40	36	29	41	29
FA120	康乃馨紅羽毛	96	96				29	25	25	29	25
FA420	康乃馨粉羽毛	74	74				37	30	26	37	26
FA700	康乃馨紫	148	148				40	40	35	40	34
FA800	康乃馨多朵混色	770	618	50	102	25	22	17	15	24	15
FA810	康乃馨多朵紅	750	590		160		24	17	15	25	15
FA811	康乃馨多朵櫻紅	712	662	50		31	28	19	15	29	15
FA840	康乃馨多朵粉	445	445				22	17	15	24	15
FA870	康乃馨多朵紫	209	209				25	19	17	28	17
FA881	康乃馨中紅帶白	475	475				25	20	17	27	16
FA887	康乃馨中白帶紫	328	328				25	18	15	27	15
FA910	小可愛紅	106	106				10	10	10	10	10
FA940	小可愛粉	132	132				17	12	10	17	10
FA970	小可愛紫	146			146						
類計		4,876	4,368	100	408						
FB103	火鶴花紅青心	153	153				16	15	13	16	13
FB106	火鶴花丘比特	3,687	3,627	60		12	15	12	7	16	5
FB401	火鶴花夏粉色	170	170				10	10	9	10	9
FB404	火鶴花粉新娘	300	300				14	8	7	14	7
FB405	火鶴花粉綠邊	665	665				14	12	8	14	6
FB407	火鶴花夢幻	1,536	1,479	57		19	16	13	7	18	6
類計		6,571	6,454	117							
FC001	大菊混合色	81	81				67	49	35	67	35
FC100	大菊紅	286	286	80		75	72	62	29	72	15
FC116	大菊花御殿	88	88				71	65	47	71	47
FC300	大菊黃	123	123				58	43	27	59	26
FC301	大菊黃秀鳳	4,031	3,738	293		73	67	58	42	70	23
FC350	大菊黃乒乓	136	136				75	57	41	75	41
FC609	大菊白東洋	1,340	1,286	54		49	48	38	24	50	20
類計		6,253	5,826	427							

附圖 A4(b)：台北花卉公司大行情網頁查詢表單

台北花市花卉電子交易市集

行情下載 - 切花大行情  
資料時間: 91/08/06 01:56:36 / 重新整理

查詢品種:  FA  FB  FC  FD  FE  FG  FH  FK  FL  FO  FR  FS  FT  FU  FW  FY

查詢成交日期: 91. 8. 2 查詢品名: FA FA100.康乃馨紅 查詢

91年8月2日 總計		04911	75792	4503	4290	230						
品名	名稱	進貨量	拍賣量	訂貨量	行口量	殘貨量	訂高價	上價	中價	下價	最高價	最低價
FA100	康乃馨紅	87	87	0	0	0	0	10	10	16	10	16
類計		87	87	0	0	0						

## 附錄二：出版者軟體設計程式碼

由於彰化花卉公司與台北花卉公司出版者軟體部份十分雷同，在此僅附上彰化花卉公司軟體。其中 fm\_publisher.java 在透過 SQL\_Statement.java 取出資料庫檔案後進行訊息出版的功能。

檔名：fm\_publisher.java

```
package Flower_TIPS;

import java.io.*;
import java.util.*;
import java.sql.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.border.*;

public class fm_publisher extends JFrame implements ActionListener{
    ImageIcon title_icon=new ImageIcon("ftp.gif");
    JPanel contentPane;
    JPanel panelup = new JPanel();
    JLabel logo = new JLabel("",new
    ImageIcon(Flower_TIPS.fm_publisher.class.getResource("logo.jpg")),JLabel.CENTER);
    JPanel panelleft = new JPanel();
    GridLayout gridLayout1 = new GridLayout();
    JLabel ip = new JLabel();
    JLabel id = new JLabel();
    JLabel pw = new JLabel();
    JTextField iptext = new JTextField("localhost:2506");
    JPasswordField pwtext = new JPasswordField("123456");
    JTextField idtext = new JTextField("nctu");
    JPanel panelright = new JPanel();
    GridLayout gridLayout2 = new GridLayout();
    JButton bt1 = new JButton();
    JButton bt2 = new JButton();
    JPanel paneldown = new JPanel();
    BorderLayout borderLayout1 = new BorderLayout();
    JScrollPane jScrollPane1 = new JScrollPane();
    JTextArea publishContent = new JTextArea();
    JButton bt3 = new JButton();
    JLabel monitor = new JLabel();
    //Flower_TIPS 變數設定
    private static final String PROPERTY_NAME = "Department";
    private static final long message_lifespan =165900000;
    private javax.jms.QueueConnection connect = null;
    private javax.jms.QueueSession sendSession = null;
    private javax.jms.QueueSession receiveSession = null;
    private javax.jms.QueueSender sender = null;

    //SQL 語法使用變數設定
    static boolean runflage=false;
    static int total_is=0;//出版小行情總筆數
    static int total_gs=0;//出版大行情總筆數
    static int total_bas=0;//出版結帳清單總筆數
    SQL_Statement sqlstatement=null;
```

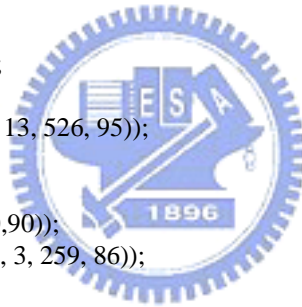
```
Mail email=new Mail();//發送 Mail
```

```
//Construct the frame
```

```
public fm_publisher() {  
    enableEvents(AWTEvent.WINDOW_EVENT_MASK);  
    try {  
        System.out.println("gogo");  
        jbInit();  
        auto_connection();  
        Calendar nowtime=Calendar.getInstance();  
        String stime="花卉訊息出版者啟動時間：  
"+nowtime.get(Calendar.HOUR_OF_DAY)+"："+nowtime.get(Calendar.MINUTE)+"："+nowtime.get(Calendar.SECON  
D);  
        email.sendmail(stime);  
    }catch(Exception e) {  
        e.printStackTrace();  
    }  
}
```

```
//Component initialization
```

```
private void jbInit() throws Exception {  
    this.setIconImage(title_icon.getImage());  
    contentPane = (JPanel) this.getContentPane();  
    contentPane.setBackground(new Color(79, 217, 131));  
    contentPane.setLayout(null);  
    this.setResizable(false);  
    this.setSize(new Dimension(800,600));  
    this.setTitle("花卉訊息出版者(每日)");  
    panelup.setBackground(Color.pink);  
    panelup.setBounds(new Rectangle(110, 13, 526, 95));  
    panelup.setLayout(null);  
    logo.setText("");  
    logo.setBounds(new Rectangle(8,3, 120,90));  
    panelleft.setBounds(new Rectangle(120, 3, 259, 86));  
    panelleft.setLayout(gridLayout1);  
    ip.setFont(new java.awt.Font("Dialog", 0, 14));  
    ip.setForeground(Color.blue);  
    ip.setOpaque(false);  
    ip.setHorizontalAlignment(SwingConstants.CENTER);  
    ip.setText("伺服器 IP-Port : ");  
    id.setText("帳號 : ");  
    id.setHorizontalAlignment(SwingConstants.CENTER);  
    id.setForeground(Color.blue);  
    id.setOpaque(false);  
    id.setFont(new java.awt.Font("Dialog", 0, 14));  
    gridLayout1.setColumns(2);  
    gridLayout1.setRows(3);  
    pw.setFont(new java.awt.Font("Dialog", 0, 14));  
    pw.setForeground(Color.blue);  
    pw.setOpaque(false);  
    pw.setHorizontalAlignment(SwingConstants.CENTER);  
    pw.setText("密碼 : ");  
    iptext.setFont(new java.awt.Font("SansSerif", 0, 14));  
    iptext.setToolTipText("伺服器 IP-Port");  
    iptext.addActionListener(new fm_publisher_ipText_actionAdapter(this));  
    idtext.setFont(new java.awt.Font("SansSerif", 0, 14));  
    idtext.setToolTipText("輸入帳號");  
    pwtext.setFont(new java.awt.Font("SansSerif", 0, 14));  
    pwtext.setToolTipText("輸入密碼");  
    panelright.setBounds(new Rectangle(394, 10, 124, 78));
```



```

panelright.setLayout(gridLayout2);
gridLayout2.setColumns(1);
gridLayout2.setHgap(5);
gridLayout2.setRows(2);
gridLayout2.setVgap(5);
bt1.setFont(new java.awt.Font("Dialog", 0, 14));
bt1.setForeground(Color.blue);
bt1.setText("開始出版訊息");
bt1.addActionListener(this);
bt2.setText("停止出版訊息");
bt2.setEnabled(false);
bt2.addActionListener(this);
paneldown.setBounds(new Rectangle(13, 113, 766, 384));
paneldown.setLayout(borderLayout1);
publishContent.setToolTipText("出版訊息內容");
publishContent.setAutoscrolls(true);
publishContent.setEditable(false);
bt3.setToolTipText("清除畫面");
bt3.setText("清除畫面");
bt3.addActionListener(this);
monitor.setText("請輸入連線設定值");
monitor.setFont(new java.awt.Font("Dialog", 0, 14));
monitor.setForeground(Color.blue);
monitor.setBounds(new Rectangle(14, 504, 766, 53));
monitor.setBorder(BorderFactory.createTitledBorder(BorderFactory.createLineBorder(Color.red,2),
"系統狀況監控中心",TitledBorder.LEFT,TitledBorder.TOP,new Font("SansSerif",0,14),Color.black));
monitor.setText("請輸入連線設定值");
contentPane.add(panelup, null);
panelleft.add(ip, null);
panelleft.add(iptext, null);
panelleft.add(id, null);
panelleft.add(idtext, null);
panelleft.add(pw, null);
panelleft.add(pwtext, null);
panelleft.setOpaque(false);
panelup.add(logo, null);
panelright.add(bt1, null);
panelright.add(bt2, null);
panelup.add(panelleft, null);
panelup.add(panelright, null);
panelright.setOpaque(false);
paneldown.add(bt3, BorderLayout.SOUTH);
jScrollPane1.getViewport().add(publishContent, null);
paneldown.add(jScrollPane1, BorderLayout.CENTER);
contentPane.add(monitor, null);
contentPane.add(paneldown, null);
}

```



```
//Overridden so we can exit when window is closed
```

```

protected void processWindowEvent(WindowEvent e) {
    super.processWindowEvent(e);
    if (e.getID() == WindowEvent.WINDOW_CLOSING) {
        leavesystem();
    }
}

```

```

public void leavesystem(){
    String tempstring="總共送出"+total_is+"筆供應人小行情資料；"+total_gs+"筆大行情資料；"+total_bas+"筆承銷人結帳清單。";
    Server_log server_log=new Server_log();

```

```

server_log.servermessage(tempstring);
email.sendmail(tempstring);
System.exit(0);
}

public void actionPerformed(ActionEvent aevent){
    if(aevent.getSource()==bt1){
        auto_connection();
    }else if(aevent.getSource()==bt2){
        idtext.setEditable(true);
        pwtext.setEditable(true);
        bt1.setEnabled(true);
        bt2.setEnabled(false);
        runflage=false;
    }else if(aevent.getSource()==bt3){
        publishContent.setText("");
    }
}
}

//End actionPerformed method

public void auto_connection(){
    if(idtext.getText().length()==0){
        display("請輸入帳號");
    }else if(pwtext.getText().length()==0){
        display("請輸入密碼");
    }else{
        idtext.setEditable(false);
        pwtext.setEditable(false);
        bt1.setEnabled(false);
        if (connect==null) setUpPublisher(iptext.getText(),idtext.getText(),pwtext.getText());
        bt2.setEnabled(true);
        runflage=true;
    }
}

public void display(String str){//監控系統狀況方法
    monitor.setText(str);
}

public void setUpPublisher(String address,String username,String password){
    try{
        javax.jms.QueueConnectionFactory factory;
        factory = (new progress.message.jclient.QueueConnectionFactory (address));
        connect = factory.createQueueConnection (username, password);
        sendSession = connect.createQueueSession(false,javax.jms.Session.AUTO_ACKNOWLEDGE);
        javax.jms.Queue sendQueue = sendSession.createQueue ("SampleQ1");
        sender = sendSession.createSender(sendQueue);
        connect.start();
        display("連接爪哇訊息服務者成功!!!");
        bt2.setEnabled(true);
        runflage=true;
        if(sqlstatement==null){
            sqlstatement=new SQL_Statement(this);
            sqlstatement.start();
        }
    }catch(javax.jms.JMSEException jexception){
        display("連接爪哇訊息服務者失敗!!");
        bt1.setEnabled(true);
        bt2.setEnabled(false);
        idtext.setEditable(true);
        pwtext.setEditable(true);
    }
}

```





```

        runflage=false;
        jexception.printStackTrace();
    }
}

public synchronized void publishmessage1(String passmessage,String selection){//小行情出版訊息//
try{
    if(passmessage==null || selection==null){
        exitConnection();
    }else{
        progress.message.jclient.XMLMessage msg1 = ((progress.message.jclient.Session)
        sendSession).createXMLMessage();
        msg1.setText(passmessage.toString());
        msg1.setStringProperty(PROPERTY_NAME,selection.trim().toUpperCase());
        sender.send(msg1,javax.jms.DeliveryMode.PERSISTENT,javax.jms.Message.DEFAULT_PRIORITY,message_lifespan);
        total_is=total_is+1;
        display("總共送出"+total_is+"筆供應人小行情資料；"+total_gs+"筆大行情資料；"+total_bas+"筆承銷人
        結帳清單。");
    }
} catch(javax.jms.JMSEException jexception){
    jexception.printStackTrace();
}
}

public synchronized void publishmessage4(String passmessage,String selection){//小行情出版訊息
try{
    if(passmessage==null || selection==null){
        exitConnection();
    }else{
        progress.message.jclient.XMLMessage msg4 =
        ((progress.message.jclient.Session)sendSession).createXMLMessage();
        msg4.setText(passmessage.toString());
        msg4.setStringProperty(PROPERTY_NAME,selection.trim().toUpperCase());
        sender.send(msg4,javax.jms.DeliveryMode.PERSISTENT,javax.jms.Message.DEFAULT_PRIORITY,message_lifespan);
    }
} catch(javax.jms.JMSEException jexception){
    jexception.printStackTrace();
}
}

public void exitConnection(){//中斷爪哇訊息服務者連線
try{
    connect.close();
    display("停止傳送：總共送出"+total_is+"筆供應人小行情資料；"+total_gs+"筆大行情資料；"+total_bas+"
    筆承銷人結帳清單。");
} catch(javax.jms.JMSEException jexception){
    jexception.printStackTrace();
}
}
}
}

