# 國 立 交 通 大 學

## 資訊學院 數位圖書資訊學程

## 碩 士 論 文

# ITIL 事件管理自動化研究

Automatic ITIL Incident Management

研 究 生： 張峯智

指導教授： 柯皓仁 教授

中 華 民 國 九 十 八 年 六 月

# ITIL 事件管理自動化研究

# Automtic ITIL incident management

研 究 生：張峯智　　Student ： Feng-Chih Chang

指導教授：柯皓仁　　Advisor： Hao-Ren Ke

國 立 交 通 大 學

資訊學院 數位圖書資訊學程

碩 士 論 文

A Thesis

Submitted to College of Computer Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master of Science

in

Digital Library

May 2009

Hsinchu, Taiwan, Republic of China

中華民國九十八年六月

# ITIL 事件管理自動化研究

研究生：張峯智　　指導教授：柯皓仁

國立交通大學　資訊學院　數位圖書資訊學程碩士班

## 摘要

資訊科技基礎架構庫(Information Technology Infrastruceure Library, ITIL)是提供高品質IT服務的指引，其中包含事件管理(Incident Management)流程模組的指引。事件管理的目標是盡快解決事件，且讓IT服務回復到正常狀態的操作環境，確保最佳等級的服務品質。ITIL中只提出各模組的流程、步驟、方針以及要點的指引，並未提供任何工具或軟體的協助，也未探討如何進一步利用自動化方式，來達成實務上減輕人力負擔的目標。本研究提出一套ITIL事件管理平台，在符合ITIL事件管理所需的流程與步驟下，探討將事件偵測、事件分類與事件檢索等事件管理所需的步驟自動化，並使用Perl、事件關聯器軟體Simple Event Correlator (SEC)與資訊檢索等技術，來達成事件管理流程自動化的處理運作，讓第一線服務台人員可利用此實作平台快速進行事件管理流程，縮短IT服務的平均復原時間 (Mean Time to Repair, MTTR) 的時間，提高IT服務的品質。本研究並利用訪談進行本研究實作系統評估，經由先取得訪談人員現行事件管理處理流程，再展示本研究的實作系統，收集目前任職於企業IT管理部門人員的意見，瞭解本實作系統對於其任職企業的助益。從訪談人員回覆的內容中，可確認本研究對於訪談人員企業內的事件管理，有著相當大的幫助。

關鍵字：ITIL、事件管理、事件偵測、事件關聯、SEC、事件檢索

# Automatic ITIL Incident Management

Student: Feng-Chih Chang Advisor: Hao-Ren Ke

Degree of Master of Science in Digital Library

National Chiao Tung University

## ABSTRACT

ITIL (Information Technology Infrastructure Library) is a set of guidelines for ensuring high-quality IT services, which includes the incident management module. The primary goal of incident management is to, when abnormal situations occur, restore normal statuses of business operations as quickly as possible and minimize the adverse impact on business operations, thus ensuring the maintenance of the best possible quality and availability of levels of service within the limits of the service level agreement. ITIL only points out the guidelines for the processes, steps, principles and elements of the modules, and it does not provide any tools or software; furthermore, it does not probe into the use of automated approaches to achieve the ITIL's goal with minimum human burden.

In view of this, this study proposes a plafform for incident management. This plaftform is in accordance with the ITIL incident management's process and procedures. This platform auomates the process of incident detection, incident classification, and incident retrieval. This platform uses Perl, Simple Event Correlator (SEC) and information retrieval techniques to achieve the automated process of incident management. This platform speeds up the handling of abnormal situations for the operators of the first line; thus, the mean time to repair (MTTR) can be reduced and the IT services quality can be improved.

Finally, this study interviews serveral IT management personnels to discuss the benefits of the platform.

Keyword: ITIL, Incident Management, Event Detection, Event Correlator, Simple Event Correlator (SEC), Event Retrieval

# 致謝

至此是該邁入下一階段人生的時候了，從職場中到在職專班的歷程，再到完成論文的日子，現今回想猶歷歷在目，在這其中還有許多人生大大小小的事情，包含轉職、結婚與新居落成等等，也都在研究所這段期間一起發生與完成。

能完成此論文，首先要感謝我的指導教授柯皓仁博士，總是在談笑之間完成論文的指導，指導的過程也都以輕鬆氣氛進行，且讓我在每次討論完後都有接續的目標來努力，因有您的指點與協助，才讓我得以完成我的論文。

感謝我的父母，辛苦的養育拉拔我長大，沒有您們就沒有現在的我。感謝老哥、大嫂及兩位老姊們對我的提攜與照顧，其中國外學成歸國的二姊對我的英文多所指點，而老哥與兩個可愛的小姪女們(嘉嘉與齊齊)常提醒我要努力加油，也讓我於鬆懈之際再度燃起雄心與鬥志，努力朝著終點邁進，誓必要作為下一代的好榜樣。感謝好友們(阿鈞、小谷、小強與小伶)的協助，提供我論文所需的資料。最後要感謝我的老婆小蘭，一路上不管天晴或風雨都陪我一起走過，在這段期間更犧牲了許多原本已規劃的旅遊活動來陪我，也在我論文進行的這段期間幫我分擔了許多我應該做的工作，更一直鼓勵與支持我，讓我能在全心且無後顧之憂的狀況下，來完成最後階段的系統實作與論文撰寫過程。

# 目錄

# 表目錄

# 圖目錄

# 一、緒論

## 1.1 研究背景與動機

由於資訊科技的發展，辦公室自動化的作業環境早已導入企業運作。根據行政院主計處 96 年度電腦使用概況統計資料[1]，受雇員工 30~199 人的企業電腦普及率為 99.07%，而 200 人以上的企業其電腦普及率達 100%，此數據顯示許多企業早已體認：採用資訊科技不僅可以輔助企業營運工作，更是幫助企業取得競爭優勢的利器。而統計資料中受雇員工 30~199 人的企業所使用的伺服器數量在 10 台以上者為 6.13%，200 人以上企業，則為 48.37%，可見隨著企業商務與規模的成長，資訊設備與資訊系統數量的重要性隨之增加。資訊系統從最基本的電子郵件、企業網站、製造執行管理系統(Manufacturing Execution System, MES)、企業資源規劃系統(Enterprise Resource Planning, ERP)、產品生命週期管理系統(Product Lifecycle Management, PLM)至企業商務智慧系統(Business Intelligance, BI)等等，皆扮演著企業商務運作中重要的角色，而維持資訊系統正常運作並即時提供企業所需商務服務協助，是 IT(Information Technology)管理最重要的工作之一。

資訊系統架構常隨著導入的系統與應用範圍擴增而日漸龐大，IT 管理人員所需監控與管理的 IT 服務也增多，在 IT 服務發生異常狀況時，首先需釐清眾多導致異常的可能原因，例如網路、主機硬體、作業系統、應用程式或是資料庫等，需分析每一相關因素的紀錄檔案內容來確認異常原因並解決問題，架構愈龐大和複雜，則造成異常的可能影響因素愈多，需花費的問題查找時間也愈長，若無適當管理工具

或管理模型的協助，資訊服務無法正常使用的時間便無法縮短，便會使企業因商務無法正常運行而造成損失。

資訊服務異常事件的發現情境可分為被動式與主動式兩類，被動式是當使用者因無法使用某項資訊服務而通知 IT 管理人員，而 IT 管理人員再進行問題解決與服務狀態恢復的處理，此類情境的平均復原時間(Mean Time To Repair, MTTR)，常因系統架構複雜或是人員權責不同而無法有效控制，因此若能改採主動式方式，設定各種監測程式來定期檢測資訊服務的狀態，並即時通知 IT 管理人員進行處理，便可有效縮短 MTTR 的時間。當事件發生時，愈快速的事件管理處理過程，愈可減少整個 IT 服務運作的意外停機時間(Downtime)，一般企業重要性質的 IT 服務，若設定以 99.9% 的運作時間為目標，則一年累積總和意外事件停機時間不能超過 8.76 小時，在此條件下，為能達到 MTTR 的要求目標，必定需要主動式的服務異常管理。

GartnerGroup 曾做過一項調查[2]，廣泛訪問企業 CIO 有關服務或應用程式無法使用的原因，發現技術或產品(包括硬體、軟體、網路、電力失常及天災等)因素只佔了 20%(圖 1)；作業程序(Process)錯誤就佔了 40%，作業程序錯誤包括未做好異動管理(Change Management)、超載、沒有測試等等程序上的錯誤或不完整；另外作業人員疏失也佔了 40% ，作業人員疏失包括忘了做某些事情、訓練不足、備份錯誤或安全疏忽等。因此若有標準管理程序或監測管理，以避免人員疏失與作業程序的錯誤，異常事件發生率最多可降低達 80%。

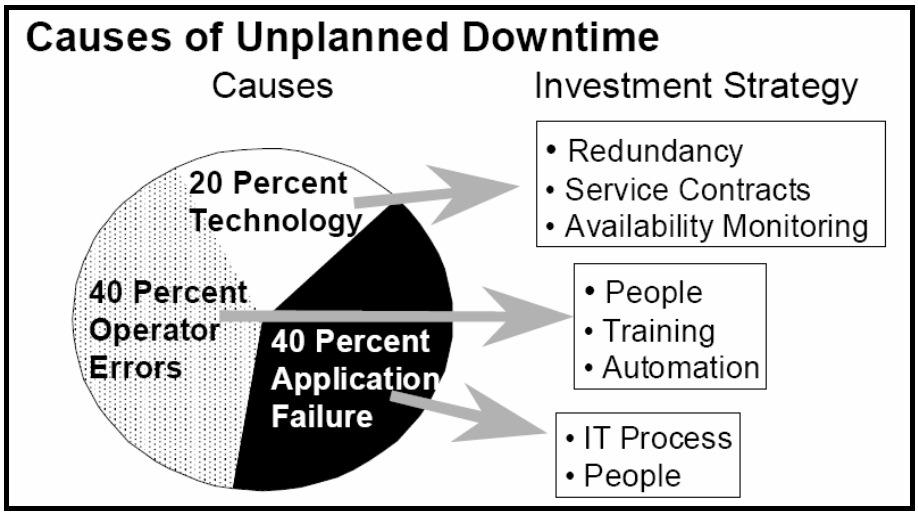


圖 1 意外當機原因統計

資料來源：GartnerGroup

根據以上所述，為維持企業商務服務穩定運作與成長的環境，必須採用適當資訊服務管理模型或管理程序來協助 IT 服務管理。資訊科技基礎架構庫(Information Technology Infrastruceure Library, ITIL)[3]是經過驗證的最佳 IT 管理實務，是全球性具體針對 IT 服務管理所訂定的品質標準，提供一套經彙整之 IT 服務管理的實務典範，以流程導向的觀點，制定 IT 服務管理的流程與步驟，以及該注意的要點與方針，供企業導入 IT 服務管理之參考，以減少和避免錯誤的發生；ITIL 亦可確保在有限的預算下，提昇系統及其服務的可靠性和可用性，進而達到降低成本、增加生產力、提升 IT 服務品質的目標。在 ITIL 中有一事件管理(Incident Management)模組[3]，主要是提供 IT 服務異常的處理指引，ITIL 事件管理指出企業需要針對任何不屬於 IT 服務標準操作程序的事件(包含可能引起資訊服務中斷或降低品質的突發事件)，進行事件記錄、分類、診斷、解決以及結案等事件管理流程的動作，IT 管理人員需要記錄事件的相關資訊，以及掌握有關的組態資訊，將事件正確分類，然後採取適當的措施，迅速地將服務回復到標準可運作的狀態。

研究者目前服務於科技產業資訊部門，有鑒於所任職企業中，IT服務尚未有系統化的架構來協助事件管理處理，另外在成本考量因素下，商用軟體費用昂貴且功能無法滿足企業內所需事件管理的需求，因此引發 ITIL 事件管理的研究動機。ITIL 雖有提出關於事件管理流程與步驟的實務指引，以及應該注意的方針和要點，至於該使用哪些工具或是該如何縮短整個事件處理流程的時效則並未有具體說明，而事件處理流程中每個步驟所需時間的加總，便是資訊系統 MTTR 時間長短的關鍵，也是 IT 部門效能與企業商務目標的影響因素。故本研究將在符合 ITIL 事件管理流程所需的步驟與指引條件下，建構出自動化的事件管理平台，讓企業應用 ITIL 事件管理時，可發揮更大的助益。

## 1.2 研究目的

根據以上研究背景與動機，本研究希望實作一系統以達成下述目的：

1. 建立集中式自動偵測機制，偵測 IT 服務相關的物件狀態。透過偵測各相關物件狀態，處理各種物件訊息，快速取得造成 IT 服務異常的問題根源因素(Root Cause)。
2. 發展符合低成本、跨平台、與設定簡易等需求的事件關聯方式。利用此事件關聯方式，處理與判斷各類自動偵測程式或是資訊系統紀錄檔分析程式所取得的訊息；依事件規則庫來分類事件與採取對應的措施，以提供事件影響程度的分類資訊，因而減少檢視相關紀錄檔的步驟與人力維護需求，避免人員因資訊技能不足，或是判斷錯誤而遺漏應檢查的資訊。
3. 整合事件與 IT 服務組態資料，便於取得物件組態狀態與異動資

訊，提供異常事件處理參考。利用 ITIL 的組態管理模組指引，建構系統組態管理資料庫，以利 IT 管理人員進行物件組態資訊的收集與稽核，減少因異動組態而導致 IT 服務異常的可能，並可供舊組態資料回復使用。

4. 利用資訊檢索技術，自動分析事件與歷史事件處理資料的相似度，取得最相似問題解決處理步驟，呈現給服務台做為異常 IT 服務事件回復解決使用，省去讓服務台人員遭遇 IT 服務異常時，需利用關鍵字搜尋相關文件的步驟，加快事件管理流程的進行。
5. 建立符合 ITIL 事件管理流程與步驟指引的事件管理系統，在本研究中將相關步驟以自動化方式進行，取代人為運作方式，並提供簡單明瞭的使用者 Web 介面，顯示簡易圖示代表資訊服務運行狀況，整合從 IT 服務到底層資源的運行狀態，協助 IT 或相關人員可快速取得系統異常關鍵與解決方式，進而快速解決事件異常，回復 IT 服務至正常使用狀況，縮短 MTTR 時間。

## 1.3 研究問題

為達成研究目的，以下為本研究需解決的問題：

1. 在 IT 管理運作上，如何得知企業所有 IT 服務的運行狀態與遭受之威脅？

   傳統的 IT 管理模式為各 IT 管理部門監測各自所屬的權責管理內容，如網路部門專注在網路環境的運作狀況，而資料庫管理部門只限於資料庫系統，資訊系統運作部門為作業系統相關範圍。但若只監測各自單位的元素，則無法確認有異常或列為警告的是影響到的哪些服務。以 IT 服務而言，需能提供管理者或一般使用者檢視現行所有資訊服務運行狀況，此為跨部門或單位的角度瞭

解資訊服務狀況。

2. 在服務異常事件上，如何快速找到問題真因？

事件的偵測可分為使用者通知異常與自動偵測事件異常，針對資訊系統服務運作，當使用者反映服務異常事件，一線人員便找尋相關標準操作程序文件(Standard Operation Procedure, SOP)中相類似的狀況，繼而判斷是否有相關異常事件，再通知二線人員確認問題真因；二線人員需檢查相關系統的紀錄檔案判斷異常原因，再進行處理或回復系統至可運作狀態。每一步驟的處理與判斷的時間加總，便是影響到系統回復時間的快慢，若能一開始便取得造成問題的真因，即可縮短處理過程與時間。

3. 如何確認 IT 服務組態狀況？

根據研究背景與動機所述 GartnerGroup 的調查，80%的 Downtime 是人為因素所造成，故若針對每一組態進行監測，自動偵測組態設定值，便可確認組態的資訊，並需有組態稽核方式，比對確認異動的組態資訊，來提供異常事件發生時的問題解決參考。

4. 有何低成本、耗用資源少、跨平台、設定簡易且功能完整的事件關聯解決方案？

HP Openview 或是 IBM Tivoli 皆是可設定監測程式的商用軟體，但從 IBM Trivoli 的規則庫設定方式[4]，便可發現其設定複雜，且一些商用軟體需安裝軟體於被監測的伺服器上，並有平台與軟硬體的需求與限制，而軟體授權費用更是 IT 管理部門的一大負擔。在另一方面，IT 部門自行開發的監測程式，無立即性的成本負擔，較商用軟體有彈性，可將監測的規則寫在程式碼中，但當這些規則隨情況而改變時，程式也往往需時常異動、維護，當這些事件日益繁多且複雜時，此時這些程式也會變得愈來愈難

以維護與理解。

5. 如何將 ITIL 事件管理流程自動化，來縮短 MTTR 時間？

ITIL 雖已提供事件管裡的流程與指引步驟，但未提供實際作法與工具，故若有實際作法來將事件管理的步驟與過程自動化，則可提供企業商務運作更大幫助。若有具備整合事件管理步驟的自動化平台，在符合 ITIL 定義的事件管理流程與指引下，運用此經驗證過的 IT 管理最佳實務架構，可協助 IT 管理人員提昇 IT 服務的事件管理品質，降低 IT 服務 MTTR 時間。

## 1.4 論文架構

本論文共分為五章，第一章為緒論，說明研究背景與動機、研究目的、研究問題以及論文整體架構。第二章為文獻探討，介紹與本研究相關的主題，包含 ITIL 事件管理、組態管理、事件關聯與資訊檢索等。第三章為系統架構與設計，介紹本研究的主要系統架構、研究範圍與限制、系統各功能內容與設計方式。第四章為系統實作與評估，說明系統實作的開發工具與平台、方式與流程、與建置出的自動化事件管理系統功能；並利用訪談企業內部 IT 管理部門人員的方式，探討基於本研究方式所建構的系統，對於其所任職企業 IT 服務事件管理的適用性與助益。第五章為結論與未來工作，總結本研究的研究成果與貢獻，並針對未來可能的研究方向加以說明。

# 二、 文獻探討

本研究欲在 IT 服務管理的環境中，設計並實作一個符合 ITIL 事件管理流程的自動化事件管理系統，讓管理事件的服務台人員可以快速發現與解決新發生的事件，提升 IT 服務的品質。在本章將對與本研究相關的主題進行探討，第一節是 ITIL 與其中的 IT 服務管理，並對 IT 服務管理中兩個與本研究相關的模組進行說明，第一個模組為本研究主要探討的事件管理模組，第二個是 ITIL 組態管理模組，是建立 ITIL 的組態管理資料庫(Configuration Management Database, CMDB)最主要的流程模組；第二節為事件關聯部分，即建構本研究中事件管理的自動化分類步驟必備項目，本節探討事件關聯相關方式與優缺點，並介紹本研究採用的 SEC 事件關聯器；第三節為資訊檢索領域探討，此部分說明本研究中會運用的資訊檢索系統，與採用之向量空間模型的概念，與文件相似度的計算方式等等，用以協助建構本研究中事件管理診斷步驟的自動化。

## 2.1 資訊科技基礎架構庫(ITIL)

ITIL 乃是由英國中央電腦與電信管理局(Central Computer and Telecommunications Agency, CCTA)所發展的一套 IT 實務管理方法，用於規範資訊技術服務管理的架構，達成有效率的 IT 服務支援，與高品質的服務傳遞與成本效益，CCTA 後來併入英國政府商務部門(Office of Government Commerce, OGC)[3]，並由 OGC 持續維護與更新 ITIL。後來，為了將 ITIL 標準化，英國政府、英國標準協會(British Standards Institution, BSI)及相關團體共同制定了以 ITIL 為核心的標

準：BS15000，由 IT 服務管理論壇(IT Service Management Forum, itSMF)[5]作為標準團體，並於 2002 年提交給國際標準化組織(International Standardization Organization, ISO)，2006 年初正式公佈以 BS15000 為主軸的 ISO 20000，正式成為 IT 服務管理的國際標準。

ITIL 是一個 IT 服務管理最佳實務(Best Practice)的參考[6]，以流程導向的觀點，制定 IT 服務管理的流程與步驟，以及該注意的要點與方針，供企業導入 IT 服務管理之參考，協助企業建立高品質的 IT 服務。最佳實務是基於超過一個人、一個組織、一個技術與一個事件以上的經驗，最佳實務所代表的是根據高品質與特定專業領域的最佳經驗指引。

ITIL 呈現的框架是由許多部分所組成，每個部分都代表一組流程(圖 2)，它包含一系列管理程序，目的是使 IT 能與企業商務目標一致，協助企業業務的成長。ITIL 的目標為降低成本、改善 IT 可持續服務狀態、調校 IT 服務的容量、增加 IT 服務可提供的總輸出量、將 IT 資源的利用最佳化，以及增加 IT 服務的可擴展性。

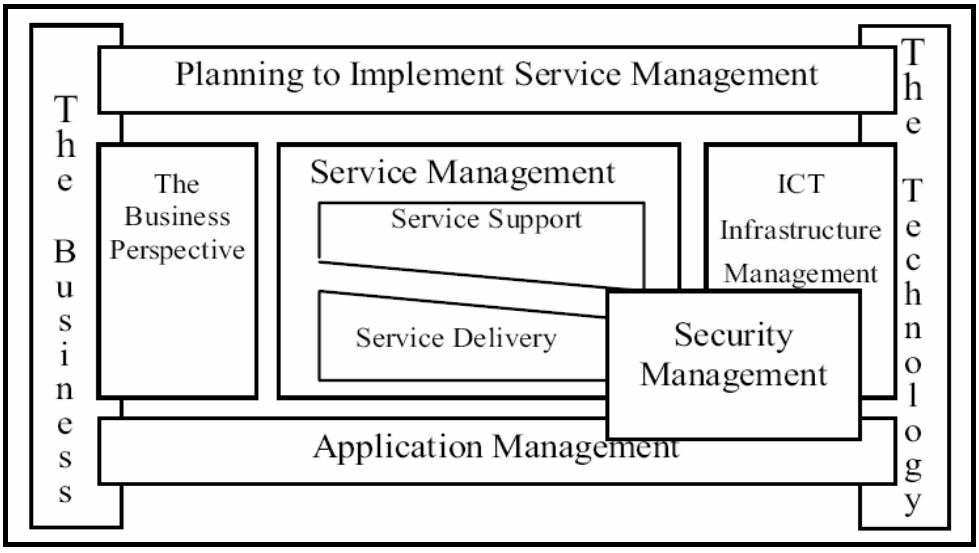


圖 2　ITIL 架構圖

資料來源：HP ITIL Foundation for IT Service Management[6]

企業導入 ITIL 可以達到降低成本、增加生產力、提升服務品質的目標，並使 IT 部門與其客戶間(各部門 IT 服務使用者)維持良好的關係。根據 IDC(International Data Corporation)機構進行統計的數據結果[7]，企業實施 ITIL 管理流程後，平均可讓企業 IT 人員的生產力提昇 53%、效率提升 26%，資訊系統 Downtime 縮減 31%，整體投資效益(Reurun on Investment, ROI)更達 1296%，此統計結果顯示出 ITIL 的導入成效相當顯著。

雖然企業導入 ITIL 有許多好處，但需注意造成實施失敗的可能狀況，Sharifi 等人[8]整理了企業導入 ITIL 失敗的因素，包括缺乏管理高層的承諾與支持、花太多的時間在複雜的流程細節、沒有建立工作的指引、沒有分配流程所屬的負責人、沒有注意品質、太過於專注與強調效能、太過於雄心勃勃想一次實施所有 ITIL 流程、未能持續實施的動能，以及未讓各部門認知需共同協同運作等因素，故企業計劃實施 ITIL 之初便需考量這些導入失敗因素，並設法避免這些狀況的發生。

### 2.1.1 服務管理(Service Management)

IT 服務管理(Information Technology Service Management, ITSM)是 ITIL 框架中的一部份。IT 服務(IT Service)的定義為[6]，一組 IT 系統所提供支援一個或一個以上之企業商務領域的相關功能，服務是由硬體、軟體與相互溝通之元件所組成的緊密實體。IT 服務管理是以服務導向的觀點，將 IT 視為一種服務，以服務管理的概念來管理企業內的 IT 內容，以提供高品質的 IT 服務方式，提昇企業的競爭優勢。

ITIL 將服務管理分為服務支援(Service Support)與服務傳遞(Service Delivery)兩大類[6]，並透過服務台(Service Desk)執行各項服

