嘉南藥理科技大學專題研究計畫成果報告

計畫編號：CN9746

計畫名稱：應用資料倉儲技術所建構之醫院主管資訊系統

執行期間：97 年 1 月 1 日至 97 年 12 月 31 日

□整合型計畫　　■個別型計畫

計畫總主持人：　　計畫主持人：洪健文

子計畫主持人：

中華民國 98 年 02 月 20 日

（一）研究計畫背景與目的

1. 序論

在傳統上，國內醫院一般以傳統的營運報表 (如門診人數統計、收入金額)作為主管經營醫院的重要參考資料，但是由於最近幾年來，醫療市場的幾個急遽變遷因素，使得醫院也必須積極的導入多元管理方法，強化醫院的經營績效。而注意醫療品質的觀念也逐漸被重視，因此越來越多提昇品質的方法被醫院所運用，其中「標竿學習」與「同儕學習」是目前常用的方法之一。

但是要有效取得競爭同行 (醫院) 的資訊並不容易，縱使費盡千方百計取得的數值，也可能因定義的不同，資料蒐集的過程有異，而降低這些數值在比較與參考上的價值。

因此，如果能夠按照醫院的需求，透過第三者定期提供其它醫療院所的相關資料，以供改善參考，不僅可以提昇資料的效度與信度，並且也能降低資料運用上的主觀性障礙，達到改善醫療品質的目的，而這就是所謂的「指標管理」。在實務上，指標管理的進行會有一些重要的困難點。本文針對指標管理的困難點，藉由 EIS 的導入具體解決這些困難，使醫院在指標管理的運作上更加落實。

本文之研究目的主要在提出一雛型 EIS 系統，以解決目前醫院在執行醫療指標計畫時所遇到諸多困難。並進而驗證此雛型 EIS 系統所使用

的相關技術，是否能發揮功效。另外也驗證以既有指標計畫 (本文為 TQIP) 為基礎建制 EIS，是否能縮短系統的建制時程。

2. 文獻探討

台灣早期的醫療發展，對品質方面一直未有真正嚴格的規定，而 QIP(Quality Indicator Project) 這個計畫是目前在國際間擁有最多醫院參予的計畫，擁有強大的資料庫進行信度極高的指標值基礎。而負責台灣醫院評鑑的非政府組織 -「財團法人醫院評鑑暨醫療品質策進會」，則於 1999 年 8 月將 QIP 引進台灣，並將之命名為「台灣醫療品質指標計畫 (Taiwan Quality Indicator Project，簡稱 TQIP)，目前 QIP 已經發展的指標有急性照護、長期照護、居家照護以及精神照護等，而 TQIP 引進的部份，則只有急性照護的住院、急門診指標，以及精神和長期照護指標，總計 250 種細項指標，而這些指標數會隨著新指標的發展而有所成長

醫療品質指標資料的蒐集，可以區分成資料蒐集的種類與蒐集頻率。而蒐集的頻率是按照指標的性質來決定。資料的收集頻率可以區分為每月收集、每季收集，或每年收集，或是因特殊事件發生或政策的施行，實施不定期的監測。例如，醫院的人員數、病床數通常在一年以內並不會有大幅度的波動，因此就可以每一年為監測間隔；而門診量、住

院人數等指標，則在不同的月份會有程度上的變化，此種比較敏感的指標則需要每月進行監測。

由於指標性的範圍可能很廣泛，資料的蒐集頻率可能也很頻繁，因此在資料蒐集上，通常會造成一個負擔，如果在資料的準確度上要求很高，那麼蒐集的負擔自然也就更重。另外，規模較小的醫院通常也比較沒有能力可以進行持續的指標監測與評估。

2.1 EIS 電腦資訊系統

如果能將回饋資料，轉入一 EIS 系統中，那麼藉由與既有的歷史指標資料的結合，加上 EIS 系統所能提供給使用者的一些功能強大而又使用簡單的查詢功能，更可以 TQIP 的架構作為模式庫，進一步提供決策支援的功能，將可更落實指標管理的實務應用。EIS 這個名詞最早是由 Rockart 和 Treacy [1982] 在「CEO goes on line」中提到，並指出 EIS 是專門為主管所開發的「資料導向 (Data Oriented)」系統，可以支援主管進行管理、規劃、控制、監督與分析的過程 [Rockart & Treacy,1982]。

