SAP 學習心得筆記

為何世界五百強企業多數採用？為何能獨霸 ERP 產業？

Author：Michael

Start Date：2019-09-24

前言

自從 2019-07-10 到三商家購報到後，就開始了瘋狂的學習與解 bug 之旅。之前的職場生涯，大都是圍繞著大大小小的 ERP 系統。只不過在後期階段，更偏向透徹的理解並提出開發 ERP 系統的所有方面的解決方案，更多是致力在開發工具、框架及商用邏輯 Store Procedure 化這 3 大面向。但是在家購，IT 是以委外為核心，所以我就試圖從頭完整的學習 SAP 的核心功能，重新以一個老兵菜鳥的心態，來搞清楚，SAP 到底是憑什麼可以有今天的地位，更讓我想了解的是，他憑什麼賣的這麼貴?

以下這系列的文章，主要是想分為下面幾個單元來談，這僅是初稿，後續會再調整。特別強調：這是一個 SAP 超級菜鳥寫的入門心得，主要面對的讀者，是已有 ERP(或 MIS) 系統經驗的相關從業人員，如果您完全沒有相關經驗，部分內容可能會不知所云，請慎入。

如何快速學習 SAP ERP?

1. 基本資料建檔(有哪些?)，如客戶、廠商、產品、人員及會計科目等
2. 參數設定(分為上線前要設好及上線後慢慢設定)，如公司聯絡資訊、單據編碼設定等
3. 開帳作業，有哪些期初值? 如何輸入期初開帳資料(要分為各系統)，如期初庫存、期初應收、應付、會計
4. 日常作業(這是企業藍圖規範的內容)
5. 每一個模組都有其核心的關鍵因素，以進銷存而言，就是四大天王，數量、成本、帳款(AP/AR) 及績效利潤
   以總帳會計而言，就是三大報表，資產負債表、損益表、現金流量表。
   只要能真的理解這些核心程式，其他再多的程式也都只是錦上添花而已。

是否有可能從這個觀點，來快速學習，並掌握 SAP 的核心功能，再順勢往外擴充?
甚至擴充到 surroundings(外圍系統) 的資料整合? 中台系統?

我先在此處隨手記下相關的學習心得，如果有可能，當作是 ERP 開發聖經的補充內容。

有一個提醒要先說在前頭，這是我繞了一點圈後的體會，Google 大神的確是無所不知，連 SAP 很多的細節都可以搜的到，但是，我還是會建議初學者要謹守本心，善用減法，不要想短期內就搞懂全部的 SAP，我原本就搞過很多年的 ERP，甚至可以獨力開發完成一套小型的 ERP 系統，所以我一開始並不認為 SAP 會多難搞，了不起就是多花點時間而已。但是，當我因為貪多嚼不爛的想要盡可能多了解細節，發現我不斷的被一篇又一篇的好文、數不盡的細節包圍，雖然對 SAP 是有點了解，但是卻發現，怎麼還有很多基本的功能沒搞清楚，對單一模組的理解並不全面。所以，有些功能就先 PASS，不要急，我的建議還是先繞著核心功能先了解，再往外擴。如果你沒有相關的經驗，那建議你跟著本文理解；如果你有自己的一套觀念，就順著你原先的觀念來學。我想這應該是較有效率的學習方法。

Table of Contents

前言

第一章 SAP 是什麼? 及各模組的簡要說明

a. ERP 是什麼? SAP 又是甚麼?
b. 各模組整合圖說明
c. Client – Company – Plant – Storage Location – 等等
d. ABAP 的重要性
e. 可自行定義每個欄位的屬性，顯示/不顯示/Enable/Disable/是否必輸入

第二章 SPRO 參數設定作業

第三章 MM/SD 的基本資料設定

第四章 SAP 基本操作及共用資料維護

a. 先整理出最基礎、最核心的 T-Code 列表
   同時記錄相關的 Table Schema
b. 基本操作說明(分為單檔及表單)
   整理幾支代表性 T-Code 的通用性基本操作，尤其是 Drill-Down 功能
c. Table 架構及命名規則
d. 上線前的準備工作-企業藍圖

第五章 MM 一些重要的單據流程及概念說明

第六章 完整的採購流程說明

第七章 完整的銷售流程說明

第八章 關於期初開帳

第九章 SAP 成本計算有哪些方式?

第 10 章 ABAP 入門基本要點

第 11 章 SAP Open SQL

第 12 章 用 SM30 撰寫單檔維護程式

第 13 章 SAP 的自定義報表工具

第 14 章 LSMW 批次處理使用方法

第 15 章 BDC 批处理使用方法

第 16 章 SAP 工作排程

附錄

a. 常用 T-Code
b. SAP 常用 Table 列表
c. 參考 SAP 相關網站列表

第一章
SAP 是什麼?及各模組的簡要說明

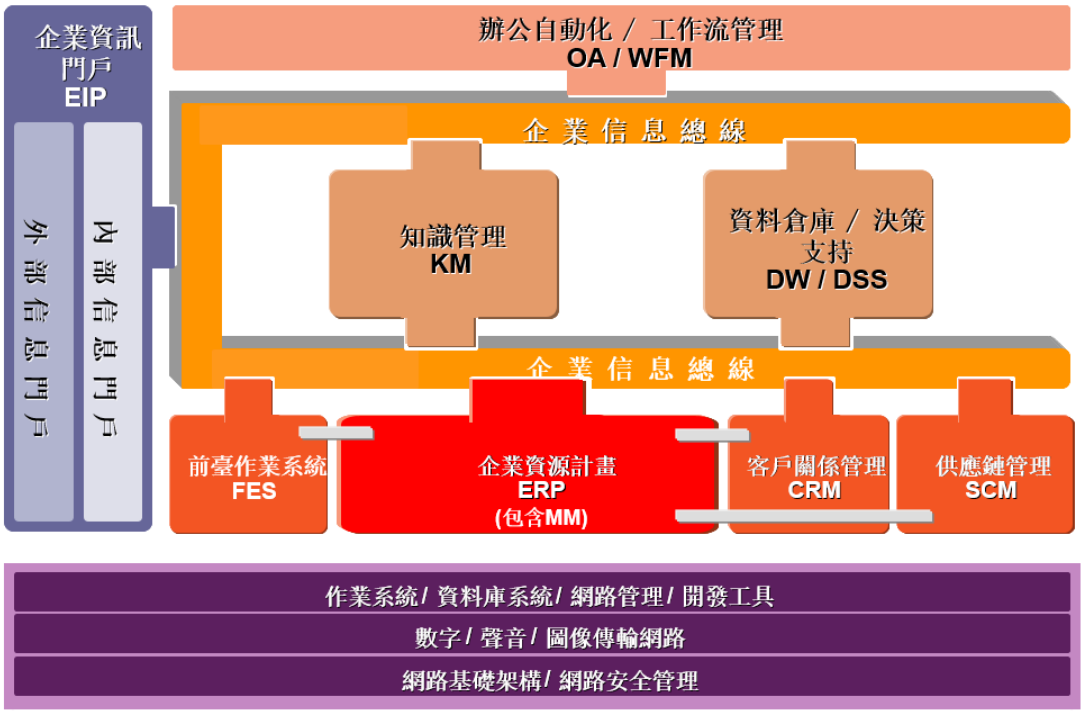
人生若只如初見，何事秋風悲畫扇。
等閒變卻故人心，卻道故人心易變。
驪山語罷清宵半，淚雨零鈴終不怨。
何如薄幸錦衣郎，比翼連枝當日願。

清 • 納蘭容若

第一章：SAP 是什麼? 及各模組的簡要說明

➢ ERP 是什麼? SAP 又是甚麼?

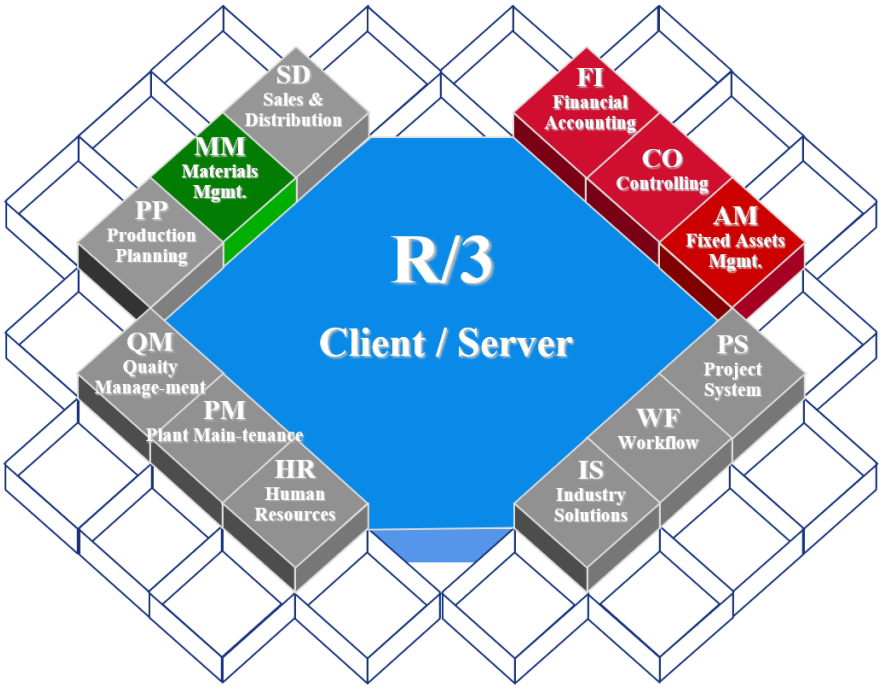
ERP系統是企業資訊系統的核心部分



➢ SAP 是什麼?

System, Application and Products (*系統，應用和產品)*

SAP中R/3標準模組示意圖



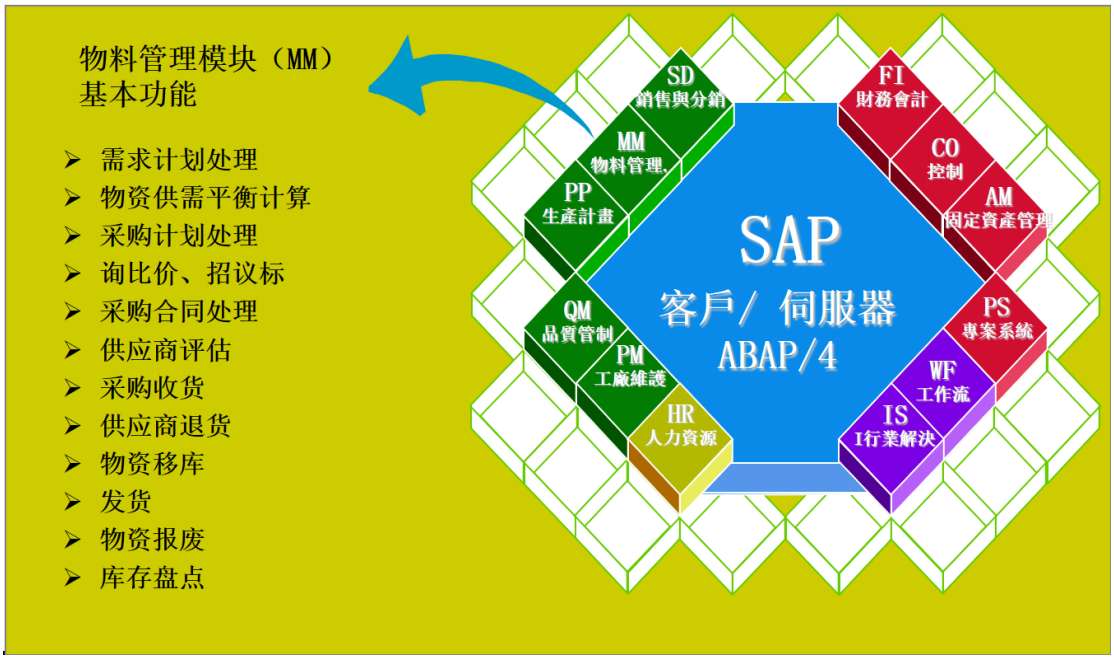
- SAP 的版本推出的順序

  SAP R/1 → SAP R/2 → SAP R/3 → SAP ECC → SAP Business Suite on HANA → SAP S/4 HANA → SAPS/4 HANA Cloud

  SAP 各版本說明

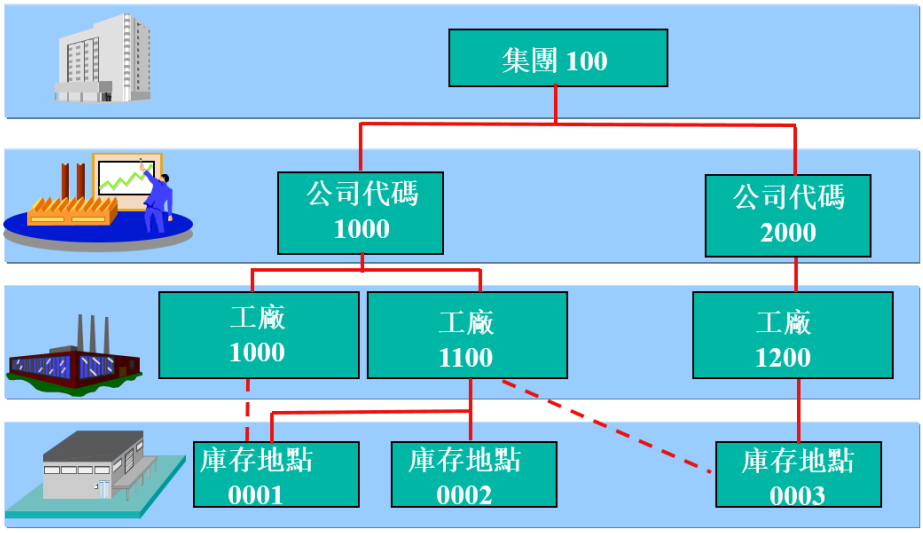
  - ✓ 基於 SAP Main frame 的 ERP 系統稱為 SAP R/2。
  - ✓ 在 1997 年，當 SAP 轉換到 client server 架構時，它被稱為 SAP R/3。也被稱為 MySAP business suite。
  - ✓ 隨後 SAP 推出了 6.0 的新版本，並將其更名為 ECC (ERP Core Component)。
  - ✓ 在 6.0 之後，SAP 推出了 HANA 數據庫，並將其 ERP 產品遷移到了 HANA。然後 ECC（ERP 產品）被重命名為 S/4 HANA。
  - ✓ 如果不想使用 S/4 HANA 但仍想使用 HANA 的功能或提高效能，則可以選擇 SAP Business Suite on HANA，這是 S/4 HANA 和 ECC 6 EHP7 之間的版本。
  - ✓ S/4 HANA 也可以在雲上使用，它被稱為 S/4 HANA cloud，它與 on premise 幾乎相同，但有一些限制 (Cloud)。

- **各模組整合圖說明**



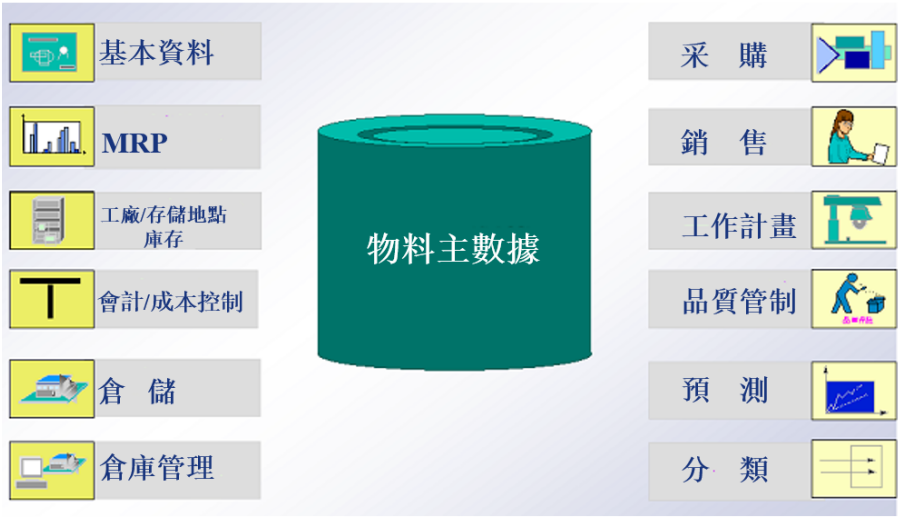
➢ Client – Company – Plant – Storage Location – 等等

庫存管理系統組織結構圖一例圖:



補充說明：在 Retail 行業中，並無 Plant 這一層，我原先以為分店(Store or Site) 的資料會建立在 Plant 中，但是後來看 NTT 文檔，發現他們是用 Site(據點 T-Code=WB01) 建立分店資料。也就是說 SAP 似乎可以依行業別，連底層架構都可以變更。此處等日後搞清楚所有架構後會再補充說明。

物料主資料一視圖:



SAP 的資料庫結構非常龐大且複雜，主要模組的資料表超過 12,000 個以上。以上面的物料主檔資料為例，每一個 View (頁籤、視圖) 都是一個(或以上)的 Table。因為實在太過龐雜，所以一般的顧問也只會熟悉主要、常用的一些 Table。其他要多利用 F1 查詢功能。又因為 SAP 設計嚴謹，基本上不應該也不允許用任何非正規的方式試圖直接異動資料庫的內容，顧問會非常嚴肅、近乎恐嚇的方式警告，如果你自己硬繞過 SAP 機制，其結果自負，SAP 完全不負相關責任。

我個人還是持一貫的立場，先熟悉核心再往外擴充。也就是先知道 MARA 是物料主檔即可，其他相關的 Table 就隨著實際的使用經驗逐步累積。所以學習 SAP 不能過於急躁，是需要一點時間的累積。

- **ABAP 的重要性**

SAP S4HANA 當然是一套非常龐大、嚴謹且完整的 ERP 系統。但是面對世界上這麼多樣的行業別及公司，也不可能可以完全適用於每一個行業、每一家公司，所以二次的開發就無法避免。**通常簡單一點的管理系統，都會內含一個開發工具，讓客戶可以依自己的需求擴充，但是內建的二次開發平臺通常都是開發一些報表，如果是功能表單，就只能以純人工的方式開發。但是 SAP 的 ABAP 可不止於此，除了報表開發還可以進行動態程式開發。甚至 SAP R/3 整套系統基本上就是用 ABAP 開發，所以它本身就構成一個完整的開發生態，不依賴於 OS 及其他廠商的開發工具(但同時支援 Java 平台)。**

如果用慣了新世界的 IDE 開發工具，初期接觸 SAP 的工作平台會不太適應，因為功能實在太陽春了，基本上就只提供 IDE 的最基本功能，畢竟它是 SAP 專用的開發平台及語言，易用性不是它的主要考量。

ABAP 是德語 *Allgemeiner Berichtsaufbereitungsprozessor* 的縮寫，意思是「通用報表預處理器」。ABAP 第一次引入了「邏輯資料庫」的概念，它在基本的資料庫層提供了更高級的抽象。

ABAP 程式語言最初被 SAP 的開發者用於開發 SAP R/3 平台。但它也被設計讓 SAP 的客戶用於增強 SAP 的軟體應用–客戶可以用 ABAP 編程開發自定義的報表和界面。這個程式語言對於程式設計師來說很容易學習但並不是一個非程序設計人員可以直接使用的工具。編寫 ABAP 程序需要良好的編程技巧和關聯式資料庫方面的知識，如果知道物件導向設計的概念更好。

雖然 SAP 最早於 1992 年就發布了 R/3，但 ABAP 現在仍可以用於為 R/3 系統編寫程序。在 20 世紀 90 年代，隨著計算機硬體的發展，越來越多的 SAP 的應用軟體和系統都用 ABAP 來實現。一直到 2001 年，幾乎所有的基本功能都是由 ABAP 編程實現的。SAP 最新的開發平台 NetWeaver 同時支持 ABAP 和 Java。

https://zh.wikipedia.org/wiki/ABAP

如果想要成為一名稱職的 SAP 顧問師，ABAP 的基本開發能力應該還是必要的基礎，而想要寫 ABAP 程式當然要先了解檔案結構及相關的語法(Open SQL)，後面會有 2 個章節對基本的 ABAP 及 Open SQL 做一個入門級的介紹，現在先有基本的概念即可。當然 ABAP 並不是 SAP 顧問必須具備的技能，顧問更多是對企業流程的了解及 SAP 功能的靈活運用，只不過如果想要真正的深入 SAP 的世界，個人還是認為 ABAP 是繞不過的坎。