```

檔名：SQL\_Statement.java

```
package Flower_TIPS;

import java.sql.*;
import java.util.*;

public class SQL_Statement extends Thread{
    fm_publisher fmp;
    Vector delcolumn1=new Vector(10);
    static int iscount=0;

    public SQL_Statement(fm_publisher fmp){
        this.fmp=fmp;
    }

    public void run(){
        while(true){
            try{
                if(fmp.runflage) Run_IS_SqlQuery();
                if(fmp.runflage && iscount==1){
                    Run_IS_End();
                    System.out.println("結束供應人 sql");
                }
                Thread.sleep(8000);
                if(iscount>2) {
                    fmp.leavesystem();
                }
            }catch(InterruptedException e){
                fmp.display("執行緒中斷");
                e.printStackTrace();
            }
        }
    }
}

public void Run_IS_SqlQuery(){//select 出資料庫尚未被送出資料
    String url1="jdbc:odbc:flowerdatabase2";
    Connection con1=null;
    Statement statement1=null;
    String selection1=null;
    try{
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        con1=DriverManager.getConnection(url1,"sa","sa");
        statement1=con1.createStatement();
        String sqlquery1="set rowcount 2000 select * from individual";//設定 SQL 語法
        ResultSet rs1=statement1.executeQuery(sqlquery1);//執行 SQL 語法
        ResultSetMetaData rsmd1=rs1.getMetaData();//找到每個欄位
        int numbercol=rsmd1.getColumnCount();//Table 表的欄位數
        while(rs1.next())&& fmp.runflage){
            int totalcost1=1;
            StringBuffer xmldoc1=new StringBuffer("<?xml version='1.0' encoding='BIG5'>\n<供應人小行情>\n<交易筆>\n");
            for(int i=1;i<=numbercol;i++){
                if (i==1){
                    int tempint=rs1.getInt(i);
                    delcolumn1.addElement(Integer.toString(tempint).trim());
                }else if(i==2){
                    String temp2=rs1.getObject(i).toString();
                    String date=temp2.substring(0,4)+"/"+temp2.substring(5,7)+"/"+temp2.substring(8,10);
                }
            }
        }
    }
}
```



```

        xmldoc1.append("<市場>彰化</市場>\n");
        xmldoc1.append("<"+rsmd1.getColumnname(i)+">"+date+"</"+rsmd1.getColumnname(i)+">\n");
    }else if(i==3){
        selection1=(String)rs1.getString(i).trim().toUpperCase();
        xmldoc1.append("<"+rsmd1.getColumnname(i)+">"+selection1+"</"+rsmd1.getColumnname(i)+">\n"
        );
    }else if(i==4){
        String temp=rs1.getString(i);
        temp=temp.substring(0,5)+"-"+temp.substring(5,7);
        xmldoc1.append("<"+rsmd1.getColumnname(i)+">"+temp+"</"+rsmd1.getColumnname(i)+">\n");
    }else if(i==5){
        String temp5=rs1.getObject(i).toString().trim();
        if(temp5.equals("1")){
            temp5="拍賣成交";
        }else if(temp5.equals("0")){temp5="殘貨";
        }else {temp5="訂貨";}
        xmldoc1.append("<"+rsmd1.getColumnname(i)+">"+temp5+"</"+rsmd1.getColumnname(i)+">\n");
    }else if(i==7){
        String temp=(String)rs1.getObject(i);
        xmldoc1.append("<"+rsmd1.getColumnname(i)+">"+temp.trim()+"</"+rsmd1.getColumnname(i)+">\n"
        );
    }else if(i==9 || i==10 || i==11){
        String tempstring2=(String)rs1.getString(i);
        if(tempstring2==null){
            totalcost1=0;
            xmldoc1.append("<"+rsmd1.getColumnname(i)+">0</"+rsmd1.getColumnname(i)+">\n");
        }else{
            totalcost1=totalcost1*(int)Float.parseFloat(tempstring2.trim());
            xmldoc1.append("<"+rsmd1.getColumnname(i)+">"+(int)Float.parseFloat(tempstring2.trim())+"</"+
            rsmd1.getColumnname(i)+">\n");
        }
    }else{
        xmldoc1.append("<"+rsmd1.getColumnname(i)+">"+rs1.getString(i).toString().trim()+"</"+rsmd1.getCol
        umnname(i)+">\n");
    }
}
xmldoc1.append("<總金額>"+totalcost1+"</總金額>\n");
xmldoc1.append("</交易筆>\n");
xmldoc1.append("</供應人小行情>\n\n");
fmp.publishContent.append(xmldoc1.toString());
fmp.publishmessage1(xmldoc1.toString(),selection1);
selection1="";
} //End while

for(int k=0;k<delcolumn1.size();k++){
    String delsqlstring="delete from individual where Number="+delcolumn1.get(k)+"";
    statement1.executeUpdate(delsqlstring);
}
if(delcolumn1.size()==0){
    iscount++;
    System.out.print("供應人計算次數:");
    System.out.println(iscount);
}
delcolumn1.clear();
statement1.close();
con1.close();
} catch(ClassNotFoundException e){
    fmp.display("連接 JDBC:ODBC 出現錯誤");
    e.printStackTrace();
} catch(SQLException e){

```



### 附錄三：供應人軟體設計程式碼

fm\_subscriber.java 為供應人軟體主要程式，包含主要 GUI 介面的呈現、使用者訊息接收、資料庫儲存、及 XML 文件剖析與儲存。其主畫面如下：



附圖 1 fm\_subscriber.java GUI 畫面

檔名：fm\_subscriber.java



```
package Flower_TIPS;
```

```
import java.io.*;
import java.util.*;
import java.net.*;
import java.awt.*;
import java.sql.*;
import org.w3c.dom.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.border.*;
import javax.swing.table.DefaultTableModel;
```

```
public class fm_subscriber extends JFrame implements javax.jms.MessageListener, ActionListener, ItemListener {
    ImageIcon title_icon = new ImageIcon("icon/ftp.gif");
    JPanel contentPane;
    JPanel allpanel = new JPanel();
    JPanel statuspanel = new JPanel(); //存放狀態列
    JPanel timepanel = new JPanel(); //時間
    JPanel panel = new JPanel(); //只放置參數設定訊息
    JPanel toolbarpanel = new JPanel(); //放置 Toolbar 與 logo 的 panel
    JPanel logopanel = new JPanel();
    JLabel logo = new JLabel("", new ImageIcon("icon/mark.gif"), JLabel.CENTER);
    JLabel logo2 = new JLabel("", new ImageIcon("icon/mark2.gif"), JLabel.CENTER);
    GridLayout gridLayout1 = new GridLayout();
    JLabel ip = new JLabel();
    JTextField iptext = new JTextField(10);
    JLabel id = new JLabel();
```



```

JTextField idtext = new JTextField(10);
JLabel topic = new JLabel();
JTextField topictext = new JTextField(10);
JLabel pw = new JLabel();
JPasswordField pwtext = new JPasswordField(10);
JScrollPane scroll = new JScrollPane();
JLabel monitor = new JLabel();
static JLabel time = new JLabel();
JLabel date = new JLabel();
FlowLayout flowLayout1 = new FlowLayout();
FlowLayout flowLayout2 = new FlowLayout();
menubar bar; //功能列
parameter_setup parametersetup = new parameter_setup(); //參數設定
viewxml view = new viewxml(); //看 XML 文件
explain explain_html = new explain(); //操作說明
About about = new About(); //關於此軟體
StringBuffer viewxmlbuffer = new StringBuffer();
//Flower_TIPS 變數設定
private static final String DEFAULT_BROKER_NAME = "localhost:2506";
private static final String PROPERTY_NAME = "Department";
private static final String DEFAULT_PASSWORD = "password";
private static final int message_lifespan = 180000; // milliseconds (30 minutes)
private javax.jms.QueueConnection connect = null;
private javax.jms.QueueSession sendSession = null;
private javax.jms.QueueSession receiveSession = null;
private javax.jms.QueueSender sender = null;
private static String App_Topic;
private static String Queue;
private static String client_id;
javax.jms.TopicConnection connect_gs = null;
javax.jms.TopicSession session_gs = null;
javax.jms.TopicSubscriber subscriber_gs = null;

//SQL 語法變數設定
static int iscount = 0;
static int gscount = 0;
static int sascount = 0;
StringBuffer stringhead;
StringBuffer stringvalue = new StringBuffer("values (");
Vector sqlvector = new Vector(); //存放 SQL 語法的 Vector
int slength1;
int slength2 = stringvalue.length(); //用來刪除 SQL 語法時，所使用的變數
//表格顯示所使用到變數
showtable stable = new showtable(); //表格顯示設定
Vector tablerow = new Vector(10); //表格每一 Row 的內容
org.w3c.dom.Node node;
//花卉交易資料存成 XML 文件檔所需變數
StringBuffer savexml doc = new StringBuffer();
String choiceTopic; // 判斷選擇訂閱主題
String endcolumn;
//存檔所使用的 Class
messagefile mfile;
String save_xsl = null;
fm_SQL fm_sql = new fm_SQL();

//Construct the frame
public fm_subscriber() {
    enableEvents(AWTEvent.WINDOW_EVENT_MASK);
    try {
        mfile = new messagefile();

```



```

        bar = new menubar();
        jbInit();
    }
    catch (Exception e) {
        e.printStackTrace();
    }
}

//Component initialization
private void jbInit() throws Exception {
    setJMenuBar(bar.buildmenubar());
    contentPane = (JPanel)this.getContentPane();
    contentPane.setLayout(new BorderLayout());
    allpanel.setLayout(new BorderLayout());
    allpanel.setSize(new Dimension(800, 600));
    allpanel.setBackground(new Color(100, 200, 150));
    toolbarpanel.setLayout(new BorderLayout());
    statuspanel.setLayout(new BorderLayout());
    flowLayout2.setHgap(1);
    flowLayout2.setVgap(1);
    timepanel.setLayout(flowLayout2);
    this.setSize(new Dimension(800, 600));
    this.setResizable(true);
    this.setIconImage(title_icon.getImage());
    this.setTitle("花卉訊息訂閱者(供應人)");
    scroll.getViewPort().setAutoscrolls(true);
    scroll.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);
    monitor.setFont(new java.awt.Font("Dialog", 0, 13));
    monitor.setText("請輸入連線設定值");
    monitor.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
    monitor.setForeground(Color.blue);
    time.setForeground(Color.blue);
    time.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
    time.setFont(new java.awt.Font("Dialog", 0, 13));
    date.setForeground(Color.blue);
    date.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
    date.setText(mfile.todaydate());
    date.setFont(new java.awt.Font("Dialog", 0, 13));
    panel.setBounds(new Rectangle(1, 0, 772, 30));
    panel.setLayout(flowLayout1);
    ip.setForeground(Color.blue);
    ip.setHorizontalAlignment(SwingConstants.CENTER);
    ip.setText("伺服器 IP-Port : ");
    iptext.setBackground(SystemColor.info);
    id.setText("訂閱者帳號 : ");
    id.setHorizontalAlignment(SwingConstants.CENTER);
    id.setForeground(Color.blue);
    idtext.setBackground(SystemColor.info);
    pw.setText("訂閱者密碼 : ");
    pw.setHorizontalAlignment(SwingConstants.CENTER);
    pw.setForeground(Color.blue);
    pwtext.setBackground(SystemColor.info);
    topic.setForeground(Color.blue);
    topic.setHorizontalAlignment(SwingConstants.CENTER);
    topic.setText("訂閱主題 : ");
    topictext.setBackground(SystemColor.info);
    iptext.setEditable(false);
    idtext.setEditable(false);
    pwtext.setEditable(false);
    topictext.setEditable(false);

```

```

flowLayout1.setHgap(10);
logo.setBorder(BorderFactory.createEtchedBorder());
logo2.setBorder(BorderFactory.createEtchedBorder());
logopanel.setLayout(new GridLayout());
logopanel.add(logo);
logopanel.add(logo2);
toolbarpanel.add(logopanel, BorderLayout.EAST);
toolbarpanel.add(bar.buildtoolbar(), BorderLayout.CENTER);
panel.add(ip, null);
panel.add(iptext, null);
panel.add(id, null);
panel.add(idtext, null);
panel.add(pw, null);
panel.add(pwtext, null);
panel.add(topic, null);
panel.add(topictext, null);
panel.setOpaque(false);
timepanel.add(time, null);
timepanel.add(date, null);
statuspanel.add(monitor, BorderLayout.CENTER);
statuspanel.add(timepanel, BorderLayout.EAST);
allpanel.add(scroll, BorderLayout.CENTER);
allpanel.add(panel, BorderLayout.NORTH);
scroll.revalidate();
monitor.setText(mfile.lastread());
contentPane.add(allpanel, BorderLayout.CENTER);
contentPane.add(toolbarpanel, BorderLayout.NORTH);
contentPane.add(statuspanel, BorderLayout.SOUTH);
//選單按鈕動作
bar.fitem1.addActionListener(this);
bar.fitem2.addActionListener(this);
bar.fitem3.addActionListener(this);
bar.fitem4.addActionListener(this);
bar.oitem1.addActionListener(this);
bar.oitem2.addActionListener(this);
bar.oitem3.addActionListener(this);
bar.vitem1.addActionListener(this);
bar.hitem1.addActionListener(this);
bar.hitem2.addActionListener(this);
bar.bt1.addActionListener(this);
bar.bt2.addActionListener(this);
bar.bt3.addActionListener(this);
bar.bt4.addActionListener(this);
bar.bt5.addActionListener(this);
bar.bt6.addActionListener(this);
bar.bt7.addActionListener(this);
//參數按鈕動作
parametersetup.choice1.addItemListener(this);
parametersetup.choice2.addItemListener(this);
parametersetup.choice3.addItemListener(this);
parametersetup.bt1.addActionListener(this);
parametersetup.bt2.addActionListener(this);
}

//Overridden so we can exit when window is closed
protected void processWindowEvent(WindowEvent e) {
    super.processWindowEvent(e);
    if (e.getID() == WindowEvent.WINDOW_CLOSING) {
        System.exit(0);
    }
}

```



```

}

public void actionPerformed(ActionEvent aevent) {
    if (aevent.getSource() == bar.fitem1 || aevent.getSource() == bar.bt1) { //參數設定
        open_frame(parametersetup);
    }
    else if (aevent.getSource() == parametersetup.bt1) {
        parametersetup.setVisible(false);
        get_parameter();
        bar.fitem2.setEnabled(true);
        bar.bt2.setEnabled(true);
    }
    else if (aevent.getSource() == parametersetup.bt2) {
        parametersetup.idtext.setText("");
        parametersetup.pwtext.setText("");
    }
    else if (aevent.getSource() == bar.fitem2 || aevent.getSource() == bar.bt2) { //啟動接收
        buttonaction("fitem2");
    }
    else if (aevent.getSource() == bar.fitem3 || aevent.getSource() == bar.bt3) { //停止接收
        buttonaction("fitem3");
    }
    else if (aevent.getSource() == bar.oitem1 || aevent.getSource() == bar.bt4) {
        buttonaction("oitem1");
    }
    else if (aevent.getSource() == bar.oitem2 || aevent.getSource() == bar.bt5) {
        buttonaction("oitem2");
    }
    else if (aevent.getSource() == bar.oitem3 || aevent.getSource() == bar.bt7) {
        buttonaction("oitem3");
    }
    else if (aevent.getSource() == bar.vitem1 || aevent.getSource() == bar.bt6) {
        buttonaction("vitem1");
    }
    else if (aevent.getSource() == bar.hitem1) {
        buttonaction("hitem1");
    }
    else if (aevent.getSource() == bar.hitem2) {
        buttonaction("hitem2");
    }
    else if (aevent.getSource() == bar.fitem4) {
        buttonaction("fitem4");
    }
} //End actionPerformed method

```

```

public void buttonaction(String button) {
    if (button.equals("fitem2")) {
        if (idtext.getText().length() == 0) { //檢驗帳號密碼是否有輸入
            display("請輸入訂閱者帳號 ");
        }
        else if (pwtext.getText().length() == 0) {
            display("請輸入訂閱者入密碼");
        }
        else if (iptext.getText().length() == 0) {
            display("請輸入伺服器 IP-Port 網址");
        }
        else if (topictext.getText().length() == 0) {
            display("請選擇訂閱的主題");
        }
    }
    else {

```

```

        checkid();
    }
}
else if (button.equals("fitem3")) { //停止接收
    if (topictext.getText().equals("大行情")) {
        exitConnection_gs();
    }
    else {
        exitConnection();
    }
    mfile.savereaddate();
    viewxmlbuffer.insert(0,"<?xml version=\"1.0\" encoding=\"BIG5\"?>\n" + "<" + choiceTopic + ">\n");
    viewxmlbuffer.append("</" + choiceTopic + ">\n");
    bar.fitem3.setEnabled(false);
    bar.bt3.setEnabled(false);
    bar.oitem1.setEnabled(true);
    bar.bt4.setEnabled(true);
    bar.vitem1.setEnabled(true);
    bar.bt6.setEnabled(true);
}
else if (button.equals("oitem1")) { //轉至資料庫
    insertDataIntoDB();
    bar.oitem2.setEnabled(true);
    bar.bt5.setEnabled(true);
    bar.oitem1.setEnabled(false);
    bar.bt4.setEnabled(false);
}
else if (button.equals("oitem2")) { //儲存 XML 文件
    savexmlfile();
    bar.fitem4.setEnabled(true);
    bar.fitem1.setEnabled(true);
    bar.fitem2.setEnabled(true);
    bar.bt1.setEnabled(true);
    bar.bt2.setEnabled(true);
    bar.oitem2.setEnabled(false);
    bar.bt5.setEnabled(false);
}
else if (button.equals("oitem3")) {
    open_frame(fm_sql);
}
else if (button.equals("vitem1")) {
    open_viewxml();
}
else if (button.equals("hitem1")) {
    open_frame(explain_html);
}
else if (button.equals("hitem2")) {
    open_frame(about);
}
else if (button.equals("fitem4")) {
    System.exit(0);
}
}

public void itemStateChanged(ItemEvent item) {
    if (item.getSource() == parametersetup.choice1) {
        iptext.setText(parametersetup.iptext.getText());
        topictext.setText("小行情");
        Queue = "SampleQ1";
        save_xsl = "<?xml-stylesheet type=\"text/xsl\" href=\"individual.xml\"?>";
    }
}