EIS 是一種電腦資訊系統，可以藉由親切的使用者界面，整合企業內外部的資訊，以使用者比較能夠接受的方式呈現資訊，並且可以達到一定的自動化、決策支援與人工智慧功能，提供管理者進行企業整體資訊的掌握，以便進一步應用到管理面 [廖云薇、梁定澎，1996]。比較新的 EIS 系統通常都設計了藉由友善的圖形界面以及 Drill down 的資料

分析，可以讓主管快速的存取與決策相關的資訊。因此 EIS 是一套資訊系統，可以透過文字、表格、圖形、影像等呈現方式，友善地提供高階主管內部經營狀況資訊、外部環境資訊以及例外情況報告，並且提供一定層級的決策支援功能。

2.2 EIS 有關的專有名詞

與 EIS 有關的專有名詞,可以在進行系統建立的時候提供定義以及技術使用上的參考：

(1). 企業內部 OLTP (Online Transaction Process) 資料庫：MIS 系統或是醫院的 HIS 系統所使用的即時線上交易資料庫，資料庫的特性必須是能夠快速的反應大量筆數資料的異動，並且資料庫系統必須具備很高的穩定性以及持續運作的能力。

(2). EIS 內部資料：EIS 必須能夠針對醫院內部的各種運作功能，如掛號、門診、住院、批價、人事、薪資、績效、品質控制，提供整合性的資訊。

(3). EIS 外部資料：EIS 的資料來源也有可能來自組織外部的資料庫，例如 TQIP 的回饋分析資料即是，以提供醫院外部大環境的資訊，能進一步應用到指標管理與標竿學習上。

(4). 親和性使用者界面：系統可以容易地由主管直接操作，而不須經由

中間者協助，使用者界面應該為圖形化的，可以提供給使用者更直覺式的操作環境。

(5). Drill Down：向下鑽取，這種方式可以快速的存取文字、圖形或是數字背後更詳細的資料。資料可以使用多層次的觀點，由彙總性質進入到細部資料。可以協助使用者進行 Top-Down 的資料方式。

(6). Roll Up：與 Drill Down 相反，可以協助使用者進行 Bottom-Up 的思考方式。

(7). Slice and Dice Query：不同的人觀察相同的資料，所持的觀點可能不同，例如內科醫師與外科醫師在查詢住院死亡率的資料時，資訊顯現的方式與意義可能不一樣，就有如在觀察一個立方體 (Cube) 那樣，將立方體進行切割以觀察不同的點，此種查詢方式稱之為 (Slicing)，也就是針對立方體或是多維度的資料，以翻頁的方式，一頁一頁的查詢，而 Dicing 則是以旋轉的方式瀏覽所觀察的立方體或是多維度資料。

(8). 趨勢分析：當累積的一段時間的歷史資料之後，便可以使用時間序列的方式，對資料進行趨勢分析，例如使用 Run Chart 進行分析。

(9). 例外報告：對於有例外情況的指標提供警示的功能，提醒使用者注意。

(10). 預測功能：提供預測功能給決策者作為決策參考。

(11). What-if 分析：針對問題進行推演。

(12). Data Warehouse：資料倉儲，可以作為 EIS 的資料基礎，大大的提昇了 EIS 系統資訊整合以及提供能力。

(13). Data Mart：為資料倉儲的一部份組成，扮演著部門、區域性或功能性資料倉儲的角色。

(14). OLAP (On-Line Analytical Processing 線上分析處理)：OLAP 可以提供給使用者對特定資料進行線上分析的功能。這個觀念是由有名的 Dr. E. F. Codd 所提出，他之前在 1985 年的時候，提出了關聯式資料庫的 12 項原則，對資料庫的發展產生了決定性的影響，後來在 1993 年的時候，由於看到企業的資料分析需求，又提出了 OLAP 的 12 項原則，包括：多維度空間、通透性、存取性、主從架構、動態分散矩陣處理、支援多使用者，以及可以提供無限制的跨空間處理、直覺式的資料產生、彈性化的結果呈現等。