務管理工作，共包含十個流程加上一個服務台功能模組。服務支援是與 IT 服務的操作性與短期管理循環有關，強調 IT 服務的立即性日常運作管理，以確保 IT 服務的品質，目的在讓使用者能夠順利使用 IT 服務來滿足企業中所需的商業運作功能。服務支援包含 5 個流程模組，描述如(1) 減少異常事件對企業商務運作造成不良影響，儘快將 IT 服務回復至正常水準的事件管理(Incident Management)模組，(2) 追蹤真正問題根源，避免相同狀況再發生的問題管理(Problem Management)模組，(3) 透過標準化的方法和程序(核准與時程等控制方式)，來管理資訊基礎架構中所有異動行為，降低變更衝擊的異動管理(Change Management)模組，(4) 利用計畫、設計、建立與測試等方式，從整體角度來確認軟硬體上線情況的上線管理(Release Management)模組、(5) 定義、控制與驗證資訊基礎架構(Infrasctucture)中之物件組態資訊的組態管理(Configuration Management)模組，另外加上一個服務台(Service Desk) 功能，服務台功能可讓使用者用單一窗口方式與提供 IT 服務的單位溝通。服務傳遞是與 IT 服務長期策略面管理有關的流程，包含財務管理(Financial Management)、服務等級管理(Service Level Management)、IT 服務永續管理(Continuity Management)、可用性管理(Availability Management)、負載管理(Capacity Management) 等 5 個流程模組。ITSM 中的每一模組皆從高角度來描述流程、功能、角色與權責，以及需要的資料庫與介面，但並無任何有關管理工具的資訊。

服務台是 ITIL 服務支援的一個重要功能模組[6]，是服務支援中各流程的運作源頭(圖 3)。企業中成立一個 IT 服務小組以接收多種管道(如電話、網頁、電子郵件、使用者面對面接觸或監測系統回報等)所傳來的服務需求、異常事件通知及問題諮詢，並經由特定程序來進

行適當的處理。ITIL 中的服務台定義為 IT 部門的單一服務窗口(Single Point of Contact, SPOC)，在日常資訊維運作業上，負責於第一線處理與記錄所有事件(Incident)及服務需求(Service Request)。服務台最重要的任務，就是維持資訊服務的正常運作狀態，若有意外事件發生時，需記錄事件與所有處理歷程，協調回復服務的相關行動，儘可能於最短時間內回復服務品質至正常水準，並且避免以後發生同樣的事件問題。在施淑鎂[9]的研究中將服務台視為支援中心，並用平衡計分卡的方式，以企業內部流程、學習與成長、顧客、財務等四個構面來評估服務台的績效。

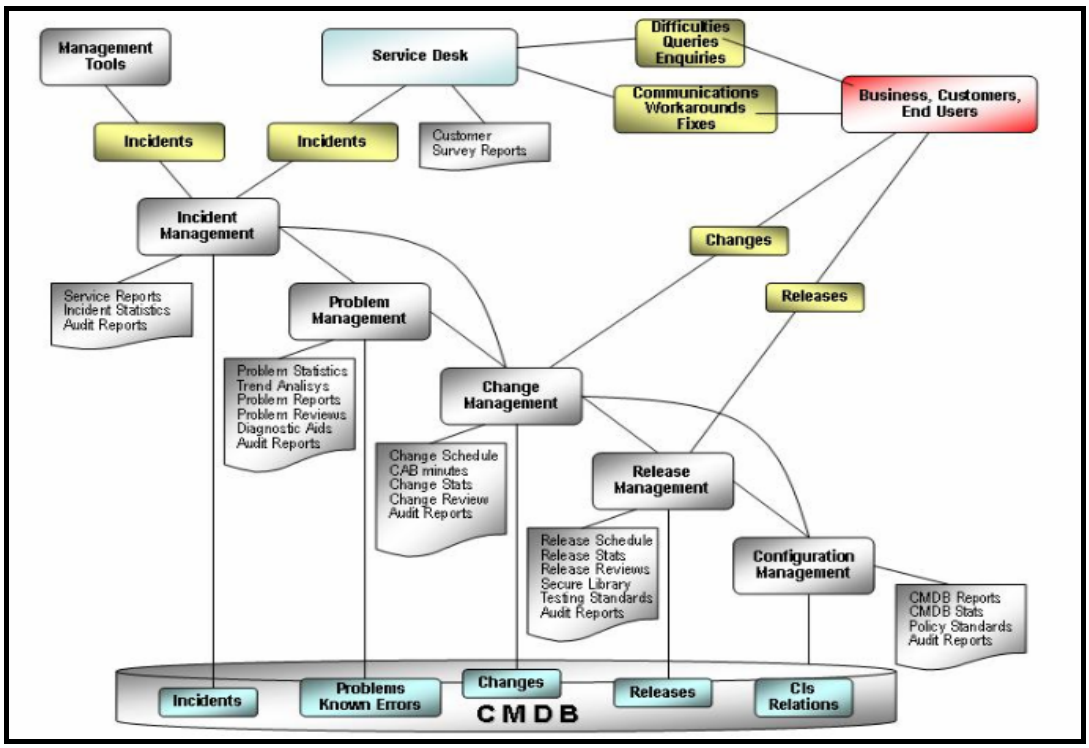


圖 3 服務支援流程架構圖

資料來源：ITIL Service Management Foundation, Sun [10]

因 ITIL 並未有自動化步驟指引，Brown 等人[11]提出依其經驗，對 ITSM 的服務支援各管理模組進行自動化的步驟指引，步驟是：(1) 先定義要自動化的流程與領域，(2) 建立可應用自動化的範圍，(3) 定義可自動指派進行的機會所在，(4) 定義指派的活動與外部活動、流

程與資料來源的相關聯性，(5) 設計介面來溝通或維護自動化 (6) 最後是實施流程自動化與指派的活動。Brenner[12]利用分析各模組於一定期間內會發生的復發率(Recurrence)、整個模組從觸發到結束的持續時間(Lead Time)、與模組所牽涉的組織之複雜程度(Organizational Complexity)、模組所造成的服務等級影響(Service Level Impact)與模組結構化程度(Structure)這五個層面，來分類 ITSM 各模組並評估工作流程(Workflow)支援工具的適用性，結論為事件管理、異動管理與問題管理此三個模組，較適合利用工作流程工具來協助模組的運作。

### 2.1.2 事件管理 (Incident Management)

在 ITIL 中，事件的定義為[6]：任何不屬於服務標準操作程序的一部分，而導致或可能會導致服務中斷或是服務品質降低的狀況，就稱為事件。因此，當使用者遭遇這些狀況時，就會向服務台提出事件需求，服務台需要採取適當的處理動作，這些動作就稱為事件管理，而這些事件管理的活動，就稱為事件管理流程。ITIL 將事件管理流程定義為[6]：當 IT 服務事件發生直到服務回復到正常運作水準的過程中，進行對事件的記錄、分類、診斷與回復等活動。事件管理的目標是盡快解決事件，且讓 IT 服務回復到正常狀態的操作環境，確保最佳等級的服務品質。

事件管理流程有相關的輸入、輸出與相關活動如圖 4 所示，其中活動的步驟包含事件偵測、事件記錄、事件分類、事件診斷、事件解決、系統回復、事件結案與事件擁有權指定等。

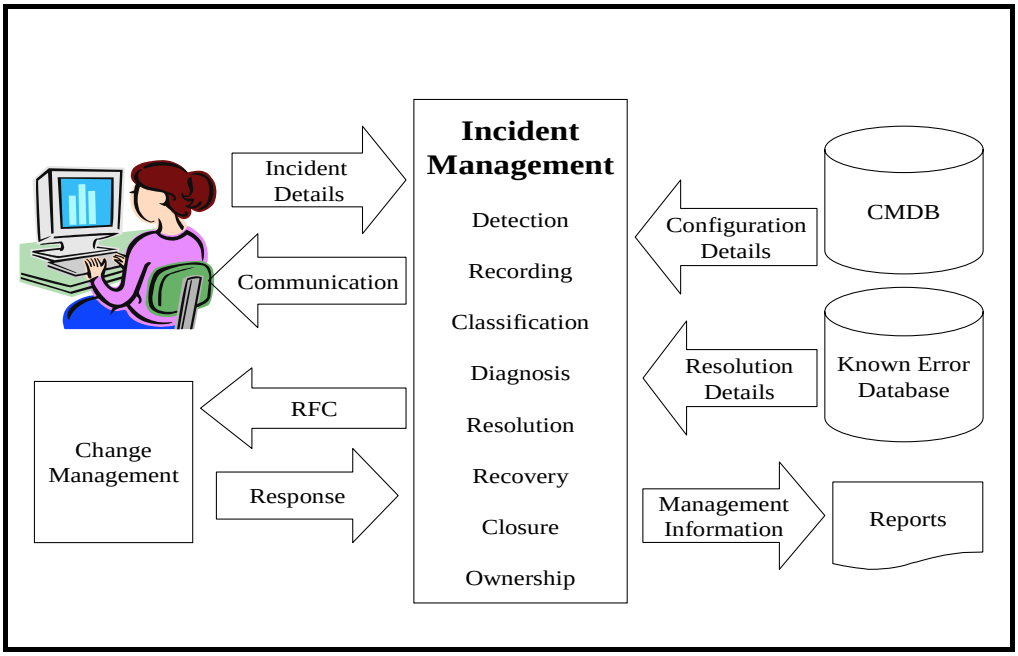


圖 4　事件管理輸入、輸出與活動

資料來源： HP ITIL Foundation for IT Service Management[6]

事件管理中的各項活動可分為以下幾點來描述[6]：

1. 事件偵測(Detection)：瞭解事件的發生情況，定義事件發生的狀況或所在，可能的來源為從服務台取得的使用者回報，或是由系統異常狀況管理偵測系統所引發，此事件詳細資料是事件管理流程的輸入，而由服務台人員所引發。事件有可能是使用者的服務需求，而非系統異常，若是服務需求則是由服務需求的處理程序來進行這個服務請求。
2. 事件記錄(Recording)：此活動需真實地記錄事件的資料，如時間、反應人員、異常系統、異常狀況與影響範圍等等。
3. 事件分類(Classification)：此活動是根據紀錄的事件細節來進行，如根據事件的起源或者是症狀，將事件辨別為某個類別的事件。另外針對事件資訊，需決定緊急程度以及影響程度，進而判斷與決定事件處理的優先程度，如為 24 小時生產使用的應用系統 MES 異常，且事件造成生產停線影響，

則優先程度為最高。

4. 事件診斷(Diagnosis)：在這個階段所要進行的動作有，(1) 仔細的評估事件的細節，(2) 收集和分析所有相關的資訊，以及(3) 嘗試從已發生並處理完成的問題處理知識庫(Known Error Database) 中，找出相類似問題的暫時性解決方案，或者是將事件安排給後端的專業人員引發一個異動需求(Request For Change, RFC)，交由問題管理(Problem Management)流程模組進行處理，來永久解決這個事件。
5. 事件解決(Resolution)：在這個階段所要進行的動作為，採取或使用取得的解決方案，或是暫時性解決方案(Workaround)來進行事件的處理與解決，讓系統啟動(System Up)。
6. 事件恢復(Recovery)：此活動是將系統回復到原先正常狀態，且讓使用者可以使用系統(User Up)。
7. 事件結案(Closure)：當事件解決後，需得到使用者或事件提出者的結案確認訊息，確定事件已被正確解決後，才可以將事件結案。
8. 所有權(Ownership)：這個階段是要確認事件從起始到結束整個過程的生命週期，所有的事件必須是沒有遺失，沒有被延遲處理或者是被忘記等等，最後還要通知使用者事件的狀況。

相關研究中，Hanemann[13]提出用反覆偵測 IT 服務狀態的模式，找出其他相關服務受到影響的狀況。Bartolini 等人[14]提出事件管理的重要性，將數種 IT 管理的效能指標(Key Performance Indicator, KPI) 對應到事件管理的結果，再對應到管理目標(Management of Business Objectives, MBO)做為整個企業與 IT 部門管理的評鑑項目。

Gupta 等人[15]也提出事件管理的自動化研究，但其事件來源範圍限定於使用者回報的事件，且使用關鍵字方式來進行，讓服務台人員利用關鍵字方式查詢相關事件的文件，或是查詢搜尋異常的根源物件。周柏村[16]的研究中提出使用知識支援來提供對事件管理的幫助，並利用資料探勘技術從事件紀錄中發掘出相關的事件管理知識，來協助服務台的事件管理，但此研究是限定在服務台針對新事件已做完初步的辨別與分類後，才可進行。李桄瑋[17]的研究中提出利用主題地圖方式，提供事件管理者事件相關知識支援，透過服務台建立事件表單方式，再由事件管理者進行判斷與事件解決。

在本研究中，將針對 ITIL 中的事件管理流程，將事件管理中原本需人為操作的事件偵測、記錄、分類，以及診斷(尋找類似解決方案)等步驟，轉化為利用自動化事件管理架構協助該等步驟的進行。採取積極主動方式，且全面性掌握每一個細節，使事件管理流程可快速處理與進行，加速回復系統至正常運行狀態，輔以 IT 服務運行最大的幫助。

### 2.1.3 組態管理(Configuration management)

ITIL 組態管理[6]是識別、控制與稽核需要管理的 IT 服務，用設計與維護一個資料庫的方式，儲存需控制的項目，與這些項目的狀態、生命週期以及相互之間的關係，加上任何可以有效管理 IT 服務品質的資訊，其中的資料庫即稱為組態管理資料庫(CMDB)。

組態管理的目標與指引可描述為以下幾項 [6]：

1. 定義與紀錄關於 IT 資產與需要被管理的 IT 服務組態資訊。
2. 在資料庫(CMDB)中控制 IT 服務組態資訊。
3. 讓 IT 部門可以證明已按成本提供最適當等級的服務給使用

者，間接可導致服務品質滿意度的改善。

4. 支援授權(License)管理。

5. 確保 IT 基礎架構資訊是有即時更新且是正確的狀態。

6. 提供一個對於 IT 服務管理的基礎管理。

7. 提供關於 IT 基礎架構組成物件的狀態資訊。

8. 提供 IT 基礎架構的管理資訊來源。

CMDB 是一個構成資訊系統所有元件的貯藏區[6]，而 CMDB 中的每一筆資料稱為組態項目(Configuration Item, CI) ，CI 可為硬體、軟體、網路、應用程式、服務、人、程序、相關的文件、事件、異動或問題等。成功建構 CMDB 的重要關鍵，就是 CI 的資料能被自動撈取且能追蹤其異動狀態，並有 CI 相互關聯關係與對應的負責人員，若 IT 服務有任何的異動，需記錄並維護於 CMDB 中。CMDB 的資料需被正確維護，方能確保 ITIL 服務支援與服務傳遞中各模組的正確運作。

如圖 3 所示，ITIL 整體是以 CMDB 為核心[6]，環繞 ITIL 其他元素如服務台、事件管理、問題管理、異動管理等模組。CMDB 好比集中的 IT 資料庫，用以存放 IT 管理中必要的元素。例如服務台人員為了協助使用者解決 ERP 系統異常，他必須要能知道(根據 CMDB 資料)公司的 ERP 系統是包含哪些 IT 物件(包含軟體、硬體、網路等)，以及這些物件之間的相互影響關係。關係的註明有助於避免意想不到的問題，例如更新資料庫系統的修補程式，便可能會對架構於其上的應用程式產生影響。企業 IT 服務運作出了任何狀況，例如製造執行系統無法使用，就可以查詢 CMDB 資訊，找出最近系統是否有進行任何異動或有組態更改狀況，確認是不是因為資料庫更新修補程式的而造成問題，以及可能還有哪些其他部分受影響。

透過 CMDB 記錄每次處理事情的過程與結果，可留下經驗值與知識資料，減少未來同樣狀況再發生時的問題釐清過程。長期而言，IT 管理者可以利用如資料探勘等方法，分析哪些應用或服務的事件發生機率高、或是何種情境下易發生事件，藉此判斷問題是出在軟硬體本身，或是使用方法不對，若是軟硬體本身則可能以升級或更換軟硬體方式改善；若是操作方式不當，則可以提供做為人員知識分享與教育訓練使用。

Sharifi 等人[18]整理出從上至下(Top-Down)、由下而上(Buttom-Up)、反覆(Iterative)與特定模式(Ad hoc)等四種方法來建立 CMDB，並提出降低 CMDB 與服務品質對應使用者需求的認知鴻溝(Gap)上需注意的觀點。在本研究中，也將根據 ITIL 的組態管理指引要點，利用自動化收集與稽核 CI 的組態資訊方式，先建立組態狀態基準(Baseline)資料，做為後續組態資料稽核或異動比較時使用，來協助事件管理流程的進行。

## 2.2 事件關聯 (Event Correlation)

根據 Jakobson 等人[19]提出的定義，事件關聯是一個概念性的轉譯程序，此程序當在事先定義的時間範圍內發現一組特定的事件時，便指定新的意義給此組事件。實現事件關聯的應用軟體稱為事件關聯器(Event Correlator)，在轉譯的程序中，關聯器可能會給定新的意義並且將原本的事件隱藏。事件關聯原先使用在重要的系統上如電力、瓦斯、水與油等分散式系統上[20]，現在被廣泛地使用於各類型用途，如錯誤診斷、安全管理、系統管理、效能監測與監控等類型[21][22]。

### 2.2.1 事件關聯方法 (Event Correlation Approaches)

目前已有多種事件關聯的方法，包括 Rule-based [23]，Graph-based [24]，Codebook-based [25]， Case-based[26] 等等，以下就這些方法說明：

- 規則庫系統(Rule-based approach):

Rule-based 的“Rule”代表為「規則」，而規則是從各領域的專家或是問題解決文件中所萃取出的專門知識而形成[27]。在規則庫系統中，會有一組規則是用來確認形成關聯的關係，這些規則會訂定“IF(條件成立)，THEN(執行特定行動)”的狀況，條件包含收到的事件連同有關系統的狀態等資訊，而最後的結論行動可能包括變更系統的行動，或是可以做為其他規則的輸入來源[23]。

規則庫系統是目前最常見的一種專家系統，也是一般常被使用作為事件關聯的方法，市面上的產品如 HP ECS 與 RuleCore 即是運用此種方法開發成[22]。規則庫系統最主要的組成元件如圖 5 所示，其中“Knowledge Base”是所有知識的儲藏庫，儲存陳述性知識(“Fact”代表所定義的物件與情況等)或程序的知識(“Rules”代表對於特定情況的建議、指示或策略)於系統中；“Inference Engine”是規則庫系統的中央控制元件，主要的用途是確認“Rules”與“Fact”如何被用來解決問題，一般都是用觸發式的方法來啟動尋找知識庫行動，並決定哪一個規則被執行，在找到可接受的解決方式時再進行確認；“User Interface”的部份是用來與其他系統或使用者溝通的介面[27]。

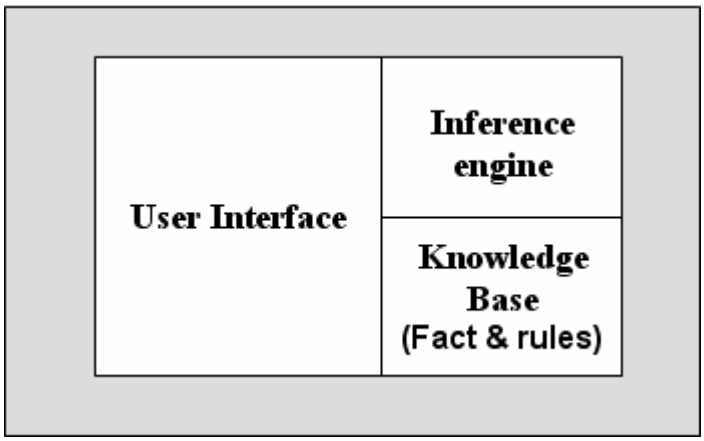


圖 5 規則庫系統組成元件

資料來源：Gardner and Harle[27]

規則庫系統的優點[22]是容易將專業知識轉化為規則，且於其他關聯方法比較而言，規則庫方法對使用者而言是較清楚且易瞭解的，因為系統中的規則或多或少是人可直接閱讀的格式，其運作過程與效果是直觀的，使用者的掌握度較高，Rich 等人[28]便提到，如果使用者不明白應用軟體為什麼以及是如何產生輸出結果，他們往往會漸漸忽略應用程式的計算結果。至於規則庫系統實際運作時可能會遭遇的問題[26][27][29]如，(1) 規則集可能會變得相當大，導致會有意想不到的規則相互影響，而使得系統難以維護，(2) 如果一個未知的情形一直沒有被規則所涵蓋，則此系統便會一直遺漏此未知情形的狀況，(3) 在規則會快速變動的領域中，規則庫系統就無法快速調整適用，很容易就被淘汰不用，(4) 沒有學習的功能，只適合應用於變化性不大的特定專業領域中。

應用規則庫系統時，需特別注意系統的易維護性，與妥善處理未知情況的這兩個要點，並且考量只使用於規則變化不大的環境中，若是變化性大的環境就需考量使用其他事件關聯系統。本研究中的 IT 服務環境變化不大，可適用規則庫系統。

- 圖形系統(Graph-based approach):

Graph-based 方法[24]是利用人力先分析與定義所有系統組件之間相依關係(如服務，主機，網絡設備等)，並以相依圖(Dependency Graph)表示。圖形中的每個節點代表著一個系統組件，圖形中每條線所連接的兩個節點代表有相依關係，而「節點 A 依靠(Depend on)節點 B」的相依關係，表示當節點 B 發生異常時，也會導致節點 A 發生異常的狀況。當一組事件發生時，該圖可用於找出事件的根本原因[30]，如 HTTP 服務器沒有回應的事件，是因為一個單一的網路連接有問題所造成的。

相依圖的運作方式如圖 6 所示[24]，一開始，事件關聯器收到原始事件並對應到相依圖中的物件，輸入的事件在此稱為故障訊息，對應到物件是為異常狀態。接下來，便開始搜索這些物件在依賴圖中有相依關係的物件狀況，有共同相依性的物件會被記錄作為濃縮事件(Condensed Event)，即不用重複檢測，在此演算法運作過程中，節點物件會被標記為正確(Correct)與異常(Faulty)兩種狀態。因此做關聯處理後，會是一個包含所有接收事件的分析結果，清單中的物件會被解釋為是否可能造成此問題之症狀的物件。如果進展順利的關聯處理，至少有一個物件被辨識是真正造成問題的點，而在相依圖中找到錯誤的品質衡量方式，是取決於在相依圖中，從原始事件一直到找到此造成根源問題之錯誤物件的路徑長度，如同 IT 管理者根據異常現象尋找事件問題根源的方式一般。

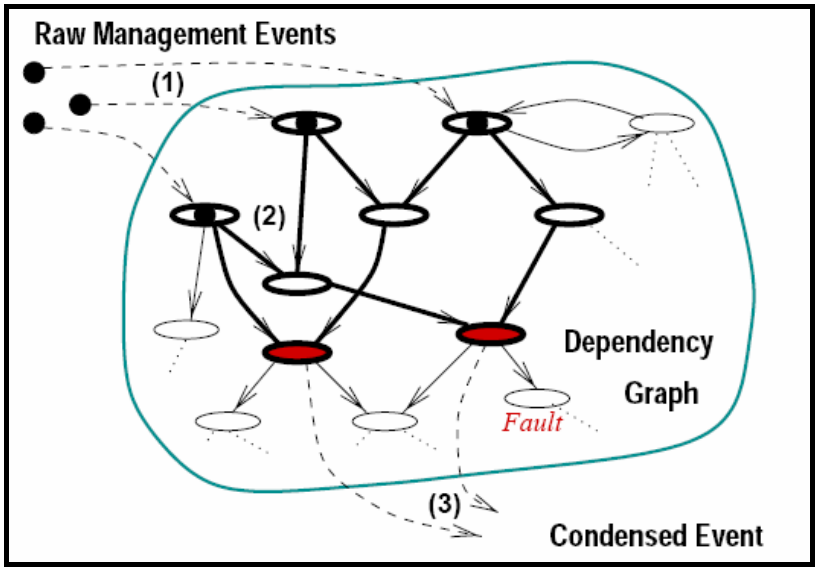


圖 6　相依圖

資料來源：Boris[24]

使用圖形系統作為事件關聯方法的好處如[24]，(1) 相依圖的圖形容易產生，(2) 可實現堅固的操作方式，如新增或刪除節點或節點之間的相依關係，不會造成關聯器整個運作異常，(3) 相依圖的管理是分散式的觀念，節點物件與相依關係的建立可由各個管理者來進行，且可為互相獨立的狀況。但相依圖對於「如果事件 A 發生就等待 30 秒看事件 B 是否發生，如果事件 B 發生就兩個事件都忽略，不然就開始進行事件 A 的處理」這種事件關聯規則的處理較難以達成。

相依圖是先由人力進行相依關係的建立，在本研究中的事件管理使用者介面資料呈現，也將運用如同相依圖的觀念，先建立 IT 服務各物件的相依關係，運用如同物料清單(Bill of Material, BOM)可展開方式，依序展開 IT 服務相關的各階物件，查看各階物件異常資料，並將底階物件異常時的影響，呈現在 IT 服務物件狀態上。

- 編碼簿系統(Codebook-based approach):

Codebook 方法[25]是根據編碼(Coding)與圖形(Graphs)的方式經驗而來，EMC SMARTS 商用軟體即是使用此種方式開發而成[22]。

編碼的概念就是[31]：每一個問題會造成許多症狀事件，症狀事件包含有問題物件本身和此事件所影響相關之物件的事件，將問題所影響的所有症狀事件進行編碼(Code)，就可用來代表此問題，而編碼簿系統的關聯動作就是將觀察到的所有症狀事件進行解碼(Decode)，確認哪一個問題有這些症狀事件，進而取得真正的問題原因。

