

## 第五章、中介軟體開發實例

中介軟體泛指 RFID Reader 和後端主機應用程式的中介媒體，除了將收集到的訊息傳遞給主機，也對訊息來源進行過濾和除錯。基於第三章所討論之目標與策略，初期在硬體效率不盡理想下，當以發展資訊系統為先。在國內 RFID 應用尚不普遍的情況下，為解決物流業初期導入 RFID 技術時，且硬體設備不提供軟體服務之困難，本研究針對特定廠牌之 RFID 設備及物流公司之資訊需求，開發應用程式介面，溝通 Reader，接收 Tag 資訊，傳送到後端資訊系統作資訊處理。本研究欲完成的工作如圖 24：

1. Tag 與 Reader 間的資訊讀寫。

研究 Tag 與 Reader 之間的溝通方式，並且確認 Tag 所攜帶之資訊是合乎個案公司所要求的。

2. Reader 與後端資訊系統的資訊交換。

研究 Reader 的基本設定，如通訊協定設定。並利用 Visual Basic 程式語言撰寫應用程式介面，完成 Reader 與後端資訊系統的資訊交換。

3. 瞭解個案公司對中介軟體程式介面之功能需求。

4. 與個案公司進行系統整合，驗證完成的中介軟體介面之可行性。

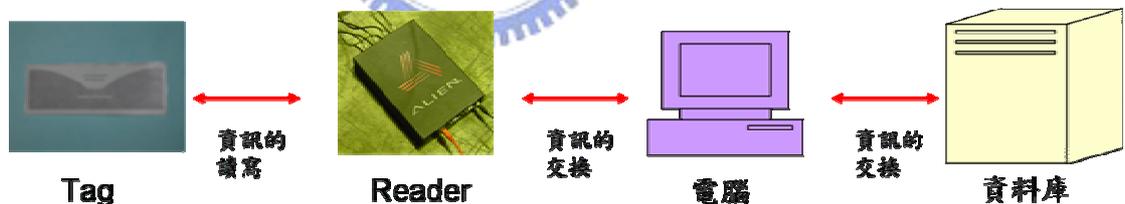


圖 24、RFID 資訊傳遞示意圖

資料來源：本研究整理

在不影響個案公司正常作業下，本研究所進行之資訊系統開發以近似個案公司之原始系統為主。其中與個案公司共同討論，針對 RFID 造成資料庫資料表之更動，設計新的資料表(如附錄三)，便於資訊系統管理。待日後個案公司實際導入 RFID 資訊系統時，期望本研究之開發經驗與所遇到之困難，能提供參考價值。而初期僅就封閉系統，僅以單一物流中心之作業為目標，進行資訊系統資料交換可行性驗證，後續可分產業別或供應鏈別，以開放性網路來建構 RFID 系統，發揮 RFID 更大之運用。

## 5.1 物流資料交換

本研究以個案公司之進貨和出貨作業流程作為研究主題，針對其需求作為程式開發。個案公司之物流資訊在進貨作業導入 RFID 前後之物流資料交換變更如圖 25 與圖 26：