(15). MDD (Multi-Deimension Database 多維度資料庫)：以多維度的觀點儲存資料，以方便使用者以不同的觀點分析資料。

(16). Metadata：中介資料，用來描述實體資料的資料，可能存放在另外一個資料庫，例如OLAP 軟體可以透過 Metadata 存取後端的資料庫，但是目前還未有普遍接受的標準。

(17). MOLAP：多維度資料庫線上分析處理，也就是 MDBMS+OLAP，

Metadata 儲存在其它專屬的資料庫中。

(18). ROLAP：關聯式線上分析處理，也就是 RDBMS+OLAP，是將 Metadata 儲存於關聯式資料庫中。

2.3 EIS 與資料倉儲

在資料倉儲的建制過程中，有幾點是必須要注意的，才能確保資料倉儲建制的成功。Silverston、Inmon 與 Graziano [1997] 認為，在將資料轉入資料倉儲之前，必須瞭解從作業環境到決策支援系統轉換流程的三個模式：

(1). **組織資料模型** (Cooperate Data Model)：此模型為開發資料倉儲的基石，用來描述組織內部不同主題彼此間的關係。

(2). **資料倉儲模型** (Data Warehouse Model)：此模式為一種整合、主題導向的策略性資訊。

(3). **部門資料倉儲設計** (Departmental Data Warehouse Design)：就是由資料倉儲按照部門區分所切割開的 Data Mart。

傳統的資料庫架構，也就是 OLTP 資料庫架構，通常都是正規劃的架構，因此整個資料庫模式並不是為了資料倉儲的需求而建制的，必須經過一些轉換程序，以建立成資料倉儲資料模型。

建立資料倉儲的步驟如下：

建制組織資料模型 (Cooperate Data Model)：

(1). 蒐集轉換的需求。

(2). 建制高階資料模型。

(3). 建制邏輯資料模型

這部份的模型我們可以使用 TQIP 的定義作為架構。

進行資料模型的轉換：

(1). 移除純粹操作性的資料。

(2). 若沒有時間因素，則必須在資料倉儲中加入時間因素，並作為鍵值。

(3). 加入適當的衍生資料 (Derived Data)。

(4). 適度的合併資料表。

(5). 適度的指定資料陣列。

(6). 組織資料倉儲中的資料。

2.4 建制資料倉儲所使用的資料模式

在 OLTP 的資料庫中，一般會採用實體關係圖 (Entity Relationship Diagrams, ERD) 和正規化技術 (Normalization Techniques)。但是前面也曾經提及，資料倉儲的特性並不適用於採用這兩種技術。在目前業界所看到的資料倉儲實際設計中，會採用所謂的星狀綱要 (StarSchema)，以

說明多維度的資料格式 (Multidemensional Data Model)。此種星狀綱要架構所建立的資料庫，會包含一個事實資料表 (Fact Table)，與多個維度資料表 (Demension Table)，在事實資料表中，會有用來指向其它維度資料表的外來鍵值欄位。另外還有一種由 Star Schema 變化而來的架構， 稱之為雪花綱要(Snowflake Schema)，與 Star Shema 不同的是，雪花綱要將比較複雜的維度資料表加以正規化，以便能夠更方便的維護維度資料表。

導出 EIS 資料與資訊需求，是 EIS 開發過程中非常重要的一部份，例如Kimball [1996] 就指出訪談使用者是設計資料倉儲的首要步驟，可以讓設計者瞭解使用者的需求與期望。在訪談使用者之後，則必須繼續訪談各資料來源系統的資料庫管理員(DBA)，因為這些 DBA 是這些來源資料庫的專家。訪談的內容可能包括了不同系統之間資料的關連性等問題。由於本雛型架構將採用 TQIP 作為資訊需求的基礎，因此在資料與資訊需求的導出上，也就直接採用 TQIP 已經清楚定義的資訊定義與需求，在使用者需求上，可以直接考慮指標管理的實際應用需求。