Table 架構及命名規則(保留)

https://help.sap.com/viewer/0ebb238af58d418dbda5fcd2d8507bf1/1908/zh-CN/7d163b5efd3e4433b41c6238836a13b0.html

https://help.sap.com/viewer/0ebb238af58d418dbda5fcd2d8507bf1/1908/zh-CN/3e950ec964f741bead7c74fae77a7f51.html

- **上線前的準備工作-企業藍圖**

https://www.sap.com/taiwan/products/roadmaps.html
https://wiki.mbalib.com/zh-tw/%E4%B8%9A%E5%8A%A1%E8%93%9D%E5%9B%BE

業務藍圖是國際 ERP 界通用的一種企業建模方法。業務藍圖用一種企業可以理解的方式來說明複雜的過程，它是一種清晰而又簡單的描述方法，只用少量不同的符號，以集合的方式進行組織，定義了什麼人必須在什麼時候，採用什麼方法去做什麼事情，這種描述方法使得非專業人員也可以理解複雜的業務過程。

業務藍圖設計是通過對公司業務現狀的調研分析，梳理出業務未來的信息流、物流和業務流的處理模型。業務藍圖階段是 ERP 實施過程中關鍵的環節，藍圖定義的過程就是企業將自身業務流程轉化為新系統業務流程的過程。在這個轉化過程中，勢必會遇到很大的阻力，甚至會在此階段出現相當大的衝突。因此，顧問必須給予這個階段相當的重視，採用適當的策略，保證項目的總體目標。同時顧問也應該清楚地知道，儘管這個問題非常棘手，但決不能繞開。如果這個階段的工作做的不扎實會造成項目目標不明，導制最終項目實施結果無法確認，項目無法驗收。

這個階段顧問方要通過標準培訓使客戶對標準業務流程有進一步深入的理解和認識，並指導他們對標準規程進行學習和研究，在此基礎之上，指導他們參照系統整理出自己的現行業務工作流程。然後，指導客戶在對照經典流程基礎上，對企業現行業務流程進行分析和整理，制訂出企業新系統的業務藍圖草案。通過引導客戶對現行的業務及其流程的整理，一方面可以理清客戶自身的管理流程。另一方面，可以使顧問獲得業務規範化調整的基礎信息和軟體客戶化需求的初步信息，為規範企業業務管理，使軟體標準化、規範化以及為下一步藍圖的實施打下堅實、可靠的基礎。

第二章 SPRO 常用參數設定作業

關於參數的說明：

一套 ERP 系統想要解決所有不同行業別的需求是不可能的。即使是 SAP、Oracle 也不可能做到，因為企業邏輯雖有基本原則，但是企業是由人所組成的、是活的生命體，管理的表單及流程完全存乎一心→簡在帝心。所以在開發 ERP 系統時，只能盡量的剖析、歸納、然後取最大公約數為標準版本。

聽起來很合理，但如果碰到較特殊、複雜的客戶業別時，無可避免會造成大量的二次開發工作，會嚴重的影響時程及成本。而且有經驗的系統整合商都會同意，過多的客製開發，絕對是專案殺手。所以對開發商而言，會根據長期累積的輔導經驗，試著降低二次開發的數量及範圍。

想解決這個問題，一般有 2 個方法

1. 盡可能將程式參數化
   A. 開發程式時，盡量將可能的變數(如稅率、單號編碼原則、成本方式、盤點方式等等)，統一設定在某幾個參數資料表中，只要變更參數，就可以隨時變更相關的邏輯。本章內容主要是著重在此，當然，SAP 的參數龐大、複雜且範圍更廣，一開始只要就基本的重要參數開始了解，最後再逐步深入擴展理解即可。
   B. 除了變數，甚至可以透過定義部分單據類型參數，達到單一表單多用途、多流程、多過帳的目的。這個要等到對系統有一點理解後才能體會(並不是只有 SAP 才有，一般 ERP 系統也有類似概念)。簡單的說，以 SAP 的採購流程來說，採購文件(P/O)中的單據類型，就基本上可以分為採購單/退貨單/調撥單，這 3 個主要基本流程，而如果是調撥單，後面的流程會跑 調撥發貨→調撥交貨→收貨，而標準的採購單，後面流程就是驗收收貨。這類的類型定義的越多，就可以用同一組表單程式，解決各種千奇百怪的流程。這就會盡可能的降低客製的需求。
2. 開放自定義畫面及檢核程式
   除了上面的作法，有很多的二次開發需求，是要在現有的表單功能增加欄位，例如要在商品主檔新增幾個進口相關欄位定義、或是磅標資訊、有效期限等。有些新型的開發工具平台，是允許讓 user 透過工具自行增減欄位(自動同步後端資料庫)、改變欄位的元件屬性、檢查某些特定條件等。SAP 也可以做到，不過必須透過 ABAP 的擴充，一般 user 無法自行設定。雖不完美，但也降低了重新開發的困難度。

如果上面的設定或少量程式調整都無法滿足客戶的需求，才需要進行二次開發。雖然無法完全避免，但是至少可以盡可能地降低二次開發的範圍。當然，隨著 Domain Know-how 及實務經驗的累積，也許會有更有效的開發方式(例如我也試著提出以資料庫為開發核心的概念)，但是以上的解法仍是目前的主流。

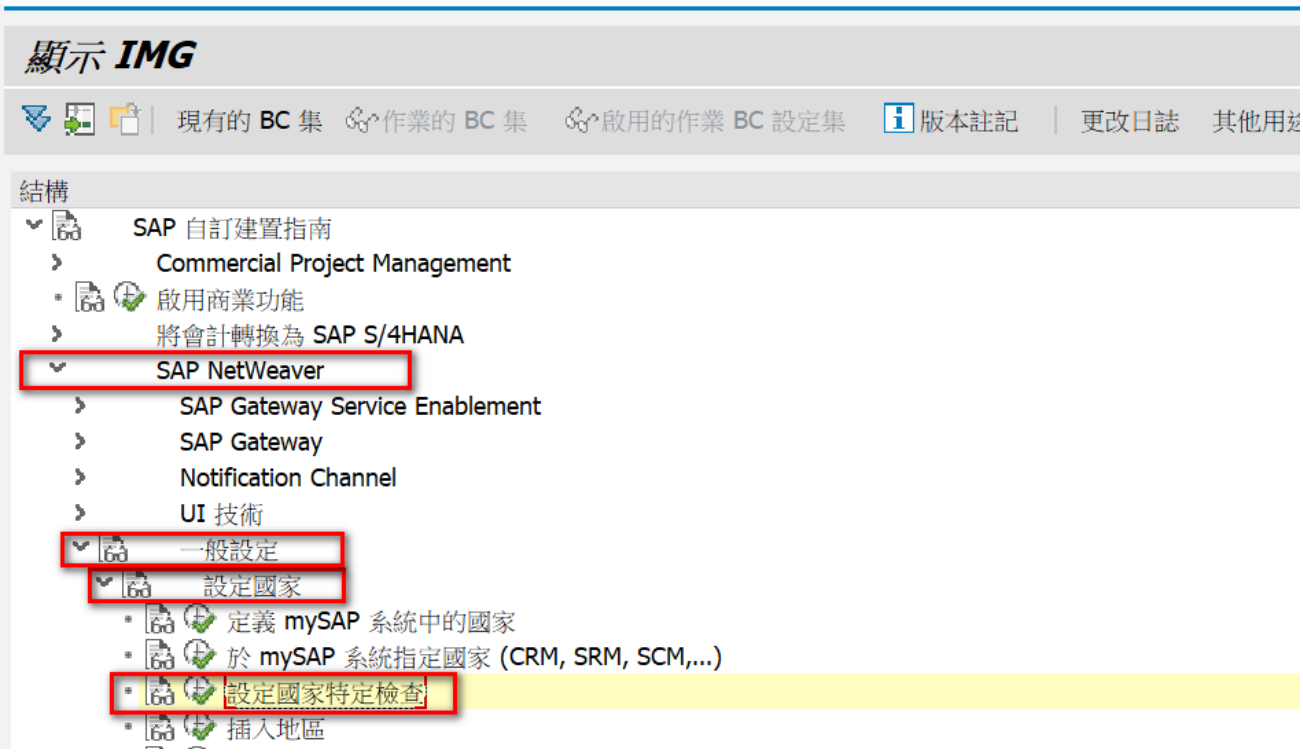
補充說明：由於 SAP 的參數多如牛毛，且繁複異常，所以本文內容僅提到一些最基本的設定，主要的目的是實際的看一下設定的方式與介面。至於詳細的參數設定內容與說明，請參閱顧問提供的參數設定手冊。

T-Code：spro 的常用設定

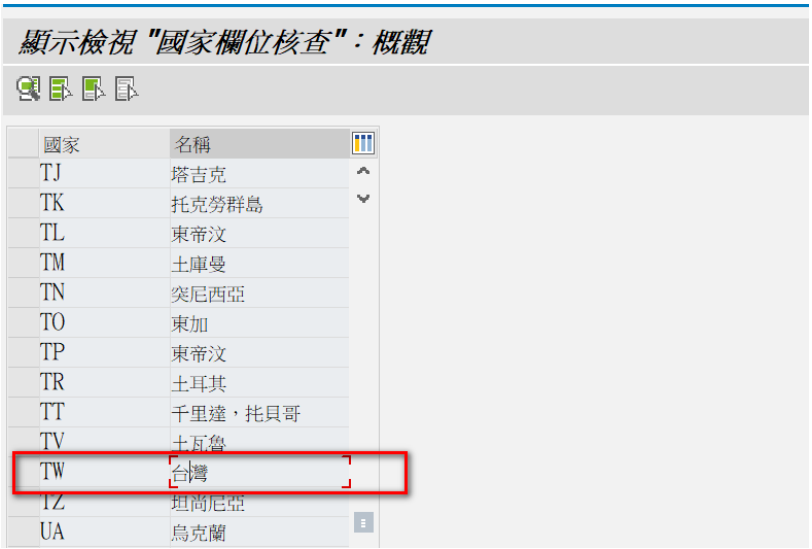
SAP NetWeaver					
	一般設定	設定國家	設定國家特定檢查		
		檢查計量單位			
		維護行事曆			
後勤系統-一般					
	物料主檔	基本設定	物料類型	定義物料類型的屬性	
		基本設定	物料類型	定義各物料類型的號碼範圍	
		欄位選擇	指派欄位至欄位選擇群組		
		設定物料主檔	定義每個畫面順序之資料畫面的結構		
企業結構					
	定義	財務會計	編輯、複製、刪除、檢查公司代碼		
			定義公司		
		後勤系統-一般	定義評價層次		
		物料管理	維護儲存位置		
			維護採購組織		

1. 國家特定檢查

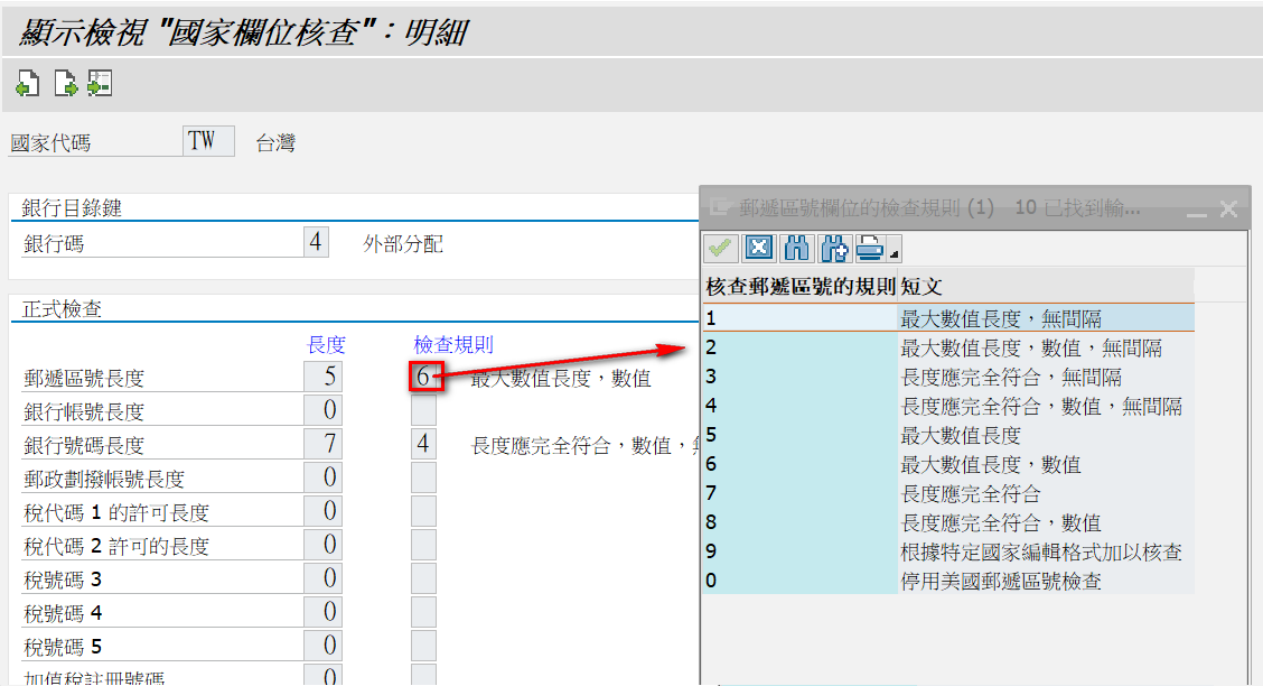
SAP 功能表	IMG→ SAP NetWeaver →一般設定→設定國家→設定國家特定檢查
交易代碼	SPRO
說明	國別的相關設定。



可以依國別設定必要的欄位檢查，如銀行帳號長度、郵遞區號長度、是否允許空白等



基本上通常不會修改，只有當舊資料匯入時，如果舊資料的設定長度，和目前設定不同，會無法匯入，此時才會需要修改此處的設定。



2. 檢查計量單位

SAP 功能表	IMG→ SAP NetWeaver →一般設定→檢查計量單位
交易代碼	SPRO
說明	單位的相關設定。

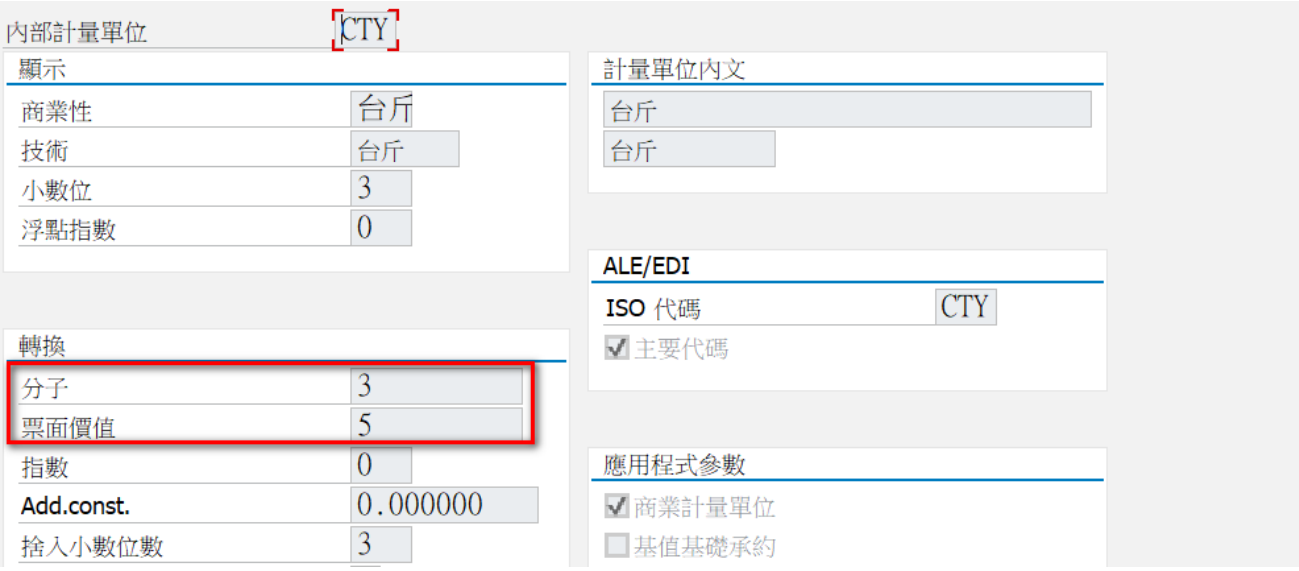
SAP 設定單位，是依維度或象限來設定(如質量代表重量單位)

選擇某一個單位後，可以 DbClick 設定或查詢明細的設定。請注意，在每一個維度的設定中，會有一列資料的顏色與其他不同，可能是紅色或藍色，該筆資料為該維度的計量(最小)單位。

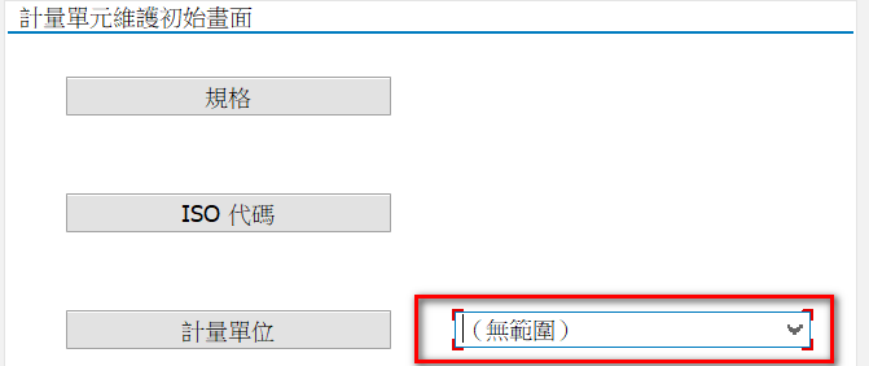
主要的設定有和最小單位的換算及換算後的小數位數設定。以 KG 為計量單位，此時票面價值和分子都是 1。

如果我們查詢 g(克) 的話，票面價值為 1000，表示 1000 g = 1 KG(計量單位) 的意思。

台斤的換算如下



關於無範圍的說明

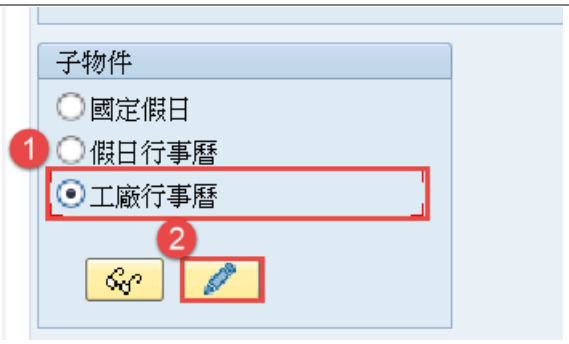


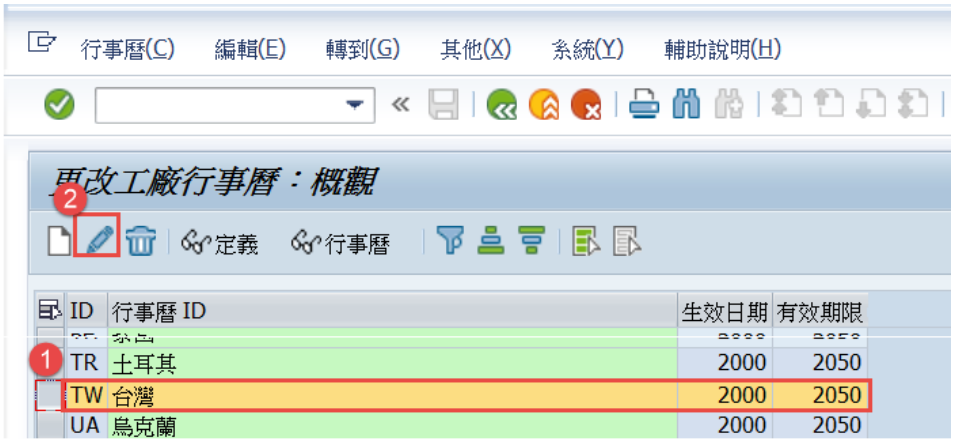
所謂的 [無範圍]，表示裡面的這些單位設定，不屬於任何維度。每個都是獨立的基本單位。