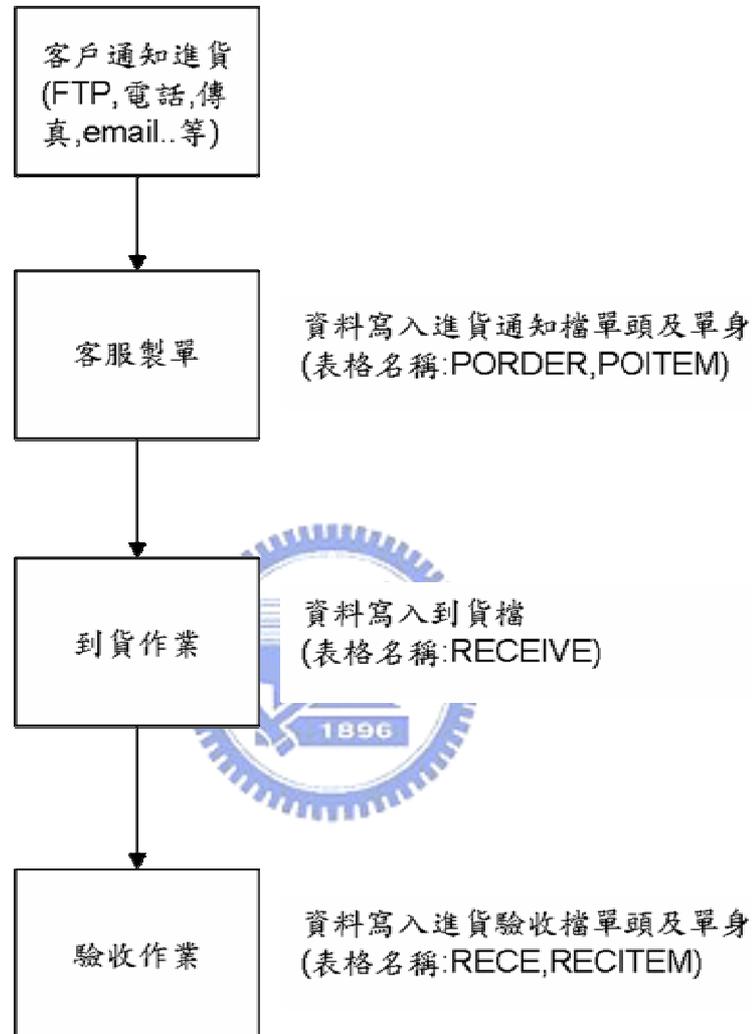


圖 25、導入 RFID 前進貨作業流程

資料來源：本研究整理

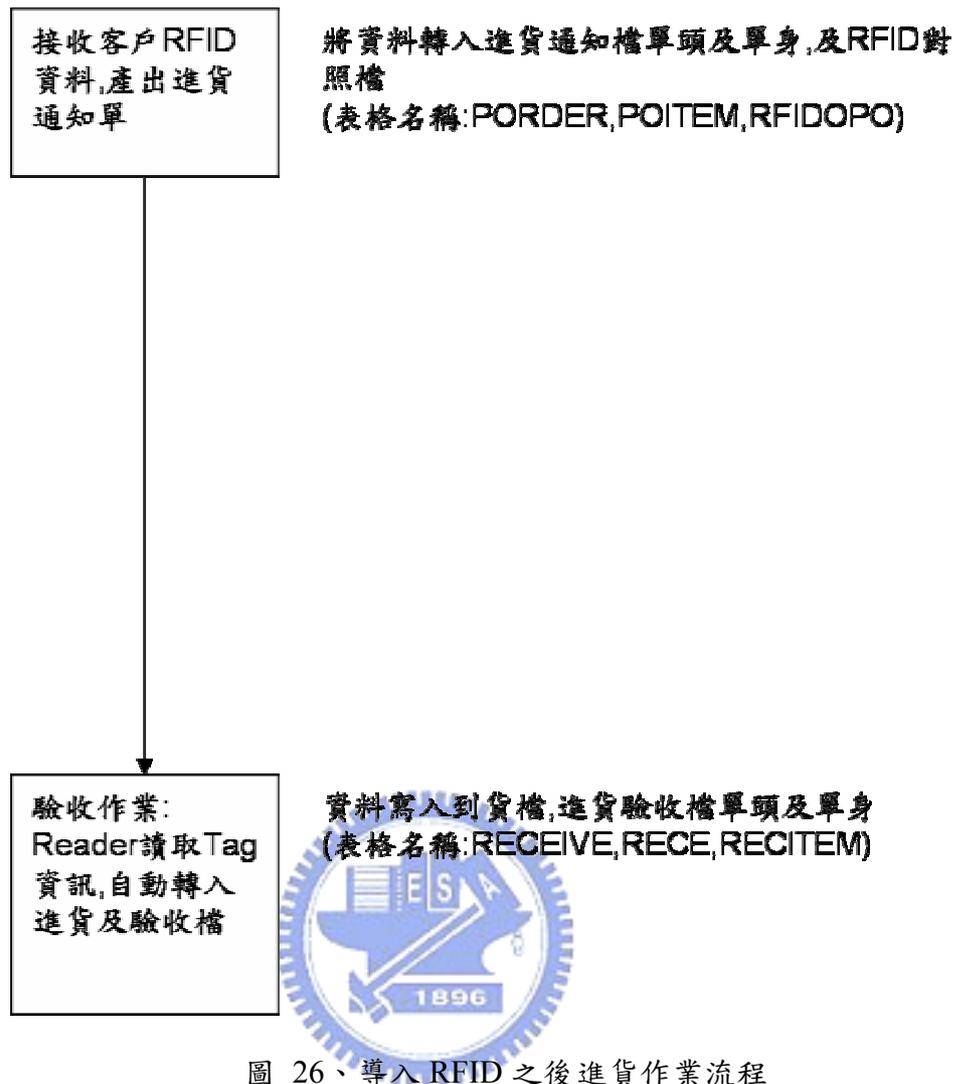


圖 26、導入RFID之後進貨作業流程

資料來源：本研究整理

1. 客戶通知進貨。

目前個案公司接受客戶訂單多以 FTP (File Transfer Protocol)、電話、傳真、E-mail 等方式進行，尤其是以 FTP 管道最多。為方便個案公司作倉儲管理，其資訊系統有一定之格式，待轉檔入系統中較為快速且便利。目前針對訂單格式處理有二種方式，一為原始訂單沒有符合個案公司之格式，待接受訂單後再轉換為個案公司所制定的訂單格式;若部分廠商願意接受個案公司所制定的訂單格式，如此一來，在客戶下訂單時，個案公司就不用再作訂單轉檔的動作。