在 EIS 的開發過程中，我們可以從目前已經存在的 EIS 系統中，看到可能存在的開發問題與開發限制，像是如何從各種不同的異質資料庫，甚至是文字檔或 Excel 這些檔案，進行資料的轉換與彙整？如何確保不同來源的資料內容是一致的，或是在欄位定義上是相同的，而不會

產生有問題的資訊 (Dirty Information)？資料彙整之後，經過一段時間的累積，會產生大量的歷史資料，如何有效率的儲存這些大量資料，並且可以快速的從大量的歷史資料中有效率的查詢所需要的資訊？如何自動化的蒐集外部資料，以提昇 EIS 在策略上的應用？如何能夠符合使用者的需求主題？

3. EIS 雛型架構與系統實作

現今企業或組織的資料通常是散佈於不同的系統中，而中大型醫院也不例外。例如醫院可能有個主要的大型主機存放重要的門診與住院資料，但是可能急診或是人事等資料則存放在個別的資料庫，甚至有些資料會以其它資料庫或文字檔案的格式，存放在小型或是個人電腦的硬碟中。如此一來，決策人員或指標資料蒐集人員想要取得的指標資料就很可能分散於不同形式的資料來源，造成取得上的困難。為了解決上述的問題，本文計畫建立具資料倉儲特色的 EIS 系統，但是有如之前所提到的，在進行資料倉儲與 EIS 系統的開發上，通常會面對的一個問題，就是資料定義是否明確的問題。而 TQIP 則可以直接解決這樣的問題，因為 TQIP 已經明確的定義各指標的定義以及資料轉換公式。因此我們可以輕易的以 TQIP 為基礎，快速的建構出 EIS 系統。

此雛型系統的構想以及想要達到的功能如下：

1. 使用 Microsoft SQL Server 2000 所提供的資料轉換服務 (Data Transformation Service, DTS)，提供資料轉換功能，將各種異質來源的資料，轉換、過濾、彙整到資料倉儲中。
2. 建構 Cube Transformation Service，將彙整的資料轉換成適合 TQIP 架構EIS 系統使用的資料結構。以 Cube 為架構的資料倉儲，適合決策人員進行指標分析、查詢 (與 OLTP 有所區隔)。
3. 藉由前面所提供的兩種服務，建立具一致性、準確性與資料儲存性的資料倉儲，可以提供長期的資料指標分析資料來源。
4. 將資料轉換與 Cube 轉換服務封裝成 COM 物件，封裝的內容可包括資料來源、轉換邏輯、資料衍生邏輯與資料轉換時間點等屬性，達到自動化的免維護機制。
5. 資料是歷史性的，不會經常異動，資料倉儲的資料通常是每隔一段時間才由 OLTP 資料庫輸入新的資料。
6. EIS 系統提供圖形的方式進行資料的呈現與分析。
7. 提供回饋資料的顯示與整合能力。
8. 除了一般 SQL 語法之外，如有效能上的需要，可針對資料倉儲，使用另外一種決策分析查詢語言 – MDX (Multidimensional Expressions – 多維度運算式)。

3.1 系統設計

在系統設計階段，會有一些重要的文件產生，並且也必須決定一些重要的原則，分別說明如下：

(1). 資料庫的設計文件：就是將邏輯資料模式轉換成實體設計綱要，有了這個文件，我們就可以根據這份文件的設計，直接建制在轉換機制 – Data Transformation Service中。