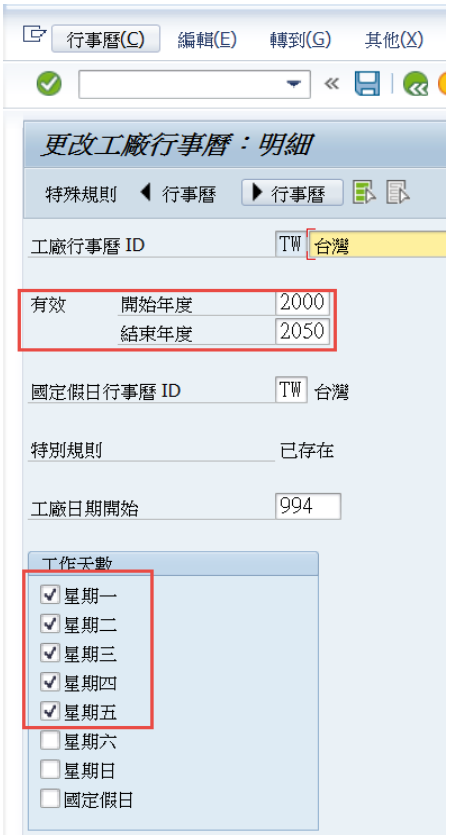
顯示規格（無範圍）的計量單位：概觀

單位	商業性	技術	計量單位內文
BAG	袋	袋	袋
BK	本	本	本
BOT	瓶	瓶	瓶
BWL	碗	碗	碗
C01	支	支	支
C02	入	入	入
C03	片	片	片
C04	付	付	付
C05	份	份	份
C06	次	次	次

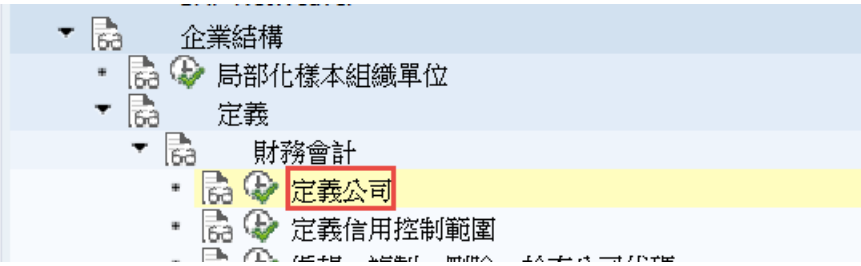
SAP 功能表	IMG→ SAP NetWeaver →一般設定→維護行事曆
交易代碼	SPRO
說明	工廠行事曆的相關設定。



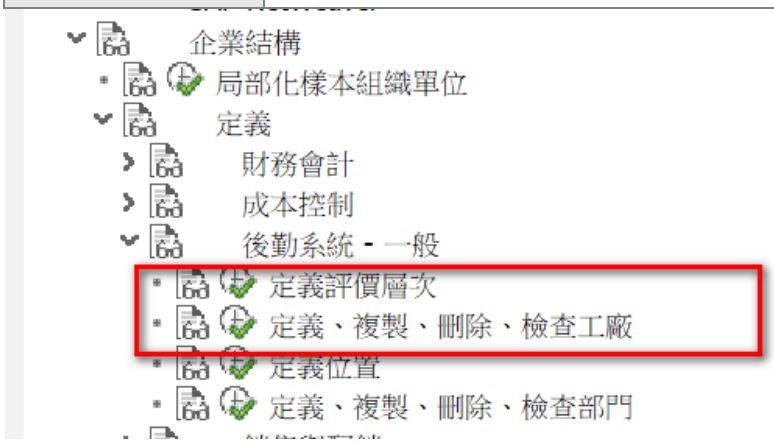




SAP 功能表	IMG→ 企業結構→定義→財務會計→定義公司
交易代碼	SPRO
說明	公司的相關設定。



SAP 功能表	IMG→ 企業結構→定義→財務會計→定義公司
交易代碼	SPRO
說明	**定義評價層次，看起來是設定成本是計算到公司或工廠(預設應該是到工廠)**



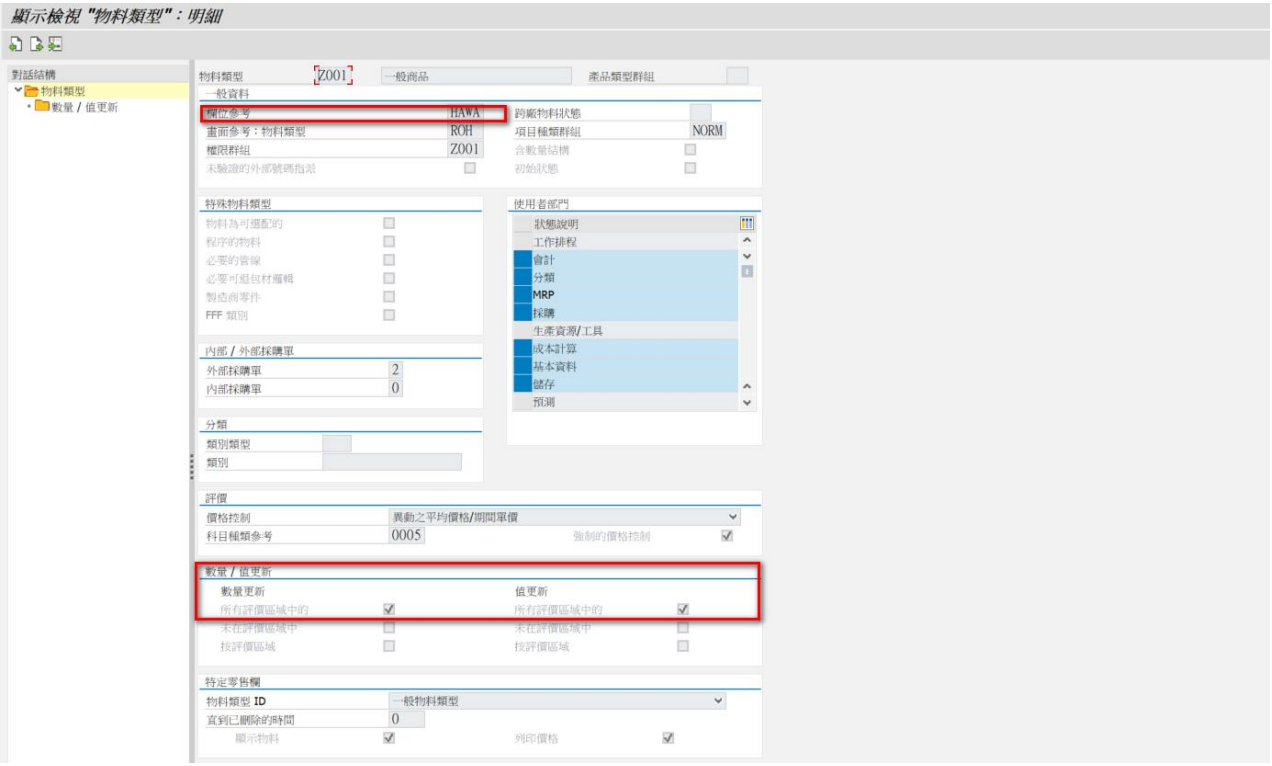
物料類型的定義

在 MM 模組中，物料類型是非常重要的基本設定，影響是否計算數量與成本，也可以根據他來調整物料主檔的 UX。

SAP 功能表	IMG→ **後勤系統-一般**→物料主檔→基本設定→物料類型→定義物料類型的屬性
交易代碼	SPRO
說明	定義物料類型的屬性

Z001 是由標準的 HAWA 複製修改而成，基本上我們不會去動原本的 SAP 設定，都是複製修改為一個新的代號，在 SAP 基本上都是這樣處理的，自行新增的類別，基本上都是以 Z 開頭來表示(三商複製新增 Z001-Z006)，這也是 SAP 的使用慣例。

欄位參考，註記本設定是由哪一個原始設定複製修改而來。



價格控制：可以選擇 (V/S) 移動平均或標準成本。

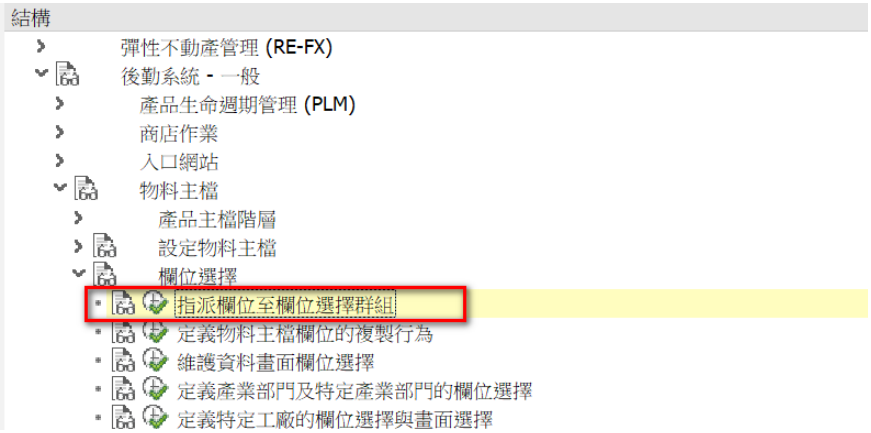
數量更新：是否要管數量，依下方的 (評價區域) 條件勾選，決定哪些要管量(這個部份我現在搞不清楚)。

值更新：是否要管成本，依下方的 (評價區域) 條件勾選，決定哪些要管成本(這個部份我現在搞不清楚)。

再來是 [號碼範圍]，這裡面有一個 [內部給號] 和 [外部給號]，應該是新增時的編碼原則、及外部轉入資料時，後面幾碼的流水號的檢查，必須要符合這個原則才能存檔(我猜的，這個我不是十分有把握)。

SAP 功能表	IMG→ **後勤系統-一般**→物料主檔→基本設定→物料類型→定義物料類型的屬性
交易代碼	SPRO
說明	定義物料類型的屬性

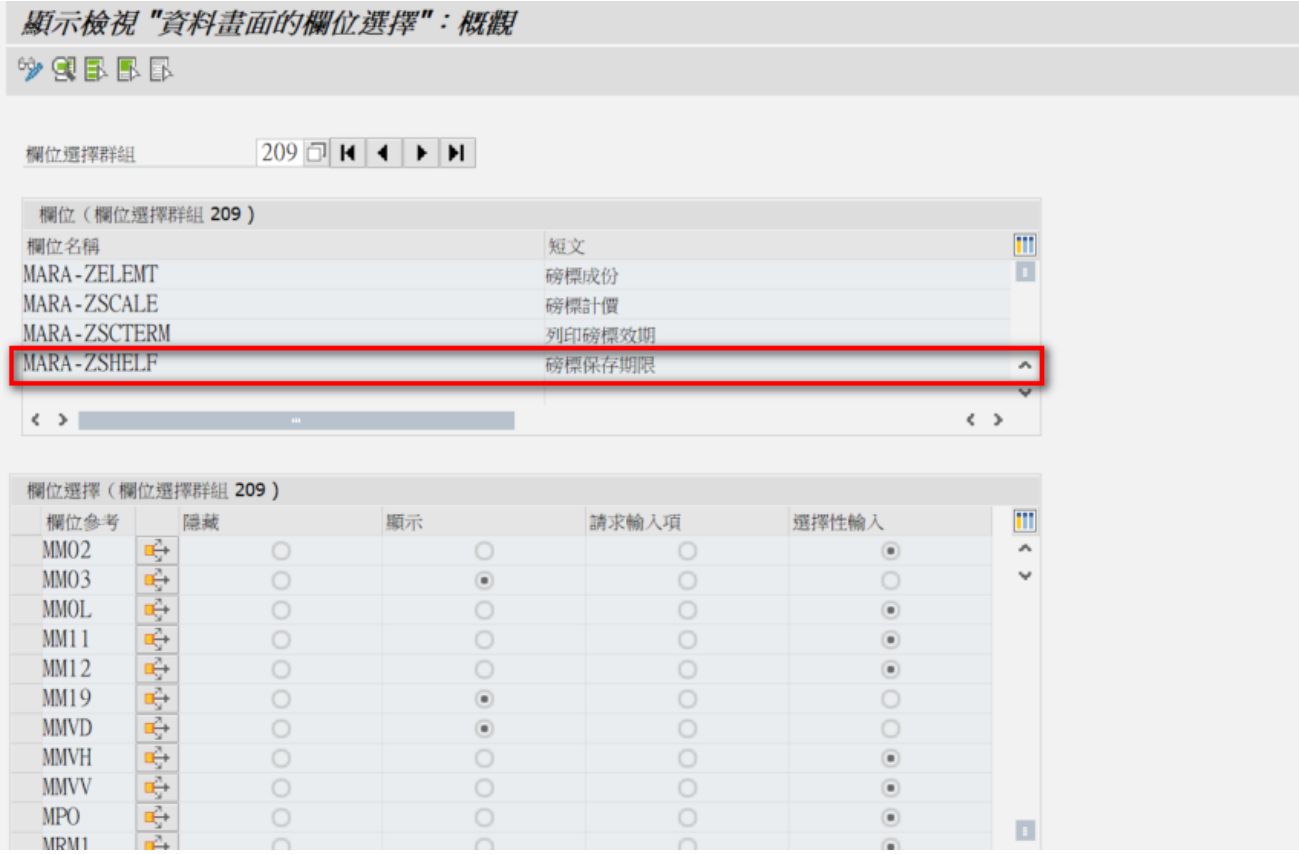
接下來聊聊神奇的 [欄位選擇] 設定功能，這個功能可以設定不同的物料類別，在物料主檔的欄位顯示樣式，可以把它看成是類似 HTML5 中的 css3 的顯示格式設定方式。以下是簡單的功能說明。



選擇 [欄位名稱] button 然後輸入欄位名稱(可以先用 F1 功能去查詢)。

基本上 SAP 是希望能以欄位群組的概念來設定，而不是單一欄位設定，這是合理的，因為這會簡化設定的複雜度，但是合理的歸屬群組就是能否順暢設定的重要先決條件。

基本上，SAP 是以欄位群組的概念來設定表單的欄位，其外在的顯示樣式，提供 4 種有限度的自定義設定，在不修改程式的原則下，提供 隱藏/顯示/請求輸入項/選擇性輸入 這 4 種設定。



其中又可分為 2 大類

隱藏/顯示：控制該欄位，在某物料類別時，是否顯示。

請求輸入項/選擇性輸入：這個是定義是否是必輸入欄位(不得空白)。

這個設定功能可以解決一部分的 UX 顯示問題，但是不夠深入。真正的複雜需求，還是得靠 ABAP 解決。

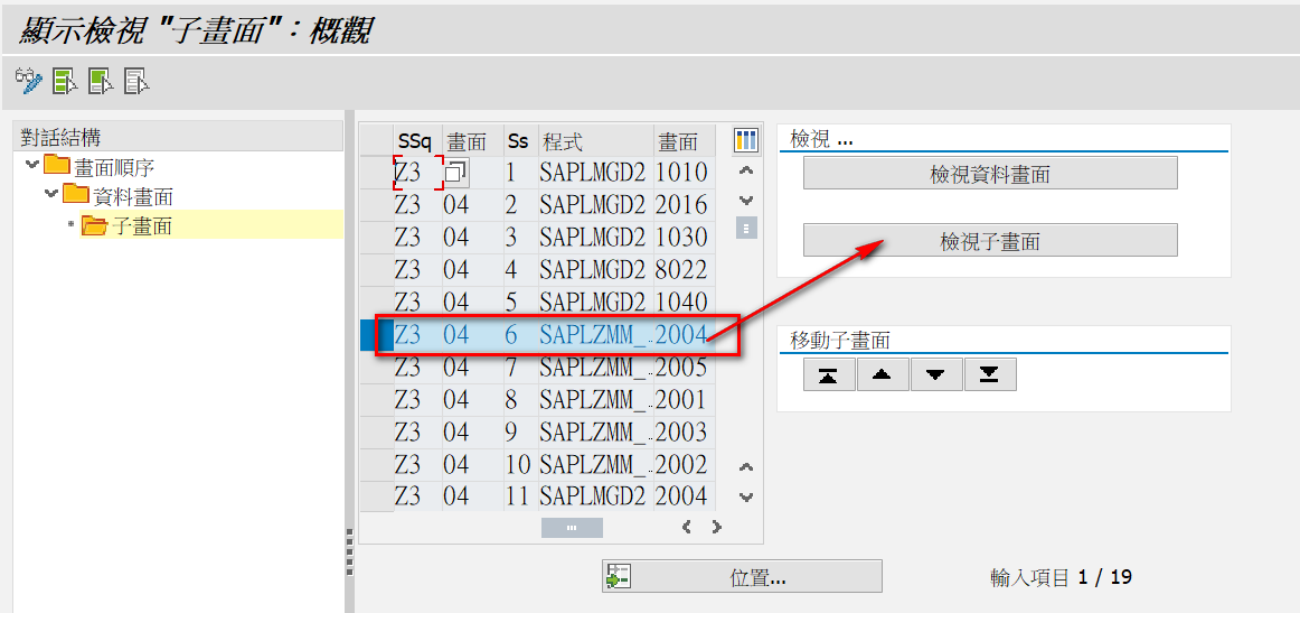
從下圖的設定就可以看出，相同的欄位(MARA-ZSHELF)，在不同的欄位參考(=物料類別)中，可以設定不同的顯示否或必輸入否。因為 Z001 Copy From HAWA，所以設定會以 HAWA 為對象。

定義畫面順序之畫面結構

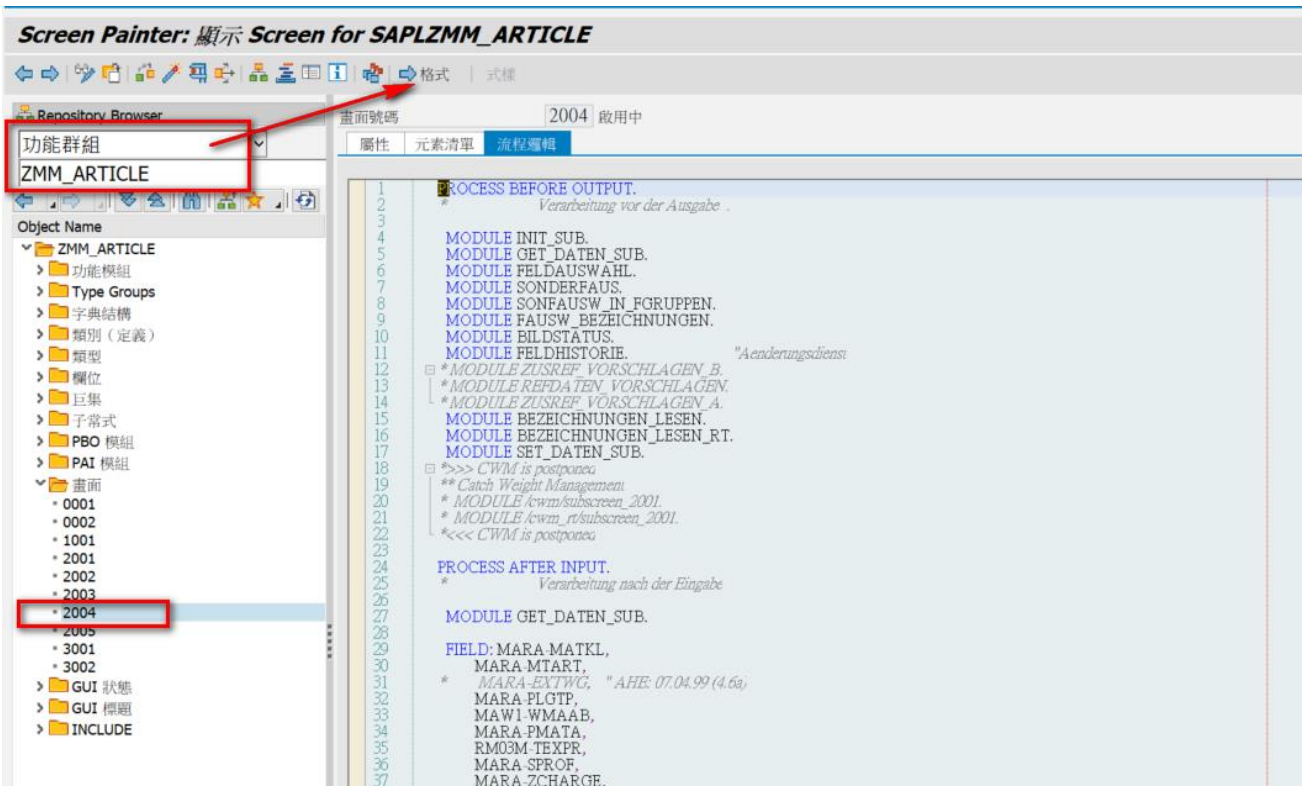
再來，我們再來玩玩 商品主檔的自定義欄位(擴充欄位) 如何擴充及定義顯示的位置。基本上是以 div 的概念來開發，此處不做深入的 abap 程式開發討論，僅追蹤一下程式的開發流程。