2. 客服製單。

客服人員接收到進貨通知後，寫入進貨通知，以進貨通知單來作收貨。

3. 到貨作業。

而在進貨時，進貨人員以進貨通知所列之明細，進行到貨作業。到貨完成再以人工動作，輸入 WMS，寫入至到貨檔。

4. 驗收作業。

以到貨檔列印成驗收單，再行驗收動作。驗收無誤後，再將資料寫入進貨驗收檔，完成整個進貨作業。

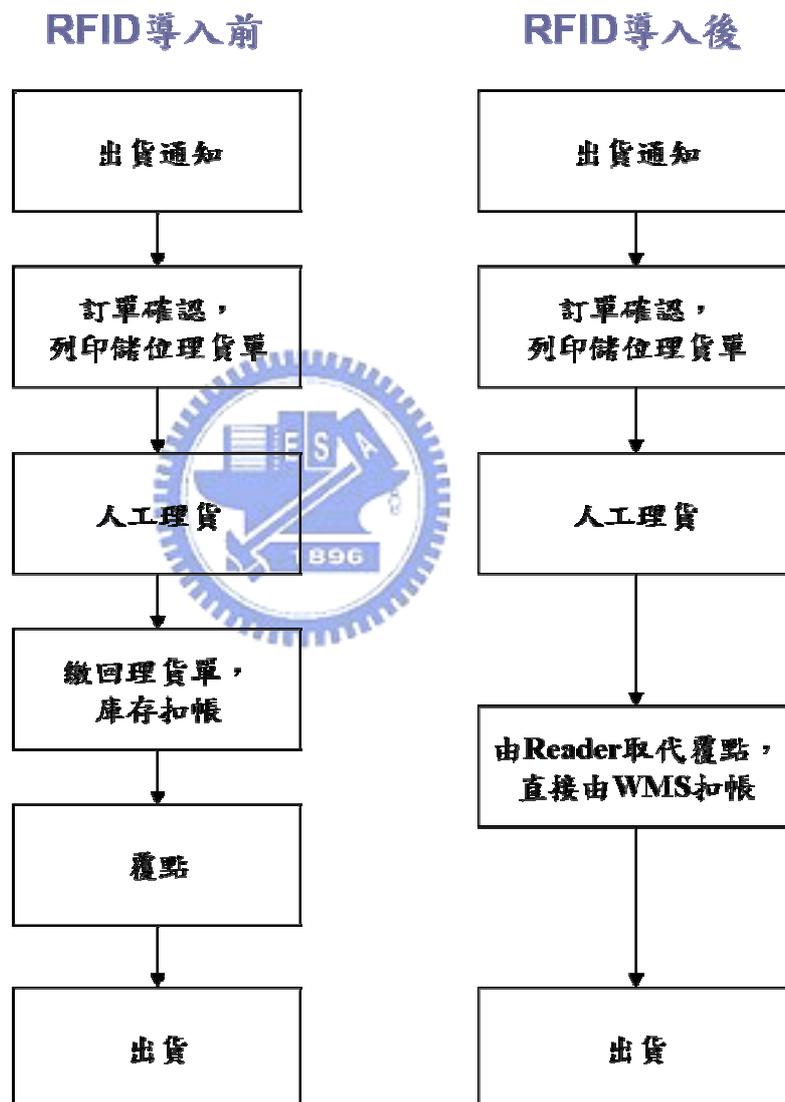


圖 27、RFID 導入前後出貨作業流程

資料來源：本研究整理

出貨作業主要影響的是 Reader 可以取代人工覆點的作業，並直接由 WMS 扣帳，無需以人工輸入理貨單之內容，再行扣帳。人工覆點是件費時的工作，其工

作時間多半決定於經驗之累積與對貨物之認知度。倘若以 RFID 技術來取代之，不僅可加速作業時間，對於覆點人員貨物之辨識度也有莫大的幫助。

接收上游客戶之 RFID 資料，在個案公司資料庫檔案中產生 RFID 對照檔，以增加 RFID ID 之辨識度。由此發現若 RFID 的資料形式有標準的格式，則供應鏈間的資訊傳遞將會更有效率，省去轉檔的時間。在個案公司中，可省略客服製單和到貨作業之轉檔動作。由此可知，一套標準的檔案格式可節省個案公司轉檔所花費的時間，而這標準化的格式也是業界所樂於看見的趨勢。

## 5.2 設備環境簡介

RFID 設備會因使用環境不同，對於所選用的器材也有所不同。適合物流中心環境使用，以 UHF 頻段較為合適。本研究針對 UHF 頻段選購 Reader 和 Tag 來作為測試，以下依硬體和軟體需求來討論。