```





```

choiceTopic = "供應人小行情";
endcolumn = "總金額";
stringhead = new StringBuffer("Insert into 供應人小行情 (");
length1 = stringhead.length();
scroll.getViewPort().add(stable.columnname(1), null);
}
else if (item.getSource() == parametersetup.choice2) {
    topictext.setText("大行情");
    iptext.setText(parametersetup.iptext.getText());
    App_Topic = "Global_Statement";
    choiceTopic = "大行情";
    endcolumn = "最低價";
    stringhead = new StringBuffer("Insert into 供應人大行情 (");
    length1 = stringhead.length();
    scroll.getViewPort().add(stable.columnname(2), null);
}
else if (item.getSource() == parametersetup.choice3) {
    iptext.setText(parametersetup.iptext.getText());
    topictext.setText("匯款清單");
    choiceTopic = "供應人匯款清單";
    endcolumn = "總金額";
    Queue = "SampleQ3";
    stringhead = new StringBuffer("Insert into 供應人匯款清單 (");
    length1 = stringhead.length();
    scroll.getViewPort().add(stable.columnname(3), null);
}
}
}

public void checkid() {
    int port = 1024;
    String result = null;
    try {
        String add = iptext.getText().substring(0, iptext.getText().indexOf(":"));
        InetAddress addr = InetAddress.getByName(add);
        Socket socket = new Socket(addr, port);
        PrintWriter s_out = new PrintWriter(socket.getOutputStream());
        BufferedReader s_in = new BufferedReader(new InputStreamReader(socket.getInputStream()));
        s_out.println("供應人登入");
        s_out.println(idtext.getText());
        s_out.println(pwtext.getText());
        s_out.flush();
        result = s_in.readLine();
        if (result.equals("訂閱者身份驗證成功")) {
            display(result);
            bar.fitem1.setEnabled(false);
            bar.fitem2.setEnabled(false);
            bar.fitem4.setEnabled(false);
            bar.vitem1.setEnabled(false);
            bar.bt1.setEnabled(false);
            bar.bt2.setEnabled(false);
            bar.bt4.setEnabled(false);
            bar.bt6.setEnabled(false);
            bar.fitem3.setEnabled(true);
            bar.bt3.setEnabled(true);
            if (topictext.getText().equals("大行情")) {
                setUpSubscriber_gs(iptext.getText(), idtext.getText().toUpperCase(), pwtext.getText(), idtext.getText());
            }
        }
        else {
            setUpSubscriber(iptext.getText(), idtext.getText().toUpperCase(), pwtext.getText(), idtext.getText());
        }
    }
}

```



```

    }
    else {
        display(result);
        bar.fitem1.setEnabled(true);
        bar.fitem2.setEnabled(true);
        bar.fitem4.setEnabled(true);
        bar.bt1.setEnabled(true);
        bar.bt2.setEnabled(true);
        bar.fitem3.setEnabled(false);
        bar.bt3.setEnabled(false);
    }
    socket.close();
}
catch (IOException e) {
    display(result);
    bar.fitem1.setEnabled(true);
    bar.fitem2.setEnabled(true);
    bar.fitem4.setEnabled(true);
    bar.bt1.setEnabled(true);
    bar.bt2.setEnabled(true);
    bar.fitem3.setEnabled(false);
    bar.bt3.setEnabled(false);
    display("建立身份確認連線失敗");
    e.printStackTrace();
}
}
}

public void display(String str) { //監控系統狀況方法
    monitor.setText("系統監控中心：" + str);
}

public void get_parameter() {
    idtext.setText(parametersetup.idtext.getText().toUpperCase());
    pwtext.setText(parametersetup.pwtext.getText());
}

public void setUpSubscriber(String address, String username, String password,String selection) {
    try {
        javax.jms.QueueConnectionFactory factory;
        factory = (new progress.message.jclient.QueueConnectionFactory(address));
        connect = factory.createQueueConnection(username, password);
        receiveSession = connect.createQueueSession(false,javax.jms.Session.AUTO_ACKNOWLEDGE);
        javax.jms.Queue receiveQueue = receiveSession.createQueue(Queue);
        javax.jms.QueueReceiver qReceiver = receiveSession.createReceiver(receiveQueue,
        PROPERTY_NAME + "=" + selection.toUpperCase() + "");
        qReceiver.setMessageListener(this);
        connect.start();
        viewxmlbuffer.delete(0, viewxmlbuffer.length());
        display("爪哇訊息服務者連線成功，準備開始接收訊息");
    }
    catch (javax.jms.JMSEException e) {
        display("爪哇訊息服務者連線失敗，請再嘗試連線一次");
        bar.fitem1.setEnabled(true);
        bar.fitem2.setEnabled(true);
        bar.bt1.setEnabled(true);
        bar.bt2.setEnabled(true);
        bar.fitem3.setEnabled(false);
        bar.bt3.setEnabled(false);
        e.printStackTrace();
    }
}
}

```



```

}

public void setUpSubscriber_gs(String address, String username,String password, String selection) {
    try {
        javax.jms.TopicConnectionFactory factory_gs = (new progress.message.
            jclient.TopicConnectionFactory(address));
        connect_gs = factory_gs.createTopicConnection(username, password);
        connect_gs.setClientID(username);
        session_gs = connect_gs.createTopicSession(false,javax.jms.Session.AUTO_ACKNOWLEDGE);
        javax.jms.Topic topic_gs = session_gs.createTopic(App_Topic);
        subscriber_gs = session_gs.createDurableSubscriber(topic_gs,"Global_statement");
        subscriber_gs.setMessageListener(this);
        connect_gs.start();
        viewxmlbuffer.delete(0, viewxmlbuffer.length());
        display("爪哇訊息服務者連線成功，準備開始接收訊息");
    }
    catch (javax.jms.JMSEException e) {
        display("爪哇訊息服務者連線失敗，請再嘗試連線一次");
        bar.fitem1.setEnabled(true);
        bar.fitem2.setEnabled(true);
        bar.bt1.setEnabled(true);
        bar.bt2.setEnabled(true);
        bar.fitem3.setEnabled(false);
        bar.bt3.setEnabled(false);
        e.printStackTrace();
    }
}

public void onMessage(javax.jms.Message aMessage) {
    try {
        if (aMessage instanceof progress.message.jclient.XMLMessage) {
            progress.message.jclient.XMLMessage xmlMessage = (progress.message.
                jclient.XMLMessage) aMessage;
            org.w3c.dom.Document doc = xmlMessage.getDocument();
            node = doc.getDocumentElement();
            printTree(node);
            if (topictext.getText().equals("小行情")) {
                iscount++;
            }
            else if (topictext.getText().equals("大行情")) {
                gscount++;
            }
            else if (topictext.getText().equals("匯款清單")) {
                sascount++;
            }
            display("總共接收到：" + iscount + "筆小行情資料；" + gscount + "筆大行情資料；" + sascount +
                "筆匯款清單資料");
        }
    }
    catch (javax.jms.JMSEException jmse) {
        display("花卉交易訊息接收出現錯誤");
        jmse.printStackTrace();
    }
    catch (java.lang.RuntimeException rte) {
        rte.printStackTrace();
    }
}

public void exitConnection() { //中斷爪哇訊息服務者連線
    try {

```

```

        connect.close();
        display("停止接收訊息資料，總共接收到：" + iscount + "筆小行情資料；" + gscount + "筆大行情資料；"
            + sascount + "筆匯款清單資料");
    }
    catch (javax.jms.JMSEException jexception) {
        jexception.printStackTrace();
    }
}

```

```

public void exitConnection_gs() { //中斷爪哇訊息服務者連線
    try {
        connect_gs.close();
        display("停止接收訊息資料，總共接收到：" + iscount + "筆小行情資料；" + gscount + "筆大行情資料；"
            + sascount + "筆匯款清單資料");
    }
    catch (javax.jms.JMSEException jexception) {
        jexception.printStackTrace();
    }
}

```

```

public void insertDataIntoDB() { //轉進 Access 資料庫
    String url = "jdbc:odbc:subscriber_db";
    Connection con = null;
    Statement statement = null;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        con = DriverManager.getConnection(url, "", "");
        statement = con.createStatement();
        for (Enumeration enum = sqlvector.elements(); enum.hasMoreElements();) {
            statement.executeUpdate(enum.nextElement().toString()); //執行 SQL 語法
        }
        statement.close(); //關閉連線
        con.close();
        sqlvector.clear();
        for (int rowcount = 0; rowcount < stable.def.getRowCount(); rowcount++) {
            stable.def.removeRow(rowcount);
            stable.def.setRowCount(rowcount);
        }
        display("花卉交易資料轉進資料庫成功");
    }
    catch (ClassNotFoundException e) {
        display("花卉交易資料轉進資料庫失敗 JDBC 或是 SQL 語法有問題");
        e.printStackTrace();
    }
    catch (SQLException sqle) {
        display("花卉交易資料轉進資料庫失敗 JDBC 或是 SQL 語法有問題");
        sqle.printStackTrace();
    }
}

```

```

public void printTree(Node node) { //解析 XML 文件的 Tree
    switch (node.getNodeType()) {
        case Node.DOCUMENT_TYPE_NODE:
            break;
        case Node.ELEMENT_NODE:
            if (node.getNodeName().equalsIgnoreCase(choiceTopic)) {
                break;
            }
            else if (node.getNodeName().equals("交易筆")) {
                tablerow.removeAllElements();
            }
    }
}

```

```

        savexml doc.append("<" + node.getNodeName() + ">\n");
        viewxmlbuffer.append("<" + node.getNodeName() + ">\n");
    }
    else if (node.getNodeName().equals(endcolumn)) {
        Node node2 = node.getFirstChild();
        table row.addElement(node2.getNodeValue());
        stable.def.addRow(table row.toArray());
        if (!node2.getNodeValue().endsWith("(結束資料)")) {
            stringhead.append(node.getNodeName() + " ");
            stringvalue.append(" " + node2.getNodeValue() + " ");
            sqlvector.addElement(stringhead.toString() + stringvalue.toString()); // SQL 語法轉成 String 放進 Vector
        }
        savexml doc.append("<" + node.getNodeName() + ">" + node2.getNodeValue() +
            "</" + node.getNodeName() + ">\n"); //存成 XML 文件檔
        savexml doc.append("</交易筆>\n");
        viewxmlbuffer.append("<" + node.getNodeName() + ">" + node2.getNodeValue() + "</" +
            node.getNodeName() + ">\n");
        viewxmlbuffer.append("</交易筆>\n");
        int end = stringvalue.length(); //清空 StringBuffer 的內容
        int headend = stringhead.length();
        stringvalue.delete(slength2, end);
        stringhead.delete(slength1, headend);
    }
}
else {
    Node node2 = node.getFirstChild();
    table row.addElement(node2.getNodeValue()); //架構顯示在表格的陣列
    stringhead.append(node.getNodeName() + ","); //架構轉進資料庫 SQL 語法
    stringvalue.append(" " + node2.getNodeValue() + " "); //架構轉進資料庫 SQL 語法
    savexml doc.append("<" + node.getNodeName() + ">" + node2.getNodeValue() +
        "</" + node.getNodeName() + ">\n"); //存成 XML 文件檔
    viewxmlbuffer.append("<" + node.getNodeName() + ">" +
        node2.getNodeValue() + "</" + node.getNodeName() + ">\n");
}
} //end of switch
for (Node child = node.getFirstChild(); child != null;
    child = child.getNextSibling()) {
    printTree(child);
}
}
}

```

```

public void savexmlfile() { //花卉交易資料存成一份 XML 文件檔
    String savefiledate = null;
    Calendar right = Calendar.getInstance(); //存檔日期
    int year = right.get(Calendar.YEAR) - 1911;
    int month = right.get(Calendar.MONTH) + 1;
    int day = right.get(Calendar.DAY_OF_MONTH);
    savefiledate = Integer.toString(year) + "_" + Integer.toString(month) + "_" + Integer.toString(day);
    File xmlfile = new File(savefiledate + choiceTopic + ".xml");
    try { //以日期做為檔名尚未完成
        if (!xmlfile.exists())
            xmlfile.createNewFile();
        BufferedWriter output = new BufferedWriter(new FileWriter(xmlfile));
        if (choiceTopic.equals("大行情")) {
            savexml doc.insert(0, "<?xml version='1.0' encoding='BIG5'?>\n" + "<?xml-stylesheet type='text/xsl' "
                href="global.xsl"?>\n" + "<" + choiceTopic + ">\n");
        }
        else if (choiceTopic.equals("供應人小行情")) {
            savexml doc.insert(0, "<?xml version='1.0' encoding='BIG5'?>\n" + "<?xml-stylesheet type='text/xsl' "
                href="individual.xsl"?>\n" + "<" + choiceTopic + ">\n");
        }
    }
}

```



```

    }
    else {
        savexml.doc.insert(0,"<?xml version=\"1.0\" encoding=\"BIG5\"?>\n" + "<?xml-stylesheet type=\"text/xsl\"
        href=\"sas.xsl\"?>\n" + "<" + choiceTopic + ">\n");
    }
    savexml.doc.append("</" + choiceTopic + ">\n");
    output.write(savexml.doc.toString());
    output.flush();
    display(savefiledate + choiceTopic + ".xml 文件產生成功");
    savexml.doc.delete(0, savexml.doc.length());
}
catch (IOException e) {
    display("寫入檔案有錯誤");
    System.out.print(e.getMessage());
}
}
}

```

```

public void open_frame(JFrame showframe) {
    boolean packFrame = false;
    if (packFrame) {
        showframe.pack();
    }
    else {
        showframe.validate();
    }
    Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
    Dimension frameSize = showframe.getSize();
    if (frameSize.height > screenSize.height) {
        frameSize.height = screenSize.height;
    }
    if (frameSize.width > screenSize.width) {
        frameSize.width = screenSize.width;
    }
    showframe.setLocation( (screenSize.width - frameSize.width) / 2,(screenSize.height - frameSize.height) / 2);
    showframe.setVisible(true);
}

```

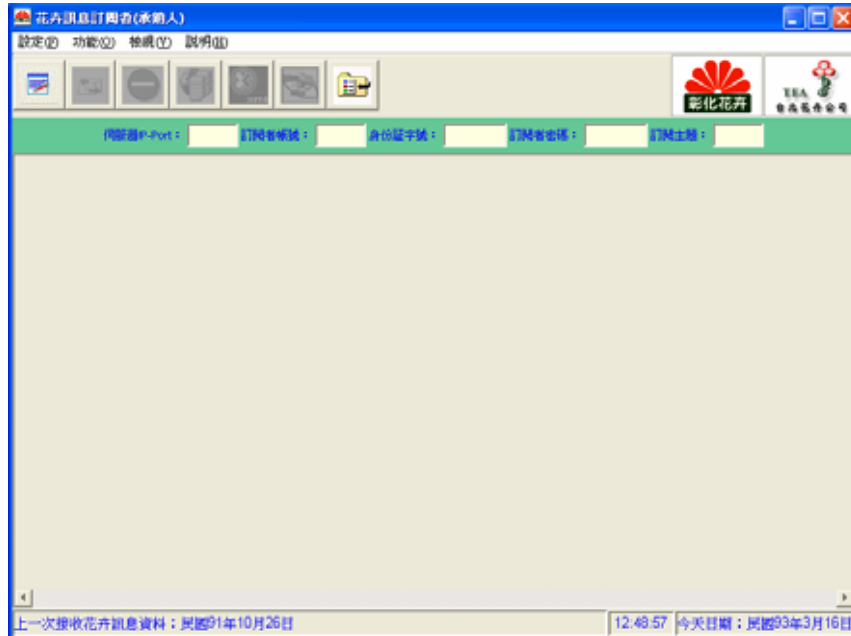
```

public void open_viewxml() {
    boolean viewxmlframe = false;
    if (viewxmlframe) {
        view.pack();
    }else {
        view.validate();
    }
    Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
    Dimension viewxmlsize = view.getSize();
    if (viewxmlsize.height > screenSize.height) {
        viewxmlsize.height = screenSize.height;
    }
    if (viewxmlsize.width > screenSize.width) {
        viewxmlsize.width = screenSize.width;
    }
    view.setLocation( (screenSize.width - viewxmlsize.width) / 2,(screenSize.height - viewxmlsize.height) / 2);
    view.setVisible(true);
    view.xmlcontent.setText(viewxmlbuffer.toString());
    view.statusbar.setText("檢視花卉交易資料 XML 文件");
}
}
}