此處的 [畫面] 指的是 Tab，也就是物料主檔顯示的多頁籤。選定[子畫面]後，會顯示該 Tab 中的欄位。

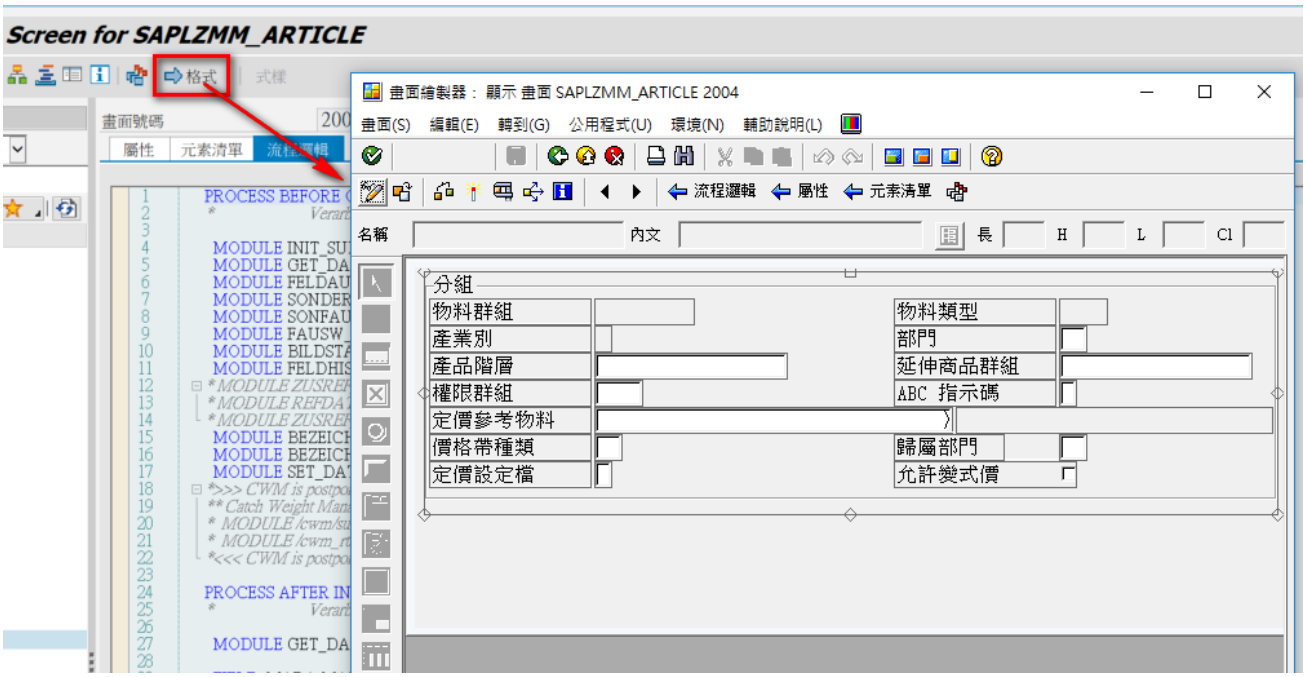
Ss 是表示 Tab 頁籤內的的幾個 Div，多個欄位組成的 block 區塊(請參考下下圖)。Div 的順序可以使用 [移動子畫面] 的 button 調整，就可以控制 Div 的顯示順序。



執行 T-Code=se80，選擇 [功能群組] 並輸入 ZMM_ARTICLE(NTT 客制程式) 後 Enter 會出現下面的 Tree，選擇 2004，請參考前頁畫面中的程式-畫面中的代號。



按下 [格式] button，可以叫出 Screen Design



詳細的 abap 程式目前還不理解，但是基本上可以透過上面的說明，做一些客制欄位及 ux 的自定義。

第三章 MM/SD 的基本資料設定

說明：SAP 的設計理念中，參數和基本資料設定是同一組概念，都是屬於後台設定，因為他會根據基本資料的特性，往下深度設定跟他有關的參數設定，例如國別，單就國別而論是基本資料，但是國別內的各項檢查設定，就是參數設定的範圍。我以往的習慣這 2 者是不同的，此處我仍沿用我的習慣來記錄。簡單的說，類似像部門、物料類型這樣的設定，我認為將其視為基本資料設定，應該是較容易理解，只不過問題還是一樣，SAP 太大了，功能太多了，已經有記不完的 T-Code 如果將所有的功能都放到 menu，menu 真的會爆掉，這應該是 SAP 會將 2 者整合的根本原因。但是無論如何，在觀念上我還是分為 2 個單元來分別討論(較容易理解)，雖然都是 SPRO 的後台功能。

顧問也有提供一份完整的各系統參數說明，本章內容僅就幾個重要的項目說明，也建議初學 SAP 者，可以先不急著一定要了解全部的參數內容及意義，否則很容易陷入其中。

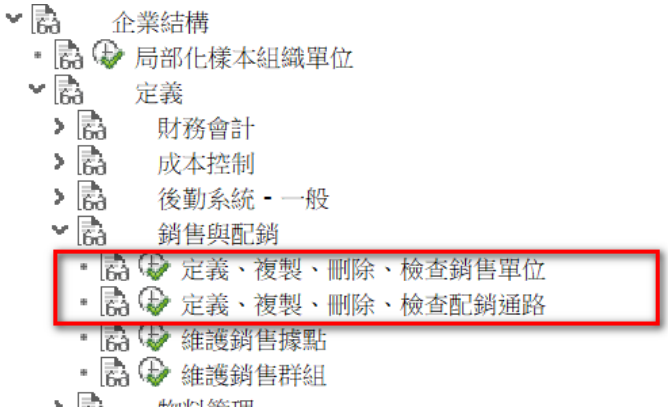
相關的設定列表

企業結構					
	定義	銷售與配銷	定義/複製銷售單位		
		銷售與配銷	定義/複製配銷通路		
		銷售與配銷	維護銷售據點		
		銷售與配銷	維護銷售群組		
		物料管理	維護儲存位置		
		物料管理	維護採購組織		
		後勤系統執行	定義/複製出貨點		
	指派	銷售與配銷	指派銷售組織到公司代碼		
		銷售與配銷	指派配銷通路給銷售組織		
		銷售與配銷	建立銷售範圍		
		銷售與配銷	指派銷售辦公室到銷售範圍		
		物料管理	指派採購單位到公司代碼		
		物料管理	指派採購組織到工廠		
		後勤系統執行	指派出貨點到工廠		
銷售與配銷					
	主檔資料	業務夥伴	客戶	銷售	定義銷售地區
		業務夥伴	客戶	銷售	定義客戶群組
		定義共同配銷通路			

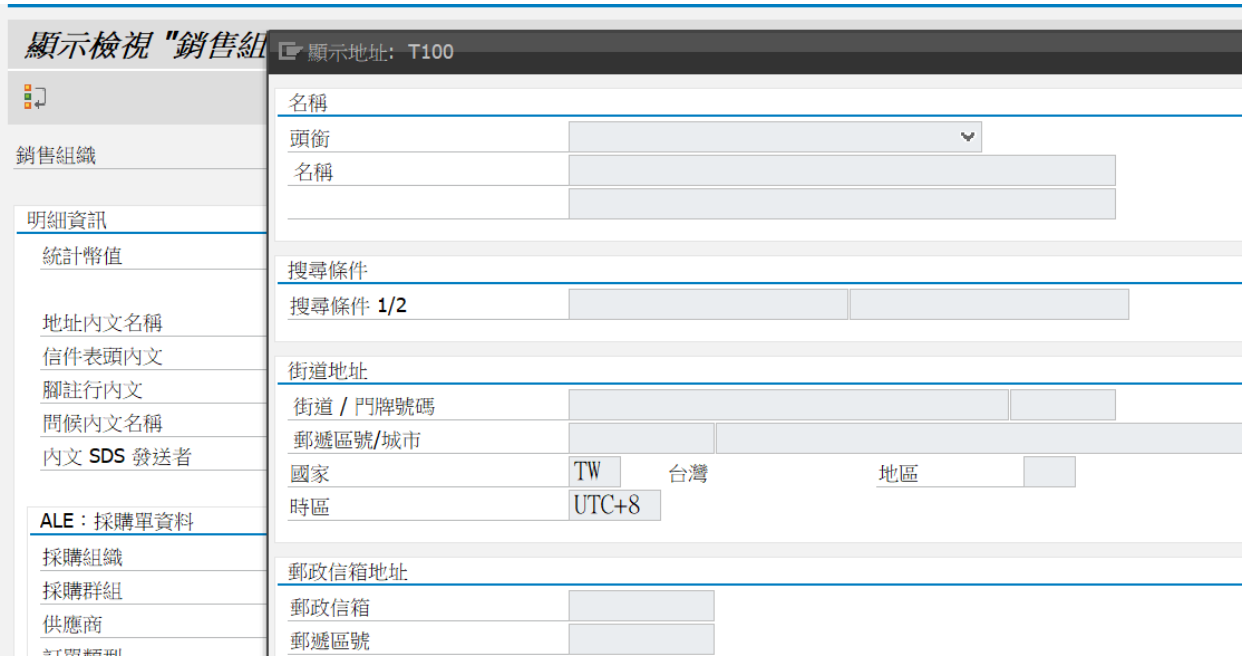
➢ **接下來，設定相關的組織，基本原則是先定義、再指派**

SAP 功能表	IMG→**企業結構**→定義→銷售與配銷→定義、複製、刪除、檢查銷售單位
交易代碼	SPRO
說明	定義、複製、刪除、檢查銷售單位(配銷通路)

➢ **銷售單位設定，因為 SAP 的關聯檔眾多，盡量都是使用複製再修改的方式建檔。**



➢ 點選 [地址] button 後，可以開啟明細設定，於輸入採購相關單據時，會自動帶入預設。



➢ 配銷通路的相關設定同上。

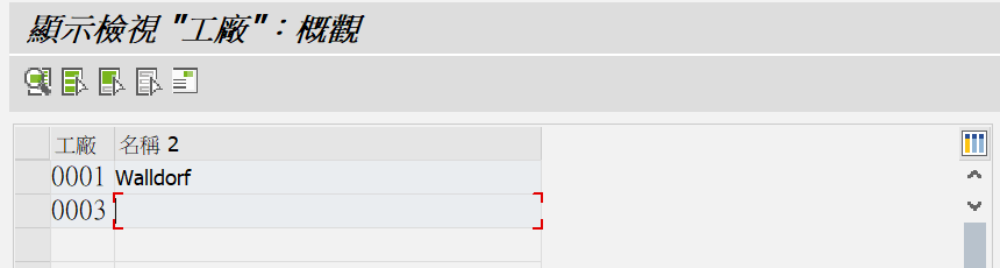
➢ 銷售群組目前沒有使用。

關於銷售據點及工廠或門市主檔 WB03 的設定說明：

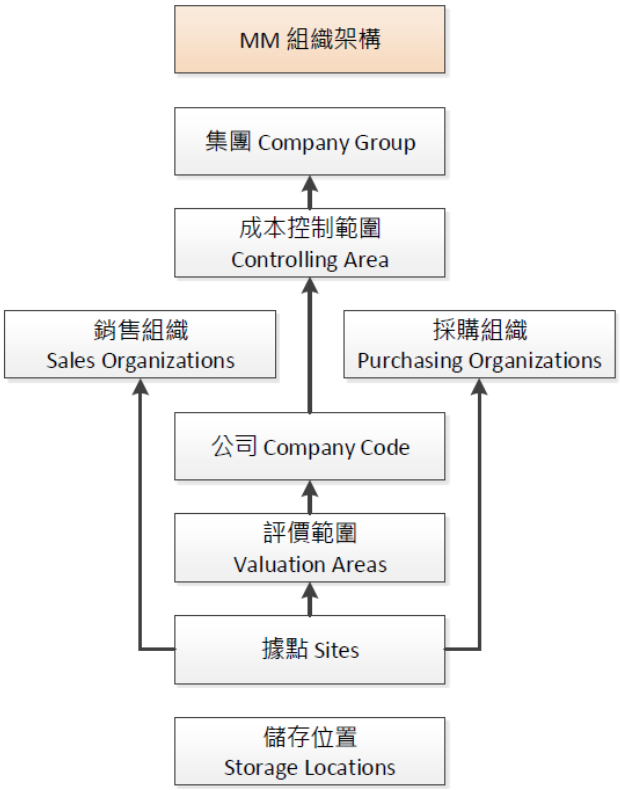
SAP 的標準架構是 Client – Company – Plants – Storage Location

目前看來並沒有設定到 Plant 工廠層(OX10)，而是使用 WB03 門市主檔(這 2 個應該是一樣的)

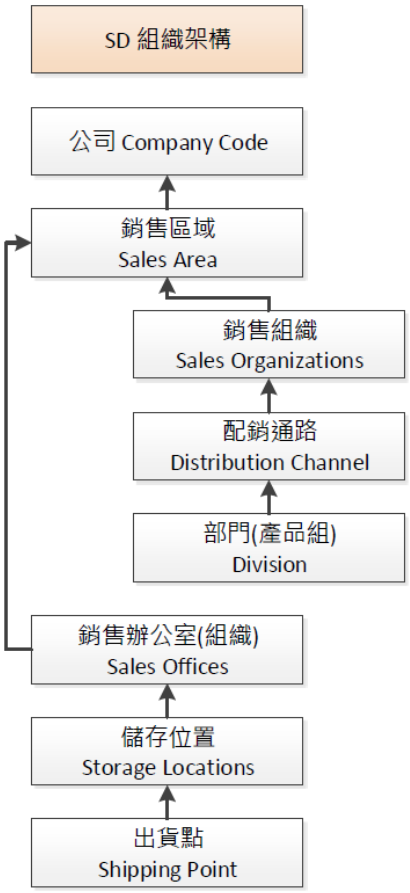
目前還是看不懂組織架構的用途，暫時先記錄疑問。



依顧問的文件說明，目前 MM 的組織架構如下



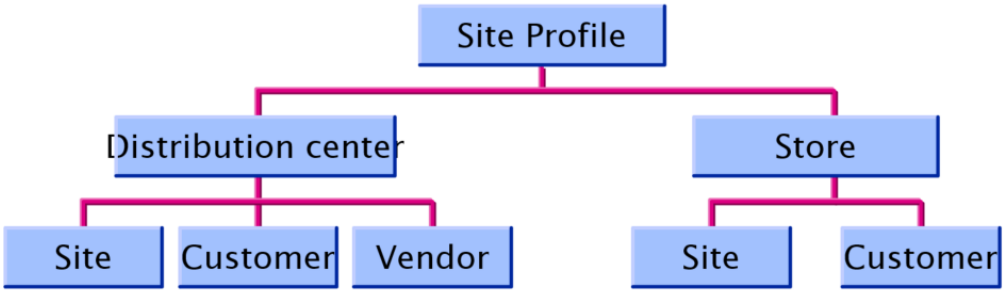
SD 的組織架構如下



這個組織架構圖目前還不了解其用途及相關的影響，為避免見樹不見林，此處先不深入理解，等基本的單據流成搞通後再回來繼續奮鬥。

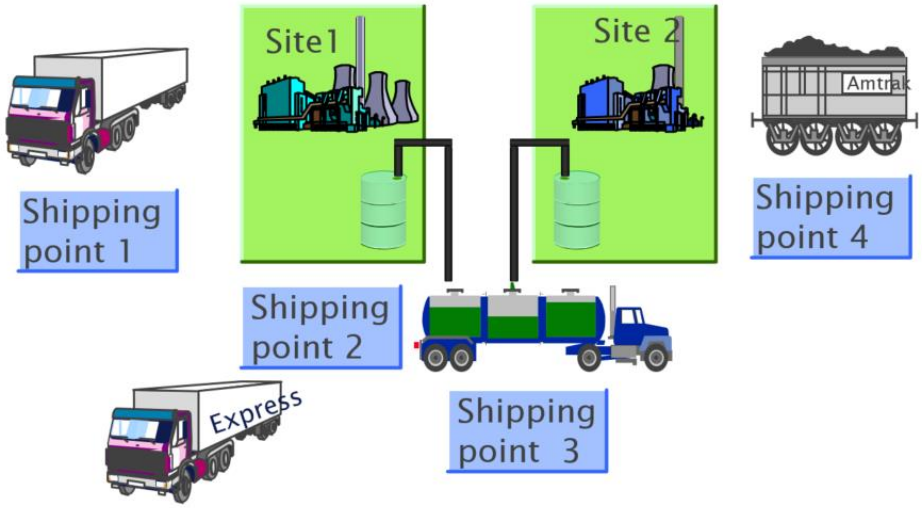
Site 據點的定義說明

- Each site master is assigned a site profile, which is used to differentiate the various site categories and control maintenance functions. For example, distribution centers, stores without internal settlement, stores with internal settlement, stores with external settlement, wholesale customers or franchisee.



IS-Retail 定義上，門市同時也具有客戶的身份。
配銷中心則同時具有客戶及供應商的身份。

關於出貨點的定義



出貨點 (Shipping point) 是處理出貨相關事項的組織單位。通常一個據點須指派一個出貨點，但一個據點也可以依商品出貨條件不同劃分多個出貨點。
每次產生出貨文件時，都需要指派出貨點。

關於店點(門市主檔) 的相關設定

WB03 設定門市主檔(以 6111 為範例)，相同的代號至少要同步設定到下面 4 個設定檔

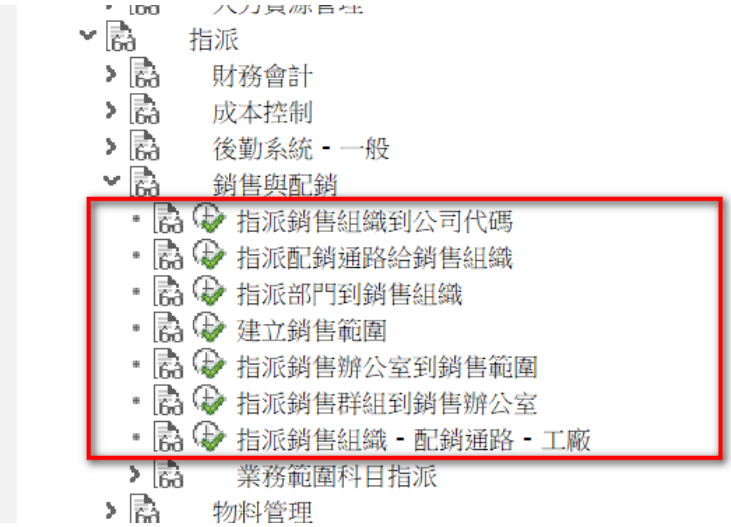
業務夥伴、客戶、銷售據點及出貨點

不曉得為何要如此設定? 設定工廠會有什麼 issue 導致要設定這麼多的關連檔?

