

# 網際網路及物件導向技術於修漏通報系統開發之研究

## A Study of Internet and Object-Oriented Technology In Water Pipe Leaking Repair Information Systems Developing

洪敏育\*

台灣省自來水公司資訊中心工程師

### 摘要

一個軟體系統的成功與否，其關鍵不在於系統所提供功能的多寡，或所使用開發技術的難易，而是在於能否有效滿足使用者作業需求，以及能否提供一個穩定的作業環境；本研究採用 UML 物件導向技術，提供使用者一個良好的溝通介面，以及穩定的系統動態及靜態結構，並利用網際網路收集修漏資料，將其與地理資料庫相結合，轉化為高附加價值之供水營運資訊及管理知識，將可提供做為系統開發者及管理決策者之參考。

### 一、前言

台灣省自來水公司目前管線埋設總長度達五萬二千五百一十五公里（統計至八十九年底），服務轄區包含台北市以外之台灣省各縣市及高雄市，用戶數達五百二十七萬戶，一千八百三十七萬人口，每年配水量二千八百四十五百萬立方公尺，抄見量一千九百四十七百萬立方公尺，修漏件數超過三十萬件，以如此數量龐大之用戶數，及遍佈全省各地成綿密網狀之輸配水管線，如何開發一電腦系統，即時處理用戶修漏作業需求通知，提升用戶服務品質，以及有效處理戶外管線漏水，避免水資源浪費，提升抄見率，確保水壓穩定，順利供水，為本研究之主要目的。

在自來水公司三層式（廠、所，區管理處，總管理處）組織架構下，管線管理單位（廠、所）遍佈全省各地，如何利用目前網路技術，加速公司內部資訊流通，提昇

基層單位之控管，避免基層人員漏報，延誤修漏作業時間，引發民眾抱怨，造成水資源浪費，甚至災害損失，乃本研究所要探討技術之一。

另對於本研究所使用之管線圖檔資料庫，以及網路傳輸作業開發，其作業型態不同於以文數字處理為主的傳統資訊系統，如仍以傳統之結構化方法進行系統設計，將有其困難度存在，因此，如何以另外一種較為簡便且合適的方法進行規劃與設計，以提昇系統的成功率，即成為本研究之另一重點。

## 二、理論討論

### (一) 網際網路之應用

網際網路 (Internet) 的興起，去除了空間與地域的障礙，使得區域之間的距離在一夕之間被縮短了，藉由其所使用的 TCP/IP 通訊協定，使得全世界的網路得以網網串聯起來，達到了所謂天涯若比鄰的理想境界，除此之外，其所蘊含的技術與應用尚包含下列各項功能：

- 1、World Wide Web：簡稱 WWW (或 Web)，提供了一個多媒體的網路世界，其藉由 HTTP (Hypertext Transfer Protocol) 傳輸通訊協定，以及 HTML (Hypertext Markup Language) 文件格式的開發，使得 Web 文件及相關圖形檔得以經由網際網路傳送至使用者的瀏覽器，而使用者也可以經由超鏈結，存取到全球的網頁資訊。
- 2、電子郵件：透過 SMTP (Simple Mail Transport Protocol) 及 POP3 (Post Office Protocol) 通訊協定，可以讓用戶端將訊息從某部主機傳送至另一部主機，也可以讓您從遠端系統讀取電子郵件。
- 3、FTP 檔案傳送：Internet 上的 FTP (File Transfer Protocol) 機能，讓使用者可以連結到檔案伺服器上進行檔案的讀取及上傳功能。
- 4、Intranet：為一種使用 Internet 通訊標準和工具，在私人網路上提供資訊給企業內部使用者的 TCP/IP 網路，藉此，一個公司可以設立一個只有公司內部員工才能存取的 Web 伺服器，如此就能將公司的信件，營運數字，以及其它公司文件公佈在上

面，而不用擔心受到外界網路世界的干擾。【6】

5、其它與 Internet 有關的技術應用尚有 Telnet、BBS、NEWS 及視訊會議等，但因非屬本研究應用範疇，將不另予討論。

藉由上述的探討可以瞭解 Internet 在資訊的發佈、傳送及分享上有其不可思議的潛力存在，因此本研究將利用上述通訊傳送及資訊分享技術進行實作設計，使得自來水公司之分散式組織能夠達到資訊互通之目的。

## (二)、物件導向方法概念

物件導向的概念是在 1967 年，由挪威學者 Ole-Jone Dahl 所提出，在其設計的 Simula 67 模擬語言之中，包含了物件、封裝、繼承及資料抽象化的觀念，而物件導向方法論 (Methodology) 則是在 1986 年由 Grady Booch 所率先提出，至今已有五十種以上的方法論出現。其主要的概念係將軟體視為一群物件的組合，每一物件包含屬性資料及行爲 (Behavior) 兩大部份，這與傳統的設計方式不一樣，傳統的設計方法以功能為導向，將功能與資料分開設計，而物件導向則以物件為主體，強調物件的封裝性 (encapsulation)，將屬性資料及行爲封裝成一個完整的物件，更能反應真實世界的個體事件；並且由於以物件做為組成系統的元件，在本質上並不會隨著時間而有太大的改變，因此系統的穩定性方面將會提高，這對於一些大型且複雜系統的開發，在其成本及維護度方面將會有相對的助益，因此，本文即是利用此一概念，嘗試以物件導向的方法，對於具有空間特性的管網地理資訊及網路傳輸進行研究，並以自來水公司的修漏通報系統為例進行系統分析。

## (三) 物件導向的基本要件

### 1. 物件 (Object)

物件是真實世界中的任何事物，具有抽象資料形態的類似表示法，將資料屬性與物件外在行爲結合在一起，它可以是 (1) 一個具體可見的東西 (2) 角色 (3) 可理解的