```

#### 附錄四：承銷人軟體設計程式碼

fm\_subscriber.java 為承銷人軟體主要程式，包含主要 GUI 介面的呈現、使用者訊息接收、資料庫儲存、及 XML 文件剖析與儲存。其主畫面如下：



附圖 2 fm\_subscriber\_buyer.java 主要畫面

檔名：fm\_subscriber\_buyer.java



```
package Flower_TIPS;
```

```
import java.io.*;
import java.util.*;
import java.net.*;
import java.awt.*;
import java.sql.*;
import org.w3c.dom.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.border.*;
import javax.swing.table.DefaultTableModel;
```

```
public class fm_subscriber_buyer extends JFrame implements javax.jms.MessageListener, ItemListener, ActionListener
{
    ImageIcon title_icon = new ImageIcon("icon/ftp.gif");
    JPanel contentPane;
    JPanel allpanel = new JPanel();
    JPanel statuspanel = new JPanel(); //存放狀態列
    JPanel timepanel = new JPanel(); //時間
    JPanel panel = new JPanel(); //只放置參數設定訊息
    JPanel toolbarpanel = new JPanel(); //放置 Toolbar 與 logo 的 panel
    JPanel logopanel = new JPanel();
    JLabel logo = new JLabel(new ImageIcon("icon/mark.gif"));
    JLabel logo2 = new JLabel(new ImageIcon("icon/mark2.gif"));
    GridLayout gridLayout1 = new GridLayout();
```

```

JLabel ip = new JLabel();
JTextField iptext = new JTextField(6);
JLabel id = new JLabel();
JTextField idtext = new JTextField(6);
JLabel topic = new JLabel();
JLabel ident = new JLabel();
JPasswordField identtext = new JPasswordField(10);
JTextField topictext = new JTextField(6);
JLabel pw = new JLabel();
JPasswordField pwtext = new JPasswordField(10);
JScrollPane scroll = new JScrollPane();
JLabel monitor = new JLabel();
static JLabel time = new JLabel();
JLabel date = new JLabel();
FlowLayout flowLayout1 = new FlowLayout();
FlowLayout flowLayout2 = new FlowLayout();
menubar bar = new menubar(); //功能列
parameter_setup_buyer parametersetup = new parameter_setup_buyer(); //參數設定
viewxml view = new viewxml(); //看 XML 文件
explain explain_html = new explain(); //操作說明
About about = new About(); //關於此軟體
StringBuffer viewxmlbuffer = new StringBuffer();
//Flower_TIPS 變數設定
private static final String DEFAULT_BROKER_NAME = "localhost:2506";
private static final String PROPERTY_NAME = "Department";
private static final String DEFAULT_PASSWORD = "password";
private static final int message_lifespan = 180000; // milliseconds (30 minutes)
private javax.jms.QueueConnection connect = null;
private javax.jms.QueueSession sendSession = null;
private javax.jms.QueueSession receiveSession = null;
private javax.jms.QueueSender sender = null;
private static String App_Topic;
private static String client_id;
javax.jms.TopicConnection connect_gs = null;
javax.jms.TopicSession session_gs = null;
javax.jms.TopicSubscriber subscriber_gs = null;
//SQL 語法變數設定
static int gscount = 0;
static int bascount = 0;
StringBuffer stringhead;
StringBuffer stringvalue = new StringBuffer("values (");
Vector sqlvector = new Vector(); //存放 SQL 語法的 Vector
int slength1;
int slength2 = stringvalue.length(); //用來刪除 SQL 語法時，所使用的變數
//表格顯示所使用到變數
showtable stable = new showtable(); //表格顯示設定
Vector tablerow = new Vector(10); //表格每一 Row 的內容
org.w3c.dom.Node node;
//花卉交易資料存成 XML 文件檔所需變數
StringBuffer savexml doc = new StringBuffer();
String choiceTopic; // 判斷選擇訂閱主題
String endcolumn;
//存檔所使用的 Class
messagefile mfile;
fm_SQL_buyer fm_sql = new fm_SQL_buyer();

//Construct the frame
public fm_subscriber_buyer() {
    enableEvents(AWTEvent.WINDOW_EVENT_MASK);
    try {

```

```

    mfile = new messagefile();
    jbInit();
}
catch (Exception e) {
    e.printStackTrace();
}
}

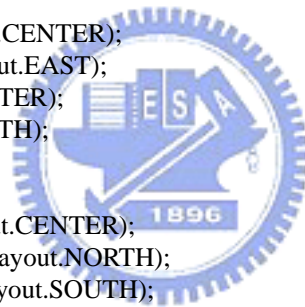
//Component initialization
private void jbInit() throws Exception {
    setJMenuBar(bar.buildmenubar());
    contentPane = (JPanel)this.getContentPane();
    contentPane.setLayout(new BorderLayout());
    allpanel.setLayout(new BorderLayout());
    allpanel.setSize(new Dimension(800, 600));
    allpanel.setBackground(new Color(100, 200, 150));
    toolbarpanel.setLayout(new BorderLayout());
    statuspanel.setLayout(new BorderLayout());
    flowLayout2.setHgap(1);
    flowLayout2.setVgap(1);
    timepanel.setLayout(flowLayout2);
    this.setSize(new Dimension(800, 600));
    this.setResizable(true);
    this.setIconImage(title_icon.getImage());
    this.setTitle("花卉訊息訂閱者(承銷人)");
    scroll.getViewports().setAutoscrolls(true);
    scroll.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);
    monitor.setFont(new java.awt.Font("Dialog", 0, 13));
    monitor.setText("請輸入連線設定值");
    monitor.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
    monitor.setForeground(Color.BLUE);
    time.setForeground(Color.BLUE);
    time.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
    time.setFont(new java.awt.Font("Dialog", 0, 13));
    date.setForeground(Color.BLUE);
    date.setBorder(BorderFactory.createBevelBorder(BevelBorder.LOWERED));
    date.setText(mfile.todaydate());
    date.setFont(new java.awt.Font("Dialog", 0, 13));
    panel.setBounds(new Rectangle(1, 0, 772, 30));
    panel.setLayout(flowLayout1);
    ip.setForeground(Color.blue);
    ip.setHorizontalAlignment(SwingConstants.CENTER);
    ip.setText("伺服器 IP-Port : ");
    iptext.setBackground(SystemColor.info);
    id.setText("訂閱者帳號 : ");
    id.setHorizontalAlignment(SwingConstants.CENTER);
    id.setForeground(Color.blue);
    idtext.setBackground(SystemColor.info);
    ident.setText("身份証字號 : ");
    ident.setForeground(Color.blue);
    ident.setHorizontalAlignment(SwingConstants.CENTER);
    identtext.setBackground(SystemColor.info);
    pw.setText("訂閱者密碼 : ");
    pw.setHorizontalAlignment(SwingConstants.CENTER);
    pw.setForeground(Color.blue);
    pwtext.setBackground(SystemColor.info);
    topic.setForeground(Color.blue);
    topic.setHorizontalAlignment(SwingConstants.CENTER);
    topic.setText("訂閱主題 : ");
    topictext.setBackground(SystemColor.info);

```

```

iptext.setEditable(false);
idtext.setEditable(false);
identtext.setEditable(false);
pwtext.setEditable(false);
topictext.setEditable(false);
FlowLayout1.setHgap(3);
logo.setBorder(BorderFactory.createEtchedBorder());
logo2.setBorder(BorderFactory.createEtchedBorder());
logopanel.setLayout(new GridLayout());
logopanel.add(logo);
logopanel.add(logo2);
toolbarpanel.add(logopanel, BorderLayout.EAST);
toolbarpanel.add(bar.buildtoolbar(), BorderLayout.CENTER);
panel.add(ip, null);
panel.add(iptext, null);
panel.add(id, null);
panel.add(idtext, null);
panel.add(ident, null);
panel.add(identtext, null);
panel.add(pw, null);
panel.add(pwtext, null);
panel.add(topic, null);
panel.add(topictext, null);
panel.setOpaque(false);
timepanel.add(time, null);
timepanel.add(date, null);
statuspanel.add(monitor, BorderLayout.CENTER);
statuspanel.add(timepanel, BorderLayout.EAST);
allpanel.add(scroll, BorderLayout.CENTER);
allpanel.add(panel, BorderLayout.NORTH);
scroll.revalidate();
monitor.setText(mfile.lastread());
contentPane.add(allpanel, BorderLayout.CENTER);
contentPane.add(toolbarpanel, BorderLayout.NORTH);
contentPane.add(statuspanel, BorderLayout.SOUTH);
//選單按鈕動作
bar.fitem1.addActionListener(this);
bar.fitem2.addActionListener(this);
bar.fitem3.addActionListener(this);
bar.fitem4.addActionListener(this);
bar.oitem1.addActionListener(this);
bar.oitem2.addActionListener(this);
bar.oitem3.addActionListener(this);
bar.vitem1.addActionListener(this);
bar.hitem1.addActionListener(this);
bar.hitem2.addActionListener(this);
bar.bt1.addActionListener(this);
bar.bt2.addActionListener(this);
bar.bt3.addActionListener(this);
bar.bt4.addActionListener(this);
bar.bt5.addActionListener(this);
bar.bt6.addActionListener(this);
bar.bt7.addActionListener(this);
//參數按鈕動作
parametersetup.choice2.addItemListener(this);
parametersetup.choice4.addItemListener(this);
parametersetup.bt1.addActionListener(this);
parametersetup.bt2.addActionListener(this);
}

```





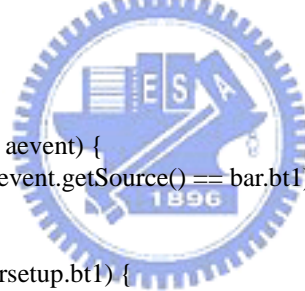
```

//Overridden so we can exit when window is closed
protected void processWindowEvent(WindowEvent e) {
    super.processWindowEvent(e);
    if (e.getID() == WindowEvent.WINDOW_CLOSING) {
        System.exit(0);
    }
}

public void itemStateChanged(ItemEvent item) {
    if (item.getSource() == parametersetup.choice2) {
        topictext.setText("大行情");
        App_Topic = "Global_Statement";
        choiceTopic = "大行情";
        endcolumn = "最低價";
        scroll.getViewPort().add(stable.columnname(2));
        stringhead = new StringBuffer("Insert into 承銷人大行情 (");
        slength1 = stringhead.length();
        scroll.getViewPort().add(stable.columnname(2), null);
    }
    else if (item.getSource() == parametersetup.choice4) {
        topictext.setText("結帳清單");
        App_Topic = "Buyer_Account_Statement";
        choiceTopic = "承銷人結帳清單";
        endcolumn = "總金額";
        scroll.getViewPort().add(stable.columnname(4));
        stringhead = new StringBuffer("Insert into 承銷人結帳清單 (");
        slength1 = stringhead.length();
    }
} //End itemStateChanged method

public void actionPerformed(ActionEvent aevent) {
    if (aevent.getSource() == bar.fitem1 || aevent.getSource() == bar.bt1) { //參數設定
        open_frame(parametersetup);
    }
    else if (aevent.getSource() == parametersetup.bt1) {
        parametersetup.setVisible(false);
        get_parameter();
        bar.fitem2.setEnabled(true);
        bar.bt2.setEnabled(true);
    }
    else if (aevent.getSource() == parametersetup.bt2) {
        parametersetup.idtext.setText("");
        parametersetup.identtext.setText("");
        parametersetup.pwtext.setText("");
    }
    else if (aevent.getSource() == bar.fitem2 || aevent.getSource() == bar.bt2) { //啟動接收
        buttonaction("fitem2");
    }
    else if (aevent.getSource() == bar.fitem3 || aevent.getSource() == bar.bt3) { //停止接收
        buttonaction("fitem3");
    }
    else if (aevent.getSource() == bar.oitem1 || aevent.getSource() == bar.bt4) {
        buttonaction("oitem1");
    }
    else if (aevent.getSource() == bar.oitem2 || aevent.getSource() == bar.bt5) {
        buttonaction("oitem2");
    }
    else if (aevent.getSource() == bar.oitem3 || aevent.getSource() == bar.bt7) {
        buttonaction("oitem3");
    }
}

```



```

else if (aevent.getSource() == bar.vitem1 || aevent.getSource() == bar.bt6) {
    buttonaction("vitem1");
}
else if (aevent.getSource() == bar.hitem1) {
    buttonaction("hitem1");
}
else if (aevent.getSource() == bar.hitem2) {
    buttonaction("hitem2");
}
else if (aevent.getSource() == bar.fitem4) {
    buttonaction("fitem4");
}
} //End actionPerformed method

public void buttonaction(String button) {
    if (button.equals("fitem2")) {
        if (idtext.getText().length() == 0) { //檢驗帳號密碼是否有輸入
            display("請輸入訂閱者帳號 ");
        }
        else if (pwtext.getText().length() == 0) {
            display("請輸入訂閱者入密碼");
        }
        else if (iptext.getText().length() == 0) {
            display("請輸入伺服器 IP-Port 網址");
        }
        else if (topictext.getText().length() == 0) {
            display("請選擇訂閱的主題");
        }
        else {
            checkid();
        }
    }
    else if (button.equals("fitem3")) { //停止接收
        if (topictext.getText().equals("大行情")) {
            exitConnection_gs();
        }
        else {
            exitConnection();
        }
        mfile.savereaddate();
        viewxmlbuffer.insert(0, "<?xml version=\"1.0\" encoding=\"BIG5\"?>\n" + "<" + choiceTopic + ">\n");
        viewxmlbuffer.append("<" + choiceTopic + ">\n");
        bar.fitem3.setEnabled(false);
        bar.bt3.setEnabled(false);
        bar.oitem1.setEnabled(true);
        bar.bt4.setEnabled(true);
        bar.vitem1.setEnabled(true);
        bar.bt6.setEnabled(true);
    }
    else if (button.equals("oitem1")) { //轉至資料庫
        insertDataIntoDB();
        bar.oitem2.setEnabled(true);
        bar.bt5.setEnabled(true);
        bar.oitem1.setEnabled(false);
        bar.bt4.setEnabled(false);
    }
    else if (button.equals("oitem2")) { //儲存 XML 文件
        savexmlfile();
        bar.fitem4.setEnabled(true);
        bar.fitem1.setEnabled(true);
    }
}

```



```

        bar.fitem2.setEnabled(true);
        bar.bt1.setEnabled(true);
        bar.bt2.setEnabled(true);
        bar.oitem2.setEnabled(false);
        bar.bt5.setEnabled(false);
    }
    else if (button.equals("oitem3")) {
        open_frame(fm_sql);
    }
    else if (button.equals("vitem1")) {
        open_viewxml();
    }
    else if (button.equals("hitem1")) {
        open_frame(explain_html);
    }
    else if (button.equals("hitem2")) {
        open_frame(about);
    }
    else if (button.equals("fitem4")) {
        System.exit(0);
    }
}

public void checkid() {
    int port = 1024;
    String result = null;
    try {
        String add = iptext.getText().substring(0, iptext.getText().indexOf(":"));
        InetAddress addr = InetAddress.getByName(add);
        Socket socket = new Socket(addr, port);
        PrintWriter s_out = new PrintWriter(socket.getOutputStream());
        BufferedReader s_in = new BufferedReader(new InputStreamReader(socket.getInputStream()));
        if (parametersetup.market1.isSelected()) {
            s_out.println("台北承銷人登入");
        }
        else {
            s_out.println("彰化承銷人登入");
        }
        s_out.println(idtext.getText());
        s_out.println(identtext.getText());
        s_out.println(pwtext.getText());
        s_out.flush();
        result = s_in.readLine();
        if (result.equals("訂閱者身份驗證成功")) {
            display(result);
            bar.fitem1.setEnabled(false);
            bar.fitem2.setEnabled(false);
            bar.fitem4.setEnabled(false);
            bar.vitem1.setEnabled(false);
            bar.bt1.setEnabled(false);
            bar.bt2.setEnabled(false);
            bar.bt4.setEnabled(false);
            bar.bt6.setEnabled(false);
            bar.fitem3.setEnabled(true);
            bar.bt3.setEnabled(true);
            if (topictext.getText().equals("大行情")) {
                setUpSubscriber_gs(iptext.getText(), idtext.getText().toUpperCase(), pwtext.getText(), idtext.getText());
            }
        }
        else {
            setUpSubscriber(iptext.getText(), idtext.getText().toUpperCase(), pwtext.getText(), idtext.getText());
        }
    }
}