(2). 維度資料表的設計：例如，在「新生兒」這個指標中，會有像是「時間維度」、「出院狀態」以及「體重」等不同的維度。

(3). 資料萃取與轉換演算法則：本文使用 MS SQL Server 2000 所提供的 Data Transformation Services - DTS，在 SQL Server 與其它支援 OLE DB、ODBC 的資料庫 (如Oracle) 或純文字檔間，進行資料的 Export、Import 和 Transformation。藉由所建立的文件，使用 MS SQL Server 2000 所提供的 DTS Designer，將來源資料進行「萃取」與「轉換」成儲存在資料倉儲中的「多維度資料」。另外在建議轉換機制的時候，也必須決定資料的載入時間以及維護策略。

(4). 使用者界面的設計：設計使用者界面的功能、配置，另外也必須決定採用的開發工具以及技術。

(5). 回饋機制設計：針對回饋的資料，要採用何種回饋機制匯入到資料倉儲中；在何處匯入資料等。

3.2 系統實作

系統實做階段則是將設計好的文件轉換成實際可運作的資訊系統，而程式的撰寫與測試是此階段的主要工作。實做階段的產出項目包括：轉換服務系統的建制、資料倉儲資料庫、前端界面功能開發、可運作的最終軟硬體架構、使用者的訓練與使用說明書。

3.2.1 建制資料倉儲資料庫

由於僅是雛型系統，故於一般的 PC 上安裝 MS SQL Server 2000 建立資料倉儲。在實體資料庫上，安裝並設定與異質資料來源進行連線的中介軟體(Middle Ware，如 OLE DB、ODBC 等)。然後使用 SQL Server 的 DTS Designer建立資料的過濾與轉換機制，並封裝成 COM 物件，以達到完全自動化的能力。另外計畫使用 IIS 建制 Web Server，以支援前端瀏覽器的程式讀取與執行，並建立與資料倉儲的資料連線能力。

3.3 系統之維護與運作

本文的一個重要目標，是希望能夠建立自動化的機制，因此系統的維護和運作是應該要探討的問題。本研究將針對下列幾點探討：

(1). 運作流程的自動化能力。

(2). 系統效能的調校。

(3). 系統維護。

3.3.1 運作流程之自動化

使用 Microsoft SQL Server 2000 的 Data Transformation Server 與 Cube Transformation Service 提供資料的來源連線設定、萃取、轉換，以及多維度資料表的建立 (Cube)，並且設定轉換時間，最後封裝成 COM 物件，放置於伺服器上，以達到完全自動化的資料轉換運作。另外在 SQL Server 2000 的資料轉換機制中，使用 JavaScript 語法撰寫複雜的資料轉換邏輯，以建立自動化資料轉換機制。

3.3.2 系統效能調校

整個系統的調校是一個持續的流程，隨著資料量的增大，以及使用者的增加，都可能會影響系統的效能，因此應該使用下面的方法進行系統效能的調校： 適當的產生 Aggregation 或 Summary 資料表。

(1). 轉換機制 SQL 語法的運用與調校。

(2). 查詢語法的調校。

(3). 儲存空間的切割與調校。

(4). 資料表的調校。

(5). 索引的調校。

3.3.3 系統維護

在系統的生命週期中，系統的功能可能會隨著使用者的需求而改變，系統環境也會隨著新的硬體平台而改變，在系統驗收時沒有發現的錯誤，也可能會在使用一段時間之後慢慢浮現。而維護則是試圖改變系統以維持系統服務功能的過程，與 EIS 系統維護相關的類型可能有：

(1). 資料需求的增加或修改、移除：例如由於某些指標的需求，可能有新的指標定義加入，或是某些指標已經喪失意義，因此需要移除。

(2). 使用者功能上的錯誤修正，或是新增與改良等。

(3). 實體伺服器的升級：隨著使用人數的增多，或是資料量的成長，原來的硬體架構可能已經不敷使用，則必須進行實體伺服器的硬體升級。

4. 雛型系統建構說明與評估

4.1 系統設計與實作說明

4.1.1 系統網路架構

目前一般中大型醫院均架設有以太網路 (Ethernet)，而本研究所建制的雛型系統，則應用若瑟醫院既有的以太網路架構，採用的 TCP/IP (TransmissionControl Protocol/Internet Protocl) 作為通訊協定架構，這也是目前網際網路所採用的通訊協定。因此，前端的使用者透過一般的瀏覽器 (Browser)，就可以使用系統，甚至透過網際網路，在醫院防火牆