事物或觀念 (4) 互動記錄 (5) 事件 (6) 組織 (7) 結構 (8) 裝置等 [1]，因此一段文字、一張圖畫、一個員工或一個帳戶等均屬於一個物件。每一個物件都擁有唯一屬於自己的識別碼 (Identifier)，因此縱使兩個物件具有相同的屬性值 (如姓名、年齡) 等亦能被加以區別。物件在本文是以方格加以表示，如圖 2-1 所示。

## 2 · 方法 (Method) / 屬性 (attribute)

在每一物件中都包含有屬性和方法，屬性用以描述此一物件的特性，而方法為一組程式碼用以操作物件的變動；每一物件內都可以有數個方法，每一個方法都有一個名稱以指定此種方法所要進行的操作 [8]。

## 3 · 類別 (class)

相同性質的物件，可集結成一類別，類別用於定義集合中各成員的共同特質，此集合中的每一成員為此類別的一個案例 (instance)，所有的案例具有相同的屬性和方法，因此具有相同性質的物件皆可共享此類別，如張三、李四是某一公司的員工，

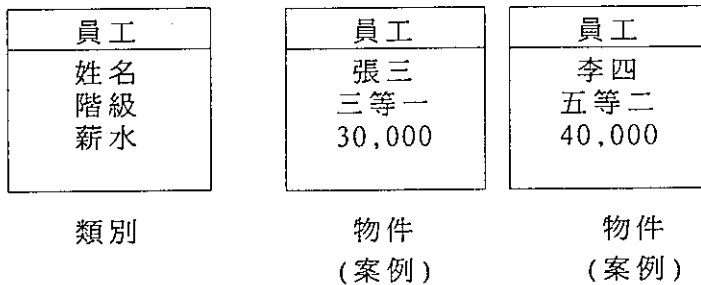


圖 2-1 物件與類別

均具有姓名、階級、薪水等屬性資料，故可用員工來代表張三和李四所屬的類別，而張三和李四

則分別是屬於員工類別的一個案例。類別在本文中同物件是以方格加以表示，如圖 2-1 所示。

## 4 · 訊息 (message)

任何物件之間的溝通以及物件之方法的啟動，是以訊息的傳遞方式來達成，一般而言，訊息包括三部份：(1) 送至那一個物件 (target object)，(2) 啟動什麼方法 (target method)，(3) 運算參數 (parameter argument)。藉由訊息的描述，則可

使靜態的物件因而改變成爲另一種形式，使我們可以得知其動態的變動。

## 5 · 類別階層與繼承 (Class hierarchy and inheritance)

物件導向系統基本上允許自既存類別 (Class) 衍生新的類別，被衍生者稱爲子類別 (Subclass)，而衍生者稱爲超類別 (Superclass)。子類別會繼承 (inheritance) 超類別之屬性及行爲。如交通工具可以衍生陸上、海上及空中交通工具，此時陸上交通工具 (子類別) 會繼承交通工具 (超類別) 的所有特性，如圖 2-2 所示。

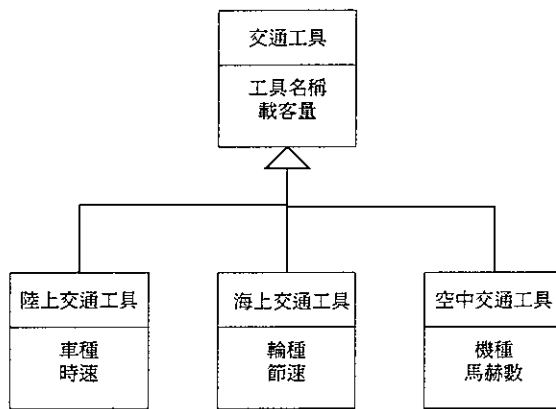


圖2-2 類別階層與繼承關係

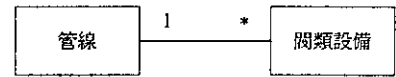


圖2-3 參考關係

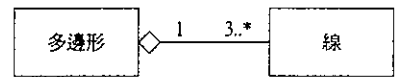


圖2-4 組合關係

6 · 關係性 (Relationship) 一般常提到物件之間的關係有下列幾種【4】【12】：

### (1) 聯結性 (Association)

聯結性 (或稱參考關係) 是指將兩個以上的物件做實體上或概念上的關聯，它可以是一對一、一對多或者是多對多之關聯，在本文是以一直線加以表示，如果爲一對多或多對多之關係則在多的一方加上星號註記。例如一條管線上有許多個閥類設備，爲一對多之表示法，可表成如圖 2-3 之範例所示。

### (2) 一般化及特殊化 (Generalization/Specialization)

一般化及特殊化是一種如前面所說的繼承關係，又稱之爲 "Is A" 關係，它是描

述一個類別和該類別精製版本之間的關係，精製版本繼承原類別之所有特性再加上該類別之專有特性而成爲另一個新的子類別。例如延性鑄鐵管是管線的一種特殊化，而管線則是所有類型水管的一般化。

### (3) 組合關係 (Aggregation)

組合關係是一種部份關係，又稱之爲“A Part Of”關係，它是一種描述物件與其組成元件之間的組合關係，它在本文是以菱形代表，如圖 2 - 4 範例所示，多邊形是由多條線所組合而成。