### 5.2.1 硬體需求

由第二章所介紹的 RFID 基本工作原理與組成，可以得知 Tag、Reader 和電腦系統是構成 RFID 運作之三大主體。為進行本研究，自行購買硬體設備，如表 12。

表 12、硬體設備

設備名稱	型號
Reader	ALR-9780
Tag	ALL-9254
Antenna	ALR-9780

資料來源：本研究整理

以下分別對硬體設備作介紹。

- Reader：UHF Port 4 Antenna，可同時讀取多個 Tags。Reader 的規格如表 13：

表 13、Reader 規格

Model:	ALR-9780
Frequency:	902-928 MHz ISM band
Specification:	EPC Class 1
Antennas:	4-ports, 50 Ohm, reverse TNC connector
Power Supply:	12VDC, 2A (unregulated)
Power Consumption:	25 Watts
Communication Interface:	RS232; 9-pin, Sub D (female)
Communication Settings:	Baud 115200, Data bits 8, Stop Bits 1, Parity None
LAN Interface:	TCP/IP
Inputs/Outputs:	4 programmable logic I/O
Indicators:	Power , RF , Sniff, Lock
Operating Temperature:	-20C to +55C

資料來源：本研究整理

- Tag：EPC Class 1 UHF Tag，64bits，"M"形 Antenna 設計，定位感應，大概的尺寸為：95mm x 30mm，良好的總體性，高性能的的標籤解決方案。
- Antenna：ALR-9780 Reader 可外接 4 個 Antenna，Reader 透過 Antenna 傳遞射頻信號，並接收 Tag 所傳回之電波。

### 5.2.2 軟體需求

以 Visual Basic 為開發語言，並以 Access 為資料庫，建立符合 RFID 資訊系統之程式應用介面。

### 5.3 程式邏輯分析

系統主要著重於與 ASN 的資料是否吻合，因此最重要的觀念在於 Reader 所讀取到的資料是否與 ASN 資料表相一致。當比對過後無誤後，再執行更新資料庫的動作。若有資料不符合，則發出一警示訊號。其程式主要邏輯如圖 28 所示。

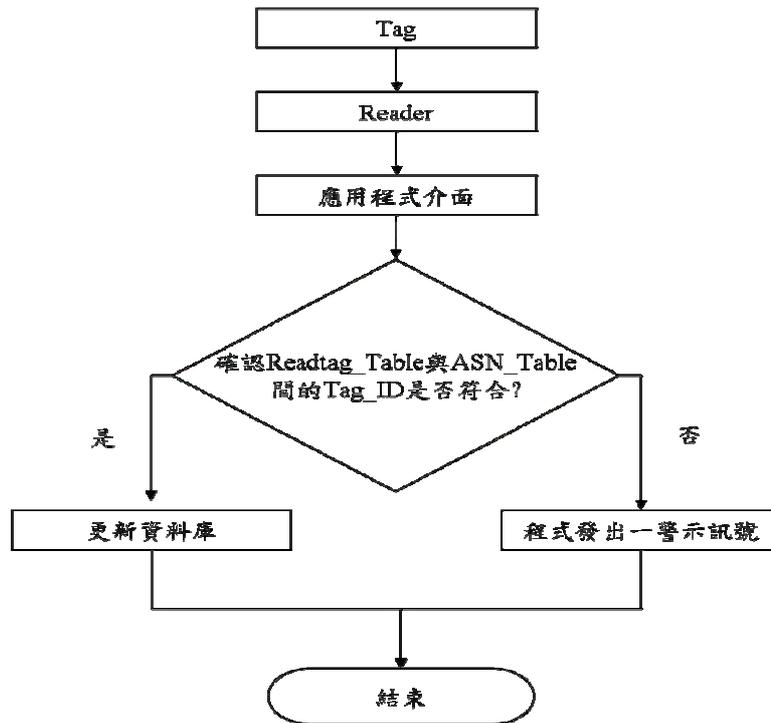


圖 28、系統分析圖

資料來源：本研究整理

本研究之精神在於協助個案公司初期導入 RFID 之準備，並參與個案公司於政府推廣 RFID 之計劃，因此許多細節皆沒有深入探討，主以個案公司之計劃需求為開發主軸，輔以本研究所學習之 RFID 知識，與個案公司人員分享並討論之，共同構築 RFID 中介軟體之應用程式介面，作為初期 RFID 導入之雛型。