的外部使用 EIS 系統。

4.1.2 轉換機制與資料倉儲設計與實作

我們開始使用 Microsoft SQL Server的「資料轉換服務 DTS」，針對來源資料進行檢查、合併或計算，然後以資料倉儲的綱要模式進行資料的轉換與儲存。針對比較複雜的轉換計算，則可以使用 DTS 所支援的 Java Script 或VBScript 語言，並且配合 T-SQL 語法建立轉換機制。本文所使用的語法為Java Script 配合標準的 SQL 語法撰寫轉換規則。現在逐步將轉換機制與資料倉儲的建立說明如下：

使用 DTS 建立轉換機制：

DTS 是 MS SQL Server 所提供的圖形界面資料轉換機制建立工具，在此以事實資料表 (FACT_TQIP4) 為例進行說明，其它維度資料表的建立方法，除了演算規則與結構不同之外，工具的建立步驟基本上是相同的。

在 DTS 中所建立的資料轉換步驟大致如下：

(1). 先將來源資料根據事實資料表的需求，以及維度的區分方式轉換到一個暫存資料表 TQIP_TEMP ，以方便進行轉換到資料倉儲。轉換前先清除TQIP_TEMP 的內容。

(2). 清除事實資料表殘存資料。

(3). 從暫存資料表 TQIP_TEMP 轉換到事實資料表，並進行適當的轉

換。

(4). 轉換其它維度資料表。

在建立轉換機制的時候，會先建立資料表，並且決定各資料表的索引鍵。設計好 FACT 資料表與其它維度資料表轉換機制的 DTS 畫面如圖 1-1。DTS 以圖形的方式，讓轉換機制的建立更為視覺化，本研究在使用 DTS 的時候，選擇使用的語法為 SQL 與 JavaScript，對於有程式設計經驗的人員，是很容易熟悉的。

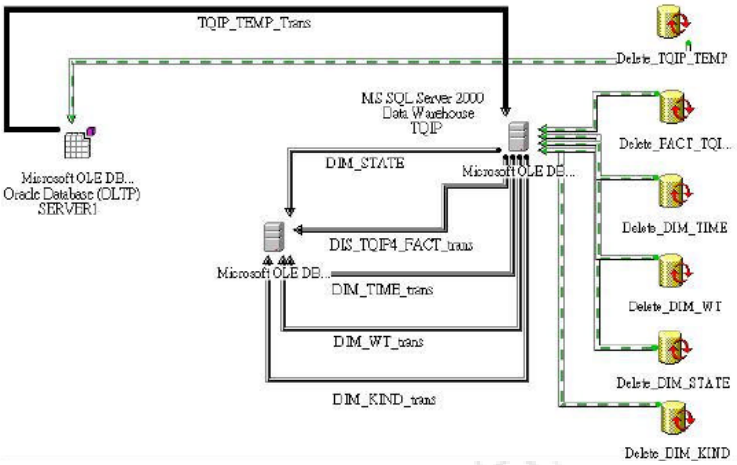

圖 1-1　資料轉換流程

4.1.3 使用者界面說明

使用者的界面使用 Web Base 的方式開發，將開發後的決策支援程式，以ActiveX 元件的方式 (.ocx 檔) 內嵌於網頁中。所提供的使用者界面，除了可以提供醫療品質主管或專員查詢 TQIP 指標相關數據之外，並且以圖形的方式顯示。另外提供選單讓使用者選擇指標的種類、

查詢指標資料的時間。

整個操作的畫面都是在 Web Browser 中進行,使用者可以選擇想要查詢的指標種類、指標年份以及區域等選項，然後按下查詢按鈕，就可以顯示出想要查詢的指標資料，如圖 1-2 的畫面。