關聯動作注重著分析事件之間因果關係[31]，而圖形方式的標記是如“A→B”這種形式，“A→B”代表事件 A 為因，而事件 B 為果，其中箭頭符號就是代表兩個事件的因果對應關係。若將所有的事件與因果關係建構起來的圖就如圖 7 所示，圖中的 S 代表症狀事件(Symptom)，P 代表問題事件(Problem)，箭頭符號代表兩節點之間的因果關係。圖 7 的優化方式為，(1) 事件 3、4 與 5 形成一個循環的事件因果關係，故可以整合為一個事件，(2) 沒有任何症狀事件或問題事件的節點亦可消除(如圖中的事件 8)，(3) 無任何問題事件直接造成症狀事件 7 與 10 的影響，故此兩個事件節點可被刪除。原始因果圖經去蕪存菁過程後的結果，就如圖 8 所示。

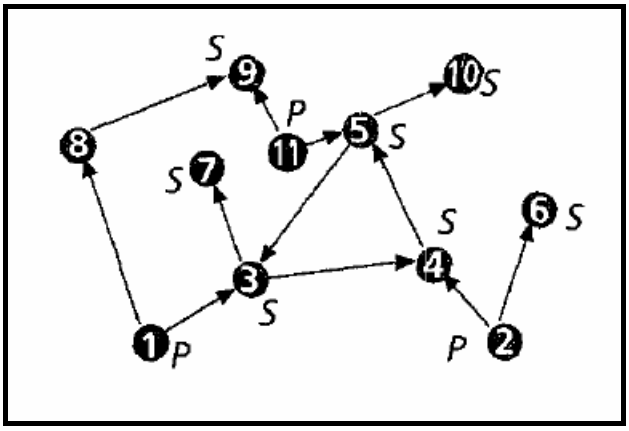


圖 7 編碼簿系統事件因果圖

資料來源：Yemini 等人[31]

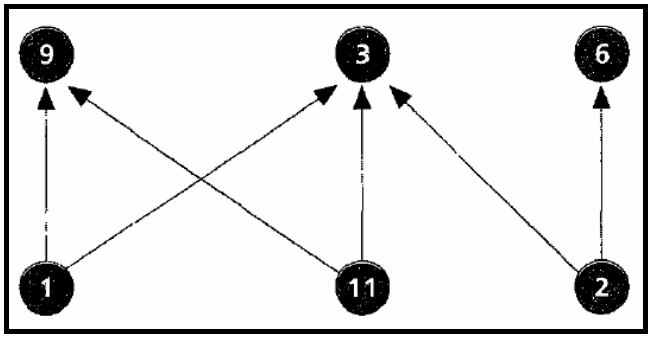


圖 8 優化後的編碼簿系統事件因果圖

資料來源：Yemini 等人[31]

整體的編碼簿系統運作方式，在輸入部分是用因果圖(Causality Graphs)方式[29] [31]，此因果圖包含事件、造成問題的事件節點與用來表達因果關係的箭頭符號，經過將圖優化的步驟後，再將此圖轉化為相關性的矩陣。此矩陣內的縱行代表問題根源的節點，橫列代表事件，資料內容為 0 或 1，用來代表事件與問題根源是否有關係存在，而 0 和 1 之間的值可用來指出相關聯的強度或可能性[29]。而編碼理論(Coding)的技術可用來將此矩陣做優化，例：如若一些事件並不會影響問題根源點的辨別，則可刪除縱行的事件資料。

與規則庫系統比較的話，此方法的優點是可以在某些情況下，來處理一些未知的組合事件狀況[29]，這些未知的組合事件，可用漢明距離(Hamming Distance)方法計算相似度，對應到已知的事件組合；相對而言，規則庫系統會比編碼方式較有彈性。Hamming Distance[32]是指兩個字碼之間相異的位元數，給予兩個任何的字碼，例如 10001001 和 10110001，即可決定有多少個相對位元是不一樣的，在此例中有三個位元不同。要決定有多少個位元不同，只需將 exclusive OR 運算加諸於兩個字碼就可以，並在結果中計算有多個為 1 的位元，各字碼與比較基準的字碼其漢明距離越少相異的位元，代表此字碼與基準字碼越相似。

- 案例庫系統(Case-based approach):

Case-based 的“Cases”代表是「問題的情況」，Aamodt 等人[33]認為案例庫方法是「根據曾發生的類似情況，重複使用以前的解決方式和知識，來解決新發生的問題」。就像一般人常使用的解決問題方式，當遇到一個新問題時，會先回想以前處理類似問題的方法與經驗，若是錯誤的方式就不嘗試，而使用能解決問題的方式或步驟來處理此新的問題狀況。案例庫系統能使用之前特定專業領域的經驗知識，將問題狀況具體化並儲存於案例庫中，而另一個特點就是能夠漸進且持續地學習問題解決狀況，當有解決新問題的經驗時，案例庫系統便可立即使用此新的經驗來解決後續發生的類似問題，也會記住解決問題的失敗經驗，避免以後再發生同樣錯誤嘗試。

一般案例庫方法的運作過程(圖 9)，可由下列 Aamodt 等人[33]所提的四個循環過程來描述：

1. 從新問題的描述資訊，利用相似度計算方式，從案例庫中檢索(Retrieve)出最相似的案例。
2. 重複使用(Reuse)過去案例的知識和資訊，來提供問題的解決方案。
3. 若之前經驗可正確解決問題時，便依此經驗修訂(Revise)先前所提議的解決辦法。若無法正確的解決問題時，則便需從專業知識中尋找可能的解決方案，並評估解決方案，若能成功解決問題，再依此經驗修訂先前所提議的解決辦法。
4. 保留(Retain)此新經驗中有用的部份至案例庫中，失敗的經驗也會被保留至系統中，做為未來的問題解決經驗。

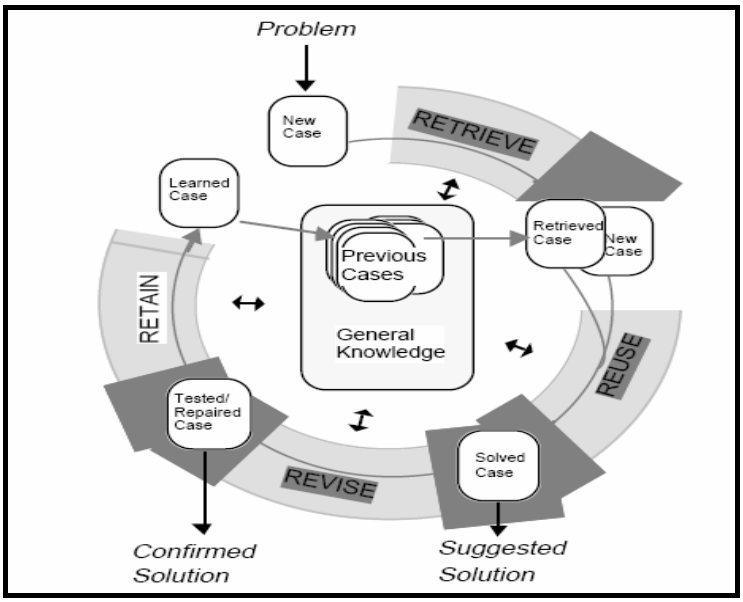


圖 9　案例推理流程

資料來源：Aamodt and Plaza [33]

案例庫系統的目標[26]是，(1) 從經驗中學習，(2) 根據過去經驗提供解決新問題的方案，(3) 只需小量的維護動作，不需頻繁且廣泛的維護操作。與其他事件關聯方法比較[22] [26][27]，案例庫系統的優點是(1) 能學習，(2) 易維護，(3) 知識的表達較無限制，(4) 能取得問題的潛在解決方案並進行評估，(5) 可調整現存解決方案的知識等。缺點是(1) 演算法效能較差，(2) 當案例不存在時，建立案例資料非常耗時，(3) 沒有一體通用的案例庫系統方法，必須針對每一個領域的應用去量身訂做。

在 Hanemann[34]的研究中，便從 Maintainability(可維護性)、Modeling(相依性與相關性的模型建構)、Robustness(強健程度，在不充分的資訊情況下，系統是否仍可進行問題解決的運作)與 Performance(執行效能)這四個層面，來比較事件關聯方法(規則庫、編碼簿與案例庫)應用於 IT 服務事件的優劣，其結論為案例庫的可維護性最好，編碼簿的模型建構最差，強健程度是案例庫較優而規則庫最差，效能則以案例庫最差而規則庫與編碼簿的執行速度優良。

綜合而言，本研究中的 IT 服務範圍是較固定且不常有異動，事件關聯項目同樣屬於相對穩定狀態，事件需處理的資料來源格式或項目，亦是由企業內部自行定義或由軟體提供的日誌資訊，就這些條件與情境上判斷，規則庫系統是適用做為 IT 服務事件關聯的運作系統，符合所需，效能也較佳。

### 2.2.2 Simple Event Correlator (SEC)

市面上已有一些功能強大的事件關聯系統(如 HP ECS, EMC SMARTS, 與 NerveCenter 等)[35]，但入門門檻高，安裝與設定過程不易且費時，價格也不便宜，一些商業用的關聯系統軟體有安裝平台的限制，有些只能應用於特定的環境中(例如 HP Openview ECS)。目前事件關聯已有很多相關的研究，並有一些雛型被開發出來，但大多都無法從網際網路上取得[30]，也沒有免費且發展成熟到足夠在企業正式環境中使用的事件關聯器。有鑑於此，Vaarandi [35]發表了一個輕量級且可運作在各平台上、使用關聯規則的事件關聯器 Simple Event Correlator(SEC)，SEC 主要的目標是建立一個開放原始碼的工具，具備處理集中化或分散在各端之事件關聯的能力，適用於多種作業系統，並能輕易地整合到其他的管理系統上，供填補自行程式開發與事件關聯商業軟體兩者之間的差距(Gap)，創造輕量級、容易自行客製調整與可廣泛使用的事件關聯器。

SEC 是一個使用 Rule-based 的事件關聯工具[36]，可從檔案系統接收一個輸入的事件，然後根據使用者指定的程式產生輸出結果。一般檔案、命名管線(Named Pipe)或者是標準輸入皆做為 SEC 的輸入。為了能夠處理任何格式與形式的輸入事件， SEC 使用常規表示式(Regular Expressions)來辨識事件，SEC 的常規表示式可用來比對

複雜且一次多行的樣式(Patterns)。此外，為了能適用於多種作業系統平台，所以 SEC 是利用 Perl 開發而成，Perl 能執行在大多數市占率高的作業系統上，且已變成許多作業系統預設會安裝的軟體，另外 Perl 程式語言中已整合了常規表示式。

一些事件關聯引擎，提供使用者事先定義好的關聯規則，可以被參數化使用，亦能像建立區塊般地定義出更複雜的事件關聯作業。SEC 採用相似的方式，來支援使用者建立自己的區塊，SEC 支援的規則類型如下所示[35][36][37]：

- **Single** – 匹配輸入的事件並執行一個行動。
- **SingleWithScript** – 匹配輸入的事件，並根據引發的外部 script 執行結果所回傳的代碼，來執行對應的行動。
- **SingleWithSuppress** – 匹配輸入的事件並執行行動，但是忽略接下來 t 秒內所匹配的輸入事件。
- **Pair** – 匹配輸入的事件並立刻執行行動，除非有接受到其他輸入的事件，否則忽略之後輸入的同樣事件，若有符合其他事件的輸入，則會執行第二個行動。
- **Pairwithwindow** – 匹配輸入的事件並等待其他輸入事件 t 秒。若其他事件沒有在給定的時間範圍內出現，則執行一個行動，若有事件於此時間範圍內抵達，則執行另一個行動。
- **SingleWithThreshold** – 在 t 秒內若匹配的事件次數有達到定義的門檻次數 n 以上，則執行一個行動。
- **SingleWith2Thresholds** – 在 t 秒內若匹配的事件次數有達到定義的門檻次數 n 以上，則執行一個行動，在執行後仍繼續統計次數，若在 u 秒內匹配的事件次數有達到定義的門檻次數 m，則執行另一個行動。

- **Suppress** – 忽略匹配的輸入事件。
- **Calendar** – 在特定的時間時執行一個行動。

而在 SEC 中所觸發的行動，不只是可以設計來產生輸出的事件，也能用來設計互動、儲存與管理知識、連接到外部的錯誤管理或知識分析模組等規則。下列的動作即 SEC 目前支援的類型[35][36][37]：

- **None** – 不執行任何行動。
- **Logonly** – 只記錄訊息。
- **Shellcmd** – 執行一個外部的 shell script 或程式。
- **Pipe** – 執行一個外部的 shell script 或程式，並且將產生的輸出作為標準的輸入使用。
- **Spawn** – 此行動將執行一個外部的 shell script 或程式，提供額外的輸入事件給 SEC。
- **Create** – 建立一個內容(context)，並且選擇性地設定它的參數內容(如存活時間)為非預設值。
- **Set** – 設定內容的參數值。
- **Delete** – 刪除一個內容。
- **Add** – 增加一個事件到內容中。
- **Report** – 提供內容中的所有事件，給外部的處理使用。
- **Event** – 產生一新的輸入事件，可供其他規則做比對。
- **Reset** – 取消一個事件關聯運作，例如重設正計數中的事件統計。

SEC 的規則範例如圖 10 所示[35]，此範例是檢查字串，當字串含有“INTERFACE”與“DOWN”的 pattern 時，則執行一個 shell command “notify.sh”，並將此字串值傳給 notify.sh 作為輸入參數，接下來的 86400 秒內，若有同樣的“INTERFACE”與“DOWN”字串

pattern 傳入則不再執行任何動作，不過若有“Interface”與“changed state to up”字串 pattern 傳入時，便執行 action2 的動作。

```
# If "<router> INTERFACE <interface> DOWN" event is
# received from the previous rule, send a notification and
# wait for "interface up" event for the next 24 hours.

type=Pair
ptype=RegExp
pattern=(\S+) INTERFACE (\S+) DOWN
desc=$1 interface $2 is down
action=shellcmd notify.sh "%s"
ptype2=RegExp
pattern2=($1) \d+: %LINK-3-UPDOWN: Interface ($2), changed
  state to up
desc2=$1 interface $2 is up
action2=shellcmd notify.sh "%s"
window=86400
```

圖 10　SEC 規則範例

資料來源：Vaarandi [35]

在 SEC 規則[35]中允許執行 shell 程式作為行動，也允許建立或刪除內容(Contexts)，在此內容中可以啟動或取消規則、可以產生新事件作為其他事件的輸入，也可以重設事件關聯的操作(如重設正在進行中的事件計數值)。若有需要，包含內容與布林表示式皆可被設定為規則的定義，利用結合數個有行動與布林表示式的規則，即能建立出符合複雜規則的事件。

SEC 已被使用於多種作業系統平台上如 LINUX、 Solaris、HP-UX、AIX、FreeBSD、Tru64 UNIX 與 Windows 等[30]，應用的領域包含：網路問題與效能管理、入侵偵測、日誌檔監控與分析[37]、詐騙偵測等等[35]；此外，SEC 的主程式只有 250 KB 的大小。因 SEC 事件關聯軟體規則設定簡易，利用常規表示式與其提供的規則類型，可建構出多樣性的事件關聯方式，本研究將使用 SEC 軟體，來協助事件管理中的自動分類步驟。

## 2.3 資訊檢索(Information Retrieval, IR)

根據 Baeza-Yates 等人[38]所述，資訊檢索是：「關於資訊項目(Information Item)的表示(Representation)、儲存(Storage)、組織(Organization)與存取(Access)，而資訊項目的表示與組織需能夠讓使用者根據其資訊需求(Information Need)進行簡易存取」。狹義的資訊檢索可以定義為[39]：「從大量未結構化的文件集合中(通常儲存在電腦)，取出符合資訊需求的文件資料」。而資訊檢索的演進[40]，可說自 1940 年代 Bush 所發表的開創性文章“As We May Think”後，便產生自動存取大量儲存知識的想法，接下來如利用電腦尋找文字、使用單字作為文件的索引與檢索系統評估工具 SMART (System for the Mechanical Analysis and Retrieval of Text) 的開發等研究，持續為資訊檢索領域注入顯著的發展；而 1990 年代由美國政府相關機構所贊助支持研究大量文件集合的 TREC(Text REtrieval Conference)，則發展出了許多資訊檢索領域的分支，包含如口語資料的檢索、非英語系語言的檢索、資訊過濾，以及使用者與檢索系統間之互動等。現在資訊世界不可或缺的網頁搜尋功能，便是由資訊檢索的演算法所達成。

### 2.3.1 資訊檢索系統

從資訊檢索的定義可延伸而知，資訊檢索系統是：「從大量未結構化的文件中，取出符合使用者資訊需求之文件的系統」，但如何讓電腦系統知道「使用者資訊需求」？以搜尋引擎的搜尋為例，若一個使用者的資訊需求為：「找出所有資訊檢索的研究資料，這些資料中需包含資訊檢索的定義，且必須是台灣研究者所發表的文章」，以現在搜尋引擎的技術而言仍無法直接按照使用者語句取得資料，必須將該使用者的資訊需求轉化為搜尋引擎查詢語句，此查詢語句一般稱為關鍵字(“Keywords” or “Index Terms”) [38]，再讓搜尋引擎從大量文件

中找出相關資料。資訊檢索系統的目標就是將使用者有需要或有相關的資料取出，其運作可簡略描述如圖 11 所示[41]，其中系統的輸入是使用者檢索(Queries)與文件(Documents)，經檢索處理後的輸出為相關的文件，並回饋給使用者。

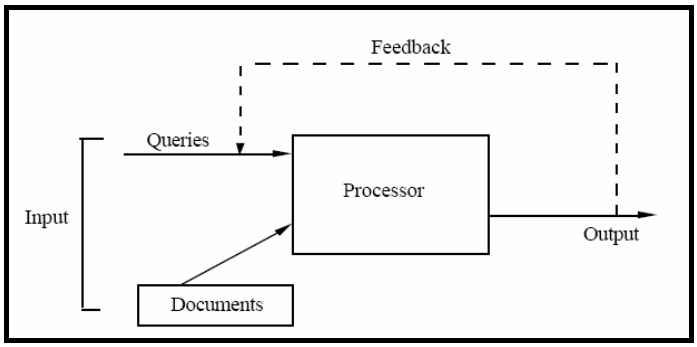


圖 11 資訊檢索系統

資料來源：Van Rijsbergen[41]

至於如何從未結構化的文件集合中取出符合資訊需求的文件資料？常用的方式就是用一組關鍵字來代表文件集合中的某份文件[41]，再透過辨識文件的關鍵字來快速確認此份文件是否符合資訊需求。一般對於未結構化文件的關鍵字處理如圖 12 所示[38]，文件經過斷詞切字、去除停用字(Stopwords)、去除名詞組(Noun Groups)、詞幹處理(Stemming)與索引(Indexing)等文件處理步驟後，即可取得代表此文件的關鍵字。

大部分資訊檢索系統的建置都是使用轉置檔(Inverted Files)的資料結構，轉置檔中存放的為關鍵字在於哪些文件中，並且包含其他相關的資訊[40]。文件處理中的索引步驟，就是在產生、建立與儲存文件特徵，而索引也就是對關鍵字進行處理，故經索引處理後產生的轉置檔稱為轉置索引(Inverted Index)[40]。使用轉置索引方式的優點是比較容易建置，且檢索效能好；而缺點是有額外儲存空間的負擔，且更新或重組索引的成本高[42]。

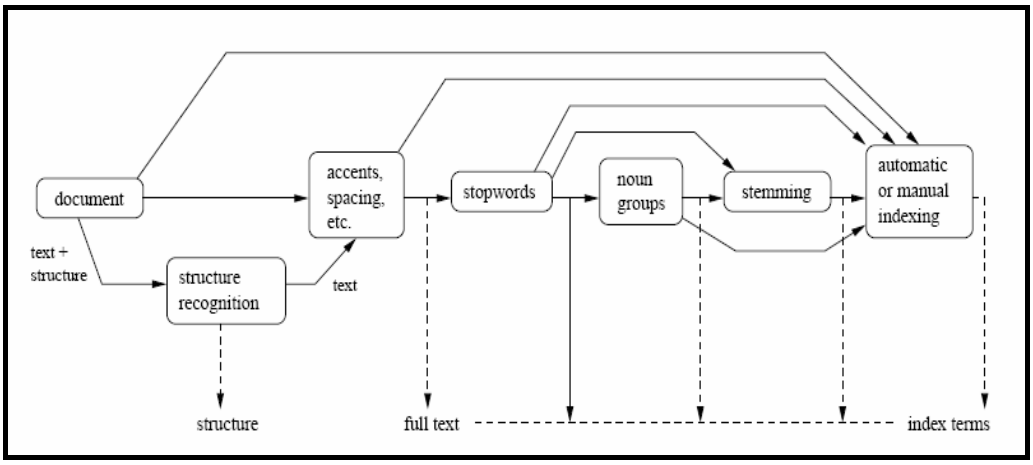


圖 12 文件轉化出關鍵字的邏輯檢視圖

資料來源：Baeza-Yates [38]

以本研究而言，事件診斷步驟中的新事件訊息可視為資訊需求，需從已發生的歷史事件訊息資料中做檢索，取出符合所需的歷史事件資訊。本研究環境中會遭遇事件含有中文字的情況，中文字無空白符號作為斷詞切字的依據，如將每個中文字單字作為關鍵字，會出現較多無相關但卻有相似字的狀況，若只以標點符號做為斷詞切字的依據，關鍵字詞則會包含過多中文單字，比對時容易遺失可能的相似資料，故若採用 N-gram 斷詞切字方法則可改善上述狀況[43]。在 N-gram 中，將相鄰的任意兩個字組合起來成為一個詞稱為 Bigram，任意三個字的組合稱為 Trigram，依此類推，任意 N 個字的組合便為 N-gram。

## 2.3.2 向量空間模型(Vector Space Model, VSM)

早期的資訊檢索系統是使用布林模式(Boolean Model)作為檢索模式，布林模式是一種以布林代數與集合理論為基礎的簡易檢索模式，其主要概念，即考慮檢索關鍵字是否存在於文件中。此模式允許使用者利用 AND、OR 與 NOT 等布林運算元建構出想要查詢語句，使用者感覺較能控制整個檢索處理，但不足的地方是[40]，(1) 無法

對取得文件做相關性排序，(2) 一般使用者不容易組合出適合的查詢語句。

向量空間模型是解決布林模式不足之處的一個替代方案[44]，在此資訊檢索模型中[45]，若文件空間中包含文件$D_i$，則有一個以上的關鍵字$T_j$來代表此文件$D_i$，這些關鍵字會根據其重要性被指定權重(Weight)等級，再轉化為向量表示。如圖 13，為三維的文件向量空間表示法，每一個文件都有三個不同的關鍵字代表。若將圖 13 擴展為$t$維度（$t$個不同的關鍵字），每個文件$D_i$的表示方式為$D_i=(d_{i1},d_{i2},...,d_{it})$，其中$d_{ij}$表示第$j$個關鍵字在$D_i$的權重。若將使用者的查詢語句也做文件向量處理，兩向量之間的夾角角度，則代表兩向量之間的相異性，夾角愈小代表兩向量愈相似，一般會用此夾角的餘弦(Cosine)作為量化相似度的依據[40]，因餘弦數值介於 1.0 到 0.0，而兩向量餘弦值為 1.0 代表兩向量完全相似，反之餘弦值為 0.0 代表兩向量完全不相似，故便可依此進行查詢語句與文件空間中各文件的相似度(Similarity)比對。

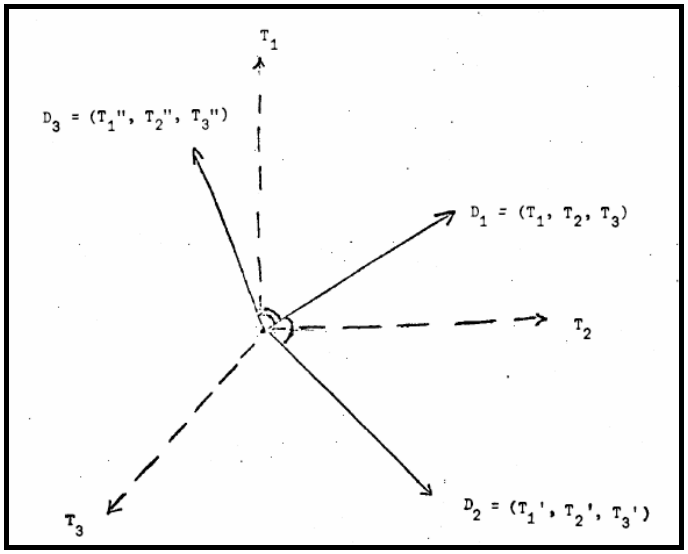


圖 13　文件空間向量表示方式

資料來源：Salton [45]