#### 5.4 介面程式開發

本研究所設計的 RFID 資訊系統主要包括四個功能表單：進貨、出貨、資料維護和查詢(圖 29)。以下分別就這四項功能表單分節討論。

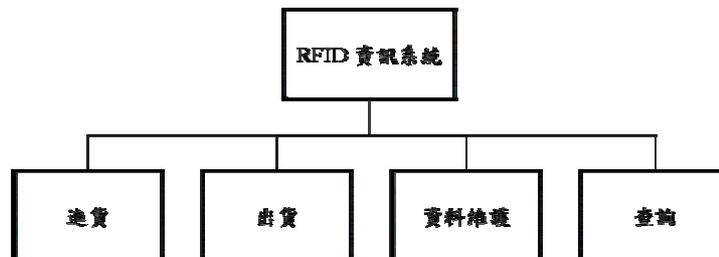


圖 29、RFID 資訊系統

資料來源：本研究整理

### 5.4.1 進貨表單

初期規劃以進貨與出貨作業為主。進貨作業如第三章所示，進貨之關鍵在於比對貨品與貨單是否正確，因此在此部分設計「進貨表單」作為進貨時資訊之判讀。

進貨表單設計主要設計三個表格：Taglist、沒有 ASN 記錄之 ID 和 Tag ID 詳細資料。進貨處理的關鍵在於如何過濾掉重覆讀取的 ID，而所讀取到的 Tag ID 必須與 ASN 作比對的工作，確認在進貨通知內有該筆記錄，以免無訂貨記錄而收錯貨。另一方面也可以由每張的 ASN，檢查貨物是否皆已驗收。於是設計兩張資料表來呈現與 ASN 比對後之記錄，作為進貨管理之便，其效果如圖 30。而為提高進貨時驗貨人員對於貨物之辨識度，在讀取的第一時間將 ID 所代表之詳細貨物資料顯示出來。

RFID進貨作業			
<b>Taglist</b>			
ID	DiscoveryTime	Count	Antenna
A5A5 8005 5600 3242	03:01:01AM	1	3
A5A5 8005 5600 1005	03:01:02AM	1	3
A5A5 8005 5600 1000	03:01:01AM	1	3
A5A5 8005 5600 2345	03:01:01AM	1	3
A5A5 8005 5600 6467	03:01:01AM	1	3
A5A5 8005 5600 0819	03:01:01AM	1	3
A5A5 8005 5600 3409	03:01:01AM	1	3
A5A5 8005 5600 2234	03:01:01AM	1	3

沒有ASN記錄之ID	
ID	A5A5 8005 5600 2236

Tag ID 詳細資料			
ID	進貨日期	產品名	讀取狀態
A5A5 8005 5600 3242	2005/5/20 上午 08:12:10	家樂氏玉米片	已讀取
A5A5 8005 5600 1005	2005/5/20 上午 08:12:10	家樂氏玉米片	已讀取
A5A5 8005 5600 1000	2005/5/20 上午 08:12:10	家樂氏玉米片	已讀取
A5A5 8005 5600 2345	2005/5/20 上午 08:12:10	吉列刀片	已讀取
A5A5 8005 5600 6467	2005/5/20 上午 08:12:10	吉列刀片	已讀取
A5A5 8005 5600 0819	2005/5/20 上午 08:12:10	吉列刀片	已讀取
A5A5 8005 5600 3409	2005/5/20 上午 08:12:10	吉列刀片	已讀取
A5A5 8005 5600 2234	2005/5/20 上午 08:12:10	金頂電池2入	已讀取
A5A5 8005 5600 9053	2005/5/20 上午 08:12:10	金頂電池2入	已讀取
A5A5 8005 5600 1007	2005/5/20 上午 08:12:10	金頂電池2入	已讀取

儲存      回主表單

圖 30、進貨表單

資料來源：本研究整理

### 5.4.2 出貨表單

出貨表單作業與進貨表單作業功能大致上相同，出貨如同進貨之扣帳，主要之別為當 Reader 讀取到 Tag 時，與出貨通知作比對。若資料庫內有該筆，則從資料庫中擷取出資料，並將庫存量減掉。若資料庫無該筆資料，表示該筆資料有異

常，系統會發出一警示訊息，物流中心相關人員再行處理。

另外設計「資料維護表單」與「查詢表單」(圖 31、圖 32)，供資訊部門與倉儲人員作進一步資料管理。

### 5.4.3 查詢表單

查詢表單設計可由欄位名稱和運算符號二大屬性來搜尋想要的資料。將二個屬性設定完畢後，按下「查詢」，即可找出該筆資料;若要該筆資料，則跳出訊息方塊「查無該筆資料」。「回復資料」鍵是設計完成查詢後，回復成資料庫原始之資料。

The screenshot displays the search form interface. At the top, there are two dropdown menus: '欄位名稱' (Field Name) and '運算符號' (Operator). The '欄位名稱' dropdown is open, showing a list of fields: ID, 庫別, 儲位, 驗收單號, 驗收序號, 儲位存放序號, 客戶編號, and DC品號. The '運算符號' dropdown is also open, showing options: =, <>, >, <, ^, and LIKE. Below these are three buttons: '查詢' (Search), '回復資料' (Restore Data), and '回主表單' (Return to Main Form). At the bottom, there is a data table with the following columns: ID, 庫別, 儲位, 驗收單號, 驗收序號, and 儲位. The table contains 15 rows of data.