### (四) 物件導向分析法

物件導向分析 (OOA) 是利用物件導向的觀念，將使用者需求予以抽象化並轉化爲具體模型的一種展示方法，由於這個領域的方法論仍處於百家爭鳴的情況，故尙無一定標準，比較著名的有 Coad/Yourdon 及 Booch 的物件導向分析及設計 (OOA/OOD)，Ivar Jacobson 的物件導向軟體工程 (OOSE)，Rumbaugh 等人的物件導向模型技術 (OMT)，Martin/Ode11 的物件導向分析及設計，以及 Booch, Rumbaugh, Jacobson 所共同發展的 UML (Unified Modeling Language)，而其中 UML 於 1997 年 11 月被 OMG 組織 (Object Management Group) 採納爲標準語言，由於它所代表的不只是一種語言符號表示法 (notation) 的統一，而且更呈現了一種哲學以及思考方式的革命，象徵了我們追求問題解決方案的演進過程，因此可望成爲未來資訊業界的標準模型語言。

**【2】【3】【4】【5】【9】【12】**

### (四) UML 模型架構

UML (Unified Modeling Language) 是一種模型語言，用來指定、呈現、建構和記錄系統化流程中的各種元件，它分別從使用者 (User)，結構 (architecture)，行爲 (Behavior)，實作 (Implementation)，以及環境 (Environment) 等五個不同觀點 (View)，

來進行系統知識的傳達，圖 2-5 是 UML 所呈現的模型觀點架構圖【7】【12】，茲分別簡述如下：

- 1、使用者模型觀點：從使用者角度描述系統功能性需求，找出使用個案（use case）及系統參與者（actor）。
- 2、結構模型觀點：描述系統的靜態邏輯觀點，包括類別圖及物件圖。
- 3、行為模型觀點：描述系統各組成部份的動態行為，以及整體運作的程序，包括循序圖、合作圖、狀態圖、活動圖。
- 4、實作模型觀點：描述系統軟體實作元件間的組織架構和依存關係，它是由元件圖所組成。
- 6、環境模型觀點：描述系統實體設備之間的連結關係，以及軟體實作元件之間的對應關係。

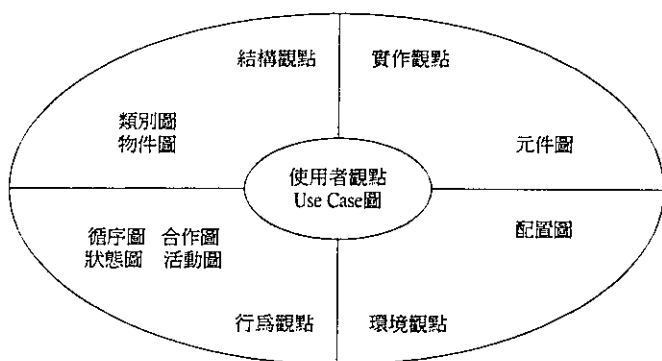


圖2-5 UML 模型觀點架構圖

由於 UML 之設計是以達成溝通為目標，故其提供了足夠的語法，能夠清楚的描繪某個主題的知識（語意），以及展示關於此主題之知識，因此很適合應用在複雜的系統工程領域；而本研究分析之對象包含以圖形為主之地理資訊，以及網路應用為主之通訊傳遞，如以傳統之結構化方法，在語意之表達上將有一定之困難度存在，但如以 UML 做為發展工具，將能夠以其所提供之充份的語意（semantics），使系統開發得

以在合乎經濟效益的方式下，直接解決目前的模式問題，而本研究亦基於此一原因而採此技術做為分析模式。

### 三、修漏通報作業需求分析

本系統所定義之修漏項目含括民眾通告及申請檢修之所有作業項目，如水管破裂、地面漏水、路面積水、沒水、水壓偏低、水濁等供水情況，系統除了建立修漏資料登錄及通報上傳作業機制外，並需規劃建立應用支援系統，透過地理圖形資料庫來處理修漏地理空間資訊，提供管理決策者作為管線供水及檢修規劃之參考，其內容包括如下：

#### 1、修漏情形管制作業

修漏單位受理報修案件應即時記錄案件處理單，交修漏人員派工修理及控管作業時程，並定期產生修漏月報表及修漏記錄簿。

#### 2、修漏稽核管制作業

為配合管理單位稽核及使民眾瞭解修漏作業進度，修漏單位應將修漏資訊上傳總管理處，並上網公告，提供各單位查詢。

#### 3、修漏頻率分析

依據案件處理單內容資料，自動分類及統計，並配合管線圖形資料庫進行各項修漏頻率分析作業，提供管理決策者參考。

### 四、研究結果

#### (一) 研究範圍及工作目標

本研究之範圍是針對前章所分析之功能需求，結合網際網路及地理資訊系統之應用，嘗試以物件導向分析方法將修漏通報作業作一整體規劃，並以 UML 架構模式建立物件導向之動態及靜態模型，以提供作為實體設計之參考，其中為了讓表達過程易於接受與瞭解，及受限於篇幅，部份系統屬性資料及功能將於本研究報告中省略，

但其結果仍不會對系統的完整性造成影響，有關本研究的工作目標詳列如下：

- 1、從使用者觀點整理修漏通報作業應有的功能及其所需相關資料。
- 2、將有關功能及資料予以抽象化，並轉成物件表達方式。
- 3、建立靜態資料之結構模型。
- 4、建立動態觀點之行爲模型。
- 5、分析軟體組成元件，並建立實作模型。
- 6、從系統實體運作觀點找出元件對應之實體配置，並建立實體環境模型。
- 7、對非空間展示之作業部份，建立一雛型（Prototype）先導系統（Pilot Systems），以驗證系統實作之可行性。

## （二）系統功能

依據第三節之修漏需求分析結果，可將修漏通報系統之功能需求以使用個案圖（Use Case Diagram）表示如圖 4-1，橢圓形所示是此系統的使用個案（Use Case），人形圖表示系統參與者，包括使用者及其它系統，如派工及修漏資料維護所產生的結果，會移轉給修漏案件計費子系統去計算人員及材料費用，而系統之維護則需要與備份子系統相結合，以確保系統運作安全性。