在 Leo 顧問的教學中有提到銷售據點要設定的和門市主檔一樣，主要是為了 FI 的分公司申報使用。

➢ 出貨點設定

➢ SAP 的設定原則：先定義後指派，指派是指將銷售組織給公司代碼。



➢ 逐項指派設定，要先看 ppt 先搞懂非常複雜的關聯設定後，這邊才看得懂到底在幹嘛??? 目前一頭霧水。

顯示檢視 "指派銷售組織 - 公司代碼"：概觀

指派銷售組織 - 公司代碼

SOrg.	名稱	CoCd	公司名稱	狀態
0001	售組織 001	0001	SAP A.G.	
T100	三商家購銷售組織	T100	三商家購股份有限公司	
T200	心樸市集銷售組織	T200	心樸市集股份有限公司	

➢ 指派銷售組織給配銷通路，這是一個關聯設定作業(母子的階層關係)，當關聯設定好後，應該有 2 個用途

1. 方便後續的 BI 明細查詢與查詢
2. 輸入相關資料時的正確(合理)性檢查

顯示檢視 "指派銷售組織 - 配銷通路"：概觀

指派銷售組織 - 配銷通路

SOrg.	名稱	通路	名稱	狀態
T100	三商家購銷售組織	01	美廉社	
T100	三商家購銷售組織	02	大美超市	
T100	三商家購銷售組織	03	心樸市集	
T100	三商家購銷售組織	04	美廉城超	
T100	三商家購銷售組織	06	Office Mart	
T100	三商家購銷售組織	99	業務部	
T100	三商家購銷售組織	DC	物流	
T100	三商家購銷售組織	EC	Go 美廉	
T200	心樸市集銷售組織	05	Yolé	
T200	心樸市集銷售組織	99	業務部	
T200	心樸市集銷售組織	DC	物流	

➢ SD 的組織結構

銷售組織(Sales Organization)：銷售和分銷產品的組織單元(organizational unit)，根據銷售和分銷的要求對企業進行分組。協商銷售條款，負責銷售事務。

銷售通路 (Distribution Channel)：物料或服務用以抵達客戶的管道。典型銷售通路包括電子商務、批發、零售和直銷。用戶可以將一個銷售通路分配給一個或多個銷售組織。客戶可以從多個銷售通路 得以交貨。物科主記錄會使用不同的銷售組織和銷售通路視圖來維護，這樣就可以訪問不同的數據，例如交貨工廠。

產品組 (Divisíon)：便用產品組來對物料和服務進行分組, 可將多個產品組分配給一個銷售組織。客戶的價格協定可限定於特定的產品組內；就部分交付、定價和支付條款達成特殊客戶協議，用戶可以執行統計分析，或者建立獨立的市場營銷。

銷售區域：銷售組織、銷售通路和產品組的組合。它定義銷售組織用於銷售特定產品組中的產品的銷售通路，每個銷售和配銷憑證都恰好分配到一個銷售區域。此分配無法更改。因為一個銷售區域只能屬於一個公司代碼。此關係通過分配銷售組織來創建。處理銷售和分銷憑證時，系統將根據銷售區域存取各種主資料。例如，這些主資料包括客戶主資料、物料主資料、價格和折扣。系統還將根據銷售區域，就某些輸入的有效性執行一系列檢查。每一個銷售區域，都可以定義所有與銷售相關的資料。例如，可以基於銷售區域進行定價，或基於銷售區域進行銷售資訊的分析。使用者可以根據銷售區域來控制配置情況，例如，對於某銷售區域(例如產品樣本)，使用者可能會支援某一類銷售流程，但對於另一個銷售區域，可能就不支援。

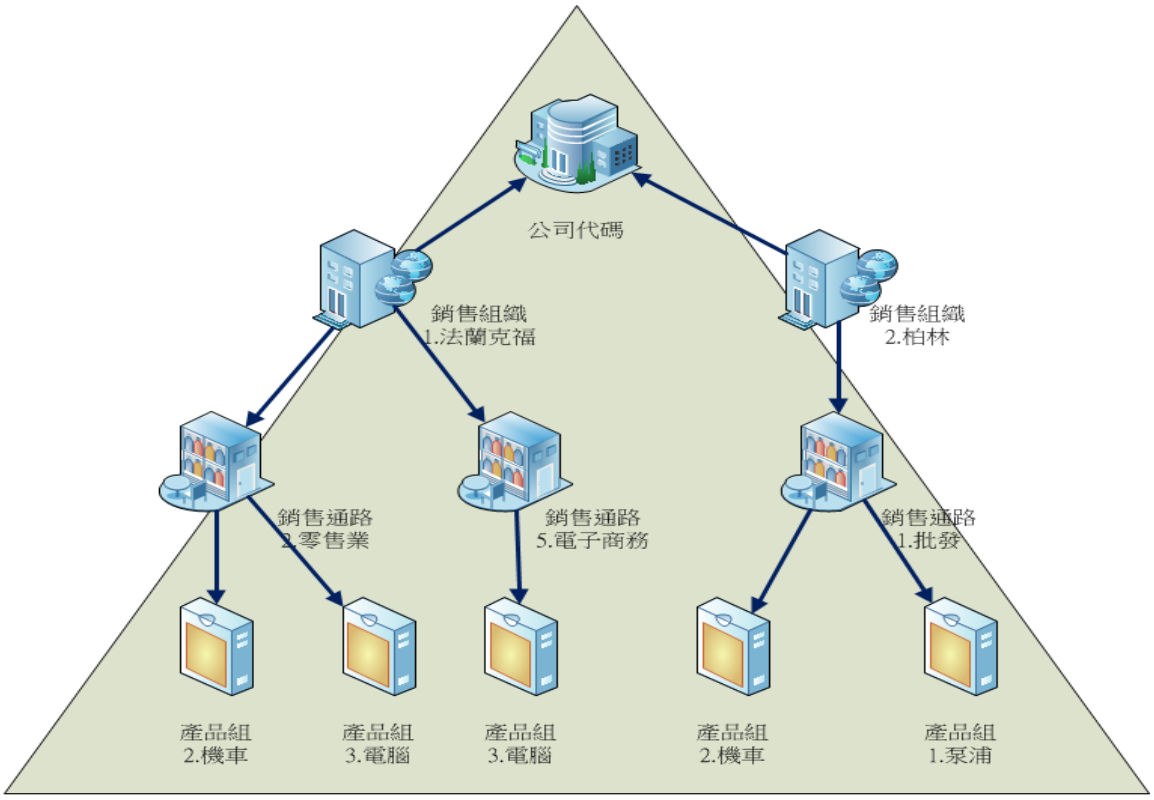
銷售據點 / 銷售辦公室(Sales Office) : 在 SAP SD 的組織結構中有可能需要銷售團隊按照組織的地理區域來分解組織。這時可以利用建立銷售辦公室來使得規畫組織結構變得非常容易。銷售辦公室需分配給銷售區域，一個銷售辦公室可能會分配給多個銷售區域。

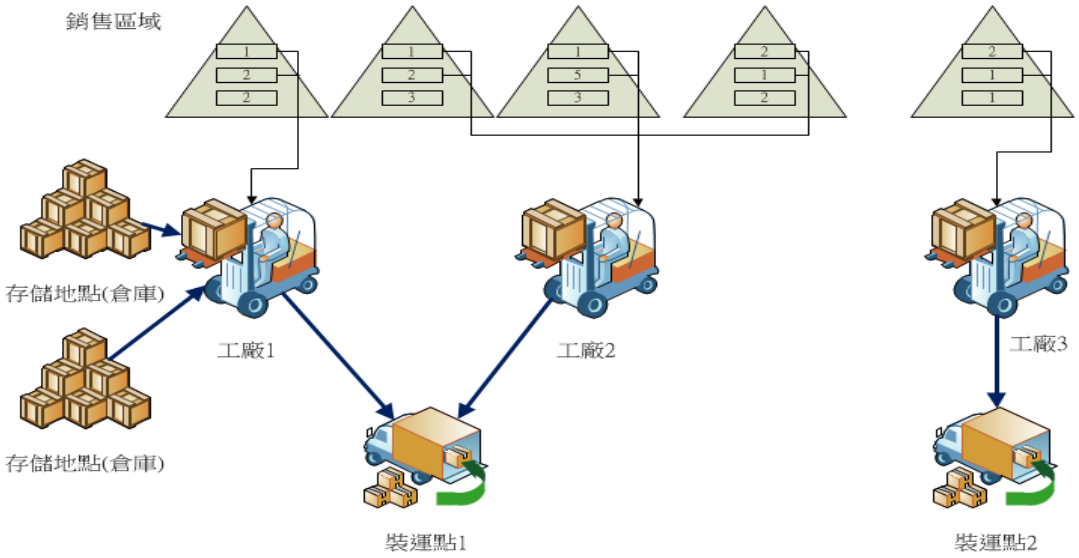
維護銷售群組(Sales Group)：銷售據點/銷售辦公室的人員可以再劃分成維護銷售群組(Sales Group)。例如可以針對獨立的產品組在銷售群組內定義。

銷售人員(Salespersons) ：可以使用獨立的員工主記錄來管理與銷售人員有關的資料。在員工主記錄中，可以把銷售人員分配給銷售群組。

工廠 (plant)：是生產物料、貨物或提供服務的一種後勤組織單元，也是生產、採購和物料計畫執行的地方。 一種物料可能會屬於多家工廠。而且，庫存還要從工廠交付于客戶，因此工廠內部也要執行可用性檢查 ATP（Available-to-Promise），需求會從銷售憑證傳入工廠的庫存需求清單(即生產或採購需求的清單)。

- 定義銷售組織/銷售單位(Sales Organization)
- 定義配銷通路/分銷渠道(distribution channel)
- 定義部門/產品組(Division)
- 定義工廠(Plant)
- 指派銷售組織到公司代碼


公司代碼
銷售組織
1.法蘭克福
銷售組織
2.柏林
銷售通路
2.零售業
銷售通路
5.電子商務
銷售通路
1.批發
產品組
2.機車
產品組
3.電腦
產品組
3.電腦
產品組
2.機車
產品組
1.泵浦


銷售區域
1
2
2
1
2
3
1
5
3
2
1
2
2
1
1
存儲地點(倉庫)
存儲地點(倉庫)
工廠1
工廠2
工廠3
裝運點1
裝運點2

第四章：SAP 基本操作及共用資料維護

學習一套系統的最有效方法，絕對不是拿著厚厚的使用說明書，從頭到尾努力的 K 完一遍。手冊當然是要看的，但是光看手冊是不可能熟悉系統的，就算你有其他系統的多年深厚經驗也一樣。

我個人建議學習的步驟如下：

1. 先看一遍快速入門的手冊或教學影片

   先對要學習的新系統(如 SAP 或 Oracle) 建立基本的概念。因為 SAP 系統實在太龐大了，一不小心就會迷失在其中，最好先選好一個模組(如果沒有意外，建議先從 MM 開始)，這類文件不多，如果你所在的公司已經導入或正在導入 SAP，導入顧問一定會建立企業藍圖和基本的操作手冊，這是最好的切入點。我一開始入門也是由此開始。如果想要深入學習，這樣當然是不夠的，所以我同步 Google 了大量的 Blog 內容，本文件基本上就是我重新整理後的心得，主要的目的是協助新人能快速理解 SAP 的概念。

   https://v.qq.com/x/page/b0121j768m0.html

2. 建立一個完整的 UAT 測試環境，將前一步驟的文檔，逐一操作過一次

   基本上，我的建議是先搞清楚最核心的基本功能及流程，再往外擴充學習。例如商品主檔，我不會在一開始就想了解每一個欄位的用途，如果看不懂某些欄位的用途，我建議先跳過，先輸入必要欄位後存檔即可，先跑過整個業務流程，對系統有概括的認識比較重要。這個過程盡量減少後台的設定，主要是熟悉前台業務邏輯的操作。

3. 依照使用手冊內容，邊看內容邊實際操作

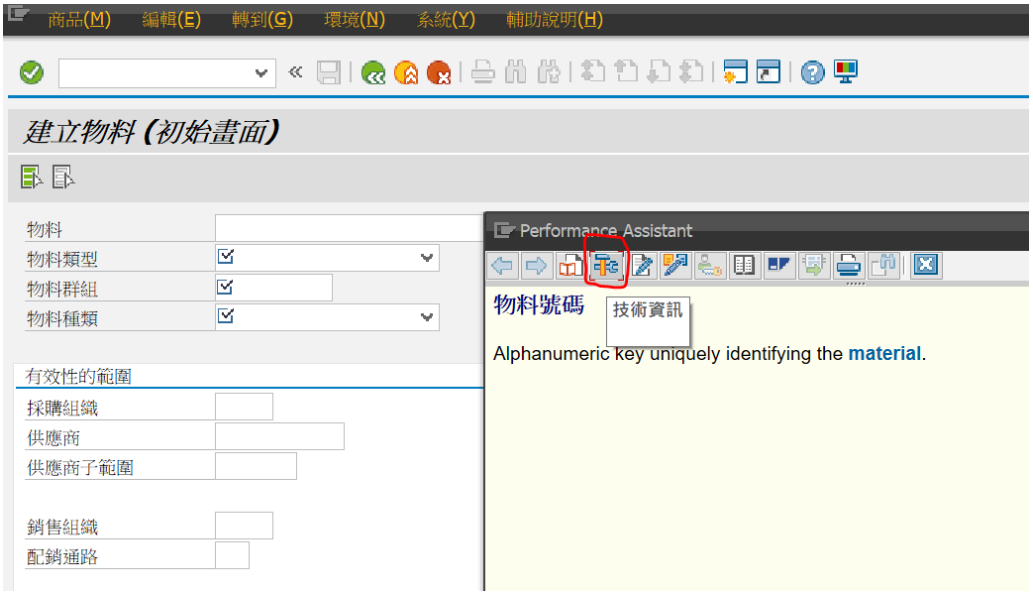
   最後是不斷的練習、實踐，主要還是要有實際的工作環境，透過實務的不斷解題，來提升自己的程度，最後就會成為高手，所以以下的內容，請務必要逐項的實作。

Session 1：如何識別某功能、某輸入元件對應的 Table 及 Field?

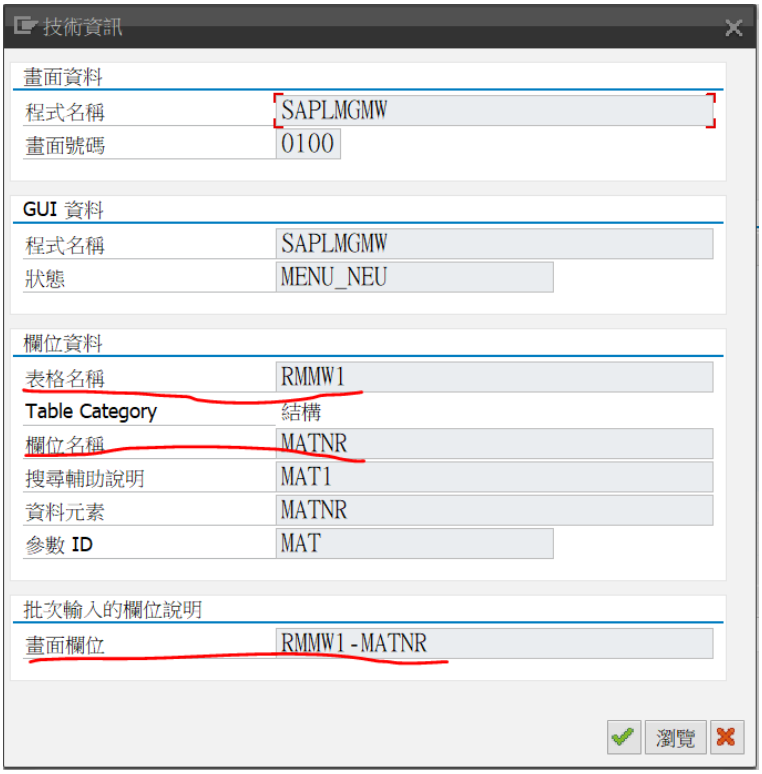
方法一：Google，網路上有很多大神有整理出常用的 Table 內容

方法二：利用 SAP 的 F1 功能

在要查詢 Table Schema 的表單上的任意欄位，按下 F1 會出現下圖



然後按下 [技術資訊] 會開啟一個新視窗



另外，如何查詢在 SPRO 執行某功能後，查詢其 T-Code 代號為何?

只要執行該功能後，在 UI 最下方的狀態列中，查詢交易，這就是 T-Code。

Session 2：整理出最基礎、最核心的 T-Code 列表 - 同時記錄相關的 Table Schema

序號	功能名稱	T-Code	Table Schema
以下是基本資料維護(序號 1-29)			
1	國別	SPRO-SAP NetWeaver–一般設定–設定國家–設定國家特定檢查	T005 / T005T
2	公司代碼	SPRO-企業結構-定義-財務會計-編輯、複製、刪除、檢查公司代碼	T100
3	幣別	SPRO-SAP NetWeaver–一般設定–幣別	TCURT / TCURC
4	單位	SPRO-SAP NetWeaver–一般設定–檢查計量單位	T006A
5	地區	SPRO–銷售與配銷-主檔-業務夥伴-客戶-銷售-定義銷售地區	V_T171
6	通路	VOR1 或 SPRO–銷售與配銷-主檔-定義共同配銷通路	TVTA
7	客戶資料維護	BP (BUT000 ??)	KNA1 客戶主文件的一般資料 KNB1 客戶主記錄公司代碼資料 KNBK 客戶主資料 (銀行細目) KNVK 客戶主要聯繫夥伴 KNVV 客戶主記錄銷售資料 KNVP 客戶主記錄夥伴功能
8	客戶類別	SPRO–**銷售與配銷**-主檔-業務夥伴-客戶-銷售-定義客戶群組	V_T151
9	廠商資料維護	BP	LFA1 供應商主資料 (一般地區) LFB1 供應商主資料 (公司代碼) LFB5 供應商主資料 (催款資料) LFM1 供應商記錄採購組織資料 LFM2 供應商文件記錄:採購數據 LFBK 供應商主資料 (銀行細目)
10	廠商類別	OMSG 供應商科目群組	V_T077K_M
11	商品主檔維護	以下是標準版本的 T-Code MM01-MM04 Create/Modify/Display / 顯示 Material 資料 三商有擴充很多的 SKU 欄位，所以有另外修改程式，T-Code 如下 **MM41-MM44**	MARA 一般物料資料 MAKT 商品描述 MARC 物料的工廠資料 MARD 物料的倉儲位置資料 MBEW 商品評估 MVKE 物料銷售視圖資料
12	商品大類 商品中類	OMSF 或 SPRO–**後勤系統-一般**-關鍵欄位設定-定義	V023

	商品小類(物料群組)	物料群組	
13	物料類型	SPRO–**後勤系統-一般**-物料主檔-基本設定-物料類型-定義物料類型的屬性	T134
14	倉庫資料維護 Storage Location	OX09 MMSC(彙總輸入) OVL3(查詢)	T001L
15	部門資料維護	VOR2 OVXB(查詢) 或 SPRO–**後勤系統-一般**-關鍵欄位設定-定義部門	V_TSPA
16	員工資料維護	BP ~~XD01 或 VPE1~~ 員工/組織/群組	BUT000(Type=?)
17	據點資料維護(Site)	WB01 新增 WB02 修改 WB03 查詢	T001K
18	工廠(Plant)	本來以為沒有使用，但查詢 se16n T001W 還是有建立所有的店點資料 所以我完全搞不清楚到底發生甚麼事了 因為在建立 Site 時，會同步建立業務夥伴及供應工廠，所以我猜，應該是在存檔時，會自動新增一筆 Plant 資料 OX10 目前只有 2 筆資料??? Why ???	T001W is a standard SAP Table which is used to store Plants/Branches data and is available within R/3 SAP systems depending on the version and release level.
19	銷售據點	SPRO–**企業結構**-定義-銷售與配銷-維護銷售據點	V_TVBUR
20	銷售單位 配銷通路	SPRO–企業結構-定義-銷售與配銷(menu 下)	V_TVKO
21	出貨點	SPRO–企業結構-定義-後勤系統執行-出貨點	TVST
	FICO 常用透明表		SKA1 總帳科目主記錄 BNKA 銀行主記錄 BKPF 會計憑證抬頭 BSEG 會計憑證行專案 BSAD 已清應收帳款 BSID 未清應收帳款 BSAK 已清應付帳款 BSIK 未清應付帳款 BSAS 已清總帳科目