ID	庫別	儲位	驗收單號	驗收序號	儲位
A5A5 8005 5606 4133	CTOD	DB071-1B	10029	11	0
A5A5 8005 5606 4134	CTOD	DC027-1A	10029	12	0
A5A5 8005 5606 4135	CTOD	DC028-1A	10029	13	0
A5A5 8005 5606 4136	CTOD	DC051-1A	10029	14	0
A5A5 8005 5606 4137	CTOD	DC045-1B	10029	15	0
A5A5 8005 5606 4138	CTOD	DC039-1A	10029	16	0
A5A5 8005 5606 4139	CTOD	DB023-1A	10029	17	0
A5A5 8005 5606 4140	CTOD	DC011-1A	10029	18	0
A5A5 8005 5606 4141	CTOD	DC024-1A	10029	19	0
A5A5 8005 5606 4142	CTOD	DC076-1A	10029	20	0
A5A5 8005 5606 4143	CTOD	DC054-1A	10029	21	0
A5A5 8005 5606 4144	CTOD	DC087-1A	10029	22	0

圖 31、查詢表單

資料來源：本研究整理

### 5.4.4 資料維護表單

除了資料移動之外，另可在資料表上新增與刪除資料，作資料之修改等功能。

ID	庫別	儲位	驗收單號	驗收序號
A5A5 8005 5606 4133	CTOD	DB071-1B	10029	11
A5A5 8005 5606 4134	CTOD	DC027-1A	10029	12
A5A5 8005 5606 4135	CTOD	DC028-1A	10029	13
A5A5 8005 5606 4136	CTOD	DC051-1A	10029	14
A5A5 8005 5606 4137	CTOD	DC045-1B	10029	15
A5A5 8005 5606 4138	CTOD	DC039-1A	10029	16
A5A5 8005 5606 4139	CTOD	DB023-1A	10029	17
A5A5 8005 5606 4140	CTOD	DC011-1A	10029	18
A5A5 8005 5606 4141	CTOD	DC024-1A	10029	19
A5A5 8005 5606 4142	CTOD	DC076-1A	10029	20

第一筆 上一筆 下一筆 最後一筆 回主表單  
資料修改 取消修改 新增資料 刪除資料

圖 32、資料維護表單

資料來源：本研究整理

## 5.5 功能模擬測試

本研究的目的是在於成功串連 Reader 與前端資訊系統，亦即 RFID 設備能夠透過本研究設計出來的應用程式介面來與資料庫作溝通。因此測試首要目的必須確認應用程式介面確實能扮演 Reader 與資料庫傳遞的橋樑，而經測試後，本研究應用程式介面確實能正常運作完成。

在協助個案公司於初期 RFID 導入時，除了解中介軟體所運作的功能，亦發現另外一項值得個案公司思考的問題：導入 RFID 時，激增之資料量對於資料庫存取時間的影響。因此延伸出本節所設計之模擬實驗。

### 5.5.1 測試目的

物流中心的貨物量是很龐大的，以個案公司而言，其貨物量高達 8,000 多種，未來若導入 RFID Tag，其資料量相當驚人。RFID 有自動資料擷取之特色，可加速貨物處理及節省人工處理資料的動作，因而在物流業之發展性相當被倚重。而在資訊大量且快速的被處理時，資料是否能即時傳遞，是物流中心業者值得關切的問題。因此本研究欲探討當資料量經由 Reader 接收，透過本研究設計之程式應用介面，隨著資料量的增加，傳送至資料庫作存取的時間之影響程度，且是否為個案公司之資訊人員所能接受之處理時間，以不致於影響後續物流作業。在這實驗當中，希望能找出本研究所開發之資訊處理系統所能夠處理資料量的臨界值，再

行改善。

### 5.5.2 測試議題

由 3.4 節的分析，現階段個案公司導入 RFID 以「Case」較適合個案公司 RFID 作業模式。以目前個案公司之作業模式，現行以條碼為倉儲管理時，條碼號可能代表同一效期的產品，若 50 箱為同一效期，即以一個條碼號代表之，在資料庫中僅需一個 record 記錄即可。但 RFID Tag 希望作到唯一性，即一個 Tag ID 代表一箱產品，此時資料量將爆增至 50 個 records。根據個案公司的訪談，若導入 RFID，以目前倉庫之貨物量約為 26,000 個儲位，每個儲位有三個棧板，一個棧板平均約 40 箱來估算的話，資料量將增加至 3,120,000，為原本的 40 倍之多。