### 2.3.3 相似度(Similarity)

因資訊檢索中的查詢語句與文件集合中的文件可用一組關鍵字來代表，若以 $D_i=(d_1,d_2 \ldots d_n)$ 表示文件的關鍵字詞，$|D_i|$ 表示文件關鍵字詞的總數量，則 $|D_i \cap D_j|$ 代表同時出現在文件 $D_i$ 與文件 $D_j$ 的關鍵字詞數量，即 $D_i$ 與 $D_j$ 所共用的關鍵字詞數量，這乃是最簡單的相似度比對方式，但此最簡單的相似度方式未考慮 $D_i$ 與 $D_j$ 的總數，若應用於檢索系統中，$D_i$ 代表查詢語句關鍵字詞，$D_j$ 代表文件集中的每個文件的關鍵字詞，除非文件集合中的每個文件的關鍵字詞數量都一樣，否則所取出的共用關鍵字數量便無法作為有效的相似度衡量。而一般用來作為資訊檢索的相似度衡量的 Cosine coefficient、Dice coefficient、與 Jaccard coefficient 等[41]，皆有考量到關鍵字詞總數的部份，基本概念是利用文件之間共同擁有的關鍵字個數來計算相似度[46]，相似度的計算數值稱為係數(Coefficient)，範圍為 0.0 到 1.0 之間，1.0 的值代表完全相似的狀況，二元向量(Binary Vectors)相似度公式如下所示。

Dice coefficient $$\frac{2|D_i \cap D_j|}{|D_i|+|D_j|} \quad (1)$$

Cosine coefficient $$\frac{|D_i \cap D_j|}{|D_i|^{1/2} \times |D_j|^{1/2}} \quad (2)$$

Jaccard coefficient $$\frac{|D_i \cap D_j|}{|D_i|+|D_j|-|D_i \cap D_j|} \quad (3)$$

相似度計算公式中[46] [47]，Dice 係數及 Jaccard 係數比較偏集合的交集與聯集觀點，兩向量關鍵字詞交集數愈多，其相似度愈高，若是相同交集數之下，Dice 係數則比 Jaccard 係數有更高相似度值。

在 ITIL 事件管理模組流程中，事件診斷步驟是由服務台人員取得新事件相關資訊後，查找歷史相類似問題與解決方案，再經由所取得的解決方案來進行事件的處理與 IT 服務回復動作。本研究中將使用資訊檢索觀念，協助將事件管理診斷步驟自動化，新事件資訊將作為查詢文件，歷史事件知識資料庫作為比對檢出的文件集合，然後經文件邏輯處理步驟，進行去除符號字元與停用字的處理步驟，並建置新事件與歷史事件知識資料庫中的轉置索引，最後經由相似度計算公式，計算新事件與歷史知識事件的相似程度。本研究將會把事件檢索用於協助事件診斷，依相似度由高至低排序後取得的前三筆歷史事件資料，採用 Jaccard 相似度計算方式，從資料庫中找出相類似事件，呈現於前端 Web 介面供服務台於發現事件時，同時可取得事件解決方案。

# 三、 系統架構與設計

本章將闡述本論文所設計之自動化 ITIL 事件管理系統的架構與運作模式。第一節介紹整體系統架構與設計，第二節說明事件偵測的設計，第三節說明組態管理的設計，第四節敘述事件關聯的設計，第五節介紹本系統的事件資訊檢索設計。

## 3.1 系統概述

本節說明本研究系統實作目標、自動化事件管理架構與設計與研究範圍、限制。

### 3.1.1 系統實作目標

本研究以 ITIL 中的事件管理流程為研究對象，建構符合 ITIL 事件管理流程的指引與需求，且將事件管理所需活動內容，在可進行自動化的步驟上，如圖 14 事件管理活動中的前四項活動(Detection、Recording、Classification 與 Diagnosis)，探討運用相關技術與方式建構此系統，並提供 Web 管理介面給服務台人員查詢運用。

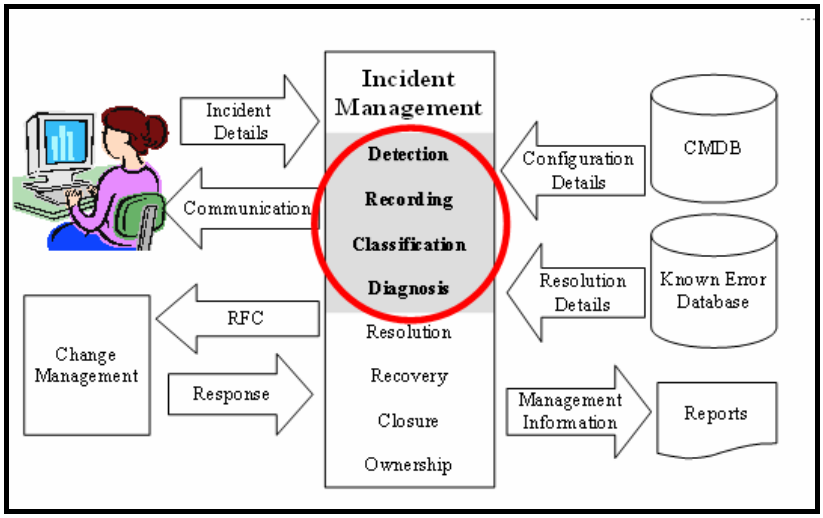


圖 14　本研究針對事件管理的前四項活動進行自動化

資料來源：本研究整理

### 3.1.2 系統架構與流程設計

本研究的系統架構圖如圖 15 所示，系統透過四個主要的代理人(Agent)來進行事件管理自動化系統的資料處理：

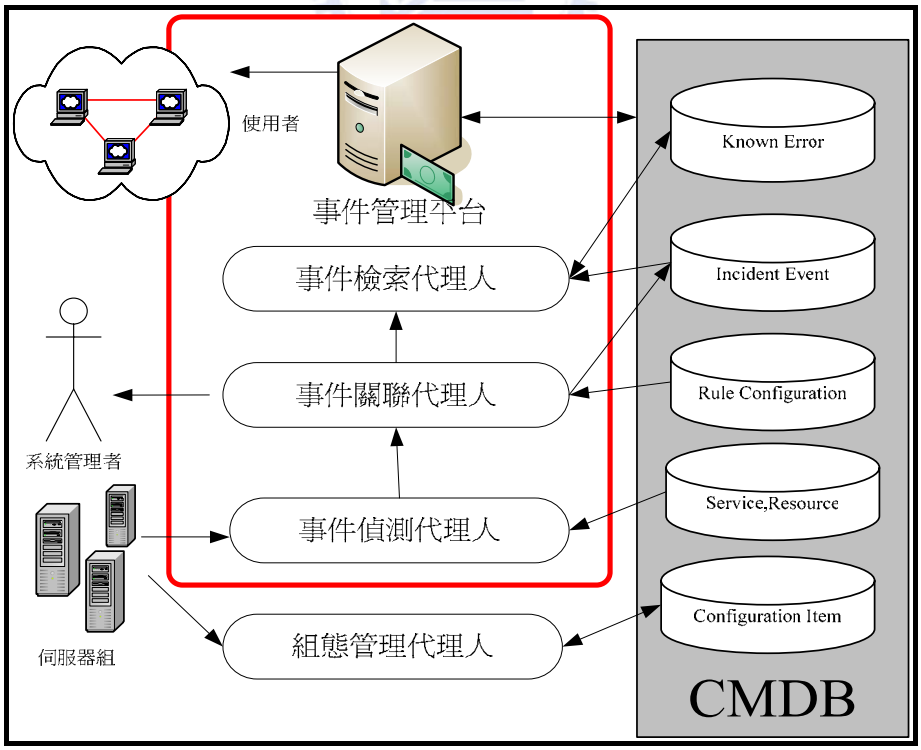


圖 15　系統架構圖

資料來源：本研究

1. 事件偵測代理人：負責偵測 IT 服務相關伺服器的服務與資源運作狀態，狀態包含程序(Process)、日誌檔(伺服器、資料庫、應用程式等)、物件網路回應、伺服器 CPU、伺服器記憶體與伺服器磁碟使用狀況等狀態，發現程序不存在於系統或是 Ping 物件無回應等等異常資訊。
2. 組態管理代理人：負責收集與稽核物件組態的狀態。組態收集程式根據收集的組態清單，進行物件組態資料的檢查與收集，並存放所取得的組態值至組態資料庫中。另外，物件若有組態異動狀況，組態稽核程式便會回報此物件組態異動資訊到物件組態庫中，並保留組態異動狀態資料，供服務台查詢物件組態異動前與異動後的設定值。
3. 事件關聯代理人：程式根據事先於本研究中所定義的 SEC 規則進行事件關聯處理。事件關聯程式會根據設定的事件規則，過濾與判定此事件的影響範圍、等級與嚴重程度，然後根據規則發送 Email 通知管理者，並儲存關聯處理後的事件資料至事件資料庫。
4. 事件檢索代理人：負責處理事件相似度資訊，並進行事件相似度比對。事件資訊檢索代理人取得經事件關聯處理後的新事件資料，便進行此新事件的檢索處理，一開始先將新事件的資訊去除停用字與進行索引處理，接下來會檢索先前已發生過，且被定義為可利用之事件資料知識庫(Known Error DB)中的事件資料，這些事件資料是由 IT 管理者妥善處理曾發生、已解決並有事件解決方式的事件資料資訊，代理人依據相似度計算公式，進行新事件與事件知識庫中資料的索引比對，再將相似程度最高的前三筆事件知識庫的資訊，回應至事件資料庫中的新事件資料。

系統整體架構流程設計以下列五個主要步驟來進行，其中包含將 ITIL 定義的事件管理如偵測、記錄、分類與診斷等活動自動化範圍：

步驟一：由事件偵測代理人定期自動進行 IT 服務相關物件的事件偵測，若有偵測到異常事件，便將事件資料傳遞給事件關聯步驟進行處理。此步驟可達成事件管理活動中，對於 IT 服務的 Detection 活動自動化。

步驟二：事件關聯代理人接收事件偵測代理人所傳遞的事件資料後，進行事件資料的關聯與分類處理，依定義規則將事件處理完成後，便將事件資料記錄到事件資料庫，並寄送 Email 給相關的 IT 管理人員，接著將事件交由事件檢索代理人進行檢索處理。此步驟可達成事件管理活動中的 Classification 與 Recording 活動的自動化。

步驟三：事件檢索代理人取得經事件關聯代理人處理的新事件後，進行新事件的檢索處理，再將相似度最高的前三筆已知問題事件知識資訊更新至事件資料庫中的新事件資料。此步驟可達成事件管理活動中 Diagnosis 步驟的自動化。

步驟四：服務台人員透過事件管理平台，從 Web 介面檢視 IT 服務異常事件，取得造成異常問題的物件問題根源，同時取得新事件的解決方案。而經由組態管理代理人取得的組態資料與組態異動稽核資料，也可經由事件管理平台來檢視與運用。

步驟五：新事件處理完成後，IT 管理者透過事件管理平台進行新事件知識處理，若此新事件是用現有的事件知識資料解決方式處理完成，則透過事件管理平台將此事件結案；若此新事件是以前未曾有過的經驗，或沒有適用的解決方式，則可經由管理平台將此新事件存入事件知識資料庫，此時事件檢索程

式便會將此事件知識資料進行索引處理，作為日後新事件的比對使用。

本研究架構後端資料庫中存放的 CMDB 資料，內容包含組態資料、IT 服務物件、事件與已知問題處理資訊等等。事件管理平台為資料呈現的介面，是從 CMDB 資料庫取得資料並提供使用者檢視與操作。事件管理平台的 Web 介面，是以 IT 服務為觀點，IT 服務底下的物件若有異常，問題根源的物件處會產生異常訊息的資料，影響範圍是利用 Bottom-Up 方式，由異常物件往上影響到所有相關聯的物件，一直到最上層物件，也就是使用者所使用的 IT 服務。IT 服務相關物件關係與物料清單(BOM)結構方式一樣，可從成品階依序展開下階料件，直到展開最底層的料件為止。

### 3.1.4 研究範圍與限制

本研究主要為探討 ITIL 事件管理自動化，另有部分組態管理的 CMDB 範圍，其他 ITIL 功能領域未進一步深入討論。事件管理偵測之研究範圍限定於 IT 服務中由系統自動偵測的事件，未包含 ITIL 事件管理中的其他事件來源，如使用者提出需求或使用者回報錯誤等等。使用者回報的錯誤，有部分屬於客戶端軟體使用問題，如個人電腦故障或 Office 軟體使用異常問題等等，不屬於本研究的範圍，其他使用者回報的 IT 服務異常錯誤，皆屬於本研究自動偵測所涵蓋範圍內，本研究以主動式進行自動偵測事件的處理，可更快速解決問題。

為符合低成本、跨平台與設定簡易需求，本研究使用開放原始碼(Open Source)的 SEC 軟體來建構與處理事件關聯，未採用商業型事件關聯處理器，因 SEC 是為採用 Rule-based 的事件關聯器建置，故只限定有定義的控制項目事件，且需由管理人員事先定義企業內欲處

理的規則。

另外，因事件檢索是以曾發生過的事件知識資料做相似度比對依據，故本系統需先取得或建置歷史事件資料，事件檢索部分的運作才能發揮功能。

## 3.2 事件偵測

本系統的事件偵測代理人，是使用安裝於 Windows 平台的 Activeperl、利用 Perl 程式語言開發而成。程式的配置方式，可於一台負責監測功能的伺服器上安裝所有事件偵測代理人程式，或者也可將代理人程式安裝在多台主機上進行監控。本系統事件偵測代理人程式使用的關鍵技術包含：

1. 利用 Perl 中的 Net::Telnet 模組建立代理人與各 UNIX 與 LINUX 平台主機的連線，並進行主機相關資訊的取得與處理；
2. 使用 IO::Socket::INET 模組，將代理人從監控的各類物件日誌檔，或是物件與服務相關狀態或設定所取得的資訊，透過 Socket 傳遞至 SEC 事件關聯器，交由 SEC 按照設定的規則進行關聯處理；
3. 使用 Win32::OLE 模組與 WMI(Windows Management Instrumentation ) 技術，使代理人可從遠端與各 Windows 作業系統溝通，取得系統上的各物件狀態，例如透過 Windows WMI 功能，可使用 Win32_Processor 方式取得 CPU 資訊，Win32_Service 則用來取得 Windows 服務的設定與目前啟動狀態資訊。

在這些技術其中，WMI 是取得物件資訊重要的部分，一般 IT

日常維護管理工作的第一步，都是取得系統的各項狀態資訊，Web Based Enterprise Management(WBEM)就是在此種考量下產生出來，它是由產業界的幾家軟硬體廠商，如 Intel，Microsoft，IBM，Oracle，Sun，BMC 等[48]，所共同訂定的一個有彈性、不限平台環境與可重複使用的基礎架構、工具與應用，目的是建立可在各類軟硬體的企業環境取得管理資訊的方法。它實際上是一組管理技術與 Internet 標準的組合，將許多已被業界廣泛使用的管理協定如 SNMP 等予以整合。WBEM 所採行的資訊管理架構，是由 DMTF(Distributed Management Task Force)所制定的 CIM(Common Information Model)。CIM 是一個在企業網路環境中，描述所有電腦系統與網路設備整體管理資訊的模型，包括了一組規格書(Specification)及一組綱要(Schema)。綱要定義了 CIM 模型的詳細描述，包括所有被管理物件及其描述方式。而規格書則定義了 CIM 模型與其他管理模型的整合。由於 CIM 只是一個物件的模型，各廠商均可以此模型提出各自的實作，而 Microsoft 所提出的實作，就稱之為 WMI。WMI 是一個三套式架構的模型[49]，由下而上分別是 WMI 資料提供者與管理物件(WMI Providers and Managed Objects) 、CIM 物件管理器(CIM Object Manager)以及 WMI 資訊請求者(WMI Consumers 與 Management Applications)，如圖 16 所示。

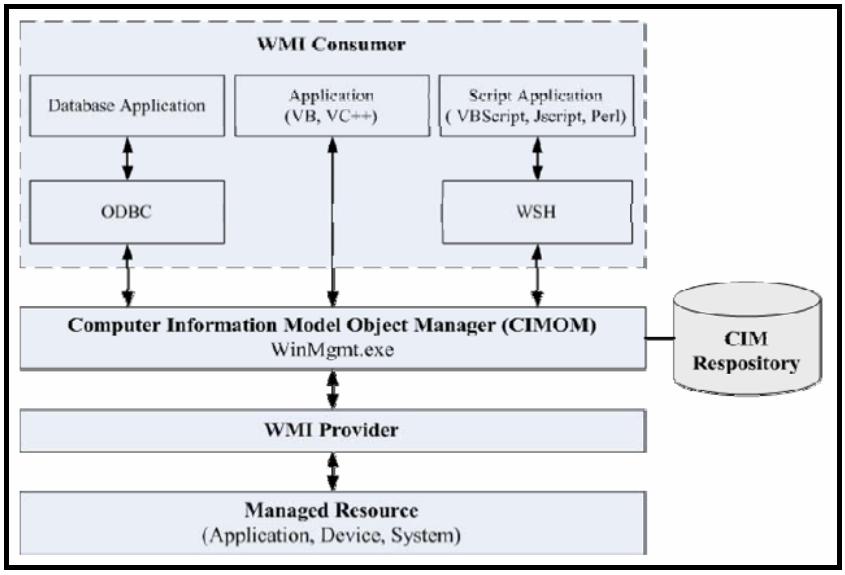


圖 16　WMI 架構圖

資料來源：Microsoft Corporation [49]

本研究之事件偵測物件類型如表 1 所示，分為網路、作業系統與應用程式三類，利用 Ping 伺服器或網路設備方式，判斷網路狀態回應；檢查作業系統相關的日誌與資源狀態，來辨識作業系統相關物件是否異常與確認是否為符合要求的品質環境；檢查應用程式日誌、相關服務與程序的狀態，確認是否異常。

表 1　事件偵測物件類型

| TYPE | CATEGORY | OBJECT |
|---|---|---|
| Network | Ping | Ping |
| OS | Log<br>Resource | Messages<br>Eventlog<br>Diskspace<br>CPU<br>Memory<br>Files |
| Application | Service<br>Log | Process<br>Service<br>SQL Server<br>Oracle DB |

資料來源：本研究

## 3.3 組態管理

組態管理代理人之開發方式與事件偵測代理人大致相似，於一台負責監測功能的伺服器上安裝所有組態代理人程式，再經由查詢取得 CMDB 中的物件項目，便可收集所需的物件相關資訊與進行組態資料稽核。所使用的關鍵技術也與事件偵測代理人類似，特別之處在於：

1. 使用 Win32::OLE 模組與 ADODB 技術，取得各種類的資料庫參數設定值(Oracle、MS SQL 資料庫等等)；
2. 使用 Win32::OLE 模組與 WMI 方式取得物件的組態資料，如使用 Win32_BIOS 取得 BIOS 資訊、Win32_OperatingSystem 取得 Windows 作業系統資訊(記憶體配置、作業系統版本等等)、Win32_Processor 方式取得 CPU 資訊(CPU 數量、速度與型號等)、Win32_Service 則用來取得 Windows 服務的設定資訊(啟動模式與啟動帳號等)；
3. 使用 Net::Telnet 模組來建立與 UNIX 或 LINUX 的作業系統連線，再執行查看系統資訊的指令，取得相關組態設定值。

組態收集與稽核程式，需監控的類型與項目如表 2 所示，不論是硬體(如 CPU 與記憶體)、作業系統的版本與修正程式、資料庫的參數值設定，以及應用軟體的設定或檔案異動日期等等，皆是組態管理代理人程式需管理的資料內容，利用此自動化組態監控，可提供系統組態資訊，減輕管理者的管理負擔，協助瞭解物件詳細資料，而組態異動稽核，則可以確認是否有未被允許的異動動作，提昇系統的穩定性。

表 2　組態管理物件

| TYPE | CATEGORY | NAME | CONFIG_NAME | VALUE |
|---|---|---|---|---|
| Server | Hardware | CPU<br>Disk<br>Memory | CPU Count<br>CPU Speed<br>Drive<br>Memory | Values |
| Server | Software | Service<br>OS | OS Version<br>Patch Version<br>Pagefile | Value |
| Database | Oracle | DBName | Parameter Name | Parameter |
| Database | SQL Server | DBName | Configure_Name<br>DB Option | Configure Value |
| File | Database Application System | DBName<br>Oracle AP<br>OS | init{SID}.ora<br>listener.ora<br>orapwd<br>hosts<br>passwd<br>httpd.conf<br>jserv.conf | Modified Date |

資料來源：本研究

## 3.4　事件關聯

本研究是利用 SEC 開放原始碼軟體進行事件關聯處理，利用 SEC 事件關聯規則方式，進行 ITIL 事件管理中事件分類步驟的自動化。例如，設定網路 Ping 監控規則，若無回應代表可能為網路異常或主機異常，若此服務為 7*24 小時必須正常運作的工廠製造執行生產系統，則嚴重程度為高，規則中便定義此事件為嚴重等級，並存放至事件資料庫，同時依照設定的 Email 寄送規則，寄送郵件給 IT 管理者通知此事件的發生，而管理平台查詢便可用圖形依嚴重等級呈現出易於辨識的資訊。其他如監控主機 CPU 使用率，若在 10 分鐘內連續兩次的使用率皆高於 80%，則為 YELLOW 等級，若 10 分鐘內的使

用率皆高於 90%，則為 RED 等級嚴重程度。事件嚴重程度分類如表 3 所示，其中 GREEN 代表 IT 服務或相關物件無異常，或是事件所有異常皆已修正或復原；YELLOW 表示系統未無法使用的狀態，但可能有隱藏的威脅存在，需要進行解決處理；RED 代表系統已有嚴重的錯誤，需立即進行事件解決。

表 3　事件關聯狀態

| 狀態 | 嚴重程度 | 是否需進行回復處理 |
|---|---|---|
| RED | Down | 需 |
| YELLOW | Warning | 需 |
| GREEN | No error<br>RED + Fixed<br>YELLOW+Fixed | 不需 |

資料來源：本研究

本研究使用 SEC 之規則類型如表 4 所示，其中 Single 規則用於處理本研究中大部分所需分類原則，如網路 Ping 狀態處理與 Process 異常處理等；SingleWithThreshold 用來設定 CPU 使用率於 10 分鐘內皆高於 90%的規則；SingleWithSuppress 規則類型則來處理與過濾設定的時間週期內，出現重複事件訊息干擾影響情況。

表 4　應用 SEC 之規則類型

| SEC 類型 | 應用內容 |
|---|---|
| Single | 網路 Ping 狀態、Process 異常狀態、Windows 服務異常狀態、網頁服務狀態異常等 |
| SingleWithThreshold | CPU 使用率於 10 分鐘內皆高於設定值 90%　等 |
| SingleWithSuppress | 作業系統日誌檔內容檢查，過濾相同重複訊息等 |

資料來源：本研究

本研究使用 SEC 之 Action 如表 5 所示，其中使用 event 來產生

一個判斷後增加新意義的輸出，作為下一個規則設定檔進行判斷處理的輸入，如嚴重程度便是經由第一個規則設定判斷處理後，增加嚴重程度到此規則的輸出字串，再交由下一個事件規則處理；使用 shellcmd 來寄送郵件通知管理者事件資訊；使用 write 將處理後的產生的事件關聯訊息寫至待處理的檔案中，再由程式依序將處理後的事件訊息資料寫入到事件資料庫。

表 5　應用 SEC 之 Action

| SEC Action | 應用內容 |
|---|---|
| event | 根據事件內容建立新事件 |
| shellcmd | 執行外部 script 內容 |
| write | 寫出事件資料至檔案 |

資料來源：本研究

圖 17 與圖 18 便是 Ping 伺服器狀態的判斷與處理規則範例，圖 17 是 SEC 判斷接收的訊息字串，若含有符合樣板(Pattern)所定義的字樣，即代表此物件 Ping 結果異常，此時便利用“event”的 action 類型，判斷與歸類此事件為 RED 等級的事件，將加入嚴重等級的字串作為輸出，再給下一個事件規則處理，在此同時執行 notify.sh 程式，此 notify.sh 負責寄送 Email 通知管理者新事件的資訊。而如圖 18 中的事件規則便負責於接收到此經判斷後的訊息，依所制定格式寫入到 FIFO 類型的檔案，再由處理程式寫入到資料庫中。

```
type=Single
ptype=RegExp
pattern=^(\S+\s+\S+)\s+(\S+)\s+NETWORK PING\s+(.*)
desc=$0
action=event 0 MONITOR:RED $2 NETWORK PING No Response at $1; \
       shellcmd /sedb/scripts/notify.sh $0 brian_chang@mail.simplo.com.tw
```

圖 17 Ping 伺服器狀態之事件規則

資料來源：本研究

```
type=Single
ptype=RegExp
pattern=^(\S+):RED\s+(\S+)\s+(\S+)\s+(\S+)\s+(.*)\s+at\s+(.*)
desc=$0
action=write %F ANL|RED|$1|$2|$3|$4|$5|$6
```