			BSIS 未清總帳科目 TKA01 成本控制範圍 TKA02 成本控制範圍分配 CSKS 成本中心主資料 CSKT 成本中心說明文本 CRCO 分配工作中心到成本中心 KALO 成本核算運行：成本核算對象(KVMK) KANZ 銷售訂單專案的分配 - 成本核算對象 COSP CO 對象：外部記帳的成本總計 COEP 成本控制物件：與期間相關的各行項目 COBK 成本控制物件：憑證抬頭 COST 成本控制物件：價格總計 FAGLFLEXA 總帳: 實際行項目 FAGLFLEXT 總帳: 總計
以下是常用單據(序號 30-49)			
30	物料異動	MIGO_GR 採購單收貨 MIGO_GI 發料 MIGO_TR 移轉過帳 MIGO_GS 外包後續調整	
以下是常用報表(序號 50-69)			
50	現有庫存查詢	MB52	

Session 3：關於商業夥伴(Business Partner：BP) 的說明

傳統的 ERP 裡用客戶和供應商來分別管理與我們發生往來業務的物件，該模型有許多限制，如：

https://www.itread01.com/content/1543482259.html

1、一個客戶或供應商僅僅只能有一個地址

2、客戶和供應商沒有關聯關係（同一個實體公司）

3、如果一家公司即是供應商也是客戶則需要建立兩個主資料，造成資料冗餘

4、屬性無時間相關性

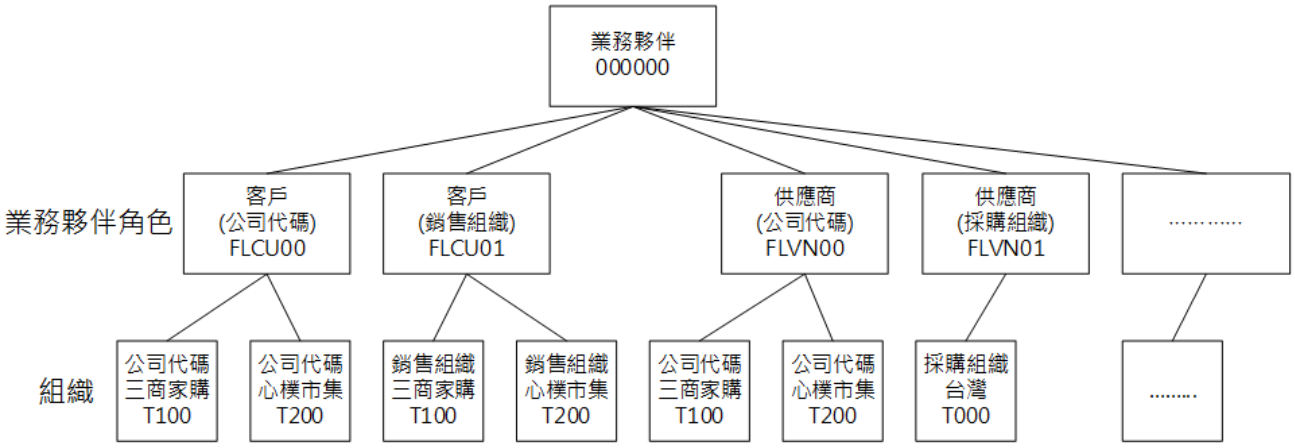
在 S4 中，SAP 提供了 Business Partner (業務夥伴)模型來解決上述問題（傳統的 FD01/02/03 和 FK01/02/03 已經被棄用），該模型有如下優點：

1、一個業務夥伴也可扮演多個角色（員工/客戶/供應商）且角色之間共享基礎資訊

2、時間相關的屬性

3、統一的資料架構，消除資料冗餘

4、CVI 模型可以確保 BP 和 customer/vendor 資料同步

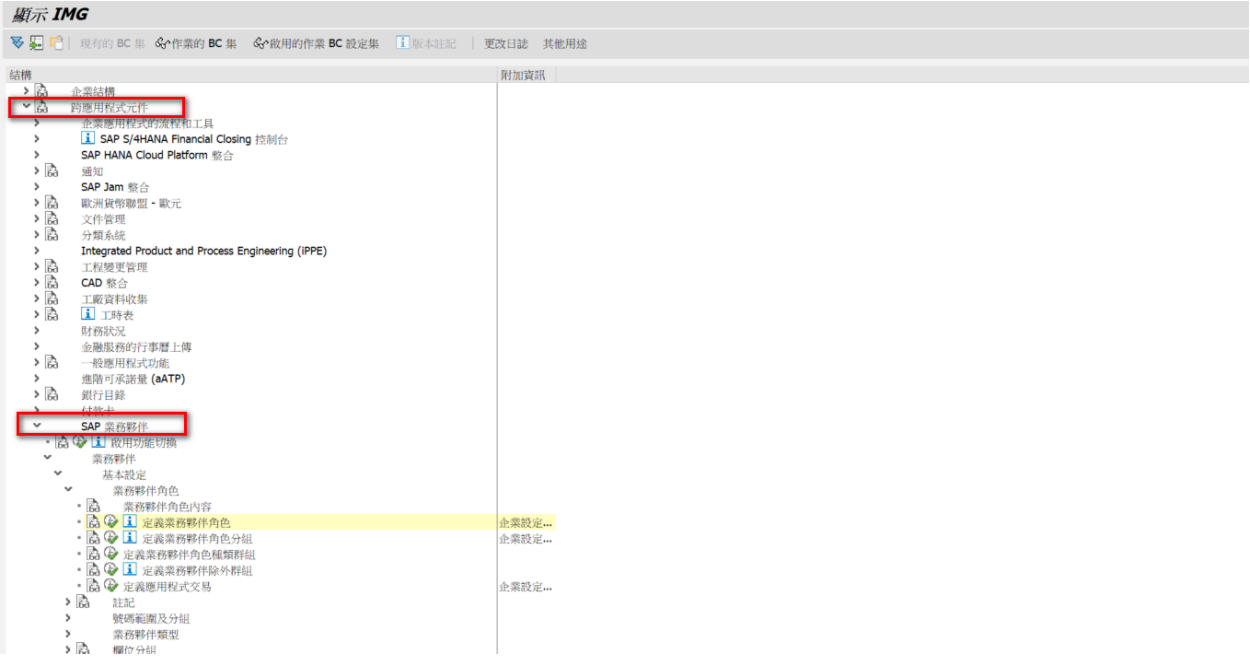


SAP S/4 Hana 版本，已經將業務夥伴與客戶主檔、供應商主檔結合，因此維護客戶主檔、供應商主檔前，需先建立業務夥伴主檔。業務夥伴主檔大部分是一般基本資料。業務夥伴也會決定客戶主檔及供應商主檔的編碼。另，客戶主檔及供應商主檔是不同的兩個主檔，可以分別獨自建立。對需要開立銷項發票並建立應收帳款者，則稱作客戶；反之，對需要取得進項發票（或收據）並建立應付帳款者，稱作供應商。

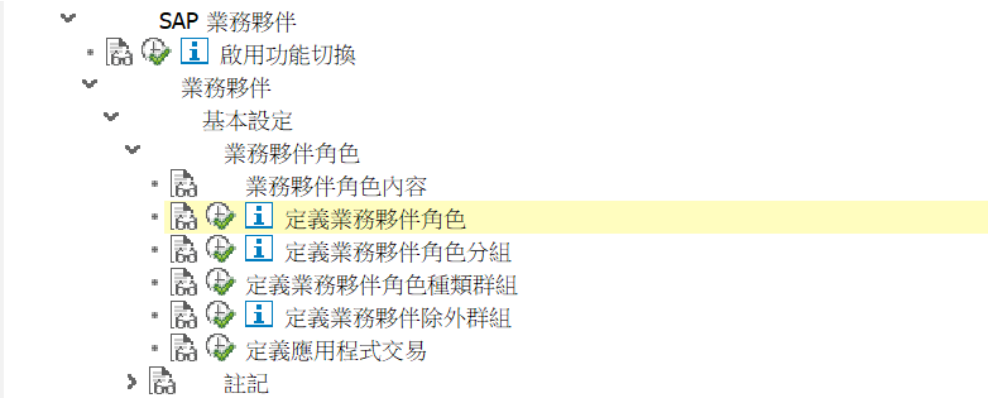
系統提供了一些列標準的業務夥伴角色，我們可以直接使用或者 copy 出自己的。不同的業務夥伴角色下有不同的屬性，代表不同的業務場景，以下四個角色為例說明：

角色	說明
FLCU00 客戶(財務)	代表業務夥伴客戶在財務的角色，屬性同於 FD01/FD02/FD03
FLCU01 客戶(銷售)	代表業務夥伴客戶在銷售的角色，屬性同於 VD01/VD02/VD03
FLVN00 供應商(財務)	代表業務夥伴供應商在財務的角色，屬性同於 FK01/FK02/FK03
FLVN01 供應商(採購)	代表業務夥伴供應商在採購的角色，屬性同於 MK01/MK02/MK03

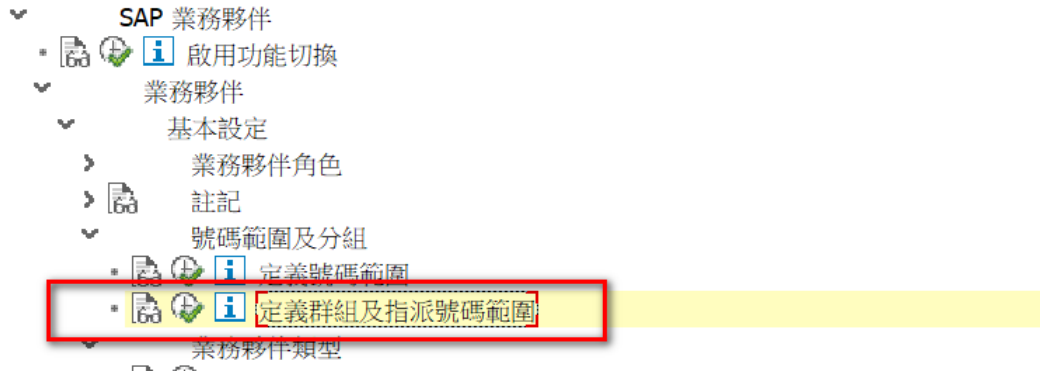
SPRO-跨應用程式元件-SAP 業務夥伴-業務夥伴-基本設定-業務夥伴角色



請逐項查詢內容及建檔，內容如下



定義業務夥伴群組



顯示檢視 "業務夥伴分組"：概觀

分組	名稱（短）	說明	號碼範圍	外部	內部標準分組	外部
S150	獲授權方	獲授權方	AB	☐		
S160	批發客戶	批發客戶	AB	☐		
SRM	利益關係人	利益關係人關係管理		☐		
TR01	財務管理夥伴	財務管理夥伴（內部）		☐		
TR02	財務管理夥伴	財務管理夥伴（外部）	AB	☐		
Z001	進貨供應商	進貨供應商：5碼系統自動編號	V0	☐	○	
Z002	其他費用廠商	其他費用廠商：A+6碼數字	A0	☑		
Z003	門市房東	門市房東：R+6碼數字	R0	☑		
Z004	加盟主	加盟主：F+5碼數字	F0	☑		
Z005	員工	員工：E+A~Z+6碼數字	EZ	☑		
Z006	門市	門市：4碼數字	10	☑		
Z007	第三方支付/B2B/B2C…	第三方支付/B2B/B2C客戶：C+6碼數字	C0	☑		
Z008	關係人	關係人：T+5碼數字	T0	☑		

在 BP 中下拉的角色，就在在此功能維護

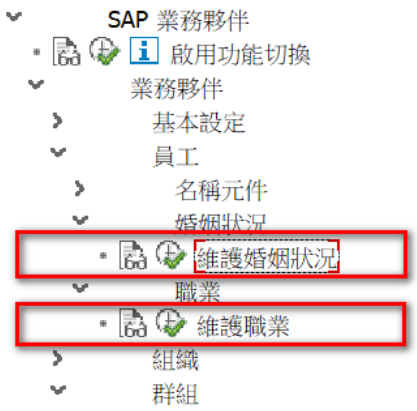
顯示檢視 "業務夥伴角色"：概觀

對話結構
- 業務夥伴角色
- 業務夥伴角色種類
  - 業務夥伴角色種類 --> 商業

業務夥伴角色

業務夥伴角色	標題	說明
CACSA1	佣金契約夥伴	佣金契約夥伴
CACSA2	佣金人員	佣金人員
CACSA3	處理人	處理人
CMS001	擔保夥伴	CMS 擔保夥伴
CRM010	承運方	承運方
FLCU00	FI 客戶	業務夥伴：FI 客戶
FLCU01	客戶	業務夥伴：客戶
FLVN00	FI 供應商	業務夥伴：FI 供應商
FLVN01	供應商	業務夥伴：供應商
FS0000	金融服務業務夥伴	金融服務業務夥伴
FS0003	業務夥伴區分	含區分的業務夥伴金融服務
FS0KNE	借款人實體	業務夥伴借款人實體

有很多細項的設定，如員工檔內的婚姻狀況、職業等下拉欄位，在其他 ERP 系統可能會用 Key-Value 參數檔統一定義內容。在 SAP 中，大量的在 SPRO 中很多設定功能。所以 SPRO 基本上是參數檔及所有次階功能的設定，會連結出一個超大的細項維護功能群組。



目前可能有使用的業務夥伴(角色) (**此處待確定)**

BP 的業務夥伴(角色)		確認目前正使用
BBP000	供應商	
BUP001	聯絡人	
BUP002	潛在客戶	
FLCU01	FI 客戶	
FLCU02	客戶	
FLVN01	FI 供應商	
FLVN02	供應商	
SLLSTL	職員	
TR0110	潛在客戶	
TR0156	當地代理商	
TR0605	所有人(客戶)	
TR0998	職員	
VLC001	最終客戶	

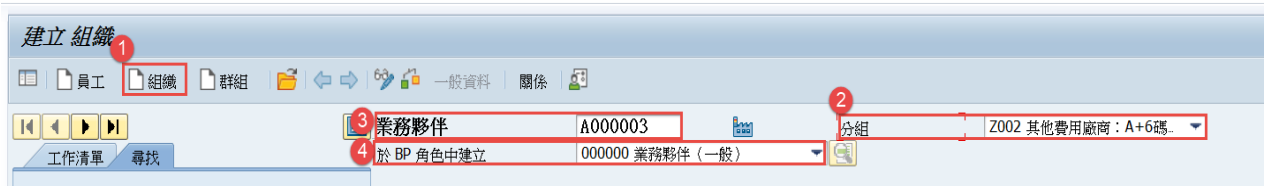
Issue：確定不使用的資料，能否刪除? 如何確定是否有正確選擇?

Session 4：基本操作說明(分為單檔及表單) – 針對通用性基本操作，尤其是 Drill-Down 功能

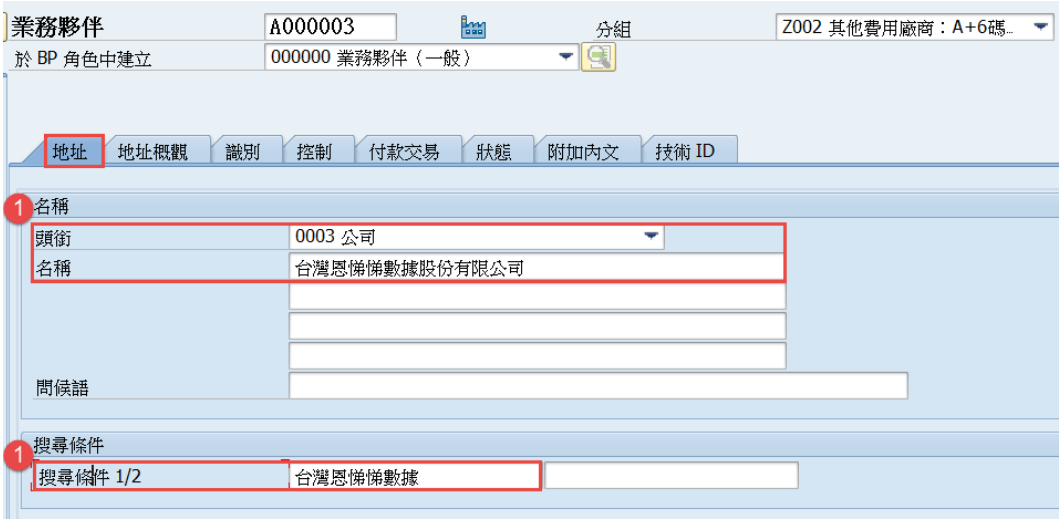
	操作重點說明(UX)
1	客戶資料維護 *BP* - 會計→ 財務供應鏈管理→ 財務與風險管理→ 業務夥伴→ 維護業務夥伴
	特殊欄位重點說明，前一節有針對 BP 相關的前置設定說明，請先看過一次。

1. 進入業務夥伴主檔初始畫面，請先查詢您需要建立的業務夥伴群組目前的編碼原則及目前已使用的編號，如果要查詢後修改已建檔的資料，也是先利用此方式先查詢到該筆資料 DbClick 後，再按 編輯。

2. 開始建立業務夥伴主檔
   (2) 選擇您要建立的業務夥伴群組。這個選擇會決定業務夥伴的號碼範圍。
   (3) 輸入業務夥伴代號
   (4) 於 BP 角色中建立：固定選擇 000000 業務夥伴（一般）。