在圖 4-1 中，每一個案均可視實際作業需要，往下展開成爲另一使用個案圖，詳述其功能及操作需求，但限於篇幅，本研究將不展示下層之使用個案。

## （三）結構模型

結構模型包含類別圖及物件圖，其目的在定義系統的各個物件，物件間的關係，以及各物件類別所擁有的屬性與運算。本模型以圖型化的方式來表示，這種圖示方法，不但系統發展人員可以輕易地瞭解，一般客戶與使用者也能夠看得懂，因此可當做一種有效的溝通工具。

由圖 4-2 所表達之類別物件關係，可以很輕易看出一張修漏頻率分析圖形是由許

多圖層所組成，而圖層內通常會包含基本地形圖、市街圖以及修漏點分佈圖，其中修漏點分佈圖是由許許多多修漏點所組成，每一修漏點則會對應到一筆修漏記錄；另外從修漏記錄及材料和修漏人員類別的參考對應關係，可以得知每一次修漏會使用到包括管線、設備及耗材在內的材料，而修漏亦需要派遣公司內部或外部的修漏人員進行實地的檢修作業。

圖 4-3 則是依照圖 4-2 類別對應關係所繪之物件圖，由該圖的實際個案，可以驗證類別圖中類別對應關係的合理性。

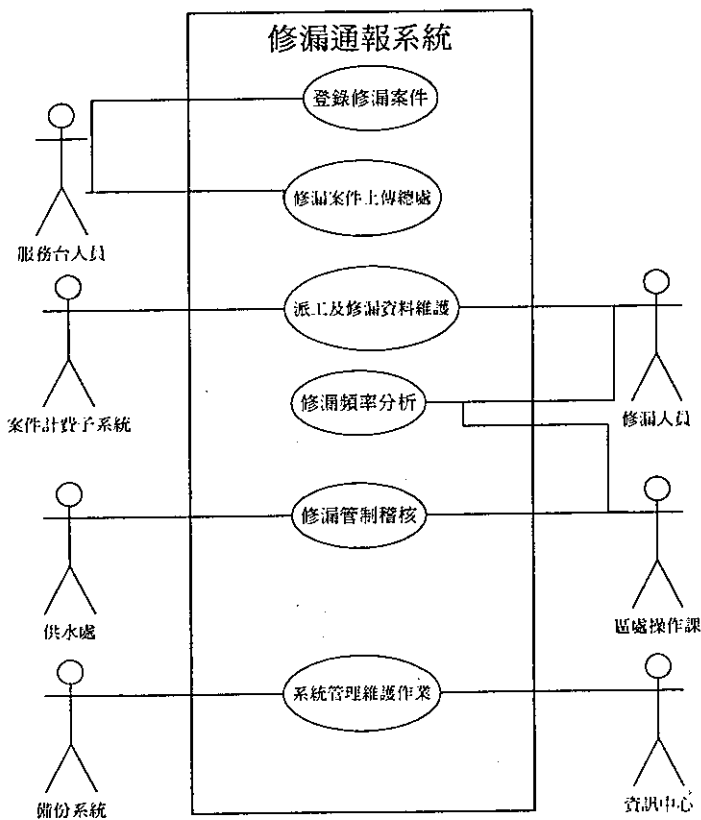


圖4-1 使用個案圖

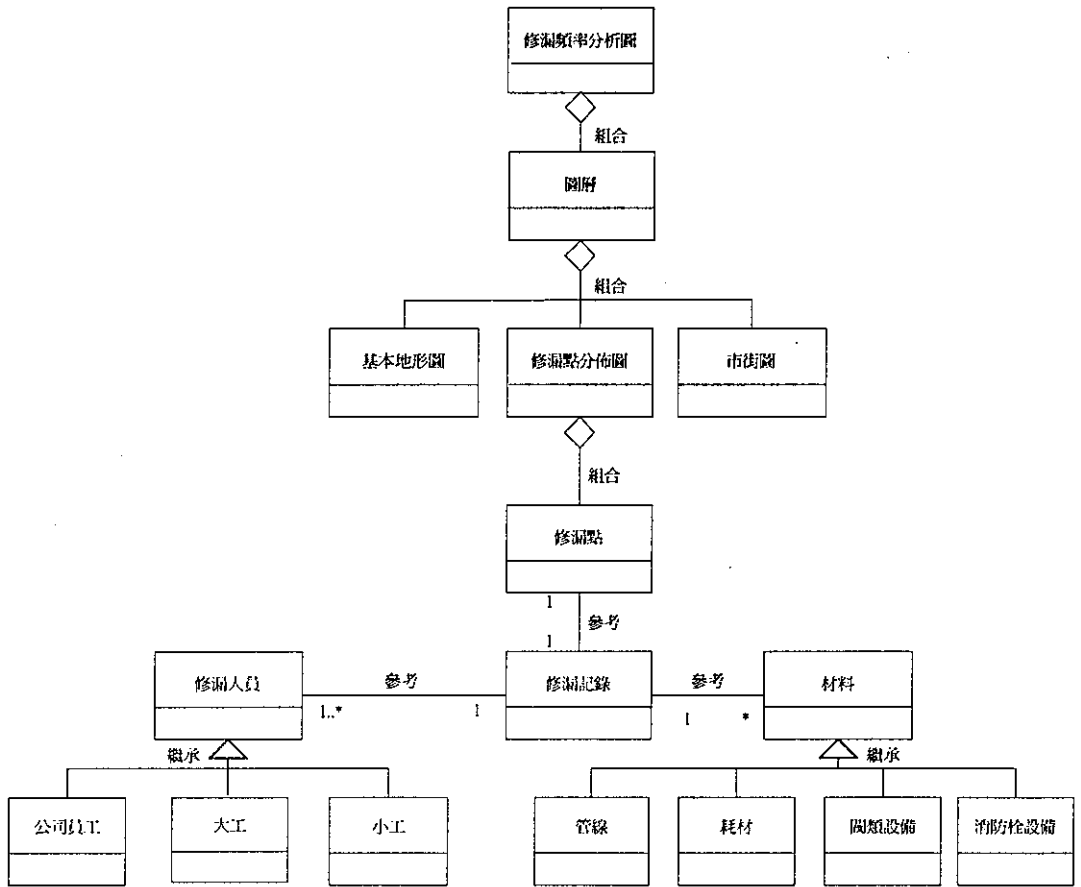


圖4-2 類別圖

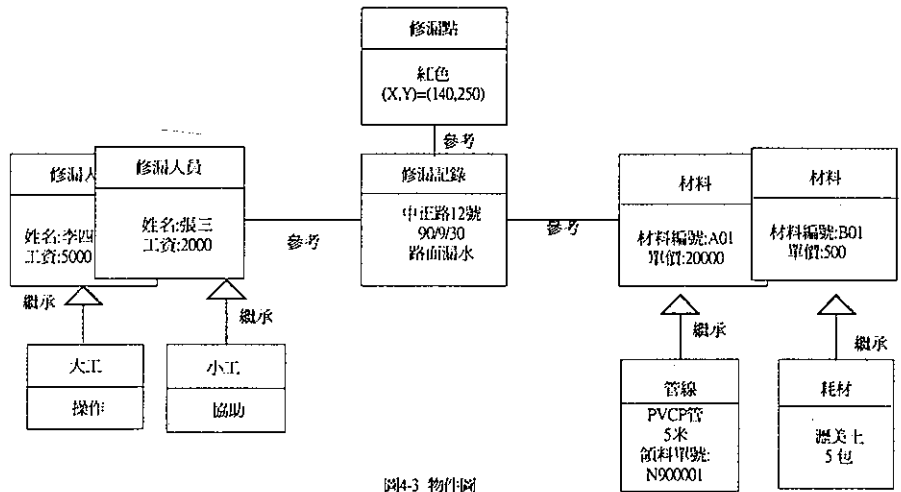


圖4-3 物件圖

#### (四) 行爲模型

要了解一個系統或應用程式，首先要檢視它的靜態結構，也就是該系統是由那些物件所構成，以及這些物件在某一時刻相互關係為何；然後再檢視各物件的狀態。物件的狀態會隨時間或事件的發生，或受其它物件的影響而發生改變。這種從時間及物件狀態改變的觀點來表示系統內部物件的狀態變化，就叫做系統的動態行爲模型（或稱爲行爲模型或動態模型）。行爲模型是利用循序圖及合作圖來描述訊息事件在時間序列及物件結構間傳送及接收所產生之變化情形，另外利用狀態圖及活動圖來描述事件所產生的系統狀態變化情況，事件表示外部的刺激，狀態表示事件本身具有的特性值，圖 4-4，4-5 所示即爲將修漏記錄點顯示在圖層上之合作圖及循序圖，圖 4-6 及 4-7 所示則爲修漏通報系統之狀態圖及修漏頻率分析之活動圖。