圖 18 Ping 伺服器狀態之寫入檔案事件規則

資料來源：本研究

## 3.5 事件檢索

ITIL 事件管理流程中的診斷步驟，是由服務平台人員，嘗試從問題處理資料庫中找出相類似問題的解決方案。一般設計都為利用關鍵字(Keyword)方式找尋相似文件或資料[15]，但在本研究中利用資訊檢索技術，來作為協助服務台人員自動尋找相類似事件的解決方案。本系統設計的事件檢索代理人程式會將事件依相似度高低程度排序，減少服務台人員進行事件搜尋與判斷處理，節省事件處理耗用過程與時間。本研究架構中的事件資料來自多種來源與格式，其中從 UNIX 與 LINUX 作業平台取得的事件訊息多為 20 個英文單字內，而 Windows 作業平台則因中英文版本皆有，故會有中英文的訊息與資料，所以在資訊檢索處理上，除了以空白、標點符號作為斷詞切字外，需再利用 n-gram 方式處理中文字串。本研究中因單一事件訊息字串內容單字數量不多，在英文字詞處理主要是使用空白與標點符號區分單字，而中文字詞的處理上使用 Bigram 方式做處理，再利用停用字消除不需的字元並作為切割依據，來減少建置索引所需的儲存空間，與查詢比對時需處理的資料量，提昇處理效能。本研究所使用的停用字如表 6 所示。

表 6　移除字元與停用字表

| 項目 | 內容 |
| --- | --- |
| 標點符號 | ，。；、( ) , : ? \| “ ” ，" 〈 〉 與空白符號等 |
| 其他 | 的，and，or |

資料來源：本研究

已知問題事件知識資料庫是新事件相似度比對的依據，這些已知的問題是 IT 服務相關物件曾發生過的事件，經由管理人員處理與復原系統完成，有詳細的事件處理步驟資訊，且被判定可作為知識保存的資料，再存放至已知問題事件知識資料庫中，事件檢索處理程式便會將此已知問題事件進行索引處理。當新事件發生並進入事件資料庫後，事件檢索程式便將此事件先進行索引處理，再利用 Jaccard 相似度計算公式，與已知問題資料庫的檢索索引資料進行相似度比對，比對後便依相似程度高低排序，將相似度最高的前三筆已知問題資訊，更新至事件資料庫中，服務台透過事件管理平台查詢檢視事件資料時，同時可取得相似度高之歷史已知問題事件資料的解決方式，依解決步驟進行事件解決。

# 四、系統實作與評估

本章將闡述本研究的系統建構成果，展示實作系統的操作畫面與相關功能說明。此外，並訪談科技業中 IT 管理人員，以取得該等人員所屬之企業的現行事件管理處理方式與步驟，瞭解訪談人員對於事件管理的認知，再經由實際展示本研究成果與說明目的與動機，取得訪談人員對於本實作系統在企業與個人工作上的助益之處，以評估本研究的效益。

## 4.1 系統實作成果

本節說明系統實作所使用的開發工具與軟體，並敘述本研究實作之自動化事件管理系統的各步驟。

### 4.1.1 相關開發軟體與功能

表 7　系統相關開發軟體

| 功能 | 相關軟體或程式語言 |
|---|---|
| 事件管理平台使用者操作 Web 介面 | Tomcat，JSP |
| 事件關聯器(Event Correlator) | Simple Event Correlator (SEC) |
| 事件偵測程式、事件資訊檢索程式、組態管理程式 | Perl (使用 Windows 平台的 Activeperl) |
| 資料庫 | MySQL |

資料來源：本研究

本系統使用 Tomcat 與 JSP 建構事件管理使用者操作平台；後端存放 CMDB 內容的資料庫，如事件資料、歷史事件資料與組態資料

等等，使用的是 MySQL 資料庫；事件關聯器利用開放原始碼的 SEC 軟體，並依本研究所需的條件與情況來設計事件規則；事件偵測、組態管理與事件資訊檢索處理等程式使用 Windows 版本的 ActivePerl 軟體開發。詳細的開發與建構軟體如表 7 所示。

### 4.1.2 自動事件偵測與組態管理

因功能或應用軟體的限制，一般的企業 IT 服務皆非全然為單一作業系統環境，故事件偵測程式必須能夠處理各種系統平台，而一般常用的伺服器作業系統為 IBM AIX、HP-UX、Linux 與 Windows 等。本研究自動事件偵測與組態管理之運作流程如圖 19 所示。

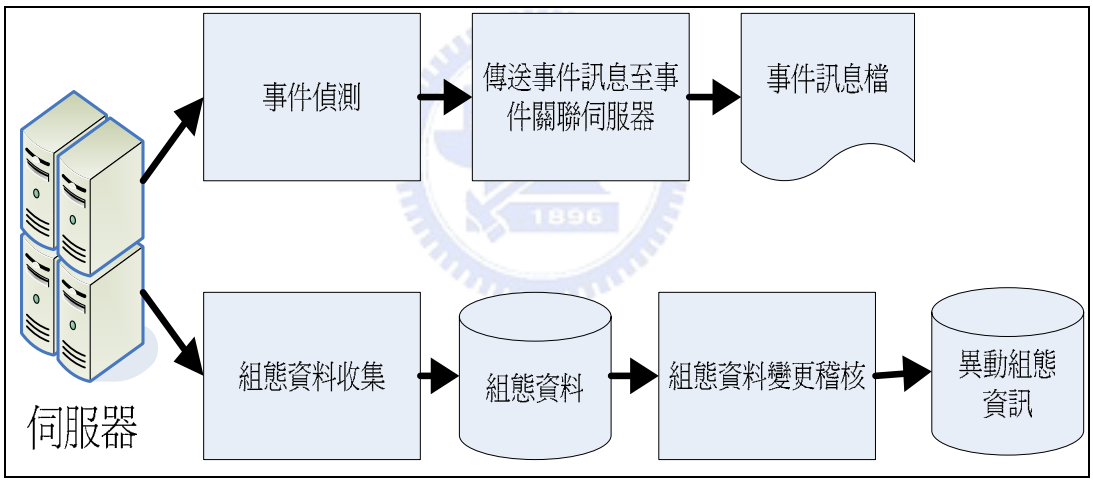


圖 19　自動事件偵測與組態管理流程

資料來源：本研究

如圖 19，自動事件偵測步驟為：

1. 透過事件偵測程式，依排程定期偵測事件。
2. 偵測到異常事件後，將事件訊息傳送到事件關聯伺服器，儲存至事件訊息檔案。

而組態管理運作步驟為：

1. 透過組態管理程式，依排程定期收集組態資料，並存放至組

態管理資料庫中。

2. 組態資料變更稽核程式依排程定期比對新舊組態資料，當有組態資料異動狀況，將組態異動資料儲存至組態管理資料庫。

實作上使用 Perl 程式語言開發事件偵測、組態管理程式，並將所有程式集中放置於一台 Windows 伺服器上，避免程式四處分散而難以維護，再依需求設定排程定期執行。本系統所開發的事件偵測與組態管理程式說明如表 8 所示。

表 8　事件偵測與組態管理程式

| 程式名稱 | 程式功能說明 |
|---|---|
| ChkCpuUsage.pl | 偵測伺服器 CPU 使用狀況是否有超過設定標準值。 |
| ChkDiskQuota.pl | 偵測伺服器 DISK 使用是否在安全的臨界值。 |
| ChkFileCount.pl | 偵測目錄中的檔案總數，是否超過設定臨界值。 |
| ChkOracleAPLog.pl | 檢查 Oracle ERP AP 的 errorlog 內容，取得有“error”的訊息資訊。 |
| ChkOracleDBLog.pl | 檢查 Oracle database 的 alertlog 內容，取得有“ORA-” 錯誤訊息的資訊。 |
| ChkOsEvent.pl | 檢查 Windows 伺服器的 eventlog 內容，取得類型為“錯誤”的訊息資訊。 |
| ChkOsMessages.pl | 檢查 UNIX 伺服器的 messages log 內容，取得“Crit、Warning”的訊息。 |
| ChkPingServer.pl | 利用 ping 伺服器方式，偵測伺服器回應狀態。 |
| ChkProcess.pl | 偵測各種重要的 Process 是否存在於 UNIX 系統中，如 Oracle 資料庫的 ora_smon process。 |
| ChkService.pl | 偵測 Windows 伺服器中的重要的服務狀態是否正常，如 MSSQL 資料庫的服務狀態必須為“啟動”狀態。 |
| ChkMssqlLog.pl | 檢查 MSSQL database 的 errorlog 內容，取得“錯誤”或“error”訊息內容資訊。 |
| ChkWebStatus.pl | 偵測網頁的資料回應內容，確認 Web-based 系統的網頁服務正常運作與否。 |

| 程式名稱 | 程式功能說明 |
| --- | --- |
| CmChangeAudit.pl | 比對各新、舊組態資料內容，若有異動則記錄至組態異動資料表中，並保存組態異動資訊內容。 |
| CmFile.pl | 取得各種重要設定檔之最後異動時間，如作業系統、資料庫、應用軟體等。 |
| CmMssql.pl | 取得 MSSQL 資料庫的組態資料，如資料庫參數與設定值。 |
| CmOracleDB.pl | 取得 Oracle 資料庫的組態資料，如資料庫參數與設定值。 |
| CmUnixOS.pl | 取得 UNIX 伺服器上，包含 CPU、記憶體、磁碟、作業系統版本與 Kernel 參數設定值等等組態資料。 |
| CmService.pl | 取得 Windows 伺服器上的所有服務資料與其設定值，如啟動模式與登入帳號等等。 |
| CmWindowsOS.pl | 取得 Windows 伺服器上，包含 CPU、記憶體、磁碟、作業系統版本等組態資料。 |

資料來源：本研究

事件偵測程式偵測到異常資訊後，會將訊息資訊利用 Socket 方式，傳送到事件關聯伺服器上，以本研究所建立之偵測伺服器 CPU 使用狀況的程式 ChkCpuUsage.pl 為例，詳細內容如表 9 所示，其中第二行 $host 代表欲監測的伺服器 IP，第三行 $threshold 為欲監測的 CPU 使用率上限值，第四行到第六行為建立所指定的遠端伺服器連線，並取得 WMI 中的 CPU 資訊，第十五行至第十九行 $avg_cpu_usage 乃取得平均的 CPU 使用率，若超過設定上線便執行寫出訊息的副程式，第二十一行至第三十一行副程式是將事件訊息，傳遞到事件伺服器的網路埠(Port)20097，由事件伺服器接收處理。本研究建構了 EventStoreListener.pl 程式在事件伺服器上，用來接收所有程式所傳遞的事件訊息，EventStoreListener.pl 使用網路埠(Port) 20097 來接收資料，再將收到的事件訊息資料存至檔案，由事件關聯進行後續的事件關聯處理。

表 9　ChkCpuUsage.pl 程式內容

| 行 | 程式碼內容 |
|---|---|
| 1 | use Win32::OLE qw(in with); |
| 2 | my $host= $ARGV[0]; |
| 3 | my $threshold= $ARGV[1]; |
| 4 | $WMI = Win32::OLE->new('WbemScripting.SWbemLocator'); |
| 5 | $Services = $WMI->ConnectServer($host); |
| 6 | $processor_set = $Services->InstancesOf("Win32_Processor"); |
| 7 | $proc_count=0; |
| 8 | $total_proc_perf=0; |
| 9 | foreach $proc (in($processor_set)) |
| 10 | { |
| 11 | $proc_perf = $proc->{'LoadPercentage'}; |
| 12 | $proc_count =$proc_count+1; |
| 13 | $total_proc_perf=$total_proc_perf+$proc_perf; |
| 14 | } |
| 15 | $avg_cpu_usage=int($total_proc_perf/$proc_count); |
| 16 | if ($avg_cpu_usage > $threshold) |
| 17 | { |
| 18 | &write_msg(); |
| 19 | } |
| 20 | sub write_msg { |
| 21 | use IO::Socket::INET; |
| 22 | my $MySocket=new IO::Socket::INET->new(PeerPort=>20097, Proto=> 'udp',PeerAddr=>'192.168.2.222'); |
| 23 | my $passstr; |
| 24 | my ($sec,$min,$hour,$day,$mon,$year,$wday) = localtime; |
| 25 | $year +=1900; |
| 26 | $mon +=1; |
| 27 | my $logmoddate="$year/$mon/$day $hour:$min:$sec"; |
| 28 | $passstr="$logmoddate\|$host\|OS\|CPU\|OVER THRESHOLD $threshold\% ,CURRENT USAGE IS $avg_cpu_usage\%"; |
| 30 | $MySocket->send($passstr); |
| 31 | } |

資料來源：本研究

組態收集程式會定期執行收集組態設定資訊，以收集 MSSQL 資料庫參數與設定值的 CmMssql.pl 程式為例，詳細內容如表 10 所示，其中第一行為欲收集組態資料的資料庫伺服器 IP，第二行為管理者名

稱，第四行到第六行是用 ADODB 建立與本研究中 MySQL 組態資料庫的連線，第七行到第九行是建立欲收集組態資料的 MSSQL 資料庫連線，第十六行到二十一行是執行 MSSQL 上的 sp_configure 預存程序，取得 MSSQL 整體資料庫於伺服器上的組態設定值，第二十二到第三十一行是將執行 sp_configure 所取得組態資料，交給 insert_data 副程式進行將資料存入 MySQL 組態資料庫動作，第三十二行到第三十七行是透過 MSSQL 的系統表 sysdatabases，取得在此資料庫伺服器的所有資料庫清單，第三十八到第五十八行是將所取得的資料庫清單，依序執行 sp_dboption 預存程序獲取各資料庫有啟動的資料庫組態設定值，第六十行到第七十二行是 insert_data 副程式內容，用來將取得的組態資料存入組態資料庫。

表 10　CmMssql.pl 程式內容

| 行 | 程式碼內容 |
|---|---|
| 1 | my $host= $ARGV[0]; |
| 2 | my $admin= $ARGV[1]; |
| 3 | use Win32::OLE; |
| 4 | my $connCMDBstr="DRIVER={MySQL ODBC 3.51 Driver};SERVER=192.168.2.222; PORT=3306;DATABASE=sec; USER=root; PASSWORD=; OPTION=3;"; |
| 5 | my $connCMDB = Win32::OLE-> new('ADODB.connection'); |
| 6 | $connCMDB-> Open($connCMDBstr); |
| 7 | my $connstr="driver={SQL Server};server=$host;AutoTranslate=No;Trusted_connection=yes;Network=dbnmpntw;Database=master;"; |
| 8 | my $conn = Win32::OLE-> new('ADODB.connection'); |
| 9 | $conn-> Open($connstr); |
| 10 | my $err = Win32::OLE::LastError(); |
| 11 | if (not $err eq "0") |
| 12 | { |
| 13 | print"FATAL: no connection, OLE error : $err\n"; |
| 14 | exit; |
| 15 | } |
| 16 | my $server_cfg = "sp_configure"; |
| 17 | if(! ($rsserver = $conn->Execute($server_cfg))) |

```
{
   print Win32::OLE->LastError();
   exit;
}
while (! $rsserver->EOF)
{
      $object_type="DATABASE";
      $object_category="SQLSERVERDB";
      $object_name="SYSTEM_CONFIG";
      $configure_name= $rsserver->Fields('name')->value;
      $configure_value= $rsserver->Fields('run_value')->value;
      &insert_data($object_type,$object_category,$object_name,$co
nfigure_name,$configure_value);
      $rsserver->MoveNext;
}
my $serverlist="select * from sysdatabases";
if(! ($rsdblist = $conn->Execute($serverlist)))
{
   print Win32::OLE->LastError();
   exit;
}
while (! $rsdblist->EOF)
{
   my $dbname=$rsdblist->Fields('name')->value;
   my $dboption="sp_dboption $dbname";
   if(! ($rsdboption = $conn->Execute($dboption)))
     {
          print Win32::OLE->LastError();
          exit;
     }
   while (! $rsdboption->EOF)
     {
        $object_type="DATABASE";
        $object_category="SQLSERVERDB";
        $object_name=$dbname;
        $configure_name= $rsdboption->Fields(0)->value;
        $configure_value="ON";
&insert_data($object_type,$object_category,$object_name,$config
ure_name,$configure_value);
        $rsdboption->MoveNext;
     }
   $rsdblist->MoveNext;
}
```

```
59
60  sub insert_data {
61  my $object_type=shift;
62  my $object_category=shift;
63  my $object_name=shift;
64  my $configure_name=shift;
65  my $configure_value=shift;
66  my $insertstr="insert into new_mo_config(host_ip,administrator,
    object_type,object_category,object_name,config_name,config_valu
    e) values('' . $host . '','' . $admin . '','' . $object_type . '','' .
    $object_category . '','' . $object_name . '','' . $configure_name .
    '','' . $configure_value . '')";
67  if(! $connCMDB->Execute($insertstr))
68  {
69     print Win32::OLE->LastError();
70     exit;
71  }
72  }
```

資料來源：本研究

### 4.1.3 自動事件關聯

事件偵測單純只是收集事件訊息傳送到事件關聯伺服器，由事件關聯伺服器處進行集中處理、判斷與關聯資訊。在本研究中的事件關聯實作運作流程如圖 20 所示，說明如下：

1. SEC 事件關聯主程式以事件偵測步驟所產生的事件訊息檔作為輸入資料，當新事件產生並傳送訊息到事件訊息檔後，SEC 主程式便會立即進行事件關聯處理。
2. SEC 根據所定義的事件關聯規則進行分析，並會寄送 Email 通知管理人員事件訊息資料，分析後所產生的事件關聯訊息，會儲存至 FIFO 檔案。
3. 由新增事件資料至資料庫的程式進行處理，將事件資訊儲存至事件資料庫中。

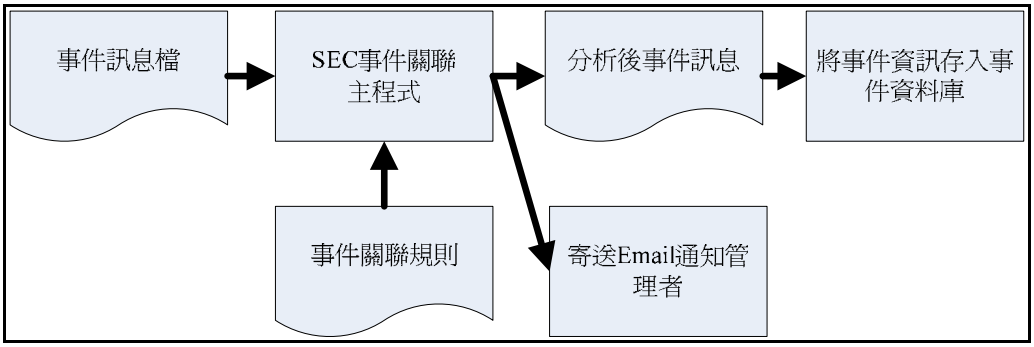


圖 20　事件關聯流程

資料來源：本研究

SEC 事件關聯啟動指令如圖 21 所示，sec.pl 為 SEC 主程式，conf 參數的兩個檔案 analyze.conf 與 anawfifo.conf 是本研究所制定的事件關聯規則檔，input 參數的 eventdata.log 檔案是伺服器上接收事件偵測的事件資料，程式啟動後即於背景執行，當有新事件產生到 eventdata.log 檔案後，SEC 程式便會馬上進行事件關聯處理。

```
perl  /sedb/sec.pl  \
-conf=/sedb/conf/analyze.conf \
-conf=/sedb/conf/anawfifo.conf \
-debug=4 \
-dump=/sedb/tmp/sec.dump \
-log=/sedb/tmp/sec.log \
-pid=/sedb/run/sec.pid \
-intevents \
-input=/sedb/event/eventdata.log \
-detach
```

圖 21　SEC 啟動指令與相關參數

資料來源：本研究

本研究中所建立兩個事件關聯規則檔案，第一個為 analyze.conf，部分內容如圖 22 所示，共設定 28 個規則。在圖 22 的 analyze.conf 規則中，action 部分是建立 event，此 event 會由下一個事件關聯規則檔案進行處理，規則中的另一個 action 便是執行 notify.sh，notify.sh 是將事件訊息資料傳送給 IT 管理人員。

```
type=Single
ptype=RegExp
pattern=^(\S+)\s+(\S+)\s+ORACLEAP ERROR_LOG\s+(.*)
desc=$0
action=event 0 MONITOR:YELLOW $2 ORACLEAP ERROR_LOG $2 at $1; \
       shellcmd /sedb/scripts/notify.sh $0

type=Single
ptype=RegExp
pattern=^(\S+\s+\S+)\s+(\S+)\s+ora_smon SERVICE PROCESS DOWN(.*)
desc=$0
action=event 0 MONITOR:RED $2 SERVICE ORACLEDB PROCESS DOWN at $1; \
       shellcmd /sedb/scripts/notify.sh $0

type=Single
ptype=RegExp
pattern=^(\S+\s+\S+)\s+(\S+)\s+tnslsnr SERVICE PROCESS DOWN(.*)
desc=$0
action=event 0 MONITOR:RED $2 SERVICE DBLISTENER PROCESS DOWN at $1; \
       shellcmd /sedb/scripts/notify.sh $0

type=Single
ptype=RegExp
pattern=^(\S+\s+\S+)\|(\S+)\|(\S+)\|WINDOWS SERVICE STOPPED(.*)
desc=$0
```

圖 22　事件規則 analyze.conf 部分內容

資料來源：本研究

在圖 22 的規則內容中定義了事件等級類別，若此事件是影響到 IT 服務無法正常運作的狀況(如資料庫 Process 不存在或是 Ping 伺服器無回應等等)，即為嚴重等級最高的事件。本研究事件關聯中將事件嚴重程度分為兩類，嚴重程度最高者會指定為“RED”，而程度次之者為“YELLOW”狀態，本研究所定義事件分類如表 11 所示。

第二個 SEC 規則設定檔為 anawfifo.conf，此設定檔中的規則是接收第一個設定檔分類與關聯處理後的資料，並將訊息以“|”符號作為欄位區分，存入 FIFO 類型的事件紀錄檔案 SEC_FIFO 中，規則內容如圖 23 所示。另外有 dbinsert.pl 程式也是啟動並於背景執行，此程式當發現 SEC_FIFO 中有新增的資料時，便將資料新增到事件資料庫中。

表 11　事件分類內容

| 類別 | 事件 |
|---|---|
| RED | 1. Ping 伺服器無回應<br>2. Linux 與 UNIX 的應用程式 Process 不存在<br>3. Windows 應用程式服務不為“啟動”狀態<br>4. 網頁服務回應異常<br>5. 備份作業失敗<br>6. Diskspace 使用率 95%<br>7. CPU 使用率持續 10 分鐘皆為 90% |
| YELLOW | 1. Windows 作業系統日誌檔錯誤訊息<br>2. Linux 與 UNIX 的 messages 日誌檔案錯誤訊息<br>3. 應用程式日誌檔錯誤訊息<br>4. Oracle 資料庫 alertlog 錯誤訊息<br>5. MS-SQL 資料庫 errorlog 錯誤訊息<br>6. Diskspace 使用率 90%<br>7. 指定目錄的檔案數超過設定上限 |

資料來源：本研究

```
type=Single
ptype=RegExp
pattern=^(\S+):RED\|(\S+)\|(\S+)\|(\S+\s+\S+\s+\S+)\|(.*)\s+at\|(.*)
desc=$0
action=write /sedb/fifo/SEC_FIFO ANL|RED|$1|$2|$3|$4|$5|$6

type=Single
ptype=RegExp
pattern=^(\S+):RED\s+(\S+)\s+(\S+)\s+(\S+)\s+(.*)\s+at\s+(.*)
desc=$0
action=write /sedb/fifo/SEC_FIFO ANL|RED|$1|$2|$3|$4|$5|$6

type=Single
ptype=RegExp
pattern=^(\S+):YELLOW\s+(\S+)\s+(\S+)\s+(\S+)\s+(.*)\s+at\s+(.*)
desc=$0
action=write /sedb/fifo/SEC_FIFO ANL|YELLOW|$1|$2|$3|$4|$5|$6
```

圖 23　事件規則 anawfifo.conf 部分內容

資料來源：本研究

### 4.1.4 自動事件檢索

本研究的自動事件檢索，是將新事件的訊息資料作為查詢語句，而比對文件資料內容為歷史已知問題且有解決方式的事件資料。實作上分為兩個部分來說明，第一部分為已解決事件問題的知識處理，第二部分是新事件的資訊檢索。