```

```

    }
}
else {
    display(result);
    bar.fitem1.setEnabled(true);
    bar.fitem2.setEnabled(true);
    bar.fitem4.setEnabled(true);
    bar.bt1.setEnabled(true);
    bar.bt2.setEnabled(true);
    bar.fitem3.setEnabled(false);
    bar.bt3.setEnabled(false);
}
socket.close();
}
catch (IOException e) {
    display(result);
    bar.fitem1.setEnabled(true);
    bar.fitem2.setEnabled(true);
    bar.fitem4.setEnabled(true);
    bar.bt1.setEnabled(true);
    bar.bt2.setEnabled(true);
    bar.fitem3.setEnabled(false);
    bar.bt3.setEnabled(false);
    display("建立身份確認連線失敗");
    e.printStackTrace();
}
}
}

public void get_parameter() {
    iptext.setText(parametersetup.iptext.getText());
    idtext.setText(parametersetup.idtext.getText().toUpperCase());
    pwtext.setText(parametersetup.pwtext.getText());
    identtext.setText(parametersetup.identtext.getText());
}

public void display(String str) { //監控系統狀況方法
    monitor.setText("系統監控中心：" + str);
}

public void setUpSubscriber(String address, String username, String password,String selection) {
    try {
        String queue = null;
        javax.jms.QueueConnectionFactory factory;
        factory = (new progress.message.jclient.QueueConnectionFactory(address));
        connect = factory.createQueueConnection(username, password);
        receiveSession = connect.createQueueSession(false,javax.jms.Session.AUTO_ACKNOWLEDGE);
        if (parametersetup.market2.isSelected()) {
            queue = "SampleQ2"; //彰化承銷人結帳清單
        }
        else if (parametersetup.market1.isSelected()) {
            queue = "SampleQ4"; //台北承銷人結帳清單
        }
        javax.jms.Queue receiveQueue = receiveSession.createQueue(queue);
        javax.jms.QueueReceiver qReceiver = receiveSession.createReceiver(receiveQueue,
        PROPERTY_NAME + "=" + selection.toUpperCase() + "");
        qReceiver.setMessageListener(this);
        connect.start();
        viewxmlbuffer.delete(0, viewxmlbuffer.length());
        display("爪哇訊息服務者連線成功，準備開始接收訊息");
    }
}

```



```

catch (javax.jms.JMSEException e) {
    display("爪哇訊息服務者連線失敗，請再嘗試連線一次");
    bar.fitem1.setEnabled(true);
    bar.fitem2.setEnabled(true);
    bar.bt1.setEnabled(true);
    bar.bt2.setEnabled(true);
    bar.fitem3.setEnabled(false);
    bar.bt3.setEnabled(false);
    e.printStackTrace();
}
}

public void setUpSubscriber_gs(String address, String username, String password, String selection) {
    try {
        javax.jms.TopicConnectionFactory factory_gs = (new progress.message.
            jclient.TopicConnectionFactory(address));
        if (parametersetup.market1.isSelected()) {
            username = "T_" + username;
        } //台北承銷人大行情
        else {
            username = "C_" + username;
        } //彰化承銷人大行情
        connect_gs = factory_gs.createTopicConnection(username, password);
        connect_gs.setClientID(username);
        session_gs = connect_gs.createTopicSession(false, javax.jms.Session.AUTO_ACKNOWLEDGE);
        javax.jms.Topic topic_gs = session_gs.createTopic(App_Topic);
        subscriber_gs = session_gs.createDurableSubscriber(topic_gs, "Global_statement");
        subscriber_gs.setMessageListener(this);
        connect_gs.start();
        viewxmlbuffer.delete(0, viewxmlbuffer.length());
        display("爪哇訊息服務者連線成功，準備開始接收訊息");
    }
    catch (javax.jms.JMSEException e) {
        display("爪哇訊息服務者連線失敗，請再嘗試連線一次");
        bar.fitem1.setEnabled(true);
        bar.fitem2.setEnabled(true);
        bar.bt1.setEnabled(true);
        bar.bt2.setEnabled(true);
        bar.fitem3.setEnabled(false);
        bar.bt3.setEnabled(false);
        e.printStackTrace();
    }
}

public void onMessage(javax.jms.Message amessage) {
    try {
        if (amessage instanceof progress.message.jclient.XMLMessage) {
            progress.message.jclient.XMLMessage xmlmessage = (progress.message.jclient.XMLMessage) amessage;
            org.w3c.dom.Document doc = xmlmessage.getDocument();
            node = doc.getDocumentElement();
            printTree(node);
            if (topictext.getText().equals("大行情")) {
                gscount++;
            }
            else if (topictext.getText().equals("結帳清單")) {
                bascount++;
            }
            display("總共接收到：" + gscount + "筆大行情資料；" + bascount + "筆結帳清單資料");
        }
    }
}

```



```

catch (javax.jms.JMSEException jmse) {
    display("花卉交易訊息接收出現錯誤");
    jmse.printStackTrace();
}
catch (java.lang.RuntimeException rte) {
    rte.printStackTrace();
}
}

public void exitConnection() { //中斷爪哇訊息服務者連線
    try {
        connect.close();
        display("停止接收訊息資料，總共接收到：" + gscount + "筆大行情資料；" + bascount + "筆結帳清單資料");
    } catch (javax.jms.JMSEException jexception) {
        jexception.printStackTrace();
    }
}

public void exitConnection_gs() { //中斷爪哇訊息服務者連線
    try {
        connect_gs.close();
        display("停止接收訊息資料，總共接收到：" + gscount + "筆大行情資料；" + bascount + "筆結帳清單資料");
    } catch (javax.jms.JMSEException jexception) {
        jexception.printStackTrace();
    }
}

public void insertDataIntoDB() { //轉進 Access 資料庫
    String url = "jdbc:odbc:subscriber_db";
    Connection con = null;
    Statement statement = null;
    try {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        con = DriverManager.getConnection(url, "", "");
        statement = con.createStatement();
        for (Enumeration enum = sqlvector.elements(); enum.hasMoreElements(); ) {
            statement.executeUpdate(enum.nextElement().toString()); //執行 SQL 語法
        }
        statement.close(); //關閉連線
        con.close();
        sqlvector.clear();
        for (int rowcount = 0; rowcount < stable.def.getRowCount(); rowcount++) {
            stable.def.removeRow(rowcount);
            stable.def.setRowCount(rowcount);
        }
        display("花卉交易資料轉進資料庫成功");
    }
    catch (ClassNotFoundException e) {
        display("花卉交易資料轉進資料庫失敗 JDBC 或是 SQL 語法有問題");
        e.printStackTrace();
    }
    catch (SQLException sqle) {
        display("花卉交易資料轉進資料庫失敗 JDBC 或是 SQL 語法有問題");
        sqle.printStackTrace();
    }
}

public void printTree(Node node) { //解析 XML 文件的 Tree

```

```

switch (node.getNodeType()) {
    case Node.DOCUMENT_TYPE_NODE:
        break;
    case Node.ELEMENT_NODE:
        if (node.getNodeName().equalsIgnoreCase(choiceTopic)) {
            break;
        }
        else if (node.getNodeName().equals("交易筆")) {
            table.removeAllElements();
            savexml.doc.append("<" + node.getNodeName() + ">\n");
            viewxmlbuffer.append("<" + node.getNodeName() + ">\n");
        }
        else if (node.getNodeName().equals(endcolumn)) {
            Node node2 = node.getFirstChild();
            table.addRow(node2.getNodeValue());
            stable.def.addRow(table.toArray());
            if (!node2.getNodeValue().endsWith("結束資料")) {
                stringhead.append(node.getNodeName() + " ");
                stringvalue.append(node2.getNodeValue() + " ");
                sqlvector.addElement(stringhead.toString() + stringvalue.toString()); // SQL 語法轉成 String 放進 Vector
            }
            savexml.doc.append("<" + node.getNodeName() + ">" + node2.getNodeValue() +
                "</" + node.getNodeName() + ">\n"); //存成 XML 文件檔
            savexml.doc.append("</交易筆>\n");
            viewxmlbuffer.append("<" + node.getNodeName() + ">" +
                node2.getNodeValue() + "</" + node.getNodeName() + ">\n");
            viewxmlbuffer.append("</交易筆>\n");
            int end = stringvalue.length(); //清空 StringBuffer 的內容
            int headend = stringhead.length();
            stringvalue.delete(slength2, end);
            stringhead.delete(slength1, headend);
        }else {
            Node node2 = node.getFirstChild();
            table.addRow(node2.getNodeValue()); //架構顯示在表格的陣列
            stringhead.append(node.getNodeName() + ","); //架構轉進資料庫 SQL 語法
            stringvalue.append(node2.getNodeValue() + ","); //架構轉進資料庫 SQL 語法
            savexml.doc.append("<" + node.getNodeName() + ">" + node2.getNodeValue() +
                "</" + node.getNodeName() + ">\n"); //存成 XML 文件檔
            viewxmlbuffer.append("<" + node.getNodeName() + ">" +
                node2.getNodeValue() + "</" + node.getNodeName() + ">\n");
        }
    } //end of switch
    for (Node child = node.getFirstChild(); child != null; child = child.getNextSibling()) {
        printTree(child);
    }
}

```

```

public void savexmlfile() { //花卉交易資料存成一份 XML 文件檔
    String savefiledate = null;
    Calendar right = Calendar.getInstance(); //存檔日期
    int year = right.get(Calendar.YEAR) - 1911;
    int month = right.get(Calendar.MONTH) + 1;
    int day = right.get(Calendar.DAY_OF_MONTH);
    savefiledate = Integer.toString(year) + "_" + Integer.toString(month) + "_" + Integer.toString(day);
    File xmlfile = new File(savefiledate + choiceTopic + ".xml");
    try { //以日期做為檔名尚未完成
        if (!xmlfile.exists())
            xmlfile.createNewFile();
        BufferedWriter output = new BufferedWriter(new FileWriter(xmlfile));
    }
}

```

```

if (choiceTopic.equals("大行情")) {
    savexml doc.insert(0, "<?xml version='1.0' encoding='BIG5'>\n" +
        "<?xml-stylesheet type='text/xsl' href='global.xml'>\n" + "<" + choiceTopic + ">\n");
}
else {
    savexml doc.insert(0, "<?xml version='1.0' encoding='BIG5'>\n" +
        "<?xml-stylesheet type='text/xsl' href='bas.xml'>\n" + "<" + choiceTopic + ">\n");
}
savexml doc.append("</" + choiceTopic + ">\n");
output.write(savexml doc.toString());
output.flush();
display(savefiledate + choiceTopic + ".xml 文件產生成功");
savexml doc.delete(0, savexml doc.length());
}
catch (IOException e) {
    display("寫入檔案有錯誤");
    System.out.print(e.getMessage());
}
}

public void open_frame(JFrame showframe) {
    boolean packFrame = false;
    if (packFrame) {
        showframe.pack();
    }else {
        showframe.validate();
    }
    Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
    Dimension frameSize = showframe.getSize();
    if (frameSize.height > screenSize.height) {
        frameSize.height = screenSize.height;
    }
    if (frameSize.width > screenSize.width) {
        frameSize.width = screenSize.width;
    }
    showframe.setLocation( (screenSize.width - frameSize.width) / 2,(screenSize.height - frameSize.height) / 2);
    showframe.setVisible(true);
}

public void open_viewxml() {
    boolean viewxmlframe = false;
    if (viewxmlframe) {
        view.pack();
    }else {
        view.validate();
    }
    Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
    Dimension viewxmlsize = view.getSize();
    if (viewxmlsize.height > screenSize.height) {
        viewxmlsize.height = screenSize.height;
    }
    if (viewxmlsize.width > screenSize.width) {
        viewxmlsize.width = screenSize.width;
    }
    view.setLocation( (screenSize.width - viewxmlsize.width) / 2,(screenSize.height - viewxmlsize.height) / 2);
    view.setVisible(true);
    view.xmlcontent.setText(viewxmlbuffer.toString());
    view.statusbar.setText("檢視花卉交易資料 XML 文件");
}
}
}

```



## 附錄五：參數設定設計程式碼

由於供應人軟體與承銷人軟體參數設定程式雷同，以下僅附上供應人參數設定軟體，使用者在設定好參數與訂閱者主題後方能與遠端伺服器建立連線，下圖為供應人軟體參數設定畫面。



附圖 3 parameter\_setup.java 程式畫面

檔名：parameter\_setup.java

```
package Flower_TIPS; //參數設定的 Frame

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.border.*;

public class parameter_setup extends JFrame {
    JPanel contentPane;
    JLabel logo = new JLabel();
    ImageIcon title_icon = new ImageIcon("icon/ftp.gif");
    ImageIcon imageicon = new ImageIcon("icon/logo.jpg");
    JPanel panelup = new JPanel();
    GridLayout gridLayout1 = new GridLayout();
    JLabel ip = new JLabel();
    JTextField iptext = new JTextField();
    JLabel id = new JLabel();
    JTextField idtext = new JTextField();
    JLabel pw = new JLabel();
    JPasswordField pwtext = new JPasswordField();
    JPanel paneltopic = new JPanel();
    JRadioButton choice1 = new JRadioButton();
    JRadioButton choice2 = new JRadioButton();
    JRadioButton choice3 = new JRadioButton();
    ButtonGroup buttonGroup1 = new ButtonGroup();
```

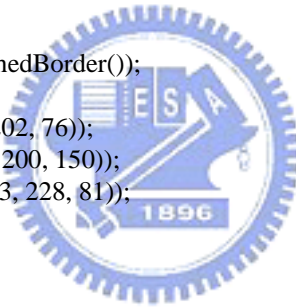
```

JButton bt1 = new JButton();
JButton bt2 = new JButton();
FlowLayout flowLayout1 = new FlowLayout();
String choice = null;
fm_subscriber flower_subscriber;

//Construct the frame
public parameter_setup() {
    enableEvents(AWTEvent.WINDOW_EVENT_MASK);
    try {
        jbInit();
    }
    catch (Exception e) {
        e.printStackTrace();
    }
}

//Component initialization
private void jbInit() throws Exception {
    this.setIconImage(title_icon.getImage());
    contentPane = (JPanel)this.getContentPane();
    contentPane.setLayout(null);
    this.getContentPane().setBackground(UIManager.getColor("ToolTip.background"));
    this.setSize(new Dimension(265, 319));
    this.setTitle("訂閱者參數設定");
    logo.setBackground(Color.white);
    logo.setForeground(Color.orange);
    logo.setBorder(BorderFactory.createEtchedBorder());
    logo.setIcon(imageicon);
    logo.setBounds(new Rectangle(26, 10, 202, 76));
    panelup.setBackground(new Color(100, 200, 150));
    panelup.setBounds(new Rectangle(14, 93, 228, 81));
    panelup.setLayout(gridLayout1);
    gridLayout1.setColumns(2);
    gridLayout1.setHgap(1);
    gridLayout1.setRows(3);
    gridLayout1.setVgap(1);
    ip.setForeground(Color.blue);
    ip.setHorizontalAlignment(SwingConstants.CENTER);
    ip.setText("伺服器 IP-Port : ");
    iptext.setFont(new java.awt.Font("SansSerif", 0, 14));
    iptext.setToolTipText("輸入伺服器 IP-Port");
    iptext.setText("140.113.58.163:2506");
    id.setForeground(Color.blue);
    id.setHorizontalAlignment(SwingConstants.CENTER);
    id.setText("訂閱者帳號 : ");
    idtext.setFont(new java.awt.Font("SansSerif", 0, 14));
    idtext.setToolTipText("請輸入訂閱者帳號");
    pw.setForeground(Color.blue);
    pw.setHorizontalAlignment(SwingConstants.CENTER);
    pw.setText("訂閱者密碼 : ");
    pwtext.setFont(new java.awt.Font("SansSerif", 0, 14));
    pwtext.setToolTipText("請輸入訂閱者密碼");
    paneltopic.setBackground(Color.pink);
    paneltopic.setAlignmentX( (float) 1.0);
    paneltopic.setAlignmentY( (float) 2.0);
    paneltopic.setBounds(new Rectangle(15, 181, 229, 62));
    paneltopic.setLayout(flowLayout1);
    paneltopic.setBackground(new Color(100, 200, 150));
    paneltopic.setFont(new java.awt.Font("Dialog", 0, 14));