#### (五) 實作模型

實作模型主要在描述系統發展過程中各模組元件的組織結構以及相互依存關係，是以元件圖來表示，元件代表實際的程式模組【11】，由一個長方形伴隨著二個小長方形來表示，如圖 4-8 所示。

#### (六) 環境模型

系統發展過程最後要交付給使用者的是一個完整的系統，配置圖可以把系統中的硬體資源及軟體元件對應關係表達出來，如圖 4-9 所示，圖中每一立方格爲一節點（node），節點代表一種計算單元，節點與節點之間則以直線相連接。

### 五、結論與建議

#### (一)、結論

一個軟體系統的成功與否，其關鍵不在於系統所提供功能的多寡，也不在於所使用開發技術的難易，而是在於能否有效滿足使用者作業需求，並提供一個穩定的作業環境；本研究採用物件導向技術及 UML 模型架構，即是在提供使用者一個良好的溝通介面，以及穩定的系統動態及靜態結構，經由本研究所開發之先導雛形系統在彰化給

水廠及員林營運所試辦半年時間，獲得使用者極為正面評價結果，可驗證物件導向方法對系統的成功性具有其正面意義和價值存在。

另經由實體開發過程發現，物件導向之繼承及多型特性，提供軟體再利用(Reuse)之機能，本研究藉由對資料庫結構做有系統之規畫設計，使得系統操作介面得以重複使用，如圖 5-1，5-2 所示，節省約一半之實體開發時間，有效降低軟體開發成本。

綜上所述，歸納本研究之研究成果如下：

- 1、以物件導向方法，將修漏通報網路傳送資料與地理圖形資料庫結合，並做一般化及抽象化之處理，提供一個高層次的概念描述，此為傳統開發方法所不易達成者。
- 2、從使用者、結構、行為、實作以及環境等五個不同構面，建構一個簡單、可擴展、並可立即使用的視覺化模型架構，本架構的核心集合將可適用於更廣泛的應用。
- 3、本研究之結果證明以 UML 物件導方法應用在地理資訊系統及網際網路系統開發之可行性及適用性，將可提供資訊業者做為規劃設計之另外一種選擇參考。