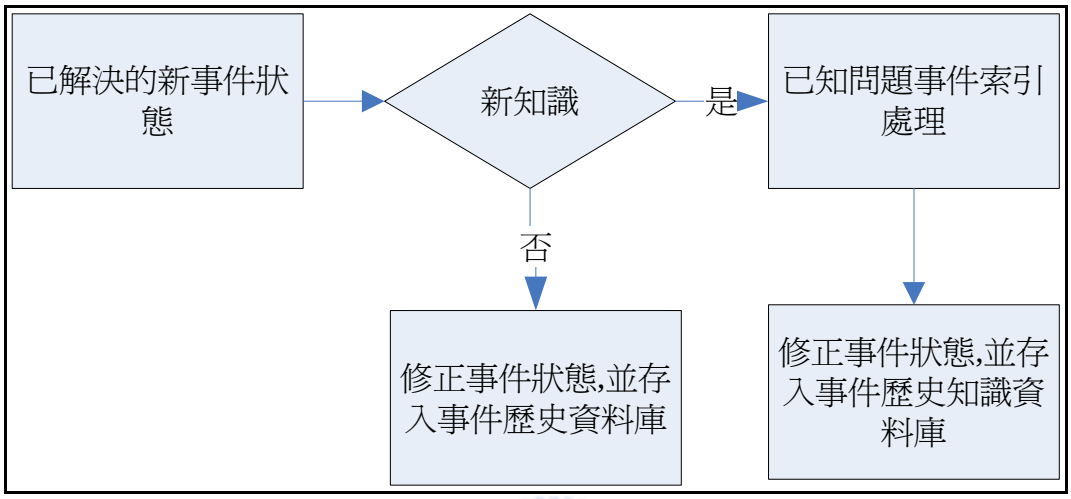


圖 24　歷史事件檢索處理流程

資料來源：本研究

第一部分流程如圖 24 所示，說明如下：

1. 經由事件管理平台的已解決事件處理介面，由 IT 管理人員透過操作介面，將事件解決的處理步驟，記錄到系統中。
2. 系統並會檢索出與此事件的相類似事件，利用類似事件的相似程度分數，管理人員可判斷此事件是否為事件知識庫中尚未具備的有用新知識。
3. 若是新的事件知識，則系統會變更事件狀態為解決狀態，並進行此事件的檢索索引處理，將事件存入歷史事件知識資料庫。
4. 若此事件為系統中已有解決方式的重複性事件，則將只變更事件狀態為解決狀態，事件存入歷史事件資料庫，不進行此事件檢索索引處理。

例如，新事件知識的訊息為 ”Source:Kerberos Event Code:4 Message:kerberos 用戶端從伺服器 THOMAS_HSUNB$ 收到 KRB_AP_ERR_MODIFIED 錯誤。”，此訊息經事件檢索處理後的索引資料如圖 25 所示內容，資料表中存放的 la_record_id 為事件資料的編號，inverted_wd 代表經處理的索引文字內容，作法是英文文字利用單字方式存放至資料表中，而中文則利用 bigram 方式處理，再存入資料表內，而 wd_count 代表此事件於索引表中的總索引字數，以便供相似度比對使用並避免新事件處理比對時的計算複雜度。

| la_record_id | inverted_wd | wd_count |
|---|---|---|
| 646 | Source | 17 |
| 646 | Kerberos | 17 |
| 646 | Event | 17 |
| 646 | Code | 17 |
| 646 | 4 | 17 |
| 646 | Message | 17 |
| 646 | kerberos | 17 |
| 646 | 用戶 | 17 |
| 646 | 戶端 | 17 |
| 646 | 端從 | 17 |
| 646 | 從伺 | 17 |
| 646 | 伺服 | 17 |
| 646 | 服器 | 17 |
| 646 | TYRONE_CHEN$ | 17 |
| 646 | 收到 | 17 |
| 646 | KRB_AP_ERR_MODIFIED | 17 |
| 646 | 錯誤 | 17 |

圖 25　事件檢索處理結果內容

資料來源：本研究

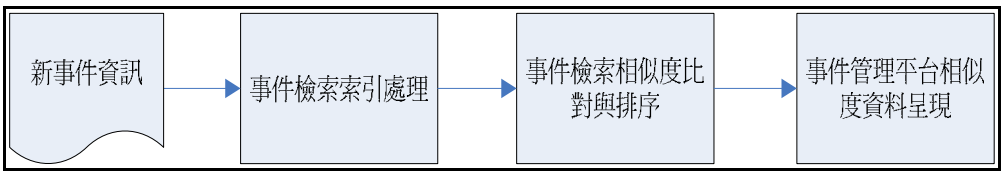


圖 26　事件檢索處理與相似度比對流程

資料來源：本研究

第二部分為新事件的事件檢索處理與比對流程，流程內容如圖 26 所示，說明如下：

1. 取得經自動偵測與事件關聯處理後的新事件。
2. 系統會將此新事件的資訊，作為查詢的資料處理，進行事件檢索索引處理步驟。
3. 索引處理後便進行與歷史事件資訊庫進行相似度比對，依照 Jaccard 相似度公式計算相似度，將最相似之前三筆歷史事件處理知識資訊，記錄為與此事件最相似的參考事件資料，更新至事件資料庫中。
4. 透過事件管理平台可查詢此新事件的相類似事件知識資料，並取得事件解決步驟。

圖 27 為新事件經本系統進行事件檢索處理後的資料內容，其中 la_hist_id1 欄位代表經檢索後相似程度最高的歷史事件，la_hist_id_score 是使用 Jaccard 相似度公式，計算新事件與 la_hist_id1 欄位中的歷史事件知識資料相似度分數，其餘 la_hist_id2 與 la_hist_id3 則與 la_hist_id1 一樣機制所取得的資料，系統會存放前三筆最相似的歷史事件知識資料，與其相似程度的資訊至事件資料庫中。

| _threat_text | la_hist_id1 | la_hist_id1_score | la_hist_id2 | la_hist_id2_score | la_hist_id3 | la_hist_id3_score | la_ir |
|---|---|---|---|---|---|---|---|
| Ource:Kerberos Event Code:4 | 23 | 100 | 646 | 94.44 | 13 | 29.27 | Y |

圖 27　事件相似度欄位資料內容

資料來源：本研究

從事件管理平台查詢時(圖 28)，可快速檢視經事件檢索處理後的事件資訊，進行事件管理中所需的診斷步驟，取得最相似的歷史事件資料，使用歷史事件解決步驟，進行 IT 服務的回復處理。在本研究中，便經由系統實作的事件檢索這兩部分，來達成事件管理中診斷步驟的自動化處理過程。

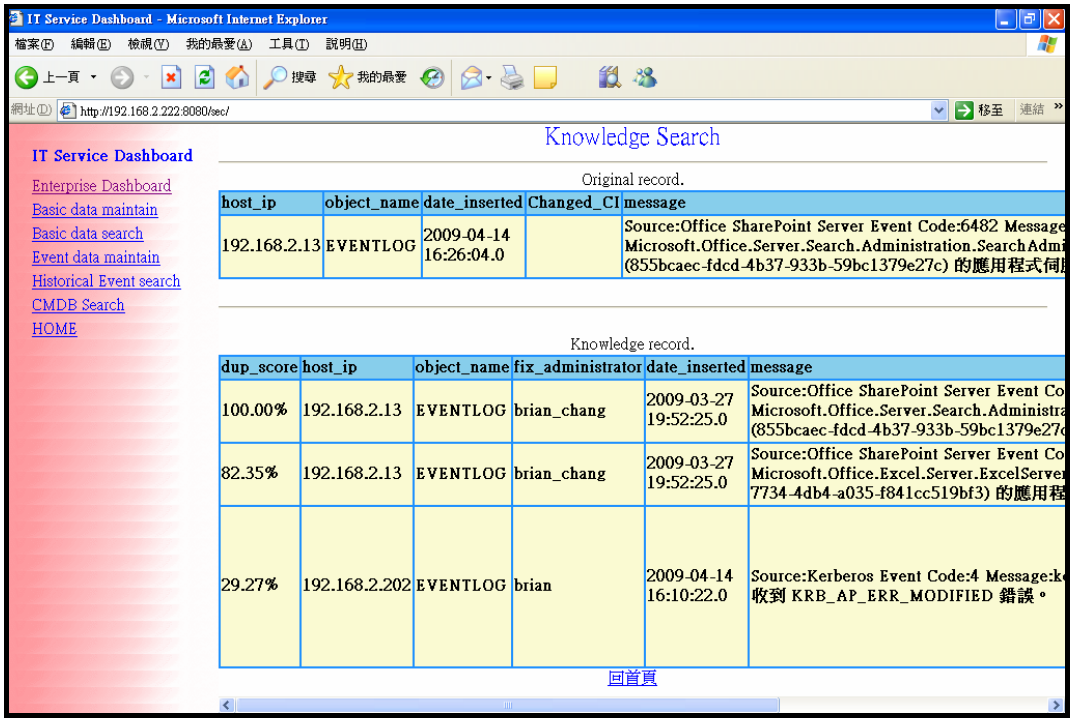


圖 28　歷史事件相似度

資料來源：本研究

### 4.1.5 事件管理操作平台

本節最後介紹本研究所建置自動事件管理操作平台的畫面與功能，圖 29 為系統起始畫面與各操作功能說明，“Enterprise Dashboard”可查看 IT 服務運行狀態，“Basic data maintain”是維護 IT 服務物件資料與階層關聯，“Basic data search”是查詢 IT 服務物件資料與階層關聯資料，“Event data maintain”是維護事件狀態與資料，“Historical Event search”用來查詢歷史事件資料，“CMDB Search”用來查詢組態資料。

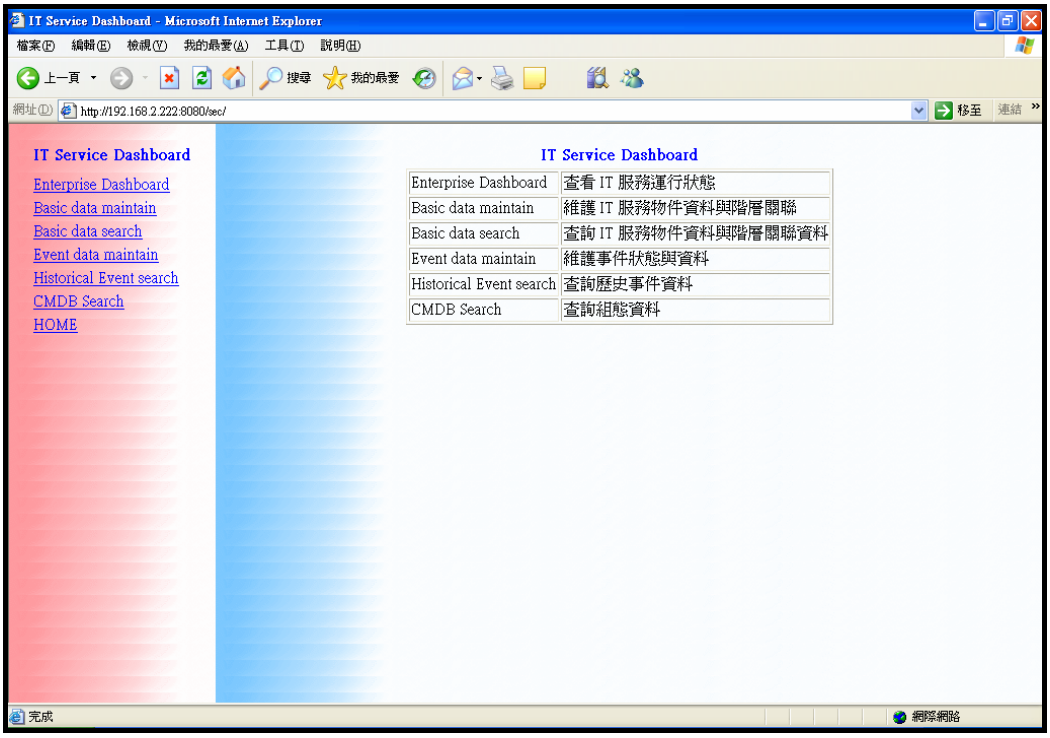


圖 29　IT 服務事件管理平台

資料來源：本研究

Enterprise Dashboard 內容如圖 30 所示，從 Enterprise Dashboard 可查得各 IT 服務目前運行狀態，在“status”欄位用紅、黃、綠燈圖示代表是否有異常事件，以及事件嚴重程度。若呈現紅燈圖示代表此服務相關物件有嚴重等級的異常事件，黃燈圖示代表此服務相關物件有需注意與處理的異常事件，綠燈圖示代表此 IT 服務相關物件運作正常，服務台人員可以經由此欄位資料，快速瞭解所有服務運行狀態。另外若與此 IT 服務相關的物件有組態異動，資料會呈現於“Changed_CI”欄位上，用來協助管理人員判斷與事件相關性，並可做為組態稽核使用，減少管理人員未經核准便異動組態資料的動作，增加系統穩定度，也避免因人為動作而造成 IT 服務無法使用的狀況發生。

接下來點選“Next_Lev”欄位查看下階物件的狀態，可依序一階一階展開來查詢相關物件，若是物件為事件來源，可看到此物件

“Messages”欄位有事件訊息內容(圖 31)，點選“KD”欄位便能查詢事件檢索的分析資料。

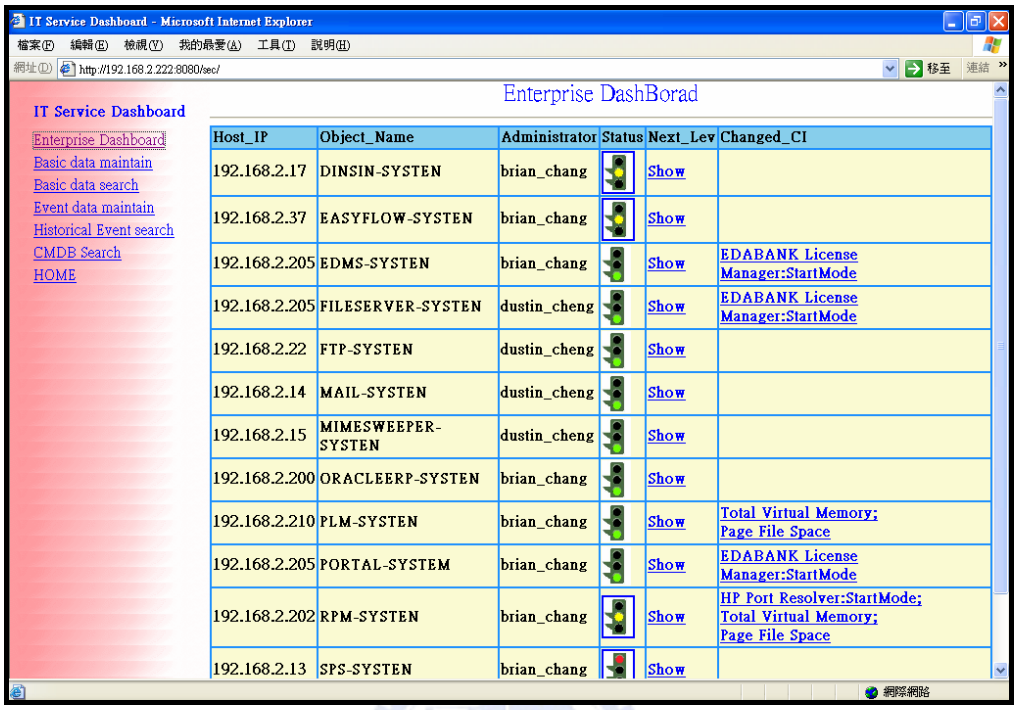


圖 30　IT 服務狀態

資料來源：本研究

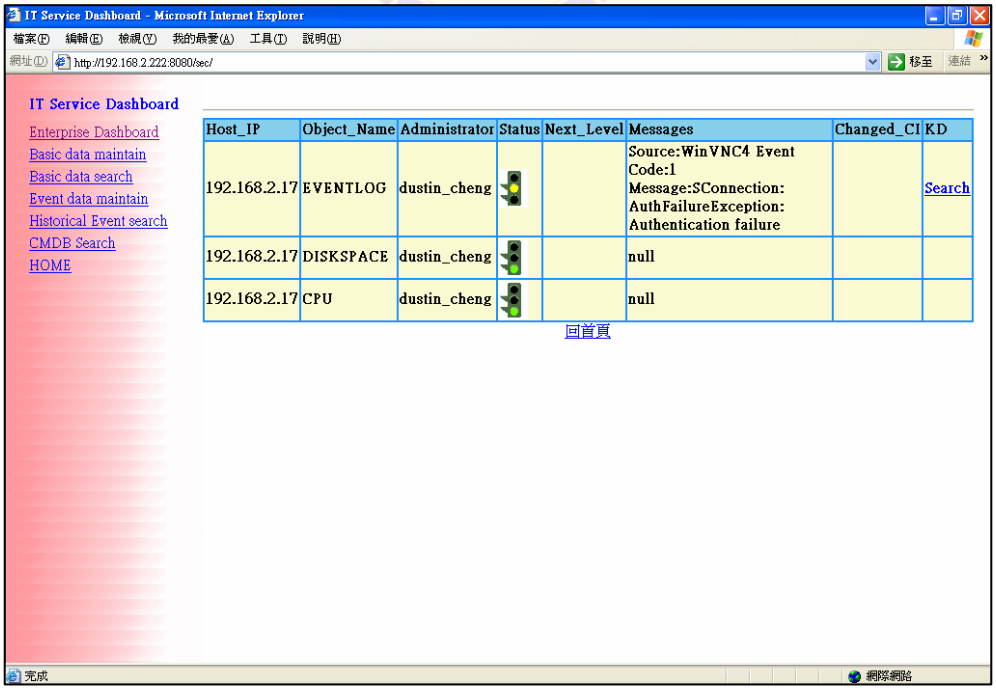


圖 31　事件資料

資料來源：本研究

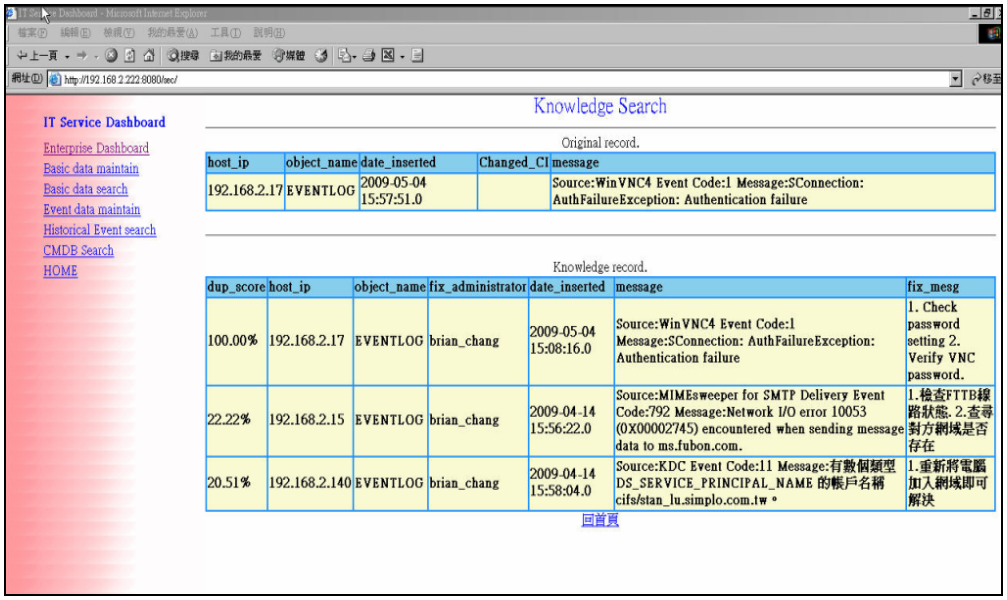


圖 32　事件檢索資料

資料來源：本研究

點選事件的“KD”欄位後，系統會呈現經事件檢索處理後的資料內容，如圖 32 所示，系統呈現此新事件訊息資料，並將相似度最高的前三筆歷史事件資料依序列出，服務台人員便可根據歷史事件相似程度，與“fix_message”欄位資料所提供的解決步驟資訊，進行事件解決處理。

Basic data maintain 功能(圖 33)，是用來建立 IT 服務物件資訊與相關物件階層式關聯，建立方式是需先建立兩物件資訊，再建立兩個物件之間的關聯，物件資訊主要是物件所處的 IP 與物件名稱，以 Oracle ERP 的 IT 服務階層第一層與第二層的關聯建立為例(圖 34)，首先利用圖 33 中的“Insert managed object”功能建立兩個物件資訊，第一個物件名稱為 ORACLEERP-SYSTEM，IP 為 192.168.2.200，第二個物件名稱為 PING，IP 為 192.168.2.200，接著再用圖 33 中的“Insert object Dependency”功能，將“TOP Object Name”與“TOP Object IP”輸入第一層的物件資訊 ORACLEERP-SYSTEM 與 192.168.2.200，而“Object Name”與“Object IP”分別輸入第二層的

物件資訊 PING 與 192.168.2.200，新增後便可建立此兩物件的階層關聯關係於系統中。Oracle ERP 此 IT 服務階層式關聯與相關物件的範例如圖 34 所示，第一層為 Oracle ERP System，第二層為 Ping Oracle ERP AP 伺服器的物件，第三層為 ERP AP 的 Process 與日誌檔案，第四層為作業系統的日誌檔、CPU、磁碟空間與 ERP DB 的 Ping 物件，第五層為 Oracle 資料庫的 Process 與日誌檔，第六層為資料庫伺服器的作業系統日誌檔、CPU 與磁碟空間物件。透過建立 IT 服務物件與階層關係的資料，若有新事件產生時，便可依循此階層關係，將有異常事件之物件所影響的範圍，依相關聯架構階層設定，利用如物料清單方式的結構呈現方式，往上層呈現事件異常狀態資訊。而 Basic data search 功能(圖 35)是供查詢已經建立的物件與階層關聯資料，便於確認資料正確性。


圖 33　IT 服務資料維護

資料來源：本研究

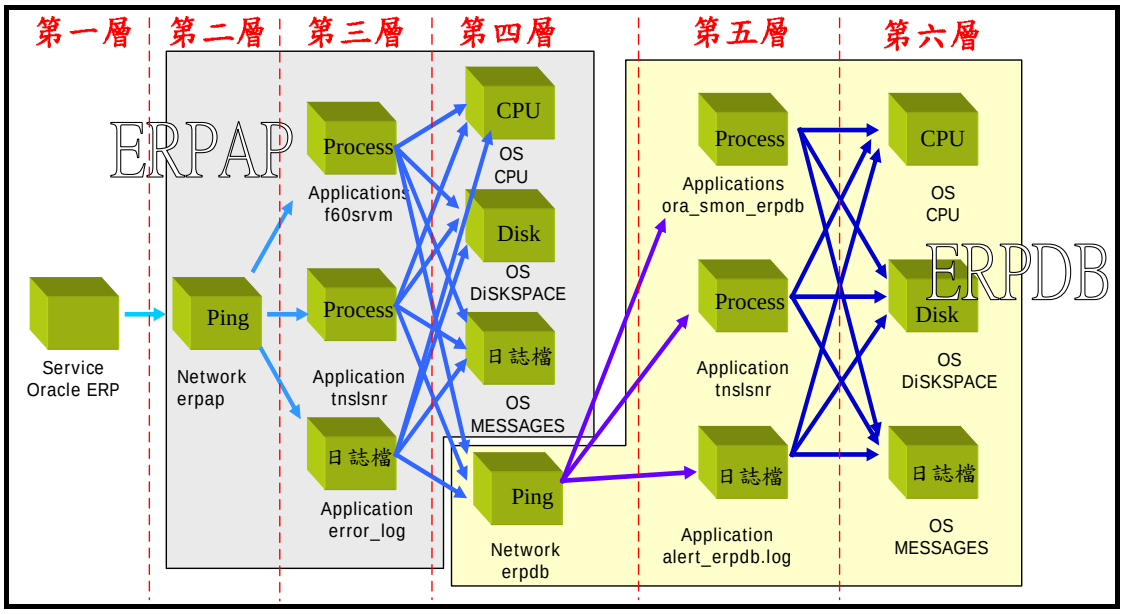


圖 34　Two Node Oracle ERP 架構範例

資料來源：本研究

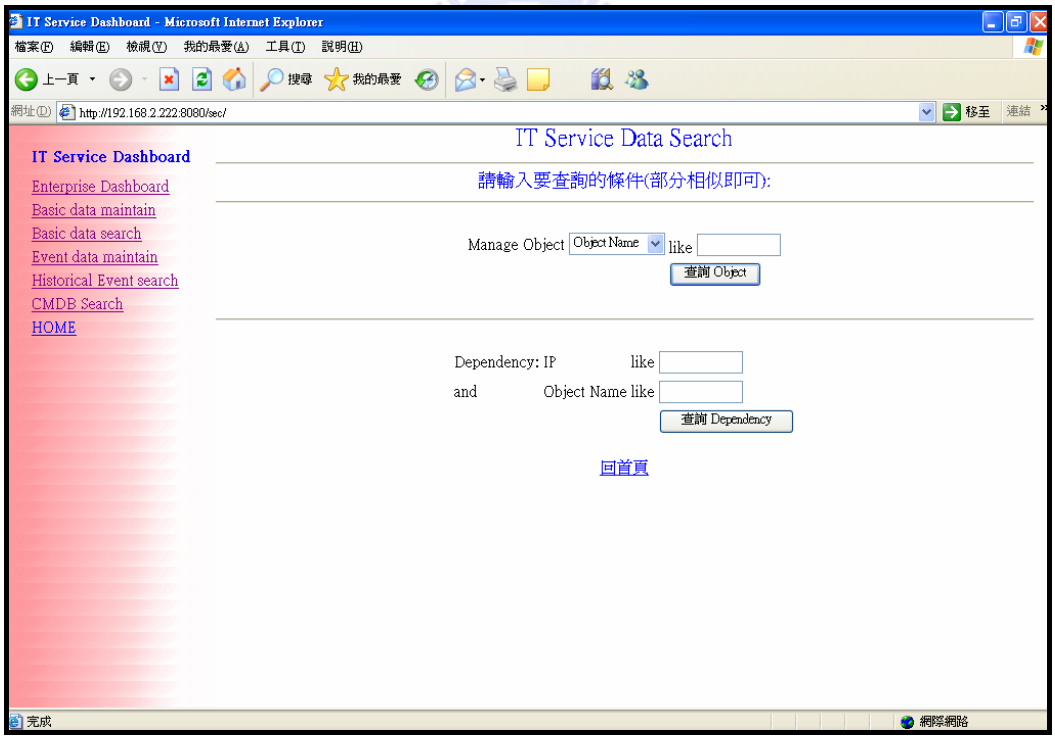


圖 35　IT 服務資料查詢

資料來源：本研究

Event data maintain 功能，則是用來處理已結案的事件資訊，透過查詢功能取得事件資訊後(圖 36、圖 37)，再點選“fix now”欄位

即可進行事件資料的處理資訊維護。如圖 38 所示，維護事件資料時，在“fix_knowledge”欄位可選擇是否為新的事件知識資料，若非新的事件資料，則選擇“Duplicate”選項，便只單純修正事件狀態；若此為新事件，未有相類似的歷史事件與適用的解決方式，則選擇“Knowledge”選項，並在“fix_description”欄位中輸入此事件的解決步驟與相關訊息，便可將事件知識資料存入歷史事件知識資料庫中，作為日後相類似事件的處理解決方式。