```





```

paneltopic.setBorder(BorderFactory.createTitledBorder(BorderFactory.createLineBorder(Color.white, 2),
"選擇訂閱主題", TitledBorder.LEFT, TitledBorder.TOP, new Font("SansSerif", 0, 14), Color.blue));
paneltopic.setToolTipText("選擇訂閱主題");
choice1.setOpaque(false);
choice1.setText("小行情");
choice2.setAlignmentY( (float) 1.0);
choice2.setOpaque(false);
choice2.setText("大行情");
choice3.setOpaque(false);
choice3.setText("匯款清單");
bt1.setBounds(new Rectangle(17, 249, 114, 26));
bt1.setToolTipText("完成參數設定");
bt1.setText("完成參數設定");
bt2.setBounds(new Rectangle(146, 250, 97, 25));
bt2.setToolTipText("重設參數");
bt2.setText("重設參數");
panelup.add(ip, null);
panelup.add(iptext, null);
panelup.add(id, null);
panelup.add(idtext, null);
panelup.add(pw, null);
panelup.add(pwtext, null);
contentPane.add(paneltopic, null);
contentPane.add(panelup, null);
paneltopic.add(choice1, null);
paneltopic.add(choice2, null);
contentPane.add(bt1, null);
contentPane.add(bt2, null);
contentPane.add(logo, null);
buttonGroup1.add(choice1);
buttonGroup1.add(choice2);
buttonGroup1.add(choice3);
}

//Overridden so we can exit when window is closed
protected void processWindowEvent(WindowEvent e) {
    super.processWindowEvent(e);
    if (e.getID() == WindowEvent.WINDOW_CLOSING) {
        this.setVisible(false);
    }
}
}
}

```



## 附錄六：自動化檔案轉移服務程式碼

以下三個為檔案轉移服務程式碼，MainFrame.java 為系統主程式包含主要 GUI 介面與 FTP 主要程式碼，Detail.java 與 Set\_Auto.java 則為自動化設定程式，下圖左為 FTP 主要 GUI 介面，右為自動化設定畫面。



附圖 4 MainFrame.java 主畫面與自動化設定畫面

檔名：MainFrame.java

```
package tw_flowers;
```

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.border.*;
import javax.swing.event.*;
import java.util.*;
import java.io.*;
import java.net.*;
import java.lang.reflect.*;
// FTP Client class
import sun.net.*;
import sun.net.ftp.*;
```

```
public class MainFrame extends JFrame {
    JList list = new JList(new DefaultListModel());
    FileDialog fd = new FileDialog(this, "選擇欲傳送檔案", FileDialog.LOAD);
    FtpClient ftpclient;
    String filechosen = "";
    int chosen_number = 0;
    ImageIcon ftp = new ImageIcon("image/ftp.gif");
    ImageIcon tw_icon = new ImageIcon("image/tw_icon.gif");
    ImageIcon tw_icon_2 = new ImageIcon("image/tw_icon.gif");
    ImageIcon red_ball = new ImageIcon("image/icon_2.gif");
    ImageIcon blue_ball = new ImageIcon("image/icon_1.gif");
    ImageIcon stop = new ImageIcon("image/stop.gif");
    JPanel jPanel1 = new JPanel();
    JPanel jPanel4 = new JPanel();
```



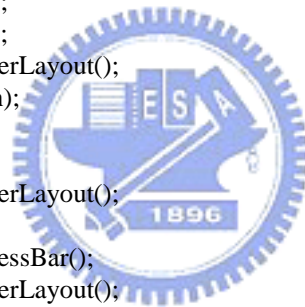
```

JPanel jPanel2 = new JPanel();
JPanel jPanel3 = new JPanel();
JPanel jPanel5 = new JPanel();
JLabel jLabel1 = new JLabel();
JLabel jLabel2 = new JLabel();
GridLayout gridLayout1 = new GridLayout();
JPanel jPanel6 = new JPanel();
BorderLayout BorderLayout1 = new BorderLayout();
JLabel jLabel3 = new JLabel();
BorderLayout BorderLayout2 = new BorderLayout();
JPanel jPanel7 = new JPanel();
JScrollPane jScrollPane1 = new JScrollPane();
JTextArea jTextArea1 = new JTextArea();
JPanel jPanel8 = new JPanel();
GridLayout gridLayout2 = new GridLayout();
JLabel jLabel4 = new JLabel();
JLabel jLabel5 = new JLabel();
JLabel jLabel6 = new JLabel();
JComboBox jComboBox1 = new JComboBox();
JTextField jTextField1 = new JTextField();
JPasswordField jPasswordField1 = new JPasswordField();
BorderLayout BorderLayout4 = new BorderLayout();
JButton jButton1 = new JButton(red_ball);
JButton jButton2 = new JButton(red_ball);
JButton jButton3 = new JButton(red_ball);
JButton jButton4 = new JButton(red_ball);
JButton jButton5 = new JButton(red_ball);
BorderLayout BorderLayout6 = new BorderLayout();
JButton jButton77 = new JButton(tw_icon);
TitledBorder titledBorder1;
TitledBorder titledBorder2;
BorderLayout BorderLayout5 = new BorderLayout();
JPanel jPanel9 = new JPanel();
JProgressBar jProgressBar1 = new JProgressBar();
BorderLayout BorderLayout3 = new BorderLayout();
BorderLayout BorderLayout7 = new BorderLayout();
TitledBorder titledBorder3;
JPanel jPanel10 = new JPanel();
BorderLayout BorderLayout8 = new BorderLayout();
JScrollPane jScrollPane2 = new JScrollPane();
JTextField jTextField2 = new JTextField();
Detail detail;
int hour, minute, second;
boolean time_flag = false;
boolean auto_flag = false;
boolean upload_end_flag = false;
String upload_time_string = "";

public MainFrame() {
    try {
        jbInit();
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        time_thread.start();
    }
    catch (Exception e) {
        e.printStackTrace();
    }
}

public static void main(String[] args) {

```



```

MainFrame mainFrame = new MainFrame();
//Center the window
Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
Dimension frameSize = mainFrame.getSize();
if (frameSize.height > screenSize.height) {
    frameSize.height = screenSize.height;
}
if (frameSize.width > screenSize.width) {
    frameSize.width = screenSize.width;
}
mainFrame.setLocation( (screenSize.width - frameSize.width) / 2,
                       (screenSize.height - frameSize.height) / 2);
}

private void jbInit() throws Exception {
    UIManager.setLookAndFeel("com.sun.java.swing.plaf.windows.WindowsLookAndFeel");
    this.setSize(new Dimension(520, 390));
    this.setTitle(" NCTU  FTP");
    this.setIconImage(ftp.getImage());
    detail = new Detail();
    titledBorder1 = new TitledBorder(BorderFactory.createEtchedBorder(Color.white
, new Color(70, 140, 105)), " Detail ");
    titledBorder2 = new TitledBorder("");
    titledBorder3 = new TitledBorder("");
    titledBorder4 = new TitledBorder("");
    this.getContentPane().setLayout(borderLayout6);
    jPanel1.setBackground(new Color(100, 200, 150));
    jPanel1.setBorder(BorderFactory.createRaisedBevelBorder());
    jPanel1.setPreferredSize(new Dimension(424, 95));
    jPanel1.setLayout(borderLayout4);
    jPanel4.setLayout(borderLayout2);
    jPanel2.setBorder(BorderFactory.createEtchedBorder());
    jPanel2.setBackground(new Color(100, 200, 150));
    jPanel2.setBorder(null);
    jPanel2.setDebugGraphicsOptions(0);
    jPanel2.setLayout(borderLayout5);
    jPanel3.setBackground(Color.lightGray);
    jPanel3.setAlignmentX( (float) 0.5);
    jPanel3.setBorder(BorderFactory.createLineBorder(Color.black));
    jPanel3.setPreferredSize(new Dimension(100, 100));
    jPanel3.setLayout(gridLayout3);
    jLabel1.setBackground(new Color(100, 200, 150));
    jLabel1.setFont(new java.awt.Font("Dialog", 0, 11));
    jLabel1.setForeground(Color.black);
    jPanel5.setLayout(gridLayout1);
    jLabel1.setBorder(BorderFactory.createLoweredBevelBorder());
    jLabel1.setMaximumSize(new Dimension(80, 21));
    jLabel1.setPreferredSize(new Dimension(50, 30));
    jLabel2.setBackground(new Color(100, 200, 150));
    jLabel2.setFont(new java.awt.Font("Dialog", 0, 11));
    jLabel2.setBorder(BorderFactory.createLoweredBevelBorder());
    jLabel2.setDebugGraphicsOptions(0);
    jLabel2.setMaximumSize(new Dimension(80, 21));
    jLabel2.setPreferredSize(new Dimension(50, 30));
    gridLayout1.setColumns(2);
    gridLayout1.setHgap(0);
    gridLayout1.setVgap(0);
    jPanel6.setLayout(borderLayout1);
    jLabel3.setBackground(new Color(100, 200, 150));
    jLabel3.setFont(new java.awt.Font("Monospaced", 0, 11));

```

```

jLabel3.setBorder(BorderFactory.createLoweredBevelBorder());
jLabel3.setPreferredSize(new Dimension(50, 21));
jLabel3.setText(" 訊息監控中心 : ");
jPanel7.setBackground(Color.lightGray);
jPanel7.setEnabled(true);
jPanel7.setDebugGraphicsOptions(0);
jPanel7.setLayout(borderLayout7);
jTextArea1.setBorder(null);
jTextArea1.setText("Message...");
jPanel8.setBackground(new Color(100, 200, 150));
jPanel8.setFont(new java.awt.Font("Dialog", 0, 11));
jPanel8.setBorder(titledBorder1);
jPanel8.setMinimumSize(new Dimension(149, 98));
jPanel8.setPreferredSize(new Dimension(285, 69));
jPanel8.setLayout(gridLayout2);
jLabel4.setBackground(new Color(255, 170, 255));
jLabel4.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel4.setForeground(Color.darkGray);
jLabel4.setText("          IP 位址  :");
gridLayout2.setColumns(0);
gridLayout2.setHgap(1);
gridLayout2.setRows(3);
gridLayout2.setVgap(1);
jLabel5.setBackground(new Color(255, 170, 255));
jLabel5.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel5.setForeground(Color.darkGray);
jLabel5.setText("          使用者帳號 :");
jLabel6.setBackground(new Color(255, 170, 255));
jLabel6.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel6.setForeground(Color.darkGray);
jLabel6.setText("          使用者密碼  :");
jTextField1.setBackground(Color.white);
jTextField1.setAlignmentX( (float) 0.5);
jTextField1.setText("yyyyyy");
jPasswordField1.setText("");
jButton1.setBackground(new Color(100, 200, 150));
jButton1.setFont(new java.awt.Font("Dialog", 0, 11));
jButton1.setForeground(Color.darkGray);
jButton1.setPreferredSize(new Dimension(73, 25));
jButton1.setToolTipText("連線主機");
jButton1.setHorizontalTextPosition(SwingConstants.TRAILING);
jButton1.setIcon(red_ball);
jButton1.setRolloverEnabled(true);
jButton1.setRolloverIcon(blue_ball);
jButton1.setText("連線");
jButton1.setIconTextGap(4);
jButton1.addActionListener(new MainFrame_jButton1_actionAdapter(this));
jButton2.setBackground(new Color(100, 200, 150));
jButton2.setFont(new java.awt.Font("Dialog", 0, 11));
jButton2.setForeground(Color.darkGray);
jButton2.setAlignmentX( (float) 0.0);
jButton2.setMaximumSize(new Dimension(81, 27));
jButton2.setMinimumSize(new Dimension(81, 27));
jButton2.setToolTipText("選擇檔案");
jButton2.setActionCommand("選擇");
jButton2.setHorizontalAlignment(SwingConstants.CENTER);
jButton2.setHorizontalTextPosition(SwingConstants.TRAILING);
jButton2.setRolloverEnabled(true);
jButton2.setRolloverIcon(blue_ball);
jButton2.setText("選擇");

```



```

jButton2.addActionListener(new MainFrame_jButton2_actionAdapter(this));
jButton3.setBackground(new Color(100, 200, 150));
jButton3.setFont(new java.awt.Font("Dialog", 0, 11));
jButton3.setForeground(Color.darkGray);
jButton3.setToolTipText("中斷連線");
jButton3.setRolloverEnabled(true);
jButton3.setRolloverIcon(blue_ball);
jButton3.setText("斷線");
jButton3.addActionListener(new MainFrame_jButton3_actionAdapter(this));
jButton4.setBackground(new Color(100, 200, 150));
jButton4.setFont(new java.awt.Font("Dialog", 0, 11));
jButton4.setForeground(Color.darkGray);
jButton4.setToolTipText("關閉應用程式");
jButton4.setMargin(new Insets(2, 14, 2, 14));
jButton4.setRolloverEnabled(true);
jButton4.setRolloverIcon(blue_ball);
jButton4.setText("結束");
jButton4.addActionListener(new MainFrame_jButton4_actionAdapter(this));
jButton5.setBackground(new Color(100, 200, 150));
jButton5.setFont(new java.awt.Font("Dialog", 0, 11));
jButton5.setForeground(Color.darkGray);
jButton5.setToolTipText("上傳至主機");
jButton5.setRolloverEnabled(true);
jButton5.setRolloverIcon(blue_ball);
jButton5.setText("上傳");
jButton5.addActionListener(new MainFrame_jButton5_actionAdapter(this));
jPanel5.setBackground(new Color(100, 200, 150));
jPanel5.setPreferredSize(new Dimension(200, 30));
jButton77.setBackground(new Color(100, 200, 150));
jButton77.setBorder(BorderFactory.createLineBorder(Color.black));
jButton77.setPreferredSize(new Dimension(200, 17));
jButton77.setRequestFocusEnabled(true);
jButton77.setHorizontalTextPosition(SwingConstants.LEFT);
jButton77.setRolloverIcon(tw_icon_2);
jButton77.setIconTextGap(2);
jButton77.addActionListener(new MainFrame_jButton77_actionAdapter(this));
jButton77.setText("");
jPanel4.setBackground(new Color(100, 200, 150));
this.getContentPane().setBackground(new Color(100, 200, 150));
jPanel6.setBackground(new Color(100, 200, 150));
jPanel6.setPreferredSize(new Dimension(50, 21));
jScrollPane1.getViewPort().setBackground(Color.lightGray);
addItem();
jComboBox1.addActionListener(new MainFrame_jComboBox1_actionAdapter(this));
jProgressBar1.setForeground(new Color(120, 220, 170));
jProgressBar1.setStringPainted(true);
borderLayout5.setHgap(0);
borderLayout5.setVgap(0);
jPanel9.setBackground(new Color(100, 200, 150));
jPanel9.setLayout(borderLayout3);
jProgressBar1.setBackground(new Color(100, 200, 150));
jProgressBar1.setEnabled(true);
jProgressBar1.setFont(new java.awt.Font("DialogInput", 0, 11));
jProgressBar1.setBorder(BorderFactory.createRaisedBevelBorder());
jProgressBar1.setPreferredSize(new Dimension(100, 20));
jProgressBar1.setRequestFocusEnabled(true);
jPanel10.setBackground(new Color(100, 200, 150));
jPanel10.setPreferredSize(new Dimension(200, 15));
jPanel10.setLayout(borderLayout8);
list.setFont(new java.awt.Font("Monospaced", 0, 12));