## (二) 建議

修漏資料的收集，並不僅在於單純記錄事件的發生經過以及處理情況，經由事後的追蹤，以及採科學化的歸納分析，將資料提升為高價值之營運資訊及管理知識，才是系統開發設計的主要目標，因此，本研究基於此一理念，透過系統化的規劃，將修漏資料與地理資料庫相結合，採分散式處理，集中式管理架構，將可提供以下之應用：

### 1、提升修漏作業服務品質及水準

- (1)、藉由資料上網公告，管理單位及民眾可有效督導修漏作業進度，避免漏報及吃案情況發生，提升為民服務效率及品質。
- (2)、由於自來水管線是埋在地底下運作，當其發生漏水情況時，除非是滲透到地表面上來，否則很難察覺其水量流失，因此要主動進行修漏作業往往有其困難度存在，但藉由本系統所提供之低壓及漏水統計數字，配合科學化歸納方法及水壓觀測站的設立，將可協助修漏人員大幅提升其檢修漏作業

成效。

## 2、提供管線汰換參考依據

經由漏水頻率分析圖可找出經常漏水路段，對於該路段可進行管材及路面載重分析，並做為管線汰換之依據，進而有效降低日後管線修漏之頻率

## 3、作為供水調配之參考

- (1)、經由修漏原因分析圖可找出水壓低或經常缺水之地區，而於該地區考量增設加壓設備，有效改善低壓地區之供水情形。
- (2)、壓力調配不當，導致經常破管地區，可經由漏水原因分析圖，找出壓力調配之策略，進而改善破管漏水頻率。

## 4、提供新進人員學習及管理工具

修漏之歷史資料及頻率分析圖，可以提供廠所主任及新進人員，對於該轄區之修漏分析數據，使其儘早掌控轄區供水情況，提升整體管理效益。

## 5、提供修漏之歷史記錄，有效維護公司形像

修漏資料電腦化管理，詳實記載每一修漏個案，方便統計分析，可避免媒體及民代對少數修漏個案事件擴大化。

經由以上的探討，可發現本研究對目前修漏作業之管理，以及供水系統抄見率之提升有相當程度的助益，有關如何以本研究為基礎，對於自來水公司供水抄見率做大幅度的提升，將是未來值得研究之發展方向。

## 參考文獻

- 1 · 江立立,個體導向系統發展技術漫談,資訊與電腦,1993,6月
- 2 · Grady Booch, Object Oriented Analysis and Design with Application, 2nd ed., The Benjamin/Comings, 1994
- 3 · James Martine & James Odell, Object-Oriented Analysis and Design, Prentice

- Hall, Inc, 1992
- 4 · James Rumbaugh etc., Object-Oriented Modeling and Design, Prentice Hall, Inc, 1991
  - 5 · Peter Coad, Edward Yourdan : Object-oriented Analysis. Prentice-Hall, 1990
  - 6 · Guy Eddon, Henry Eddon, Components with Microsoft Visual Basic 6.0 2<sup>nd</sup> Edition, 1998
  - 7 · Sinan Si Albir, UML In A Nutshell, O'Reilly, 1999
  - 8 · Paul Harmon, Brian Sawyer, UML for Visual Basic 6.0 Developers, Morgan Kaufmann Publishers, Inc, 1999
  - 9 · Ivar Jacobson, Object-oriented Software Engineering: A Use Case Driven Approach, Addison Wesley, 1992
  - 10 · James Rumbaugh, G. Booch, Ivar, Rational Unified Process, Rational Software Corporation, 1998
  - 11 · Martin Fowler, Kendall Scott, UML Distilled 2nd Edition, Addison Wesley Longman, Inc., 2000
  - 12 · Grady Booch, James Rumbaugh, Ivar Jacobson, The Unified Modeling Language User Guide, Addison Wesley, 1999