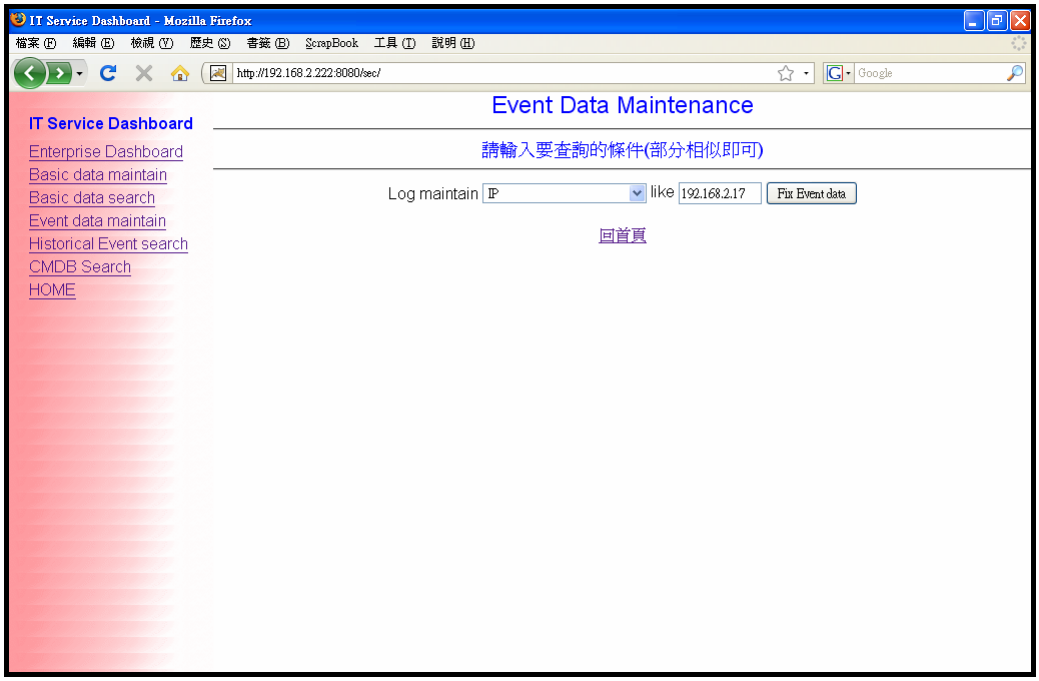


圖 36　事件資料維護查詢

資料來源：本研究

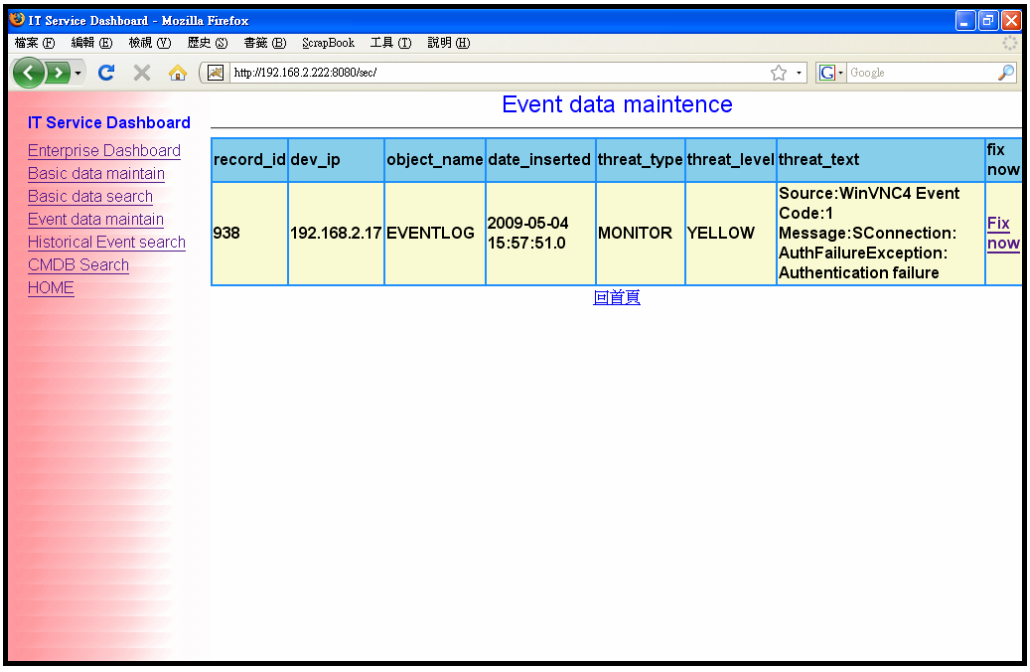


圖 37　事件資料內容

資料來源：本研究

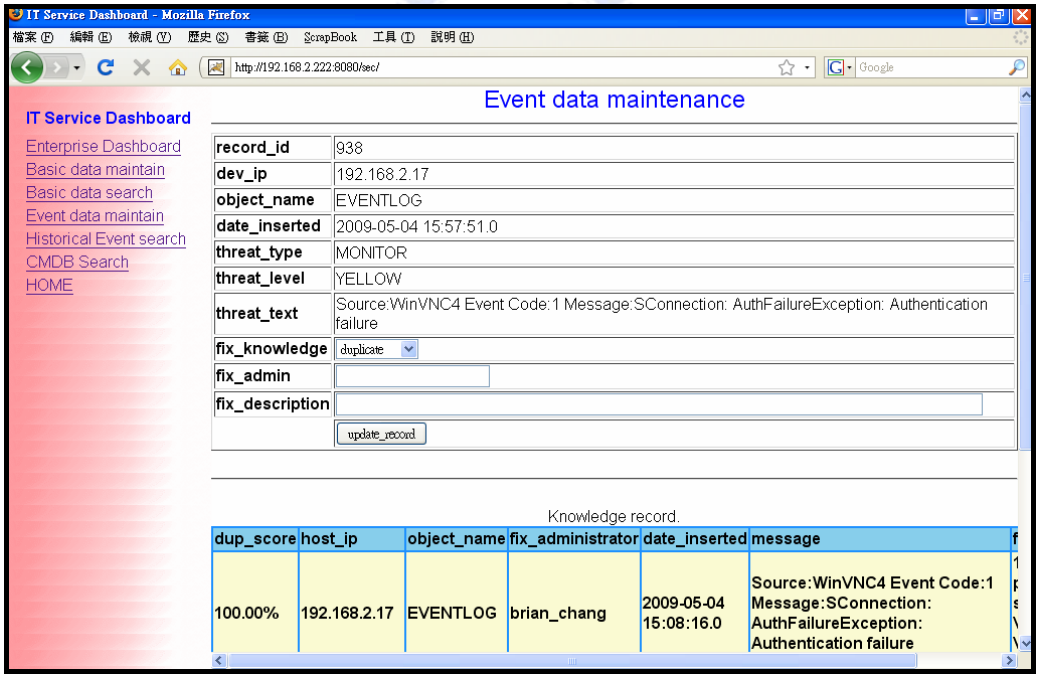


圖 38　事件資料維護

資料來源：本研究

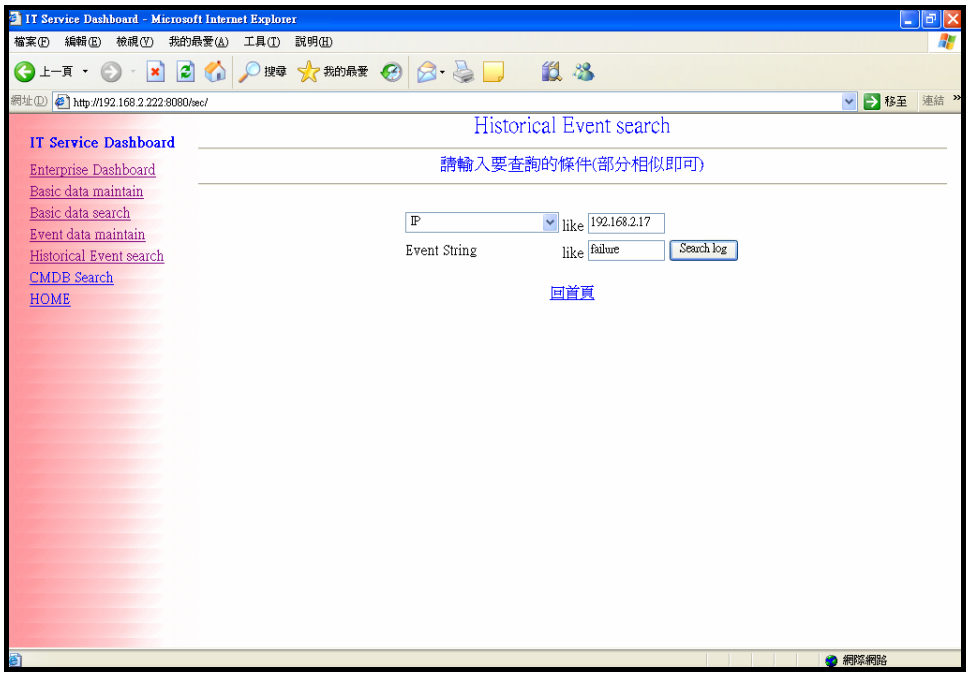


圖 39　歷史事件資料查詢

資料來源：本研究

Historical event search 功能則提供對於歷史事件資料的查詢方式，可利用 IP、物件加上事件的字串資料等等，查找歷史事件資料庫資料，與其事件解決的處理步驟資訊(圖 39 與圖 40)。

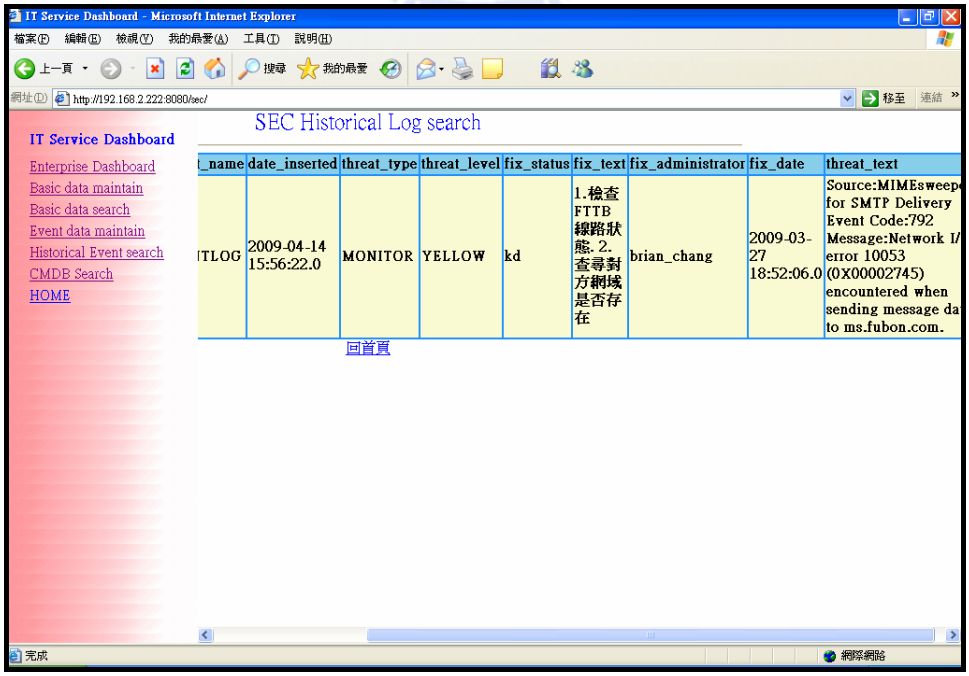


圖 40　歷史事件資料內容

資料來源：本研究

最後則為 CMDB Search，此功能可用來查詢物件組態資料(圖41)，包括目前組態、前一版本組態資料與有異動的組態資料資訊，提供管理人員查詢組態資訊，並可供組態稽核使用，避免未經核准的異動組態狀況，提高系統穩定程度。圖 42 即為組態異動資料的查詢結果，可發現此台伺服器有異動 Page file space，還有將 HP Port Resolver 服務的 startup mode 從“Manual”異動為 “Auto”狀態。

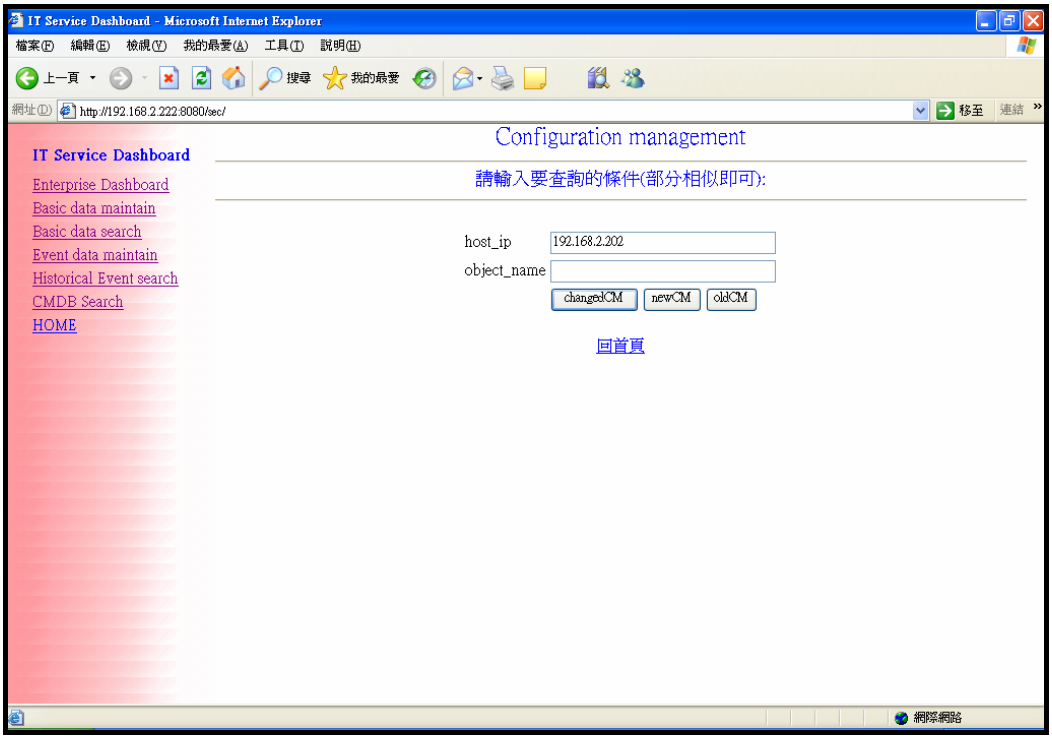


圖 41　組態資料查詢

資料來源：本研究

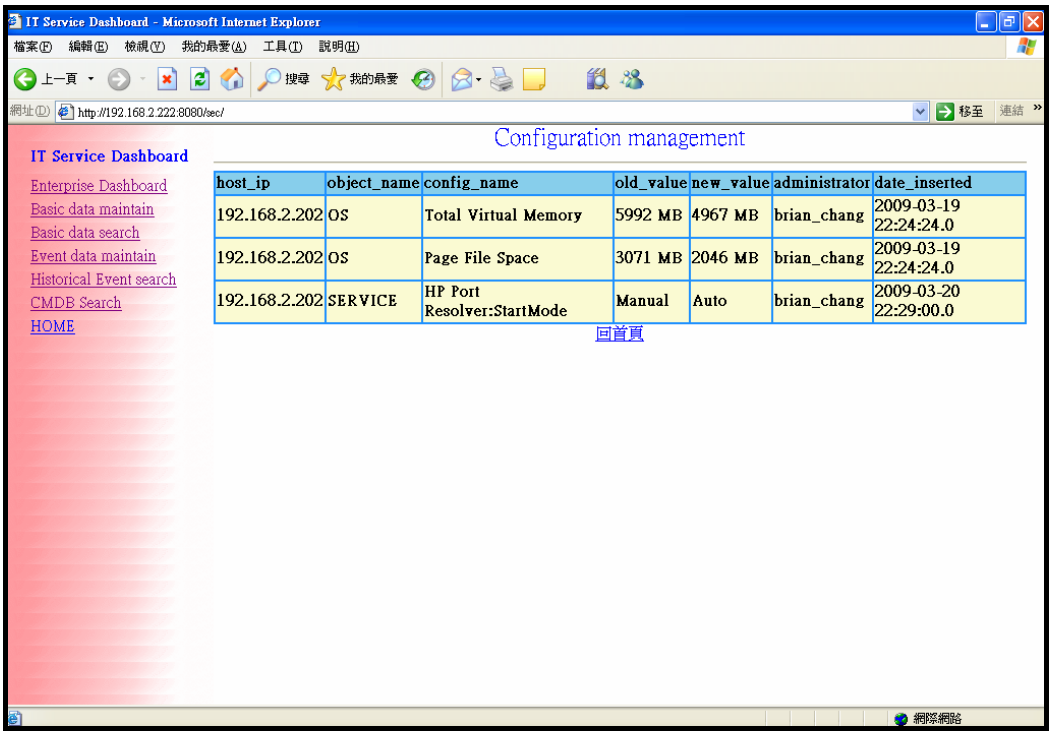


圖 42　組態異動資料內容

資料來源：本研究

## 4.2　系統評估

本節說明基於本研究動機與目的，分析本研究實作系統成果與助益，利用與企業中 IT 管理人員進行實際訪談，瞭解一般科技產業 IT 管理人員對於 ITIL 事件管理的認知，與其現行的事件管理處置方式，再實際展示本研究實作成果，讓訪談者瞭解系統功能，收集系統助益的回饋，並對訪談結果進行分析整理。

### 4.2.1 使用事件管理自動化系統與未使用前之差異

現行企業環境中的事件管理方式，大多是經由人力來運作與進行，在這種運作模式中，人員的技術能力、環境熟悉度與架構複雜程度，都是影響 MTTR 時間長短的可能變數，如管理人員的技術能力

不足，則可能會找不到導致異常原因的物件，或是不清楚該檢查的內容為何；環境熟悉度不足，則可能會誤判事件嚴重程度；架構複雜則會讓管理人員花費大量時間來確認架構中相關物件的狀態。若使用本研究所建立的事件管理自動化系統，則可將上述變數影響減輕至最低。未使用與使用事件管理自動化系統的比較如表 12 所示。

表 12 未使用與使用本研究所建置系統的操作方式比較表

| 項目 | 未使用 | 使用自動化事件管理系統 |
|---|---|---|
| 事件偵測 | 使用者發現 IT 服務無法使用時通知管理人員，IT 服務異常狀態可能已持續相當長的時間 | 自動定期偵測所有物件狀況，可發現潛藏威脅供修正異常，避免導致更嚴重的異常事件發生 |
| 事件記錄 | 管理人員手動記錄事件相關資訊，易發生未記錄確實或遺漏狀況 | 自動紀錄事件詳細資訊，如時間、異常物件相關資訊與管理人員資訊 |
| 事件分類 | 管理人員自行判斷事件緊急程度，若是新進人員易發生誤判狀況 | 自動依所定義規則進行嚴重程度分類，並寄送 EMail 通知管理人員 |
| 事件診斷 | 管理人員需手動檢查所有可能相關物件狀況、日誌檔與相關文件資料，進行異常判斷並取得解決方式，但檢查時可能會遺漏重要異常物件訊息，或未找到正確與適當的文件資料 | 自動由事件檢索功能進行診斷，取得最相似的歷史事件問題解決步驟，不需輸入任何關鍵字來查找文件，也不會發生遺漏重要參考文件的狀況 |
| 組態管理 | 異常原因查找時進行相關組態查詢，但無法知道何時曾異動，也無法知道異動前的設定值為何 | 自動取得所有物件組態與組態異動資訊，也包含異動時間資訊，管理人員可快速且確實掌握資訊充足之組態狀況，協助異常解決 |
| IT 服務狀態 Web 查詢 | 無法得知全部 IT 服務運行狀態與所遭受的威脅 | 透過事件管理平台 Web 介面可清楚瞭解所有 IT 服務之即時運行狀態與威脅 |

資料來源：本研究

### 4.2.2 訪談內容與對象

因本研究主題是與 IT 管理相關領域，故所訪談的目標對象，是目前任職於科技業的 IT 管理人員。本研究訪談對象背景基本資料如表 13 所示。

表 13　訪談人員背景資料

| 訪談對象 | 產業別 | 資本額 | 員工人數 | 任職部門 |
|---|---|---|---|---|
| A | XX 科技-電池模組 | 22 億 | 4000 | 資訊部 |
| B | XX 科技-網通 | 52 億 | 3000 | 網路管理課 |
| C | XX 電子-記憶體 | 380 億 | 2000 | 資料庫管理課 |
| D | XX 科技-IC 設計 | 51 億 | 600 | 系統管理課 |
| E | XX 科技-IC 設計 | 6 億 | 100 | 資訊部 |

資料來源：本研究

訪談內容以分析與評估本研究的動機與欲達成的目的進行設計，訪談項目設計如表 14 所示，其中第一部分內容(項目 1 至項目 4)是訪談人員所任職企業的現行事件處理運作，與組態管理方式；第二部分(項目 5)是對於 ITIL 事件管理的瞭解程度；第三部分(項目 6 與項目 7)是介紹本研究實作系統內容，並取得訪談人員對於本研究實作系統能對任職企業的助益意見。第一部分與第二部分內容是先利用 Email 寄出與取得回覆內容，避免誘導或誤導訪談人員的狀況，影響訪談回覆內容。而第三部分則後續由實地訪談方式，展示本研究實作的事件管理平台運作流程，解說研究動機與內容，再收集訪談人員回應資料。

表 14　訪談內容

| 項目 | 訪談內容 | 回覆內容 |
|---|---|---|
| 1 | 貴公司現行事件處理流程？(如何得知事件發生？如何確認異常原因？如何取得解決方式或文件？如何解決異常事件？) | |
| 2 | 貴公司現行事件處理有何不足之處？您個人認為該如何改善？ | |
| 3 | 貴公司是否有使用事件關聯軟體協助處理事件(如 HP Openview)？是否安裝設定簡易？有無平台限制？ | |
| 4 | 您現行是如何得知系統的設定資訊(如資料庫參數，作業系統參數等)？是否曾遇過因設定異動而導致系統無法使用的狀況？ | |
| 5 | 是否瞭解 ITIL 事件管理流程？ | |
| 6 | 經展示本研究系統功能後，是否瞭解本系統功能與目的？ | |
| 7 | 本系統對貴公司異常事件處理是否有助益？有可幫助哪些方面？ | |

資料來源：本研究

### 4.2.3 訪談結果與分析

訪談回覆資料內容如附錄一至附錄五所示，在訪談資料第一部分，訪談人員所任職企業，現行事件處理與組態管理方式的回覆內容中可知，訪談人員 B、C 與 D 其企業內會對系統異常狀況，利用如 HP SIM、IBM Direct、IP Check 等工具來協助事件偵測，但因無適用與滿足全面性需求的軟體，故會再自行開發監測程式；監測程式由各部門各自進行，程式散佈於各伺服器，系統管理人員異動時，易發生程式交接項目遺漏，也常因管理人員開發使用的程式語言不同，容易發生維護不易的狀況。訪談人員 B 的內部人員，曾嘗試安裝使用 HP Openview 軟體，但因軟體熟悉門檻高，設定不易，故最後未上線使用。對於現行事件管理方式，訪談人員都認為有改善的需要，異常事件根源的確認是需改善的重點。在組態管理方面，都曾遭遇組態管理問題而導致 IT 服務異常事件發生，也期盼有可以知道 IT 服務組態資訊與組態異動狀況的稽核資料狀況。