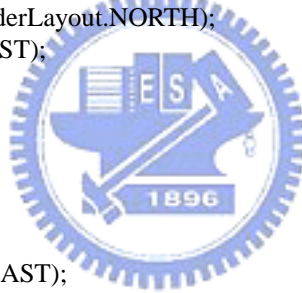
```



```

list.setBorder(null);
list.setSelectionBackground(Color.pink);
list.setSelectionForeground(Color.darkGray);
list.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
list.addMouseListener(new MainFrame_list_mouseAdapter(this));
jTextField2.setBackground(new Color(100, 200, 150));
jTextField2.setBorder(BorderFactory.createRaisedBevelBorder());
jTextField2.setDoubleBuffered(false);
jTextField2.setPreferredSize(new Dimension(57, 20));
jTextField2.setText("jTextField2");
titledBorder1.setTitle(" 連線資訊 ");
titledBorder1.setTitleFont(new java.awt.Font("Dialog", 0, 11));
titledBorder1.setTitleColor(Color.black);
jCheckBox1.setBackground(new Color(100, 200, 150));
jCheckBox1.setFont(new java.awt.Font("Dialog", 0, 11));
jCheckBox1.setForeground(Color.red);
jCheckBox1.setToolTipText("自動");
jCheckBox1.setBorderPainted(true);
jCheckBox1.setText(" 全自動");
jCheckBox1.addActionListener(new MainFrame_jCheckBox1_actionAdapter(this));
jComboBox1.setAlignmentX( (float) 0.5);
gridLayout3.setColumns(1);
gridLayout3.setHgap(1);
gridLayout3.setRows(0);
gridLayout3.setVgap(1);
titledBorder2.setTitleFont(new java.awt.Font("Dialog", 0, 11));
this.getContentPane().add(jPanel1, BorderLayout.NORTH);
jPanel1.add(jPanel8, BorderLayout.WEST);
jPanel8.add(jLabel4, null);
jPanel8.add(jComboBox1, null);
jPanel8.add(jLabel5, null);
jPanel8.add(jTextField1, null);
jPanel8.add(jLabel6, null);
jPanel8.add(jPasswordField1, null);
jPanel1.add(jButton77, BorderLayout.EAST);
this.getContentPane().add(jPanel4, BorderLayout.SOUTH);
jPanel4.add(jPanel5, BorderLayout.EAST);
jPanel5.add(jLabel2, null);
jPanel5.add(jLabel1, null);
jPanel4.add(jPanel6, BorderLayout.CENTER);
jPanel6.add(jLabel3, BorderLayout.CENTER);
this.getContentPane().add(jPanel2, BorderLayout.CENTER);
jPanel2.add(jPanel3, BorderLayout.WEST);
jPanel2.add(jPanel7, BorderLayout.CENTER);
jPanel7.add(jScrollPane1, BorderLayout.CENTER);
jPanel7.add(jPanel9, BorderLayout.NORTH);
jPanel9.add(jProgressBar1, BorderLayout.CENTER);
jPanel2.add(jPanel10, BorderLayout.EAST);
jPanel10.add(jTextField2, BorderLayout.NORTH);
jPanel3.add(jButton1, null);
jPanel3.add(jButton2, null);
jPanel3.add(jButton5, null);
jPanel3.add(jButton3, null);
jPanel3.add(jButton4, null);
jPanel3.add(jCheckBox1, null);
jTextField2.setText("NCTU : \\");
this.setVisible(true);
jScrollPane2.getViewPort().add(list);
SwingUtilities.updateComponentTreeUI(this);
jScrollPane1.getViewPort().add(jTextArea1, null);

```





```

String s = br.readLine();
while (s != null) {
    jComboBox1.addItem(s);
    s = br.readLine();
}
br.close();
}
catch (IOException ioe) {}
}

void listFTPFile() {
try {
    String token = "";
    String allfile = "";
    DefaultListModel listModel = new DefaultListModel();
    listModel.addElement("..");
    //將屬性為資料夾的檔案先行列出//
    TelnetInputStream input = ftpclient.list();
    BufferedReader br = new BufferedReader(new InputStreamReader(input));
    while (allfile != null) {
        if (allfile.startsWith("d")) {
            StringTokenizer st = new StringTokenizer(allfile);
            while (st.hasMoreTokens()) {
                token = st.nextToken();
            }
            if (!(token.equalsIgnoreCase(".") || token.equalsIgnoreCase(".."))) {
                listModel.addElement(token + " : \\");
            }
        }
        allfile = br.readLine();
    }
    input.close();
    br.close();
    // 將其他檔案列出
    TelnetInputStream input2 = ftpclient.list();
    BufferedReader br2 = new BufferedReader(new InputStreamReader(input2));
    allfile = br2.readLine();
    while (allfile != null) {
        if (!allfile.startsWith("d")) {
            StringTokenizer st = new StringTokenizer(allfile);
            while (st.hasMoreTokens()) {
                token = st.nextToken();
            }
            listModel.addElement(token);
        }
        allfile = br2.readLine();
    }
    input2.close();
    br2.close();
    list.setModel(listModel);
}
catch (IOException ie) {
    jTextArea1.setText(jTextArea1.getText() + "\n" + ie.getMessage());
    jLabel3.setText(" 訊息監控中心 : 整理遠端目錄失敗 !!");
}
}

void chooseFile() {
    String s = "";
    try {

```



```

        BufferedReader br = new BufferedReader(new FileReader(new File(
            "default_path.txt")));
        s = br.readLine();
        br.close();
    }
    catch (IOException ioe) {
        error();
    }
    fd.setDirectory(s);
    fd.setVisible(true);
    filechosen += (fd.getDirectory() + fd.getFile() + "\n");
    jTextArea1.setText("已選取 :" + "\n" + filechosen);
    jLabel3.setText(" 訊息監控中心 : 請選取檔案 ");
}

void enter() {
    String enterDir;
    enterDir = list.getSelectedValue().toString();
    if (enterDir.endsWith("\\")) {
        enterDir = enterDir.substring(0, enterDir.length() - 4);
    }
    else if (enterDir.equalsIgnoreCase("..")) {
        enterDir = "..";
    }
    else {
        enterDir = "";
    }
    try {
        ftpclient.cd(enterDir);
        if (!enterDir.equalsIgnoreCase("..")) {
            jTextField2.setText(jTextField2.getText() + "\\" + enterDir);
        }
        else if (enterDir.equalsIgnoreCase("..")) {
            jTextField2.setText(jTextField2.getText().substring(0,
                jTextField2.getText().lastIndexOf("\\")));
        }
        listFTPFile();
    }
    catch (Exception e) {
        error();
        jTextArea1.setText(jTextArea1.getText() + "\n" + e.getMessage());
        jLabel3.setText(" 訊息監控中心 : 整理遠端目錄失敗 !!");
    }
}

void upload() {
    this.setCursor(Cursor.getPredefinedCursor(Cursor.WAIT_CURSOR));
    String success_file = ""; //測試成功上傳的檔案(success_file),已選欲上傳的檔案(filechosen)
    int success_number = 0;
    StringTokenizer st = new StringTokenizer(filechosen, "\n");
    while (st.hasMoreTokens()) {
        File f = new File(st.nextToken());
        if (f.isFile()) {
            jLabel3.setText(" 訊息監控中心 : 正在上傳" + f.getName() + ", 請稍後...");
            try {
                ftpclient.binary();
                int filesize = (int) f.length();
                jProgressBar1.setMaximum(filesize);
                java.io.FileInputStream inFile = new FileInputStream(f);
                sun.net.TelnetOutputStream outFile = ftpclient.put(f.getName());

```



```

success_number = 0;
jLabel3.setText(" 訊息監控中心 : 結束上傳 !");
this.setCursor(Cursor.getPredefinedCursor(Cursor.DEFAULT_CURSOR));
upload_end_flag = true;
}

boolean haveAnyFile() {
    boolean flag = false;
    try {
        File dir = new File(this.detail.local_path);
        String[] files = dir.list();
        if ( (files.length != 0) &&
            (this.detail.set_auto.jRadioButton1.isSelected()) ) {
            flag = true;
        }
    }
    catch (Exception e) {}
    return flag;
}

void auto_run() {
    Thread t = new Thread() {
        public void run() {
            try {
                connect();
                BufferedReader br2 = new BufferedReader(new FileReader(new File(
                    "remote_path.txt")));
                ftpclient.cd(br2.readLine());
                br2.close();
                BufferedReader br = new BufferedReader(new FileReader(new File(
                    "default_path.txt")));
                File dir = new File(br.readLine());
                br.close();
                //將資料夾內全部的檔案列出(全選)
                String[] files = dir.list();
                jTextArea1.setText(jTextArea1.getText() + "\n" + "已選擇下列檔案 :");
                for (int i = 0; i < files.length; i++) {
                    filechosen += (dir.getAbsolutePath() + "\\ " + files[i] + "\n");
                }
                jTextArea1.setText(jTextArea1.getText() + "\n" + filechosen);
                upload();
                jTextArea1.setText(jTextArea1.getText() + "\n" +
                    "最後上傳時間 :\n" + jLabel2.getText() + " " +
                    jLabel1.getText());
            }
            catch (IOException ioe) {
                jTextArea1.setText(jTextArea1.getText() + "\n 自動處理失敗 : IOException");
                jLabel3.setText(" 訊息監控中心 : 自動處理失敗 !!");
            }
            catch (Exception ex) {
                jTextArea1.setText(jTextArea1.getText() + "\n 自動處理失敗 :");
                jLabel3.setText(" 訊息監控中心 : 自動處理失敗 !!");
            }
            finally {
                try {
                    ftpclient.closeServer();
                }
                catch (IOException ioe) {}
            }
        }
    };
}

```



```

    }
};
t.start();
}

```

```

boolean compareTime() {
    time_flag = false;
    if (this.detail.set_auto.jRadioButton2.isSelected() ||
        detail.set_auto.jRadioButton3.isSelected()) {
        try {
            int[] auto_time = {
                Integer.parseInt(this.detail.auto_hour),
                Integer.parseInt(this.detail.auto_minute)};
            if ( (this.second == 0) && (this.hour == auto_time[0]) &&
                (this.minute == auto_time[1])) {
                time_flag = true;
            }
        }
        catch (Exception ioe) {}
    }
    return time_flag;
}

```

```

private Thread time_thread = new Thread() {
    public void run() {
        while (true) {
            try {
                Thread.sleep(1000);
                Calendar rightnow = Calendar.getInstance();
                hour = rightnow.get(Calendar.HOUR_OF_DAY);
                minute = rightnow.get(Calendar.MINUTE);
                second = rightnow.get(Calendar.SECOND);
                String[] day = {
                    "日", "一", "二", "三", "四", "五", "六"};
                jLabel1.setText("    " + hour + " : " +
                    minute
                    + " : " + second);
                jLabel2.setText("    " + (rightnow.get(Calendar.MONTH) + 1)
                    + " / " + rightnow.get(Calendar.DATE) + " ( " +
                    day[rightnow.get(Calendar.DAY_OF_WEEK) - 1] + " )"
                    );
                //設定即時上傳
                if (auto_flag && haveAnyFile() && upload_end_flag) { //自動+有檔案+上傳是否告一段落
                    upload_end_flag = false;
                    try {
                        jTextArea1.setText("等待檔案轉換中...");
                        Thread.sleep(6000); //先等待 60 秒,等檔案轉完才開始自動化
                        auto_run();
                    }
                    catch (InterruptedException ie) {}
                }
                //設定時間自動上傳
                if (auto_flag && compareTime()) {
                    jTextArea1.setText("等待檔案轉換中...");
                    Thread.sleep(6000);
                    auto_run();
                }
            }
            catch (InterruptedException ie) {
                jTextArea1.setText(ie.toString());
            }
        }
    }
}

```

```

        }
        catch (Exception e) {
            jTextArea1.setText(e.toString());
        }
    }
}
};
JCheckBox jCheckBox1 = new JCheckBox();
TitledBorder titledBorder4;
GridLayout gridLayout3 = new GridLayout();

void jButton1_actionPerformed(ActionEvent e) {
    Thread t = new Thread() {
        public void run() {
            connect();
            try {
                BufferedReader br = new BufferedReader(new FileReader(new File(
                    "remote_path.txt")));
                ftpclient.cd(br.readLine());
                listFTPFile();
                br.close();
            }
            catch (IOException ioe) {
                ioe.printStackTrace();
            }
            catch (Exception e) {
                e.printStackTrace();
            }
        }
    };
    t.start();
}

void jButton2_actionPerformed(ActionEvent e) {
    jTextArea1.setText("");
    Thread t = new Thread() {
        public void run() {
            chooseFile();
        }
    };
    t.start();
}

void jButton5_actionPerformed(ActionEvent e) {
    Thread t = new Thread() {
        public void run() {
            upload();
            filechosen = "";
        }
    };
    t.start();
}

void jButton4_actionPerformed(ActionEvent e) {
    Thread t = new Thread() {
        public void run() {
            System.exit(0);
        }
    };
}

```



```

    t.start();
}

void jButton3_actionPerformed(ActionEvent e) {
    Thread t = new Thread() {
        public void run() {
            try {
                ftpclient.closeServer();
                list.setEnabled(false);
                jTextArea1.setText(jTextArea1.getText() + "\n" + "已中斷連線 !");
                jLabel3.setText(" 訊息監控中心 : 連線結束 !");
            }
            catch (Exception ec) {
                jTextArea1.setText(jTextArea1.getText() + "\n" + ec.getMessage());
                jLabel3.setText(" 訊息監控中心 :");
            }
        }
    };
    t.start();
}

void jComboBox1_actionPerformed(ActionEvent e) {
}

void jCheckBox1_actionPerformed(ActionEvent e) {
    if (jCheckBox1.isSelected()) {
        jButton1.setEnabled(false);
        jButton2.setEnabled(false);
        jButton3.setEnabled(false);
        jButton5.setEnabled(false);
        auto_flag = true;
        upload_end_flag = true;
    }
    else {
        jButton1.setEnabled(true);
        jButton2.setEnabled(true);
        jButton3.setEnabled(true);
        jButton5.setEnabled(true);
        auto_flag = false;
        upload_end_flag = false;
    }
}

void jButton77_actionPerformed(ActionEvent e) {
    detail.setVisible(true);
}

void list_mouseClicked(MouseEvent e) {
    enter();
}
}

```



檔名：Detail.java

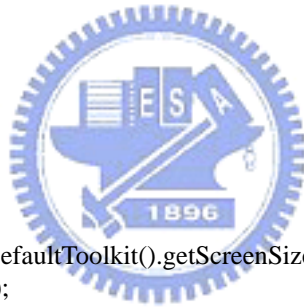
```
package tw_flowers;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import java.util.*;

public class Detail extends JFrame {
    JTabbedPane jTabbedPane1 = new JTabbedPane();
    JPanel jPanel1 = new JPanel();
    JButton jButton1 = new JButton();
    JButton jButton2 = new JButton();
    BorderLayout borderLayout1 = new BorderLayout();
    GridLayout gridLayout1 = new GridLayout();
    Set_Auto set_auto = new Set_Auto();
    Set_About set_about = new Set_About();
    ImageIcon blue_ball = new ImageIcon("image/icon_1.gif");
    ImageIcon red_ball = new ImageIcon("image/icon_2.gif");
    ImageIcon detail = new ImageIcon("image/detail.gif");
    static String local_path = "";
    static String auto_hour = "";
    static String auto_minute = "";

    public Detail() {
        try {
            jbInit();
            init_set();
            this.setSize(400, 300);
            //Center the window
            Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();
            Dimension frameSize = this.getSize();
            if (frameSize.height > screenSize.height) {
                frameSize.height = screenSize.height;
            }
            if (frameSize.width > screenSize.width) {
                frameSize.width = screenSize.width;
            }
            this.setLocation( (screenSize.width - frameSize.width) / 2,
                (screenSize.height - frameSize.height) / 2);
        }
        catch (Exception e) {
            e.printStackTrace();
        }
    }

    private void jbInit() throws Exception {
        this.setIconImage(detail.getImage());
        this.setTitle(" 相關設定");
        this.getContentPane().setLayout(borderLayout1);
        jPanel1.setLayout(gridLayout1);
        jButton1.setBackground(new Color(90, 190, 140));
        jButton1.setFont(new java.awt.Font("Dialog", 0, 11));
        jButton1.setIcon(red_ball);
        jButton1.setRolloverIcon(blue_ball);
        jButton1.setText("全部重設");
        jButton1.addActionListener(new Detail_jButton1_actionAdapter(this));
    }
}
```