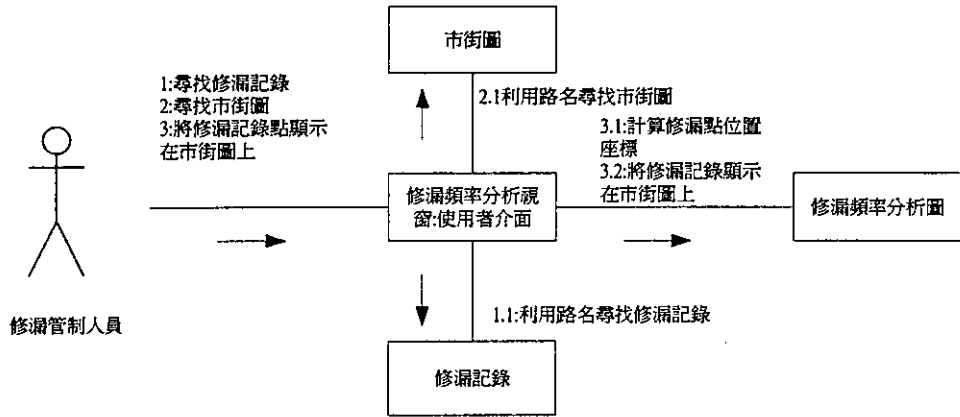


圖4-4 將修漏記錄顯示在圖層上之合作圖

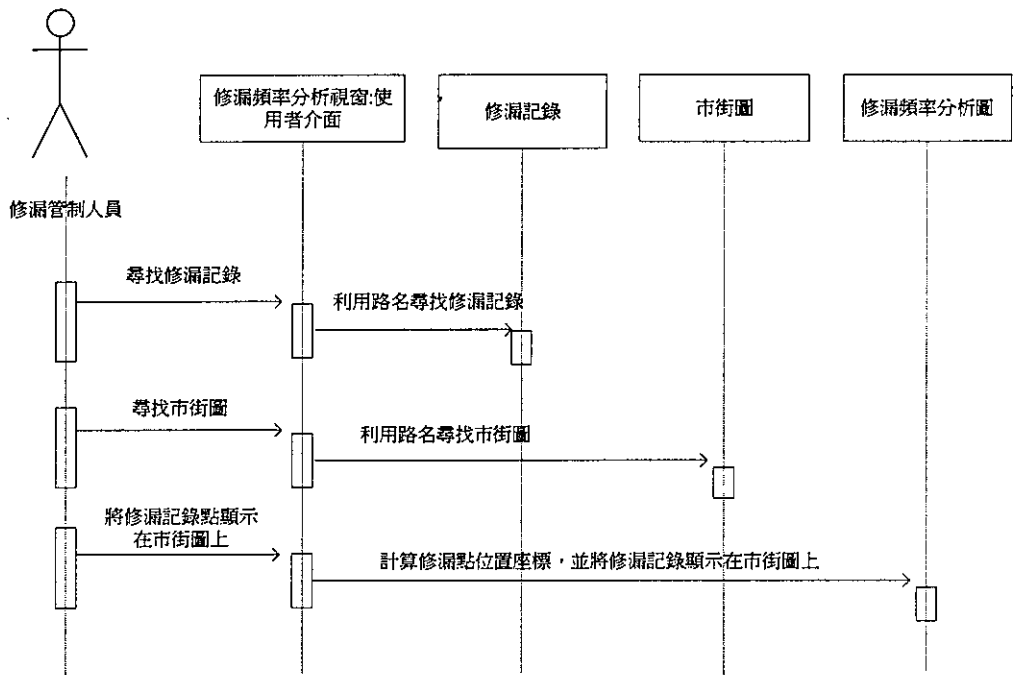


圖4-5 將修漏記錄顯示在圖層上之循序圖

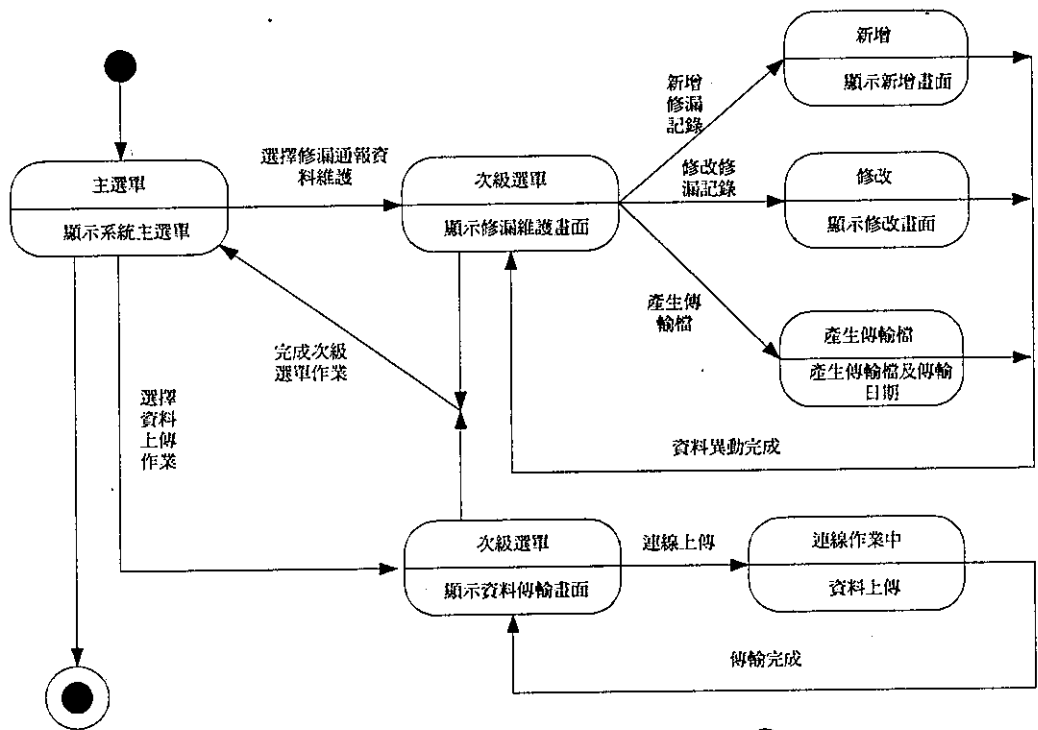


圖4-6 修漏通報系統狀態圖

- 表開始狀態
- 表結束狀態
- 表動作流
- 表資料流

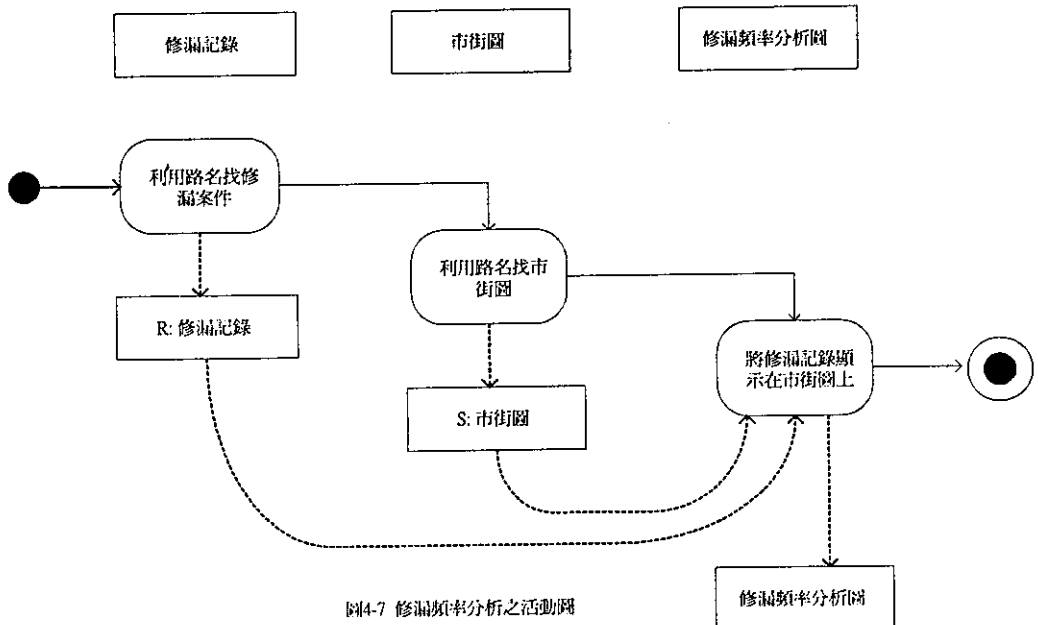