另外，過高的投資成本也是令廠商為之卻步的主因。在成本居高不下和讀取率尚未達 100%之現況下，本研究建議導入 RFID 應循序漸進，亦即採階段性導入，待技術和成本成熟時，再全盤導入。依目前個案公司庫存量來分為 2%、5%、10%、20%、40%、80%和 100%導入程度，來測試資料庫存取時間多寡。在出貨時候，由於必須在資料庫內搜尋該筆資料，在龐大的資料庫中如果要花費許久時間，則對於現行作業反而會造成負面的影響。因此在導入 RFID 之時，在出貨作業會因資料量的擴張，對於資料存取時間影響較大，而這也是個案公司必須面對的問題。有鑑於此，本研究欲透過模擬實驗，觀察巨額的資料量對於資料擷取的影響程度。

### 5.5.3 測試方法

如圖 33 所示，假設 Tag 與 Reader 資訊讀寫的時間，加上 Reader 與 API 資訊的交換時間為 T1，API 與資料庫資訊的交換時間 T2。當  $T1 \geq T2$ ，則能達到 Tag 資訊上傳與資料庫更新之即時性。

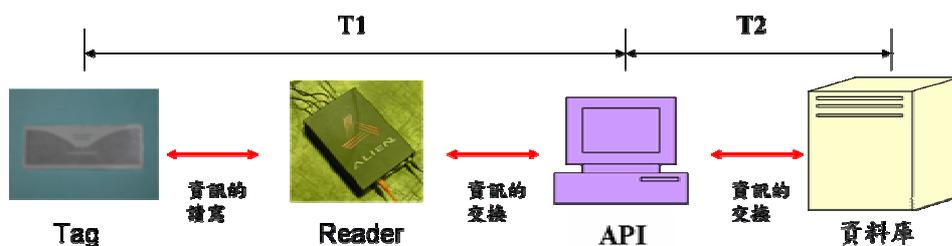


圖 33、T1 與 T2 示意圖

資料來源：本研究整理

實驗假設與條件：

1. API 每隔一秒詢問 Reader 一次，即  $T1=1$  秒，並將其所讀取到之 Tag 資訊上傳至 Taglist 資料表。
2. 以「Case」為基本單位貼附 RFID Tag，平均一個棧板約為 40 個 Tags。
3. 進貨通知與出貨通知皆以 100 筆資料來作測試。
4. 測試設備：
  - AMD Athlon XP 1.84GHz CPU
  - 512MB RAM

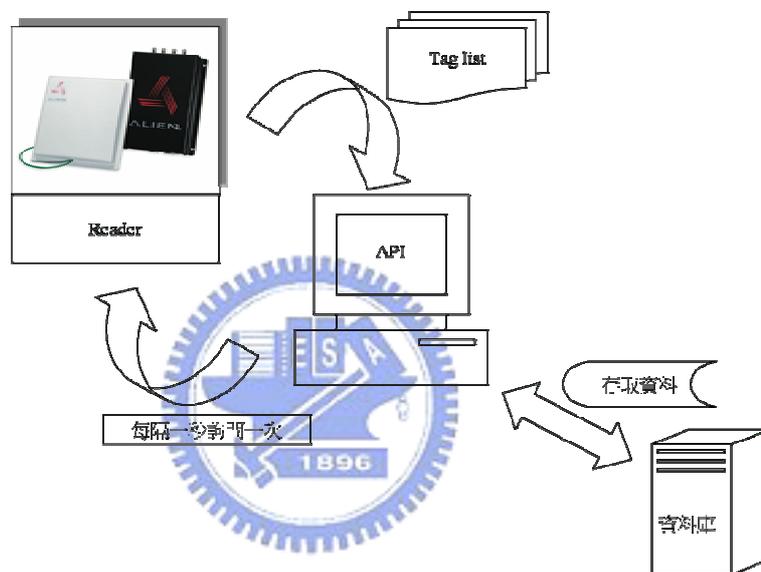


圖 34、資料處理示意圖

資料來源：本研究整理

Reader 所讀取的 Tag 會先暫存於 Taglist 中。API 每隔一秒詢問 Reader 一次，並將其暫存的 Taglist，透過 API 作過濾的動作，使 Tag ID 在資料庫中成為唯一性。確保資料無誤後，再將資料存取於資料庫。而本測試主要所量測的時間為「存取資料」所花費的時間，即  $T2$ 。

### 5.5.4 測試結果

測量得到的實驗數據如表 14。

表 14、實驗數據

資料庫筆數(千筆)	貨物導入 RFID 之百分比	進貨 記錄新增時間(秒)	出貨 記錄刪除時間(秒)
62.4	2%	0.68	0.82
156	5%	0.66	1.50
312	10%	0.72	5.19
624	20%	0.68	11.59
1248	40%	0.77	-
2496	80%	0.71	-
3120	100%	0.80	-