```

jButton2.setBackground(new Color(90, 190, 140));
jButton2.setFont(new java.awt.Font("Dialog", 0, 11));
jButton2.setMaximumSize(new Dimension(73, 27));
jButton2.setPreferredSize(new Dimension(73, 40));
jButton2.setIcon(red_ball);
jButton2.setRolloverIcon(blue_ball);
jButton2.setText("確定");
jButton2.addActionListener(new Detail_jButton2_actionAdapter(this));
this.getContentPane().setBackground(new Color(90, 190, 140));
this.setFont(new java.awt.Font("Dialog", 0, 11));
this.setForeground(Color.darkGray);
this.setState(Frame.NORMAL);
jPanel1.setBackground(Color.white);
jPanel1.setFont(new java.awt.Font("Dialog", 0, 11));
jPanel1.setForeground(Color.darkGray);
jPanel1.setPreferredSize(new Dimension(146, 40));
jTabbedPane1.setBackground(Color.white);
jTabbedPane1.setFont(new java.awt.Font("Dialog", 0, 11));
jTabbedPane1.setForeground(Color.darkGray);
set_auto.setBackground(new Color(90, 190, 140));
set_auto.setFont(new java.awt.Font("Dialog", 0, 11));
set_auto.setForeground(Color.darkGray);
set_about.setBackground(new Color(90, 190, 140));
set_about.setFont(new java.awt.Font("Dialog", 0, 11));
set_about.setForeground(Color.darkGray);
this.getContentPane().add(jPanel1, BorderLayout.SOUTH);
jPanel1.add(jButton2, null);
jPanel1.add(jButton1, null);
this.getContentPane().add(jTabbedPane1, BorderLayout.CENTER);
jTabbedPane1.addTab("設定自動化", set_auto);
jTabbedPane1.addTab("關於", set_about);
}

```

```

void init_set() { // 自動化初始化設定
try {
    BufferedReader br = new BufferedReader(new FileReader(new File(
        "default_path.txt")));
    local_path = br.readLine();
    this.set_auto(jTextField1.setText(local_path);
    br.close();

    BufferedReader br2 = new BufferedReader(new FileReader(new File(
        "remote_path.txt")));
    this.set_auto(jTextField2.setText(br2.readLine());
    br2.close();

    BufferedReader br3 = new BufferedReader(new FileReader(new File(
        "upload_time.txt")));
    StringTokenizer st = new StringTokenizer(br3.readLine(), ":");
    auto_hour = st.nextToken();
    auto_minute = st.nextToken();

    this.set_auto.jComboBox1.setSelectedItem(auto_hour);
    this.set_auto.jComboBox2.setSelectedItem(auto_minute);
} catch (IOException ioe) {}
}

```

```

void save() { // 儲存自動化設定
try {
    String s = this.set_auto(jTextField1.getText());

```

```

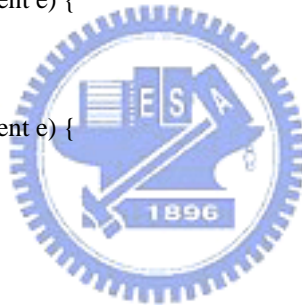
BufferedWriter bw = new BufferedWriter(new FileWriter(new File(
    "default_path.txt")));
bw.write(s);
local_path = s;
bw.close();
String s2 = this.set_auto.jTextField2.getText();
BufferedWriter bw2 = new BufferedWriter(new FileWriter(new File(
    "remote_path.txt")));
bw2.write(s2);
bw2.close();
int[] s3 = this.set_auto.getTime();
BufferedWriter bw3 = new BufferedWriter(new FileWriter(new File("upload_time.txt")));
auto_hour = String.valueOf(s3[0]);
auto_minute = String.valueOf(s3[1]);
bw3.write(auto_hour + ":" + auto_minute);
bw3.close();
} catch (IOException ioe) {}
this.setVisible(false);
}

void reset() {
    this.set_auto.jTextField1.setText("C:\\nctu_ftp");
    this.set_auto.jTextField2.setText("tw_ftp");
}

void jButton2_actionPerformed(ActionEvent e) {
    save();
}

void jButton1_actionPerformed(ActionEvent e) {
    reset();
}
}

```



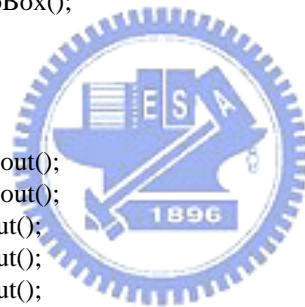


檔名：Set\_Auto.java

```
package tw_flowers;

import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.io.*;
import java.lang.Integer.*;
import javax.swing.border.*;
import com.borland.jbcl.layout.*;

public class Set_Auto extends JPanel {
    JPanel jPanel1 = new JPanel();
    ImageIcon blue_ball = new ImageIcon("image\\icon_1.gif");
    ImageIcon red_ball = new ImageIcon("image\\icon_2.gif");
    JLabel jLabel2 = new JLabel();
    JTextField jTextField2 = new JTextField();
    JPanel jPanel3 = new JPanel();
    JLabel jLabel3 = new JLabel();
    JPanel jPanel4 = new JPanel();
    JLabel jLabel4 = new JLabel();
    JLabel jLabel5 = new JLabel();
    JComboBox jComboBox1 = new JComboBox();
    JComboBox jComboBox2 = new JComboBox();
    JPanel jPanel5 = new JPanel();
    JPanel jPanel6 = new JPanel();
    JPanel jPanel7 = new JPanel();
    JPanel jPanel8 = new JPanel();
    FlowLayout flowLayout1 = new FlowLayout();
    FlowLayout flowLayout2 = new FlowLayout();
    GridLayout gridLayout1 = new GridLayout();
    GridLayout gridLayout2 = new GridLayout();
    GridLayout gridLayout3 = new GridLayout();
    GridLayout gridLayout4 = new GridLayout();
    BorderLayout borderLayout1 = new BorderLayout();
    FlowLayout flowLayout4 = new FlowLayout();
    FlowLayout flowLayout5 = new FlowLayout();
    TitledBorder titledBorder1;
    TitledBorder titledBorder2;
    TitledBorder titledBorder3;
    TitledBorder titledBorder4;
    TitledBorder titledBorder5;
    JPanel jPanel9 = new JPanel();
    GridLayout gridLayout5 = new GridLayout();
    ButtonGroup bg = new ButtonGroup();
    JRadioButton jRadioButton1 = new JRadioButton();
    TitledBorder titledBorder6;
    JPanel jPanel10 = new JPanel();
    GridLayout gridLayout6 = new GridLayout();
    JRadioButton jRadioButton2 = new JRadioButton();
    JRadioButton jRadioButton3 = new JRadioButton();
    GridLayout gridLayout7 = new GridLayout();
    TitledBorder titledBorder7;
    JPanel jPanel2 = new JPanel();
    JTextField jTextField1 = new JTextField();
    JLabel jLabel1 = new JLabel();
    TitledBorder titledBorder8;
    TitledBorder titledBorder9;
```



```
TitledBorder titledBorder10;
```

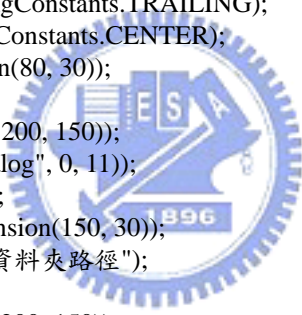
```
public Set_Auto() {  
    try {  
        jbInit();  
    }  
    catch (Exception ex) {  
        ex.printStackTrace();  
    }  
}
```

```
void jbInit() throws Exception {  
    titledBorder6 = new TitledBorder("");  
    titledBorder7 = new TitledBorder("");  
    titledBorder8 = new TitledBorder("");  
    titledBorder9 = new TitledBorder("");  
    titledBorder10 = new TitledBorder("");  
    jButton1.setFont(new java.awt.Font("Dialog", 0, 11));  
    jButton1.setForeground(Color.darkGray);  
    jButton1.setBorder(null);  
    jButton1.setToolTipText("偵測本地資料夾內檔案,自動上傳及刪檔");  
    jButton1.setHorizontalAlignment(SwingConstants.CENTER);  
    jButton1.addActionListener(new Set_Auto_jButton1_actionAdapter(this));  
    jPanel10.setLayout(gridLayout6);  
    gridLayout6.setColumns(1);  
    gridLayout6.setRows(0);  
    jButton2.setFont(new java.awt.Font("Dialog", 0, 11));  
    jButton2.setForeground(Color.darkGray);  
    jButton2.setBorder(null);  
    jButton2.setToolTipText("本地資料夾內檔案,定時上傳不刪檔");  
    jButton2.setHorizontalAlignment(SwingConstants.CENTER);  
    jButton2.setBackground(Color.white);  
    jButton2.setText("定時上傳不刪檔");  
    jButton2.addActionListener(new Set_Auto_jButton2_actionAdapter(this));  
    jButton2.setSelected(true);  
    jButton3.setBackground(Color.white);  
    jButton3.setFont(new java.awt.Font("Dialog", 0, 11));  
    jButton3.setForeground(Color.darkGray);  
    jButton3.setToolTipText("本地資料夾內檔案,定時上傳且刪檔");  
    jButton3.setHorizontalAlignment(SwingConstants.CENTER);  
    jButton3.setText("定時上傳且刪檔");  
    jButton3.addActionListener(new Set_Auto_jButton3_actionAdapter(this));  
    jPanel10.setBackground(Color.white);  
    jPanel10.setFont(new java.awt.Font("Dialog", 0, 11));  
    jPanel10.setForeground(Color.darkGray);  
    jPanel10.setBorder(null);  
    gridLayout7.setColumns(2);  
    gridLayout7.setHgap(10);  
    jTextField1.setPreferredSize(new Dimension(150, 30));  
    jTextField1.setToolTipText("變更本地資料夾路徑");  
    jLabel1.setBackground(new Color(100, 200, 150));  
    jLabel1.setFont(new java.awt.Font("Dialog", 0, 11));  
    jLabel1.setForeground(Color.darkGray);  
    jLabel1.setPreferredSize(new Dimension(80, 30));  
    jLabel1.setToolTipText("本地資料夾路徑");  
    jLabel1.setHorizontalAlignment(SwingConstants.CENTER);  
    jLabel1.setHorizontalTextPosition(SwingConstants.TRAILING);  
    jLabel1.setText("本地路徑 :");  
    jPanel2.setBackground(new Color(100, 200, 150));  
    jPanel9.setBorder(titledBorder10);
```

```

jPanel5.setFont(new java.awt.Font("Dialog", 0, 11));
FlowLayout2.setAlignment(FlowLayout.RIGHT);
bg.add(jRadioButton1);
bg.add(jRadioButton2);
bg.add(jRadioButton3);
titledBorder1 = new TitledBorder("");
titledBorder2 = new TitledBorder("");
titledBorder3 = new TitledBorder("");
titledBorder4 = new TitledBorder("");
titledBorder5 = new TitledBorder("");
this.setLayout(borderLayout1);
for (int i = 0; i < 24; i++) {
    jComboBox1.addItem("" + i);
}
for (int i = 0; i < 60; i++) {
    jComboBox2.addItem("" + i);
}
jPanel1.setBackground(Color.white);
jPanel1.setForeground(Color.darkGray);
jPanel1.setBorder(BorderFactory.createLineBorder(Color.black));
jPanel1.setLayout(gridLayout4);
this.setBackground(Color.white);
this.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel2.setText("遠端路徑 :");
jLabel2.setVerticalAlignment(SwingConstants.CENTER);
jLabel2.setHorizontalTextPosition(SwingConstants.TRAILING);
jLabel2.setHorizontalAlignment(SwingConstants.CENTER);
jLabel2.setPreferredSize(new Dimension(80, 30));
jLabel2.setToolTipText("遠端資料夾");
jLabel2.setBackground(new Color(100, 200, 150));
jLabel2.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel2.setForeground(Color.darkGray);
jTextField2.setPreferredSize(new Dimension(150, 30));
jTextField2.setToolTipText("變更遠端資料夾路徑");
jPanel3.setLayout(flowLayout4);
jPanel3.setBackground(new Color(100, 200, 150));
jPanel3.setForeground(Color.darkGray);
jLabel3.setBackground(Color.white);
jLabel3.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel3.setPreferredSize(new Dimension(80, 30));
jLabel3.setHorizontalAlignment(SwingConstants.RIGHT);
jLabel3.setHorizontalTextPosition(SwingConstants.TRAILING);
jLabel3.setText("上傳時間 :");
jPanel4.setBackground(Color.white);
jPanel4.setForeground(Color.darkGray);
jPanel4.setDebugGraphicsOptions(0);
jPanel4.setLayout(gridLayout5);
jLabel4.setBackground(Color.white);
jLabel4.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel4.setForeground(Color.darkGray);
jLabel4.setText(" 點");
jLabel5.setBackground(Color.white);
jLabel5.setFont(new java.awt.Font("Dialog", 0, 11));
jLabel5.setForeground(Color.darkGray);
jLabel5.setText(" 分");
jPanel5.setBackground(Color.white);
jPanel5.setForeground(Color.darkGray);
jPanel5.setMinimumSize(new Dimension(71, 33));
jPanel5.setLayout(flowLayout2);
jPanel6.setBackground(Color.white);

```



```

jPanel6.setLayout(flowLayout1);
jComboBox2.addActionListener(new Set_Auto_jComboBox2_actionAdapter(this));
jComboBox1.setBackground(Color.white);
jComboBox1.setForeground(Color.black);
jComboBox1.setToolTipText("");
jComboBox2.setBackground(Color.white);
jPanel7.setLayout(gridLayout2);
gridLayout2.setColumns(1);
gridLayout2.setRows(2);
jPanel8.setLayout(gridLayout3);
gridLayout3.setColumns(1);
gridLayout3.setRows(2);
gridLayout4.setColumns(1);
gridLayout4.setRows(2);
jPanel7.setBackground(new Color(100, 200, 150));
jPanel7.setForeground(Color.darkGray);
jPanel7.setBorder(titledBorder3);
jPanel8.setBackground(Color.white);
jPanel8.setBorder(titledBorder5);
jPanel9.setBackground(Color.white);
jPanel9.setLayout(gridLayout7);
gridLayout5.setHgap(0);
jRadioButton1.setBackground(Color.white);
jRadioButton1.setText("即時上傳");
jPanel7.add(jPanel3, null);
jPanel3.add(jLabel2, null);
jPanel3.add(jTextField2, null);
jPanel7.add(jPanel2, null);
jPanel2.add(jLabel1, null);
jPanel2.add(jTextField1, null);
jPanel5.add(jLabel3, null);
jPanel5.add(jPanel6, null);
jPanel6.add(jComboBox1, null);
jPanel6.add(jLabel4, null);
jPanel6.add(jComboBox2, null);
jPanel6.add(jLabel5, null);
jPanel8.add(jPanel4, null);
jPanel4.add(jPanel9, null);
jPanel9.add(jRadioButton1, null);
jPanel9.add(jPanel10, null);
jPanel10.add(jRadioButton2, null);
jPanel10.add(jRadioButton3, null);
jPanel8.add(jPanel5, null);
jPanel11.add(jPanel7, null);
jPanel11.add(jPanel8, null);
this.add(jPanel11, BorderLayout.CENTER);
}

int[] getTime() {
    String h = this.jComboBox1.getSelectedItem().toString();
    String m = this.jComboBox2.getSelectedItem().toString();
    int[] time = {
        Integer.parseInt(h), Integer.parseInt(m)};
    return time;
};

void jComboBox2_actionPerformed(ActionEvent e) {
}

```



```
void jButton1_actionPerformed(ActionEvent e) {  
    jPanel5.setVisible(false);  
}  
  
void jButton2_actionPerformed(ActionEvent e) {  
    jPanel5.setVisible(true);  
}  
  
void jButton3_actionPerformed(ActionEvent e) {  
    jPanel5.setVisible(true);  
}  
}
```