3. 接著開始維護各頁籤資料，進入業務夥伴 [地址] 頁籤，並逐一輸入各欄位內容

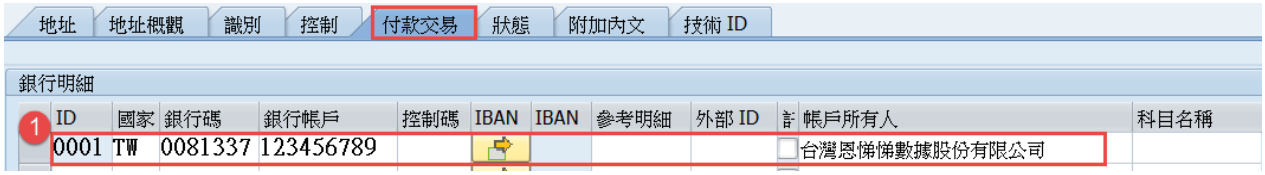


4. 進入業務夥伴[地址概觀]頁籤

若此業務夥伴有多個地址需要維護，請至[地址概觀] 頁籤後，點選建立

5. 進入業務夥伴 [識別] 頁籤，若此業務夥伴非公司組織，則可勾選自然人

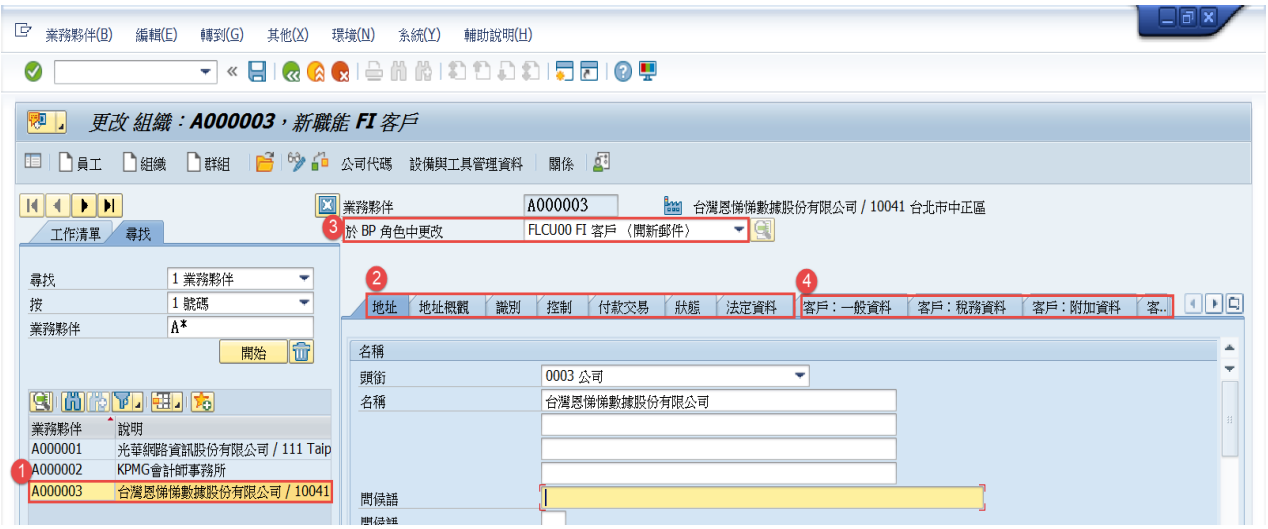
6. 進入業務夥伴[付款交易]頁籤



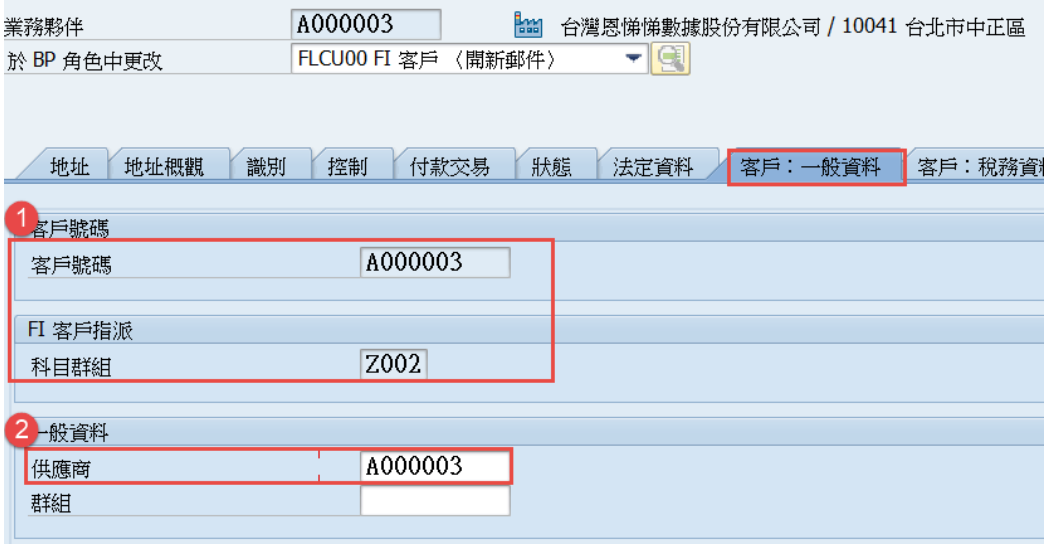
7. 最後，確認您所輸入的欄位正確無誤後，按存檔即可

以上只是建立 [業務夥伴主檔]，接下來，開始建立 [客戶(或供應商)資料主檔]

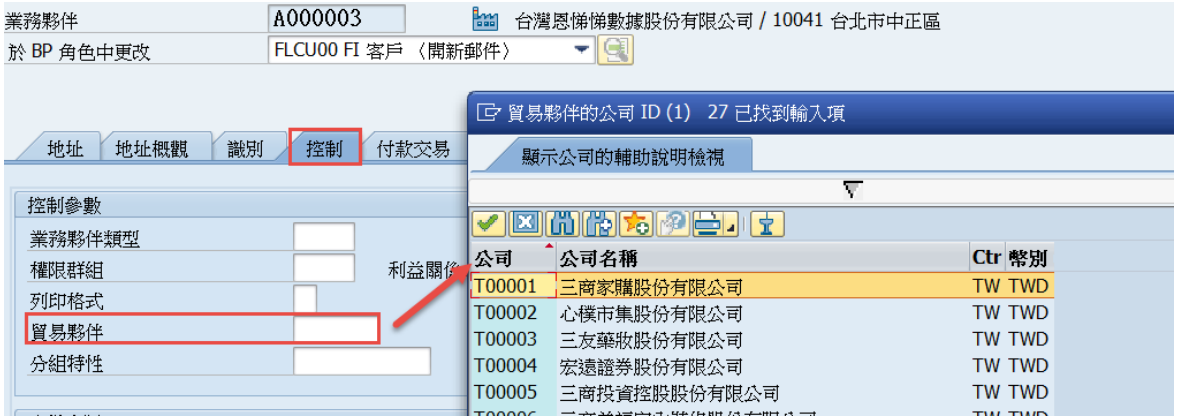
1. 進入業務夥伴主檔初始畫面，查詢到剛建檔的該筆資料，然後修改 BP 角色為 FLCU00，按 Enter 後自動開出新的客戶相關的頁籤。



2. 開始維護客戶主檔一般資料區塊各頁籤資料，如果也是供應商，供應商也一併輸入，業務夥伴編號一般而言都會和客戶編號、供應商編號設定相同，方便管理(如下圖)。



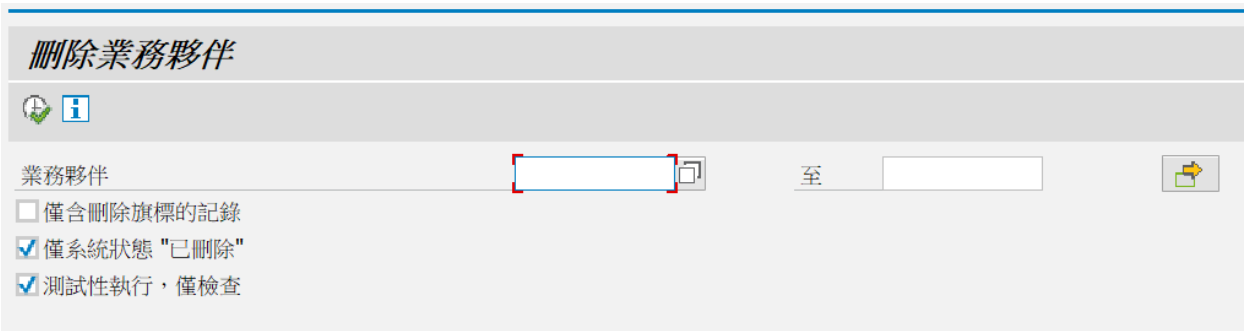
若此客戶為本公司之關係人，請於 [控制] 頁籤，於貿易夥伴欄位以下拉選單點選關係人代號。



3. 接著，開始維護客戶主檔之 [公司代碼] 區塊，會切換另一個畫面，逐一輸入個欄位資料即可。

4. 同理，如果此業務夥伴也是供應商，同上步驟，請將 BP 角色為 FLVN00，按 Enter 後自動開出新的供應商相關的頁籤，其他操作同客戶資料維護。
5. 如果要凍結/ 刪除旗標客戶(供應商)主檔，到 [客戶：狀態] 頁籤勾選相關的 checkbox 即可。請注意，這個頁籤是在 [公司代碼] 這支程式內。

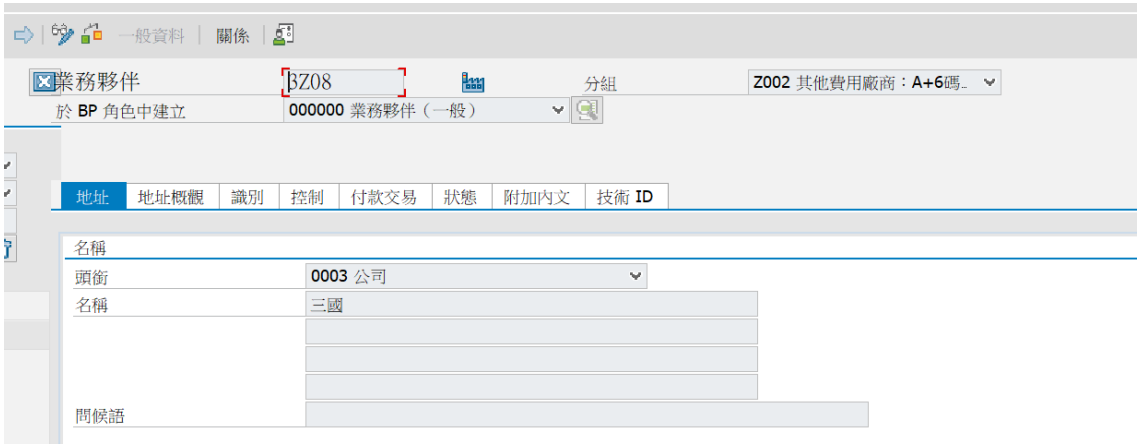
如果要真正的刪除 BP 資料，可以利用 T-Code=BUPA_DEL



特別注意事項：

在練習建檔時，曾經發生輸入的資料有錯誤，按存檔時，會發錯誤訊息警告(夥伴號碼長度錯誤)，然後該筆資料就被 Lock，無法修改任何欄位的資料，就卡在那邊無法存檔。只能放棄、重新輸入。到目前為止還不確定該如何解決這個問題，仍在尋找解決的方法。

經過 Google 和 SAP Help 後，仍無所獲。然後，在我重複操作的過程中發現一個現象，當我輸入錯誤資料，發生錯誤 Alert 後，SAP 會讓表單進入灰色的 ReadOnly 狀態



然後，我鬼使神差的移到編號欄位，按下 Enter，就會變成白色的編輯狀態，問題解決，真是神奇的操作阿。

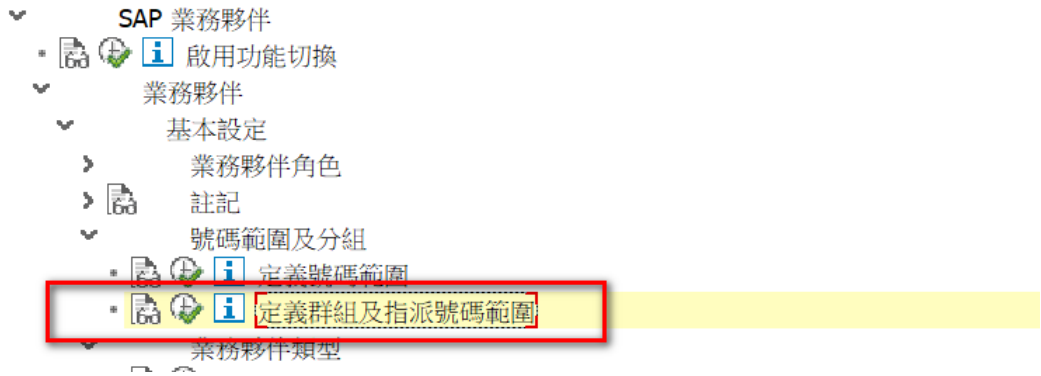
PS：後來又發現這個方法有例外，當編號的內容或長度跟後台設定有衝突時，這個方式無效。

關於單據的編碼原則設定，請參考後台設定

SPRO-跨應用程式元件-SAP 業務夥伴-業務夥伴-基本設定-號碼範圍及分組-定義群組及指派號碼範圍

在 SAP 中主要的主檔或單據都有類似的設定，而且還會區分所謂的內部及外部編號。所謂內部編號，指的是存檔時由系統自動產生編號，外部則是考慮到期初開帳(或其他會有大資料匯入時)，會有透過 txt 或 Excel 檔案批次匯入資料，因為編號在舊系統時已經產生，此時系統會判定是外部其他系統的資料要匯入，就不會檢查內部編碼原則，但是仍會根據外部編碼原則檢查是否有異常資料。

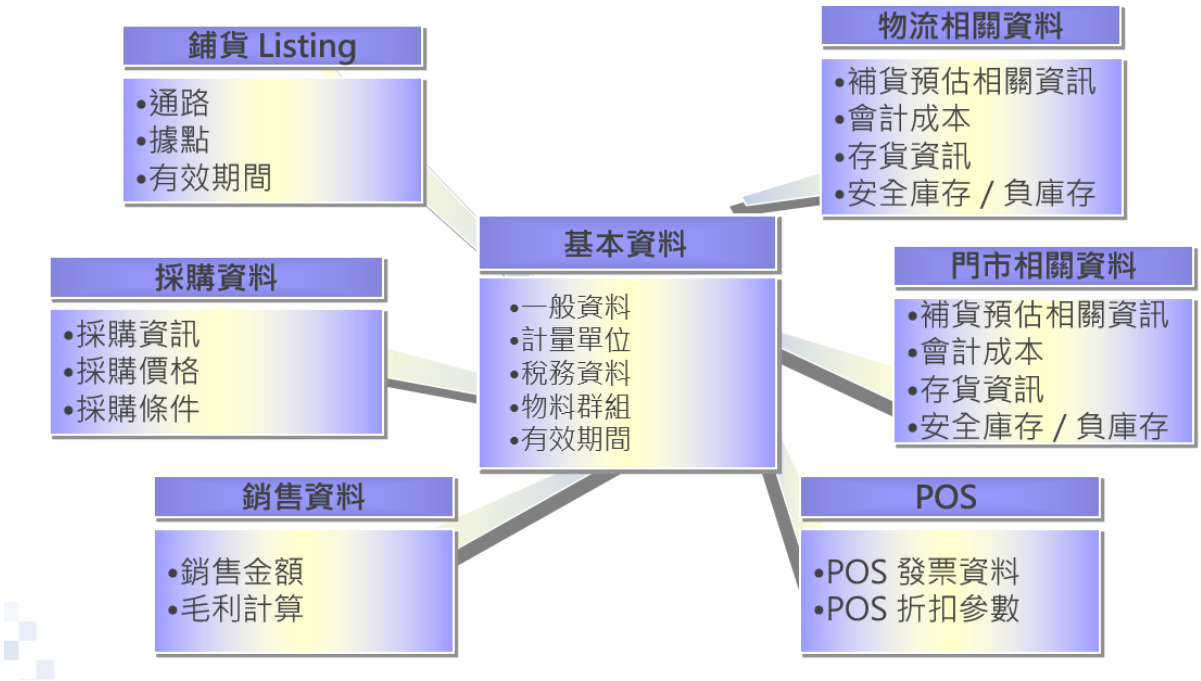
說實在的，所謂的細節裡隱藏著魔鬼，SAP 在這種細節控制上的考慮，的確是一般其他系統所望塵莫及的，不得不佩服。



2	物料主檔建檔作業(單檔維護) 主檔資料→物料資料→物料→建立 MM41 主檔資料→物料資料→物料→更改 MM42 主檔資料→物料資料→物料→顯示 MM43
	特殊欄位重點說明

物料建檔除了 MM41-MM43 之外，還有如下的主要標準功能(本文檔不另說明)。另外，商品主檔的新增主要是由 SCM 的新品提報自動匯入 SAP 完成，人工作業較少使用，特此說明。

T-Code	功能說明
ME11	建立採購資訊記錄
MEK1	維護採購價格
WRCR	商品重分類（更改商品群組）
WSP4	建立鋪貨
WSE6	刪除鋪貨
WSK1	大量複製鋪貨



1. 輸入物料(商品)類型、物料群組及物料種類，若該物料需要向外部供應商購買，則須輸入採購組織及供應商編號，右方的頁籤一般都是全選即可。

商品類型	商品類型說明	量更新	值更新	號碼起	號碼迄
Z001	一般商品（含貿易商品和原物料）	V	V	1000000	9999999
Z002	自製品 - 成品	V	V	1000000	9999999
Z003	自製品 - 半成品	V	V	1000000	9999999
Z004	營運用品（含備品、文具用品）	V	V	1000000	9999999
Z005	非一般商品（如儲值、停車費等代收代付品項）	X	X	1000000	9999999

2. 維護基本資料，輸入品名、單位、長寬高、採購群組等欄位。

若商品有多單位，可在計量單位頁面輸入其換算比例。例如商品有瓶、手、箱等三個單位，則輸入：

瓶	<=>	1	瓶
手	<=>	6	瓶
箱	<=>	2	手

3. 商品有四個預設單位可以選擇，分別為基礎計量單位（庫存單位）、訂購單位（採購單位）、D/I（調撥發貨單位）、銷售單位。

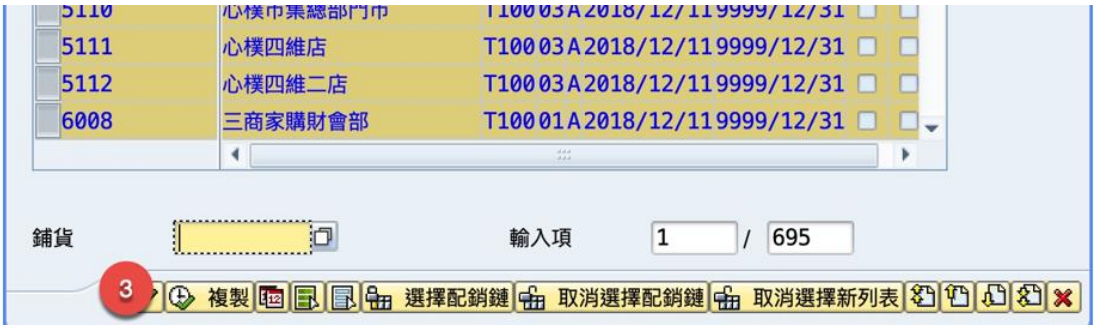
4. [鋪貨] 頁籤，是指這個商品在哪些據點可以採購及銷售。鋪貨這個功能看起來是 SAP Retail 模組的標準功能，我覺得他有點類似台帳的功能，詢問 Austin 後應該是鋪貨是 SAP 的 base 基礎，而台帳是擴充我們需求的不足(猜測的，日後確認會再補充說明)。

因為目前外圍的 SCM 有新品提報功能，所以目前大部分的商品資料，都是透過 CRM 提報後，自動匯入 SAP，程式會自動處理鋪貨作業，並不需要人工建檔。此處的細節待確認。