第二部分對於 ITIL 事件管理的瞭解部分，從訪談紀錄第一部分回覆內容中，可知各企業內事件處理與 ITIL 事件管理流程有相關部分符合，但五位訪談的 IT 管理人員，目前皆不甚清楚 ITIL 事件管理定義的流程，可見 ITIL 事件管理流程最佳實務，在台灣科技業內的推展尚有努力空間。

第三部分對於本系統實作部分的展示與研究動機，五位訪談人員皆瞭解，且都有正面的回應，對於其所處企業現行事件管理運作皆有所助益。實作功能從事件管理平台、事件自動偵測、組態管理、程式集中管控、利用 Perl 跨平台使用、事件關聯的軟體與規則設定，與歷史事件解決方式資料自動取得、知識保存等等，可滿足訪談人員現行

需求並改善事件管理的不足。

從回應資料，可得知本系統實作對於現行企業的事件管理是有幫助的，且能提供組態資訊的內容，方便系統管理運作需求。訪談過程中，也得到此實作系統後續發展的建議，這些建議是在本系統未來研究中可加入的擴充與沿伸。訪談者所提供的系統建議如下：

1. 利用資料探勘機制，增加分析異常事件趨勢。
2. 增加設計異常事件自動回復至正常運作狀態機制。
3. 希望擴展本系統，加入異動管理的部分的建議(此為 ITIL 異動管理模組內容)。

# 五、結論與未來研究方向

本章分為兩節，第一節總結本研究成果，並列舉研究貢獻；第二節則說明本研究實作系統未來的可擴充性與研究方向。

## 5.1 結論

本研究所建構的自動化事件管理系統，符合 ITIL 事件管理流程模組活動所需，主要功能如下所述：

1. 利用自動偵測事件取得事件資料，符合事件管理的偵測步驟，且將偵測程式集中統一放置，避免凌亂散佈狀況。
2. 使用 SEC 規則關聯軟體進行事件相關聯與分類，符合事件管理所需的分類與記錄步驟。
3. 用資訊檢索方式進行事件檢索，取得事件解決方案，符合事件管理中的診斷步驟。
4. 建立組態資料自動收集與稽核功能，提供 IT 服務所需的輔助資訊。
5. 建立 Web 事件管理平台，供服務台進行事件管理的操作，此事件管理自動化平台，可簡化事件管理的操作過程。

針對本研究第一章所提的研究問題，以下逐一回答：

1. 在 IT 管理運作上，如何得知企業所有 IT 服務運行狀態與遭受的威脅？

   在本研究事件管理平台上建置所有企業 IT 服務與相關物件的關係，經由本研究的事件偵測與事件關聯機制，即可確認 IT 服務運行狀態，若有發現可能會導致異常的訊息，也會呈

現於管理介面上，可即時進行修正，避免事件影響層級擴大與擴散。

2. 在服務異常事件上，如何快速找到問題真因？

   經由本研究的事件偵測、事件關聯與組態管理的機制，可找到導致異常的問題根源物件，組態管理並會呈現相關的組態異動資訊，作為事件異常的輔助參考資訊，可達成快速找到問題真因所需。

3. 如何確認 IT 服務組態狀況？

   本研究的組態管理機制，會進行組態資訊的收集，協助 IT 管理者取得管理上所必需、基本且重要的組態資訊。組態稽核方面可確認組態的異動狀況與異動日期，並留存異動前後的組態值，可供系統復原時有正確設定值可參考使用。

4. 有何低成本、耗用資源少、跨平台、設定簡易且功能完整的事件關聯解決方案？

   本研究使用開放原始碼的 SEC 事件關聯軟體進行事件關聯的處理，此輕量級的關聯軟體主程式大小只有 250KB，且是使用 Perl 語言開發而成，能被安裝於各種作業系統上，其規則類型與所觸發行動經簡易設定便能達成需求，也可交錯組合成進階的事件判定所需，功能足供企業運作使用。

5. 如何將 ITIL 事件管理流程自動化，來縮短 MTTR 時間？

   利用本研究架構中的事件偵測、事件關聯與事件檢索機制，將 ITIL 事件管理中偵測、記錄、分類與診斷等活動，以不需人為運作方式進行自動化處理，並直接提供服務台人員關於新事件的解決方案，縮短系統復原 MTTR 的時間。

此外，本實作系統所達成之目的與貢獻描述如下：

1. 建立自動化事件管理平台，使服務台或 IT 管理人員利用此操作介面，快速取得 IT 服務與相關物件的運作狀態資訊，可提高 IT 人員專業素質、服務能力與工作效率，節省人力成本，縮短 IT 服務 MTTR 時間，增加 ITIL 事件管理模組對於企業實務應用的貢獻。
2. 運用輕量級與設定方式簡易的 Rule-based 事件關聯軟體 SEC 於 ITIL 領域，建構事件管理中的分類處理步驟，達成事件關聯所需功能與目標，避免其他高門檻軟體的高昂費用與設定複雜情況。
3. 運用資訊檢索技術進行事件檢索，處理事件管理中所需的診斷步驟，簡化服務台人員需進行事件診斷與取得解決方案的過程，自動進行新事件與歷史事件處理資料的相似度比對，並取得最相似的問題解決處理步驟，呈現給服務台作異常 IT 服務事件回復解決使用，省去讓服務台需再利用關鍵字來搜尋相關標準操作文件的步驟，也避免因搜尋方式不正確而拖延異常處理時間，加快事件管理流程的進行。
4. 提供 IT 服務事件管理主動運作模式與工具，將過去零散與較無系統的事件管理方式統整並自動化，運用跨平台的事件偵測設計與實作，程式集中放置，縮短事件處理發現過程與提高事件處理時效，更快速解決 IT 服務異常，提高 IT 服務的可用性與可靠性，提供企業高品質商務運作的服務。
5. 建立組態資料庫收集與稽核方式，協助 IT 管理瞭解現行系統組態與異動資訊，提供組態稽核所需資料，用以避免任意組態異動行為，與保留異動前後的組態資訊，以利系統異常回復與判斷事件異常可能因素使用，降低導致 IT 服務

Downtime 的人為因素發生可能。

本研究並利用訪談方式進行評估,訪談對象為目前任職於企業 IT 管理部門人員，經展示本研究實作成果，來取得本研究成果對於訪談人員事件管理的助益。從訪談人員回覆資料可知，本研究實作事件管理平台系統，可改善訪談人員所處企業的事件管理流程，並提供相當大的助益。

## 5.2 未來研究方向

在未來研究方向，可從下列幾點方向進行：

1. 將 ITIL 其他模組，如：問題管理、異動管理與上線管理等流程模組整合至本系統，建構完整 ITIL 系統，協助 IT 管理單位提供高品質的 IT 服務。
2. 整合簽核系統進行組態異動稽核自動化管理，確保 IT 服務穩定性。
3. 對已發生的事件資料，利用資料探勘方法，找出事件資料的趨勢狀態，取得對 IT 服務管理有助益的事件資訊，提供 IT 管理決策分析使用。
4. 探討國內企業文化與環境對於 ITIL 各模組之適用性分析，協助企業經由合用的導入方式來運用 ITIL，改善 IT 服務環境。

# 參考文獻


1. 行政院主計處 96 年電腦應用概況報告，http://www.dgbas.gov.tw/public/Attachment/8101516265571.pdf, 2008
2. Scott, D., “Making Smart Investments to Reduce Unplanned Downtime”, GartnerGroup, March 1999.
3. ITIL, http://www.ogc.gov.uk/guidance_itil.asp, OGC
4. IBM 規則建置器手冊，http://publib.boulder.ibm.com/tividd/td/tec/GC32-0669-01/zh_TW/HTML/RBGmst02.htm#ToC_40, IBM
5. itSMF 論壇, http://www.itsmfi.org/, itSMF® International
6. ITIL Foundation for IT Service Management, HP
7. 江國順，“ITIL/ITSM 市場趨勢及分析”，http://dbmaker.syscom.com.tw/mag/114/coverstory_04.htm
8. Sharifi, M. et al, “Lessons learned in ITIL implementation failure”, International Symposium on Information Technology, Volume 1, pp. 1-4, Augest 2008
9. 施淑鎂，「資訊服務業支援中心之營運流程架構」，國立高雄第一科技大學，碩士論文，民國 96 年
10. Rossetti, P., ITIL Service Management Foundation, Sun, 2003
11. Brown, A.B. and Keller, A., “A Best Practice Approach for Automating IT Management Processes”, Network Operations and Management Symposium, Volume 3, Issue 7, pp. 34-44, April 2006
12. Brenner, M., “Classifying ITIL Processes; A Taxonomy under Tool Support Aspects”, Business-Driven IT Management, pp. 19-28, April 2006
13. Hanemann, A., ”Refining ITIL/eTOM Processes for Automation in

Service Fault Management", Business-Driven IT Management, pp. 106-107, May 2007

14. Bartolini, C., Salle, M., and Trastour, D., "IT service management driven by business objectives: an application to incident management", Network Operations and Management Symposium, Volume 3, Issue 7, pp. 45-55, April 2006.
15. Gupta, R., Prasad, K.H. and Mohania, M., "Automating ITSM Incident Management Process", Autonomic Computing, pp. 141-150, June 2008
16. 周柏村，「IT 服務管理：運用情境認知之知識支援於事件管理」，國立交通大學，碩士論文，民國 94 年
17. 李桄瑋，「IT 服務管理：運用主題地圖與資料探勘支援事件管理」，國立交通大學，碩士論文，民國 94 年
18. Sharifi, M., Ayat, M., Sahibudin, S., "Implementing ITIL-Based CMDB in the Organizations to Minimize or Remove Service Quality Gaps", Second Asia International Conference on Modelling & Simulation, pp. 734-737, May 2008
19. Jakobson, G. and Weissman, M., "Real-time telecommunication network management: Extending event correlation with temporal constraints", International Symposium on Integrated Network Management, pp. 290-301, 1995
20. Park, Y., "Event correlation", IEEE Potentials, Volume 20, Issue 2, pp. 34-35, May 2001
21. Jakobson, G., "The technology and practice of integrated multiagent event correlation systems", Integration of Knowledge Intensive Multi-Agent Systems, pp. 568-573, 2003
22. Lin, A., "A hybrid approach to fault diagnosis in network and system management", HP Technical Report , 1998

23. Jakobson, G. and Weissman ,M., “Alarm correlation”, IEEE Network, Volume 7, Issue 6, pp. 52-59, November 1993.
24. Gruschke, B., “Integrated Event Management: Event Correlation using Dependency Graphs”, Distributed Systems: Operations and Management, pp. 130-141, 1998
25. Kliger, S., et al., “A coding approach to event correlation”, International Symposium on Integrated Network Management, pp. 266–277, California, May 1995
26. Lewis, L., “A case-based reasoning approach for the resolution of faults in communication networks”, International Symposium on Integrated Network Management, pp. 671–682, 1993
27. Gardner, R.D. and Harle, D.A., “Methods and systems for alarm correlation”, Global Telecommunications Conference, Volume 1, pp.136-140, 1996
28. Rich, E. and Knight, K., Artificial Intelligence, 2nd edition, McGraw-Hill, 1991
29. Hanemann, A. and Sailer, M., “A framework for service quality assurance using event correlation techniques”, Telecommunications, pp.428-433, July 2005
30. Vaarandi, R., “Simple Event Correlator for Real-Time Security Log Monitoring”, Hakin9 Magazine , pp. 28-39 , January 2006
31. Yemini, S.A. et al, “High speed and robust event correlation”, Communications Magazine, IEEE, Volume 34, Issue 5, pp.82-90, May 1996
32. Hamming distance, http://en.wikipedia.org/wiki/Hamming_distance, wikipedia
33. Aamodt, A. and Plaza, E., “Case-Based Reasoning：Foundational issues , Methodological Variations, and System Approaches”, AI Communication. ISO Press, Volume 7, pp. 39-59 , 1994

34. Hanemann, A., “A hybrid rule-based/case-based reasoning approach for service fault diagnosis”, Advanced Information Networking and Applications, April 2006
35. Vaarandi, R., “SEC - a lightweight event correlation tool”, IP Operations and Management, pp. 111-115, 2002
36. Brown,J., http://sixshooter.v6.thrupoint.net/SEC-examples/article.html , November 2003
37. Rouillard, J. P., “Real-time Log File Analysis Using the Simple Event Correlator (SEC)”, System Administration Conference Proceedings of the 18th USENIX conference on System administration, pp.133-150, 2004
38. Baeza-Yates, R. and Ribeiro-Neto, B., Modern Information Retrieval, Addison-Wesley-Longman, 1999
39. Manning, C.D., Raghavan, P. and Schütze, H., Introduction to Information Retrieval, Cambridge UP, 2008
40. Singhal, A., “Modern Information Retrieval: A Brief Overview”, Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, Volume 24, Nomber 4, pp. 35–43, 2001
41. Van Rijsbergen, C. J. , Information Retrieval, Butterworths, 1979
42. Faloutsos, C. and Oard, D. W., “A survey of information retrieval and filtering methods”, University of Maryland at College Park, College Park, MD, 1995
43. Cavnar, W.B. and Trenkle, J. M., “N-grambased text categorization”, In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pp. 161-175, 1994
44. Salton, G., Allan j. and Buckley C., “Automatic structuring and retrieval of large text files”, Communications of the ACM, Volume 37, Issue 2, pp.97-108, February 1994

45. Salton, G., Wong, A., and Yang, C.S., “A Vector Space Model for Automantic Indexing”, Communications of the ACM, Volume 18, No.11, pp. 613-620, 1975
46. Ibrahimov, O., Sethi, I. and Dimitrova N., “A Novel Similarity Based Clustering Algorithm for Grouping Broadcast News”, Data Mining and Knowledge Discovery: Theory, Tools, and Technology IV, Volume 4730, pp.394-404, 2002
47. Erk, K., “A Simple, Similarity-based Model for Selectional Preferences”, In Proceedings of ACL, pp.216-223, 2007
48. DMTF, http://www.dmtf.org/home, Distributed Management Task Force, Inc.
49. Windows Management Instrumentation, http://msdn.microsoft.com/en-us/library/aa394582(VS.85).aspx , Microsoft Corporation , 2009

# 附錄一、訪談回覆資料內容 A

| 任職產業別 | XX 科技-電池模組 | 資本額 | 22 億 |
|---|---|---|---|
| 任職部門 | 資訊部 | 員工人數 | 4000 |

| 項目 | 訪談內容 | 回覆內容 |
|---|---|---|
| 1 | 貴公司現行事件處理流程？(如何得知事件發生？如何確認異常原因？如何取得解決方式或文件？如何解決異常事件？) | 使用者遇到異常時會通知負責管理的人員，或是由自行開發的程式進行系統檢查，並寄送 Email 通知；異常原因確認方式是先確認網路，接下來為主機，最後是資料庫狀況，並檢查 log 內容；由負責工程師依經驗進行問題解決。 |
| 2 | 貴公司現行事件處理有何不足之處？您個人認為該如何改善？ | 目前系統異常時需先確認相關的部分是否正常，事件發生時有需多 log 內容需檢查，應有自動檢查各 log 的程式，並寄送異常通知，且需要能夠很快的知道問題源頭，避免花費時間在檢查不需檢查的地方，未來將進行這一部分的檢查設計。 |
| 3 | 貴公司是否有使用事件關聯軟體協助處理事件(如 HP Openview)？是否安裝設定簡易？有無平台限制？ | 無。 |

| | | |
|---|---|---|
| 4 | 您現行是如何得知系統的設定資訊(如資料庫參數，作業系統參數等)？是否曾遇過因設定異動而導致系統無法使用的狀況？ | 登入系統進行檢查設定；遇過設定異動而系統當機頻繁狀況，進行設定異動必須先於測試環境驗證，且該有設定資料內容的檢查設計。 |
| 5 | 是否瞭解 ITIL 事件管理流程？ | 不瞭解。 |
| 6 | 經展示本研究系統功能後，是否瞭解本系統功能與目的？ | 是。 |
| 7 | 本系統對貴公司異常事件處理是否有助益？有可幫助哪些方面？ | 有很大助益；方便瞭解所有系統運作異常狀況，瞭解問題嚴重程度；可檢查各種平台與系統的 log，可不需再手動檢查不需檢查的部分，節省時間；可取得系統的設定資料內容，可進行設定異動的檢查； |

# 附錄二、訪談回覆資料內容 B

| 任職產業別 | XX 科技-網通 | 資本額 | 52 億 |
|---|---|---|---|
| 任職部門 | 網路管理課 | 員工人數 | 3000 |

| 項目 | 訪談內容 | 回覆內容 |
|---|---|---|
| 1 | 貴公司現行事件處理流程？(如何得知事件發生？如何確認異常原因？如何取得解決方式或文件？如何解決異常事件？) | 套裝軟體(IPCheck)及利用 Windows Script 自行撰寫 monitor， 解決方式則以工程師經驗處理。 |
| 2 | 貴公司現行事件處理有何不足之處？您個人認為該如何改善？ | 缺乏統一完整之平台對整個資訊環境做全盤的偵測，因為有時一個事件發生可能會牽連到許多系統或是有時從事件發生的表面無法查出根源其實是另一個系統設定異常所造成，因此需要一個能從網路架構到系統面再到應用程式層面的事件偵測機制。 |
| 3 | 貴公司是否有使用事件關聯軟體協助處理事件(如 HP Openview)？是否安裝設定簡易？有無平台限制？ | 目前尚未發現有適合之事件關聯軟體。部門內曾嘗試安裝 HP Openview 軟體測試，但因熟悉門檻高且安裝與設定不易，最後未正式上線使用。 |
| 4 | 您現行是如何得知系統的設定資訊(如資料庫參數，作業系統參數等)？是否曾遇過因設定異動而導 | 實際登入到系統檢查設定資料；是，曾遇過設定異動而發生系統無法使用狀況，異動應 |

| | | |
|---|---|---|
| | 致系統無法使用的狀況？ | 通知相關系統負責人或公告，來避免影響到其他部門負責的系統。 |
| 5 | 是否瞭解 ITIL 事件管理流程？ | 否。 |
| 6 | 經展示本研究系統功能後，是否瞭解本系統功能與目的？ | 是。 |
| 7 | 本系統對貴公司異常事件處理是否有助益？有可幫助哪些方面？ | 是，可清楚的檢視所有系統的資訊，方便系統異動管理及資產盤點管理；減輕系統維護人力成本；可清楚知道異常是哪個問題造成的；若能有事件異常趨勢分析的資訊則更好。 |

# 附錄三、訪談回覆資料內容 C

| 任職產業別 | XX 電子-記憶體 | 資本額 | 380 億 |
|---|---|---|---|
| 任職部門 | 資料庫管理課 | 員工人數 | 2000 |

| 項目 | 訪談內容 | 回覆內容 |
|---|---|---|
| 1 | 貴公司現行事件處理流程？(如何得知事件發生？如何確認異常原因？如何取得解決方式或文件？如何解決異常事件？) | 24 小時值班人員(Helpdesk)透過電話(User 通知)或電子郵件(HP SIM，各部門系統檢查程式 Alert)來得知異常事件，當有異常時便通知各部門值班人員進行檢查，並發出異常事件通報，二線人員需於 30 分鐘內回到公司處理問題，由二線人員進行系統問題解決。 |
| 2 | 貴公司現行事件處理有何不足之處？您個人認為該如何改善？ | 部門內系統異常解決 SOP 文件過多，有些文件未即時更新，Helpdesk 無法直接依 SOP 文件處理問題，需待二線回廠處理，且這段期間系統持續無法使用，若能有確實的異常解決資訊給 Helpdesk，由第一線來處理問題，則可改善問題解決時效。 |
| 3 | 貴公司是否有使用事件關聯軟體協助處理事件(如 HP Openview)？是否安裝設定簡易？有無平台限 | 無使用事件關聯軟體，由各部門自行開發監控程式檢查系統，再通知 Helpdesk 與負責的 |

| | | |
|---|---|---|
| | 制？ | 管理人。 |
| 4 | 您現行是如何得知系統的設定資訊(如資料庫參數，作業系統參數等)？是否曾遇過因設定異動而導致系統無法使用的狀況？ | 曾遇過因資料庫參數設定錯誤而造成系統重起時無法啟動資料庫狀況，後來各部門有開發程式每天上傳一次系統的設定資訊，應該在設定異動前能先通知所有相關人員進行確認的機制。 |
| 5 | 是否瞭解 ITIL 事件管理流程？ | 有聽過 ITIL，不瞭解事件管理流程內容。 |
| 6 | 經展示本研究系統功能後，是否瞭解本系統功能與目的？ | 已瞭解。 |
| 7 | 本系統對貴公司異常事件處理是否有助益？有可幫助哪些方面？ | 此設計會有幫助，Helpdesk 不需自行查找大量的 SOP 文件，而有最相似的解決步驟可解決問題；異常一發生可找到異常事件起源，不會 Call 錯二線負責人；Monitor 程式集中且跨平台方式，可解決程式於每一台伺服器上放置與維護不易的狀況；Dashboard 階層方式查詢可容易的檢查異常事件，可讓各部門人員釐清所屬責任問題；此系統若能有自動修正異常的機制更佳。 |

# 附錄四、訪談回覆資料內容 D

| 任職產業別 | XX 科技-IC 設計 | 資本額 | 51 億 |
|---|---|---|---|
| 任職部門 | 系統管理課 | 員工人數 | 600 |

| 項目 | 訪談內容 | 回覆內容 |
|---|---|---|
| 1 | 貴公司現行事件處理流程？(如何得知事件發生？如何確認異常原因？如何取得解決方式或文件？如何解決異常事件？) | 如何得知事件發生：透過軟體針對系統服務、使用 HP SIM 及 IBM Direcot 對硬體、自行開發程式對於應用流程進行監控，及部分透過人員通報。<br>如何確認異常原因：簡單的透過經驗及原廠文件說明，複雜的則會透過測試重現開問題及確認問題原因。<br>如何解決異常事件：一般事件透過系統端更改及修護、災害復原計劃來解決，複雜的透過架構或流程的更改來解決。 |
| 2 | 貴公司現行事件處理有何不足之處？您個人認為該如何改善？ | 資訊系統 Change 管理，相關人員的通知、記錄。 |
| 3 | 貴公司是否有使用事件關聯軟體協助處理事件(如 HP Openview)？是否安裝設定簡易？有無平台限制？ | 目前採用自行開發 Notes 系統做為異常事件管理系統，透過 Notes Client 即可使用，依 Notes Client 支援的平台為限。 |
| 4 | 您現行是如何得知系統的設定資訊(如資料庫參數，作業系統參數 | 異常處理時進行檢查，曾遇過設定異動而造成系統異常，應 |

| | | |
|---|---|---|
| | 等)？是否曾遇過因設定異動而導致系統無法使用的狀況？ | 該記錄異動資料，遇到異常可以回復上次正常的設定值。 |
| 5 | 是否瞭解 ITIL 事件管理流程？ | 否。 |
| 6 | 經展示本研究系統功能後，是否瞭解本系統功能與目的？ | 是。 |
| 7 | 本系統對貴公司異常事件處理是否有助益？有可幫助哪些方面？ | 對異常事件處理有幫助，可以記錄異常事件，對於曾發生過的事件可以很快的依系統內的解決步驟解決問題；系統 Change 管理可以做到檢查；Dashboard 部分可以清楚的知道系統運作情形與異常事件優先處理程度；組態管理方面可以知道異動前後的系統設定值，也方便系統回復使用。 |

# 附錄五、訪談回覆資料內容 E

| 任職產業別 | XX 科技-IC 設計 | 資本額 | 6 億 |
|---|---|---|---|
| 任職部門 | 資訊部 | 員工人數 | 100 |

| 項目 | 訪談內容 | 回覆內容 |
|---|---|---|
| 1 | 貴公司現行事件處理流程？(如何得知事件發生？如何確認異常原因？如何取得解決方式或文件？如何解決異常事件？) | 如何得知：User 回應，系統 Alarm。<br>異常原因確認：檢查回報異常部位，log 與先前經驗及紀錄。<br>Solution: 系統工具、SOP 文件、Google 查詢、重起 Service 或 Machine。 |
| 2 | 貴公司現行事件處理有何不足之處？您個人認為該如何改善？ | 交接事項未明確，應該要詳細填寫事件處理紀錄。 |
| 3 | 貴公司是否有使用事件關聯軟體協助處理事件(如 HP Openview)？是否安裝設定簡易？有無平台限制？ | 否。 |
| 4 | 您現行是如何得知系統的設定資訊(如資料庫參數，作業系統參數等)？是否曾遇過因設定異動而導致系統無法使用的狀況？ | 使用系統管理工具，Google 查詢原廠文件；曾遇過設定異動而導致系統無法使用的狀況，用回復預設值方式解決。 |
| 5 | 是否瞭解 ITIL 事件管理流程？ | 否。 |
| 6 | 經展示本研究系統功能後，是否瞭解本系統功能與目的？ | 是。 |
| 7 | 本系統對貴公司異常事件處理是 | 有幫助；解決事件建立紀錄； |

| | | |
|---|---|---|
| | 否有助益？有可幫助哪些方面？ | 若事件都是曾發生過，可不用花時間查詢網路的資料來取得問題解決方式；程式集中並有跨平台功能，可避免交接不完整狀況。 |