圖4-7 修漏頻率分析之活動圖

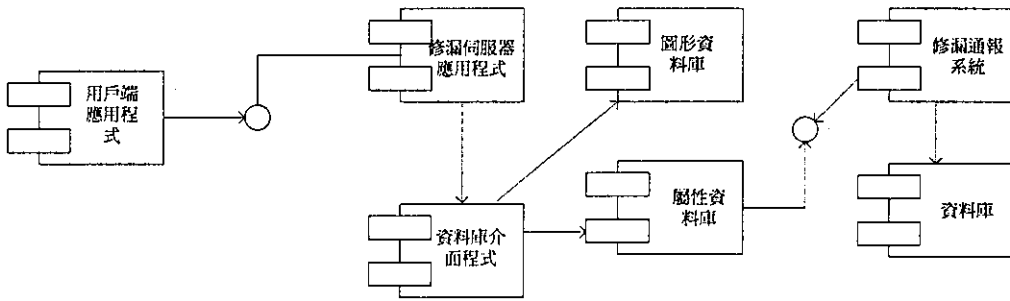


圖4-8 修漏通報系統元件圖

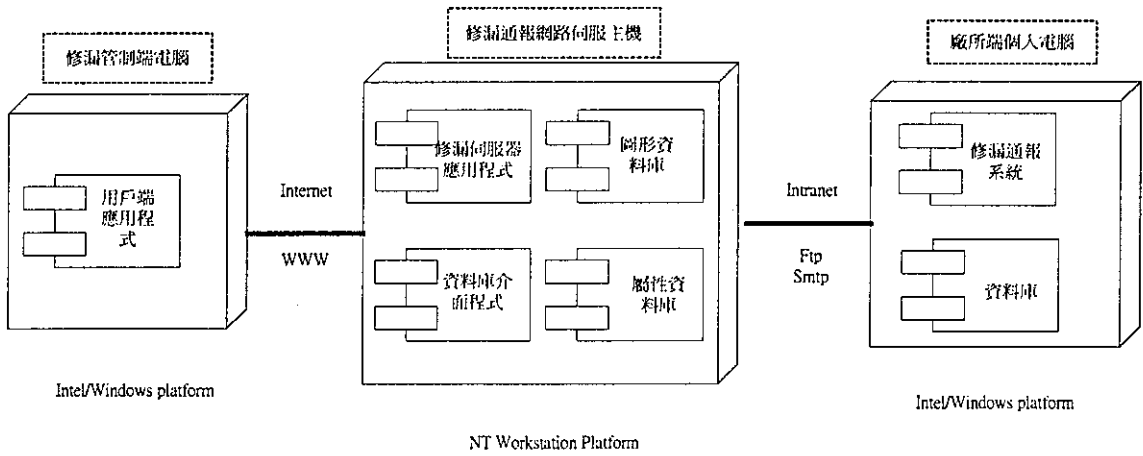


圖4-9 修漏通報系統配置圖

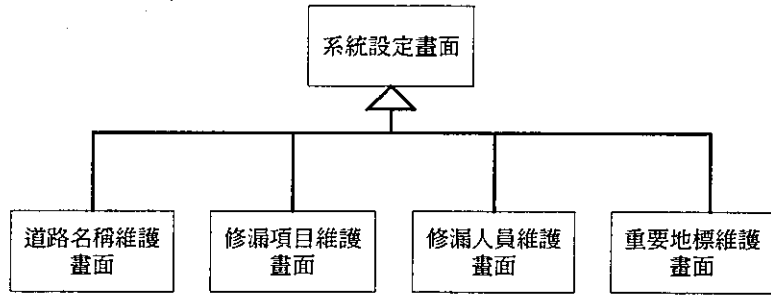


圖5-1 系統設定畫面之繼承

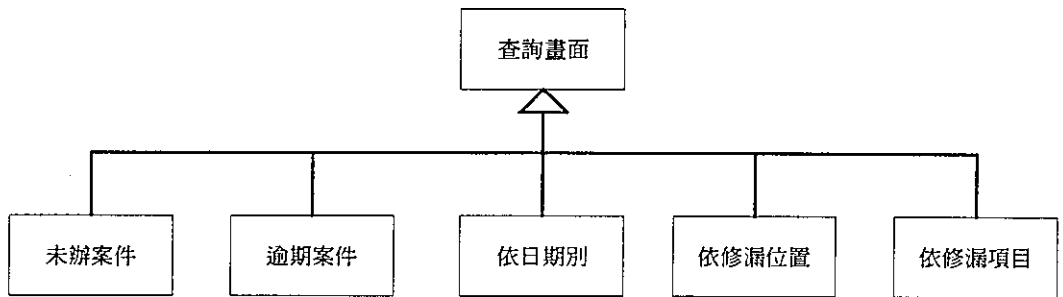


圖5-2 系統查詢畫面之繼承