圖 1-2 使用者界面

在多維資料分析上，可以讓使用者依據需求組合不同維度的資訊，並且也可以看到圖形的差異。資料可以用一維、二維、甚至三維以上的方式呈現，另外也可以對不同的維度進行 Drill-Down 請參閱圖 1-3。

圖 1-3 多維資料分析

而維度欄位則可以使用滑鼠拖曳的方式，在不同的位置 (X 軸、Y 軸) 之間切換，方便使用者從不同的角度進行資料分析。在使用者界面中提供了資料轉入的機制，可以讓使用者將 TQIP 所提供的資料匯入資料倉儲中，不過這部份的機制是使用固定程式碼撰寫的，如果萬一TQIP 回饋資料格式有變動，則會造成資料匯入錯誤的問題，因此如果能夠針對這部份進行 XML 資料交換的建立，應該是比較理想的方式。

由於使用者可以在 Grid 中看到本地醫院以及其它區域的指標資料，因此可以直接在 EIS 的使用者界面上進行更有意義的資料比較。在 Grid 的下方則提供了圖形的顯示方式，圖形的產生是與資料動態連結的，也就是 Grid 中的資料會與圖形的呈現一致。目前雛型系統只提供長條圖方式，也可以另外提出讓使用者選擇不同圖形格式的功能，讓系統功能更加完整。圖 1-4 為圖形的顯示範例。而維度欄位則可以使用滑鼠拖曳的方式，在不同的位置 (X 軸、Y 軸) 之間切換，方便使用者從不同的角度進行資料分析。

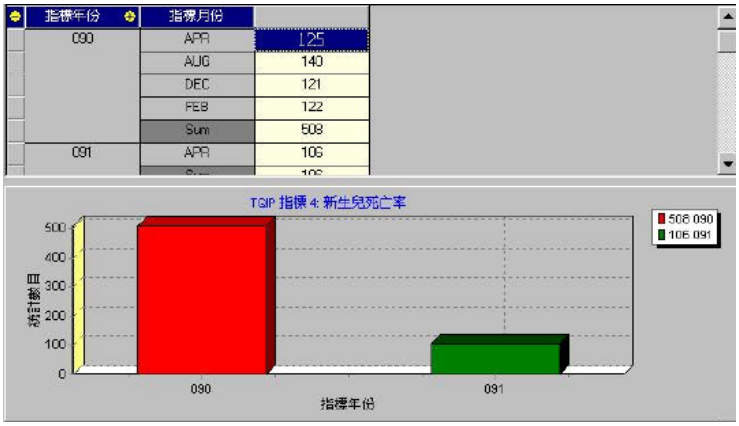

圖 1-4 顯示範例

在使用者界面中提供了資料轉入的機制，可以讓使用者將 TQIP 所提供的資料匯入資料倉儲中，不過這部份的機制是使用固定程式碼撰寫的，如果萬一TQIP 回饋資料格式有變動，則會造成資料匯入錯誤的問題，因此如果能夠針對這部份進行 XML 資料交換的建立，應該是比較理想的方式。由於使用者可以在 Grid 中看到本地醫院以及其它區域的指標資料,因此可以直接在 EIS 的使用者界面上進行更有意義的資料比較。在 Grid 的下方則提供了圖形的顯示方式，圖形的產生是與資料動態連結的，也就是 Grid 中的資料會與圖形的呈現一致。目前雛型系統只提供長條圖方式，也可以另外提出讓使用者選擇不同圖形格式的功能，讓系統功能更加完整。

4.2 系統自動化與效能

以 MS SQL Server 2000 的 DTS 建立好需要的轉換機制之後，整個機制可以儲存成「封裝」，並藉由為封裝進行排程，在指定的時間進行自動化的資料轉化，達到資料倉儲資料更新的功能。以指標 4 為例，指定系統在每月的月初 1號凌晨 1 點進行資料轉換,就可以讓資料倉儲隨時保存有最新的指標 4 資料。請參閱圖 1-4。

在建立資料轉換的時候，如果由於 OLTP 資料庫架構的關係，需要進行複雜的資料轉換與演算，那麼就要注意 SQL 語法撰寫所可能產生的效能問題。在設計資料倉儲的時候，最小資料顆粒的考慮也是很重要