資料來源：本研究整理

出貨作業當測量到 20%，其記錄刪除時間已高達 11.59 秒，因此停止實驗。將表 14 製成圖 35 如下。

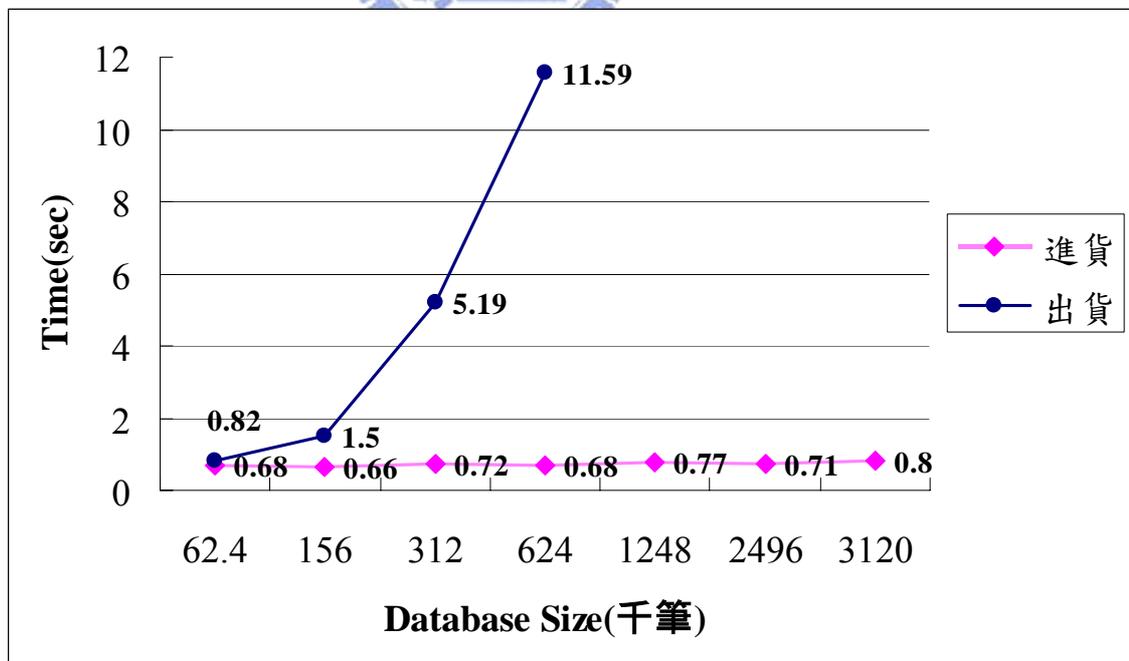


圖 35、Database Size 與更新資料庫時間之數據分析

資料來源：本研究整理

理論上而言，資料存取的速度要比 Taglist 上傳的速度快，否則會造成資料遺失，已讀取的資料未實際寫入資料庫。本研究假設 Taglist 上傳的速度為一秒鐘，即 API 向 Reader 詢問的時間。因此若資料庫存取的速度超過一秒，則可能造成部分 Tag 資料遺漏。

而由測試結果來看，由圖 35 可知，進貨作業寫入資料庫的時間皆在一秒內，意即每筆 Tag 皆能在取得下一次 Taglist 前完成儲存動作。而出貨部分，當資料庫大小約為 6 萬筆，佔總貨物量 2% 時，時間約為 0.82 秒，尚未超過一秒；但當資料量達全倉貨物量的 5%，即 156,000 筆時，則資料存取時間就超過一秒鐘。而當資料庫越大，則存取時間隨之增長。由實驗結果顯示，資料庫的大小確實會對資料存取時間造成影響，尤其是在出貨作業。因此在 RFID 加快物流中心進貨和出貨作業的同時，資料庫是否能即時且正確的完成加減帳，並維持資料庫的精準性，是未來無論是中介軟體，或是 RFID 資訊系統開發時，值得關切的議題。

由於在出貨作業測試時， $T_2 > 1$  秒，造成出貨作業與記錄刪除不能即時的問題，建議選擇在更新資料庫時前，先行建立與 ASN 作比對之緩衝機制，將  $T_2$  時間縮短至一秒內，即可解決即時之問題。雖然資料庫無法即時更新，會存在資料不是最新狀態的問題。但此一緩衝機制可以解決資料存取即時性問題，待出貨完成後再另行作資料庫更新，避免出貨作業受到直接影響，而喪失 RFID 即時的效果。

既然在初期無法將全部貨物導入 RFID，且礙於 Tag 過高的成本，本研究建議在初期個案公司可先選取合適且高價的產品來作起，降低單位成本，且對於高價貨物作管理之實質意義也較大。因此在採取階段性導入的計畫下，初期以高價產品來作為導入之對象效益會較明顯。