的，如果資料量過大，會影響使用者進行資料查詢時的效能。另外也可以使用查詢維度資料表專用的 MDX 語法，可以提昇針對資料倉儲架構的資料表查詢的效能。本研究並沒有針對 MDX 語法進行實作。

4.3 系統成效評估

就整體而言，達到下列幾點效果：

(1). 達到自動化蒐集指標資料的成效。

(2). 藉由使用 TQIP 專案簡化 EIS 架構資訊定義的步驟。

(3). 實作 Web-Base 架構的 EIS 界面，所建制的使用者界面，能夠直接讓使用者操作，並且應用於醫療品質的管理上。

5. 結論與討論

根據本文所建制的以醫療品質指標為基礎之主管資訊系統，希望能針對醫院中醫療品質指標的管理有實際的助益。一般參與 TQIP 專案的醫療院所所遇到的問題，大部分是在資料蒐集、整合的困難，以及指標數據分析與應用到實際管理面無法落實。藉由本研究雛型系統的提出、實作以及測試，得出一些結論如下：

(1) 必須要能自動化的資料轉換機制，本研究的整個系統除了第一次的系統建置需要有技術人員支援之外，後續系統便可以自動進行資料的蒐集與轉換。

(2) 要能提供方便使用者操作的 EIS 使用者界面，可以輕易地取得所需要的指標資料進行分析，以協助進行醫療品質指標的管理。

(3) 在開發 EIS 系統的過程中，如果能搭配既有的醫療品質計畫，可以增快 EIS系統的開發時程。

(4) 在使用瀏覽器為平台開發 EIS 使用者界面方面，則非為必要。另外如果採用瀏覽器為平台開發前端系統，一方面可能會對應用程式的功能產生限制，並且也會增加開發上的技術困難性。

整體而言，雛型系統所提出的解決方案，解決了指標資料蒐集的問題，並且也能藉由只儲存轉換過的彙總資料，可以使用較低的成本儲存長期的指標資料，並且應用了 TQIP 指標系統的明確定義，省卻開發 EIS 系統在資訊定義建立上冗長與沈重的工作過程。而所提供的 EIS 界面，也讓管理人員更方便的解讀指標數據，並且有助於將指標數據實際應用到醫療品質的管理面。

本文只以單一指標呈現資料建立的雛型系統，並且只採用 TQIP 所定義的資料維度，因此後續研究應該可以擴大雛型系統，加入其它 TQIP 指標，讓整個指標資料的呈現更加完整，使用者的評估也會更加完整。另外可以就醫療院所的實際需求，以 TQIP 為架構基礎，加入其它非 TQIP計畫的管理指標，可以建立更完整的 EIS 決策支援系統。由於 TQIP 是一個世界性的計畫，所有的資料是由全世界所有參與會員醫院

分享，因此跨地區資料的交換也是一個重要的問題，TQIP 除了書面的資料，也提供了電子檔的磁片，但是如果能夠藉由一些可以做到自動化的資料交換技術，如 XML，那麼對整個 EIS 系統會有更好的效果。

參考文獻

1. Barquin, R. C. and Edelstein, H., " Planing and Designing the Data Warehouse" , Prentice Hall, Inc, 2001
2. Dodge G. and Gorman T., " Oracle8 Data Warehousing" , Addison Wesley Longman, Inc, 2003.
3. Inmon, W.H., Rudin, K., Buss, C. K. and Sousa, R., " Data Warehouse Performance" , Addison Wesley Longman, Inc, 2006.
4. Kimball, R. and Merz, R., " The Data Webhouse Toolkit Building the Web-Enabled Data Warehouse" , John Wiley & Sons, Inc. 2007.
5. Kimball, R. and Merz, R., " The Data Webhouse Toolkit" , John Wiley & Sons, Inc, 2007.
6. Jacobson, R., " The Microsoft SQL Server 2005 Analysis Services Step by Step" , Microsoft Press, 2008.