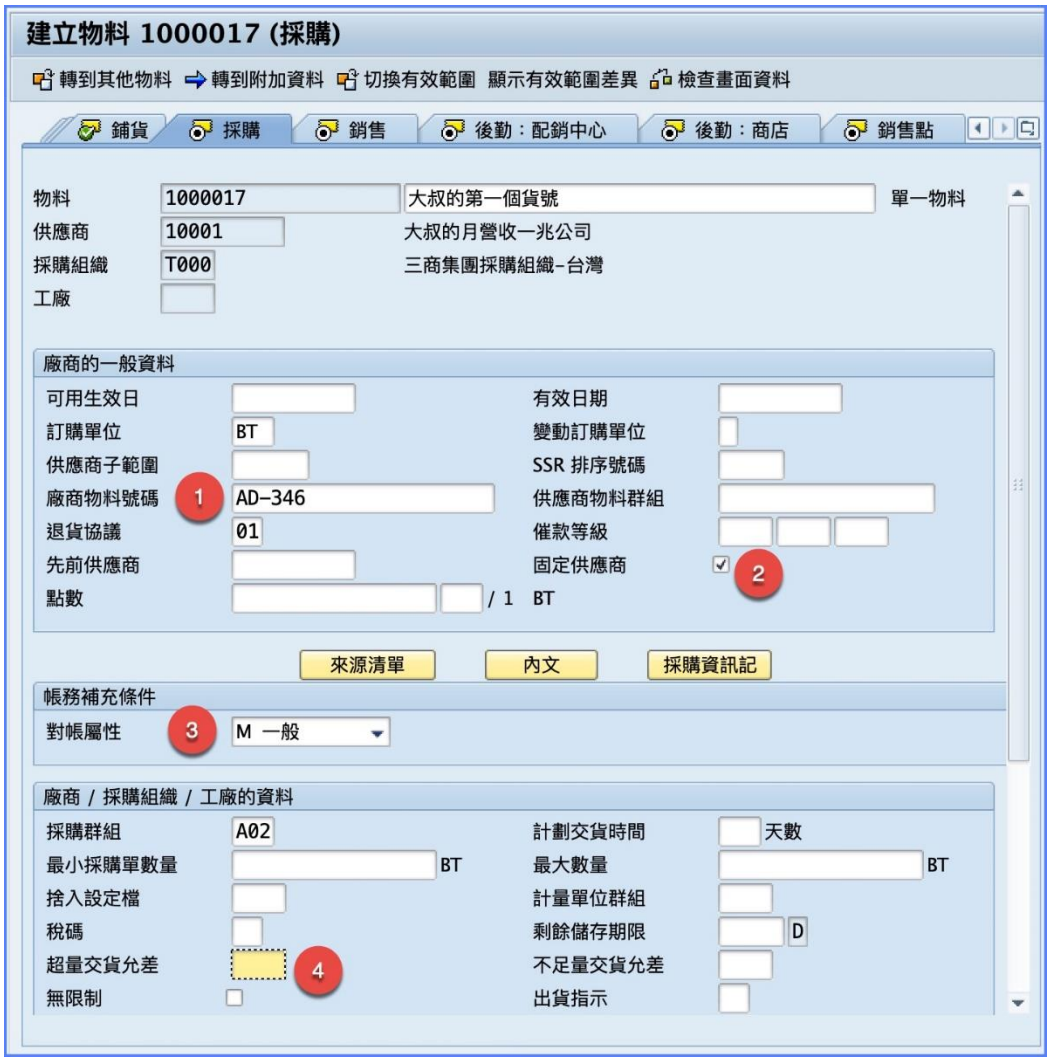
① 「鋪貨程序」請輸入 02，「儲存」和「配銷中心」的鋪貨程序都要輸入。

② 勾選「手動維護分類」及「新列示核查」，按 執行鋪貨列表 。

③ 跳出「複製鋪貨到物料指派」畫面後，按 複製 即完成鋪貨。



5. 採購頁面輸入商品的採購資訊，包含供應商、對帳屬性(寄售或月結)、採購群組、退貨協議、進貨價格。



① 「廠商物料號碼」輸入供應商貨號或型號；「供應商物料群組」則輸入供應商的商品分類。

② 商品只能有一個「固定供應商」。若同一個商品在不同據點有不同主供應商，則必須維護貨源清單。

③ 維護寄售或月結供應商。

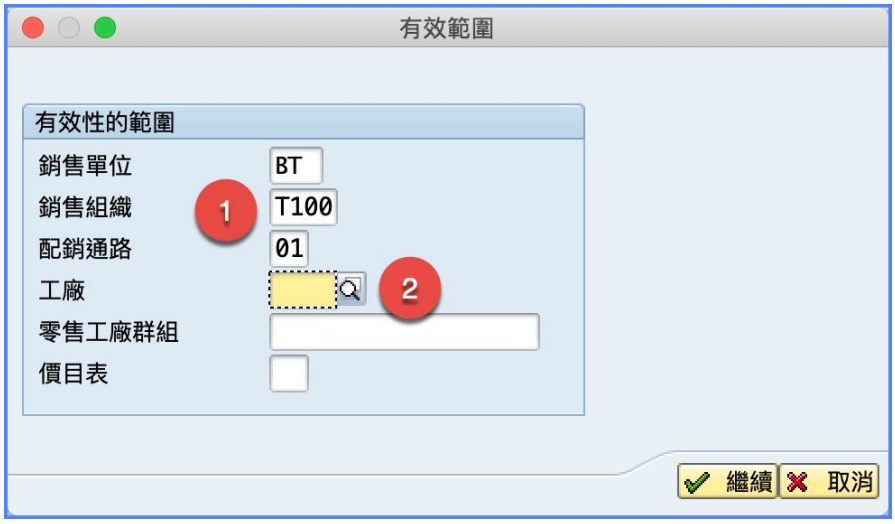
④ 若商品允許超收或短收，可在「超量交貨允差」及「不足量交貨允差」維護百分比。

⑤ 輸入淨價，也就是未稅採購價。要特別注意，採購價格的單位預設為庫存單位。

⑥ 如果為進口商品，可輸入國貿條件，按 enter 後，系統會跳出條件類型選擇視窗，請把游標點在 PB00 採購價，再按 確認。

6. 銷售頁面主要在輸入售價及銷貨收入科目

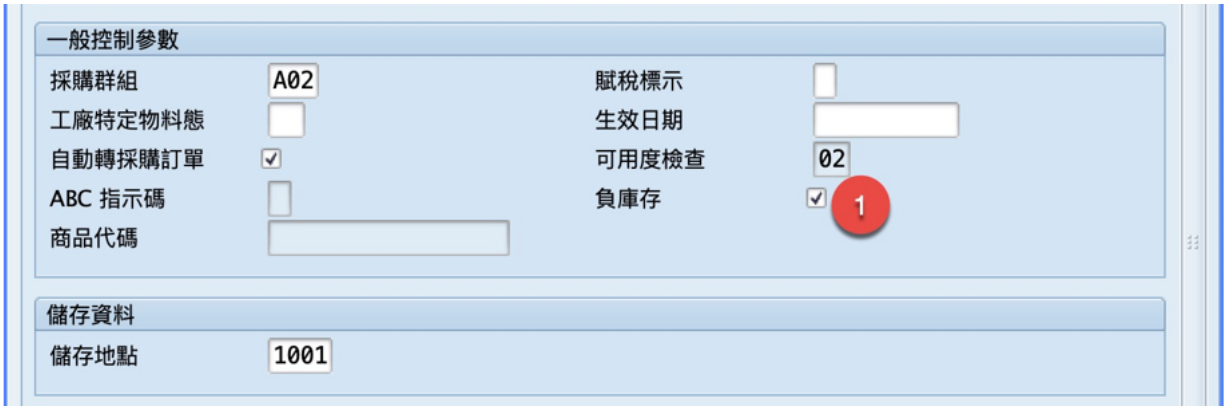
關於售價的定義，因為零售業的商店端，一定有 POS 系統，而 POS 系統一定有自己後台的促銷模組(或其他類似的系統)，所以此處的設定，應該是較標準的建議售價、或是非 POS 的銷售策略。這個也是跨系統整合的難點，畢竟要串不同的系統，有時難免會水土不服。



7. 計算售價，確認價格後，務必勾選「儲存計算的價格」，按 Enter 即完成售價設定，並進到下一頁。如果沒有勾選「儲存計算的價格」，試算的售價就不會儲存。

8. 「後勤：配銷中心」主要在控制物流及央廚的補貨相關資料

9. 「後勤：商店」在控制門市的補貨相關資料，主要是門市允許負庫存。



10. 「銷售點」頁主要在控制門市端的作業行為。

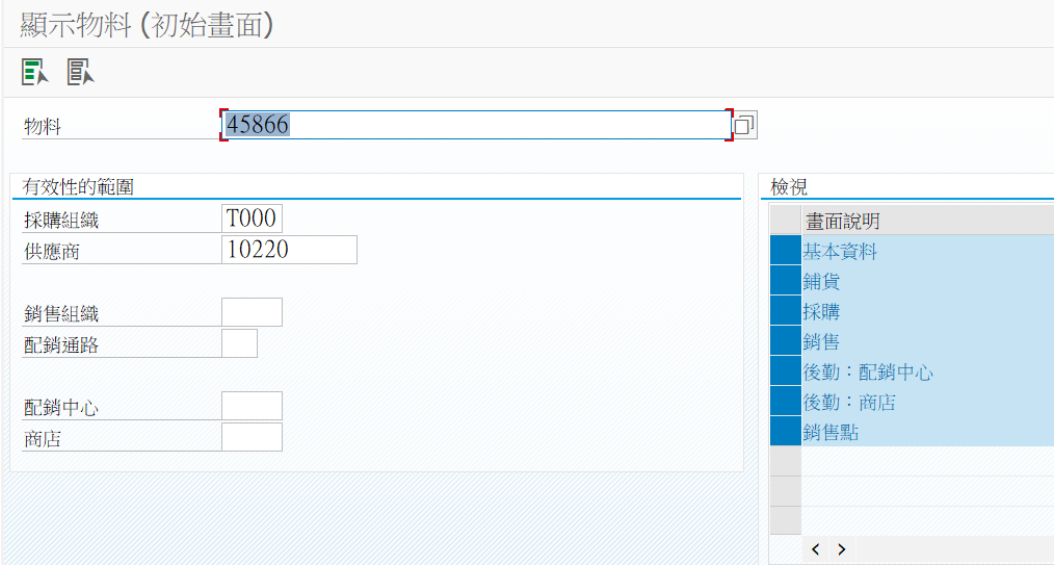
① 定義 POS 是否允許折扣、強制變價，以及各品牌負責的採購人員。

② 控制各品牌的作業行為及起迄日期。

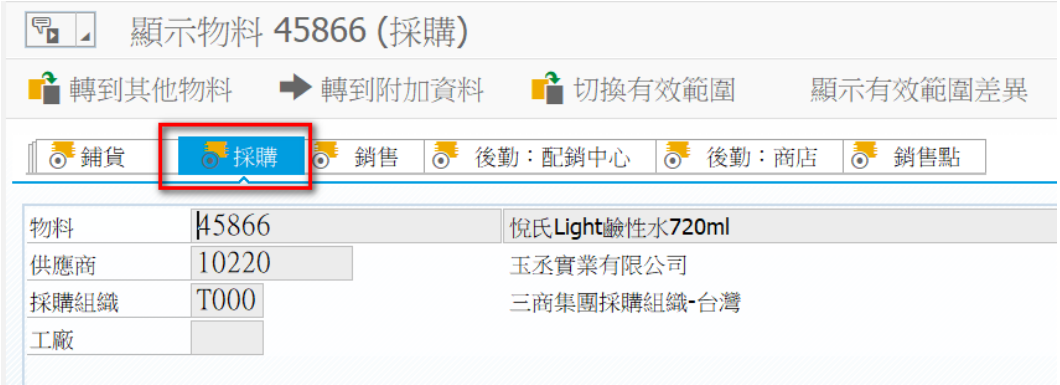
③ 「收銀機收據內文」若有維護，POS 的發票上會列印這個品名。若沒有維護，則會列印畫面最上方的正常品名。

11. 經過以上步驟，依序輸入個欄位內容後，即可存檔。

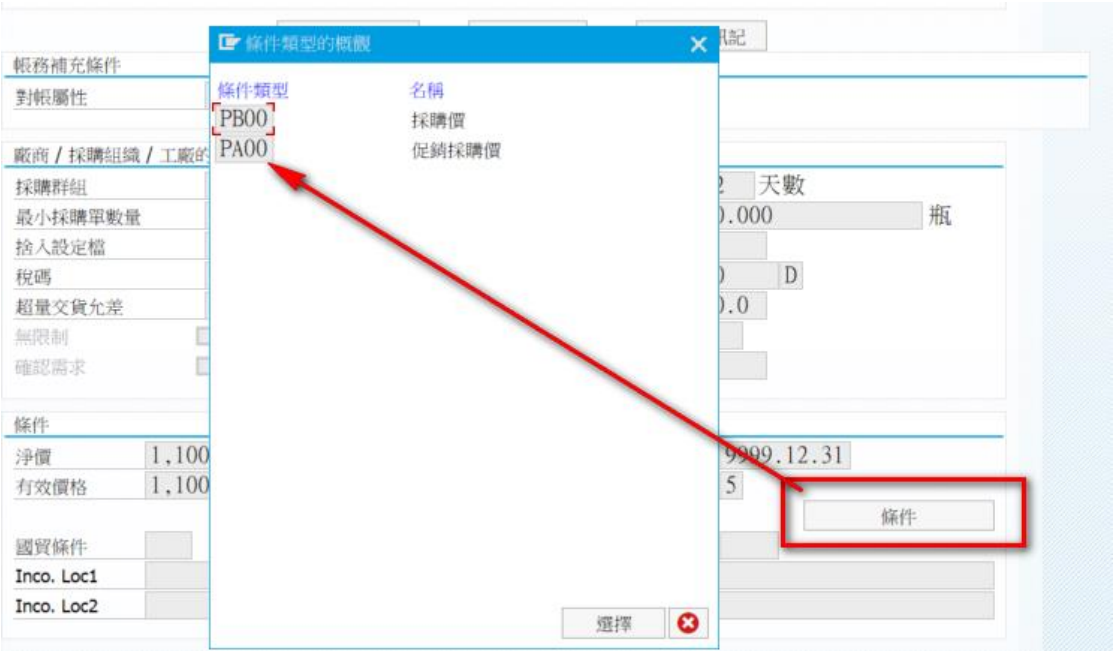
➢ 查詢 MM43 的採購促銷相關金額設定

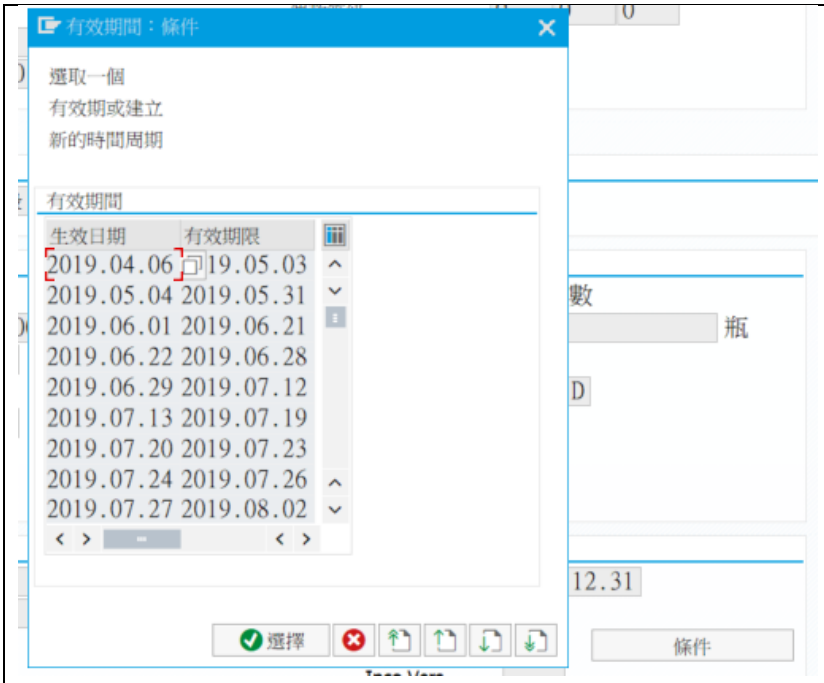


● 選擇 [採購] 頁籤

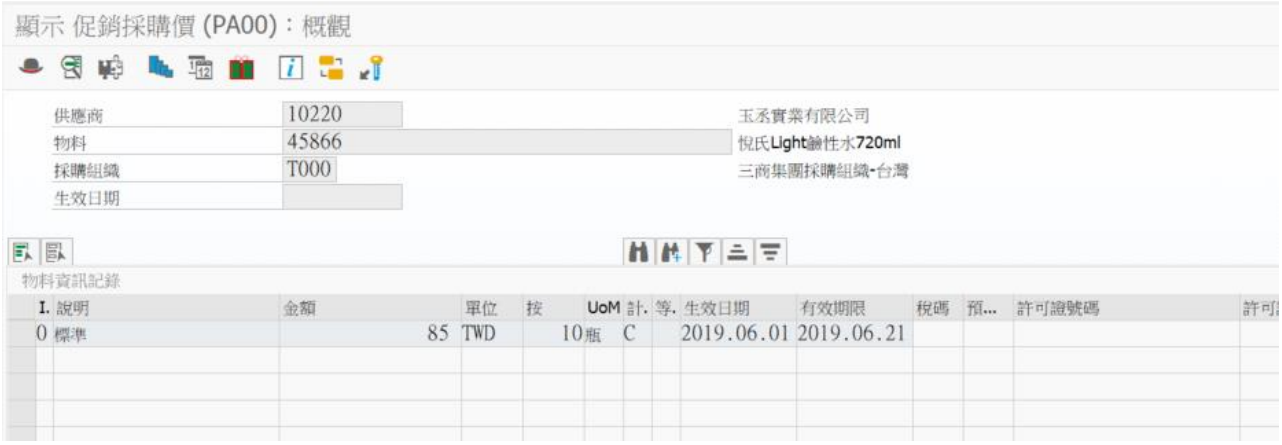


● 點選 [條件] button，並在 (PA00) DbClick





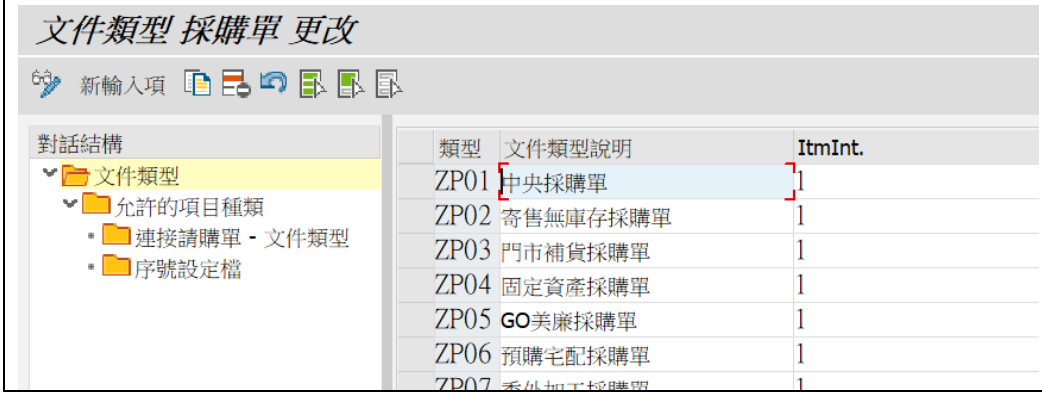
- 在某日期區間 DbClick，會另開視窗顯示相關的採購促銷設定價格



3	採購文件建檔作業(單據維護) ME21N　建立採購文件（包含採購單、退貨單、調撥單） ME22N　修改採購文件（包含採購單、退貨單、調撥單） ME23N　顯示採購文件（包含採購單、退貨單、調撥單） ME2N　查詢採購文件及收貨狀況 MIGO　採購文件收貨、發貨及退貨
	特殊欄位重點說明

本文定義的採購文件，包含採購單、退貨單及調撥單三種，調撥單的建檔的部分內容和前 2 者稍有不同，先說明採購單的建檔步驟。設定的 T-Code 如下

➢ SPRO-物料管理-採購-採購單-定義文件類型 / 定義號碼範圍(單號編碼原則)



下面是參考的單據類型(Z 開頭都是複製修改而來)

說明	單據類型	種類	號碼起	號碼迄
中央採購單	ZP01	PO	1100000001	1199999999
客訂採購單	ZP02	PO	1200000001	1299999999
門市補貨採購單	ZP03	PO	1300000001	1399999999
固定資產採購單	ZP04	PO	1400000001	1499999999
Go 美廉採購單	ZP05	PO	1500000001	1599999999
預購宅配採購單	ZP06	PO	1600000001	1699999999
帳上進貨單	ZP99	PO	1900000001	1999999999
中央指定退貨單	ZR01	RPO	4100000001	4199999999
物流退貨單	ZR02	RPO	4200000001	4299999999
門市退貨單	ZR03	RPO	4300000001	4399999999
Go 美廉退貨單	ZR05	RPO	4500000001	4599999999
帳上退貨單	ZR99	RPO	4900000001	4999999999
中央指定調撥單	ZS01	STO	6100000001	6199999999
客訂調撥單	ZS02	STO	6200000001	6299999999
門市補貨調撥單	ZS03	STO	6300000001	6399999999
店對店調撥單	ZS04	STO	6400000001	6499999999
門市退物流調撥單	ZS05	STO	6500000001	6599999999

➢ 新增採購單

① 建立採購單

② 選擇單據類型。請只選擇 Z 開頭的單據類型，ZP 開頭為採購單，ZR 開頭為退貨單，ZS 開頭為調撥單。本章節為建立採購單及退貨單，所以請選擇 ZP 或 ZR 開頭的單據類型。

③ 輸入供應商。

④ 點選「組織資料」頁籤，輸入採購組織 T000、採購群組及公司代碼。公司代碼可先不輸入，系統會自動代入採購單表身第一筆的據點所屬的公司代碼。也就是說，採購單不可以跨公司。

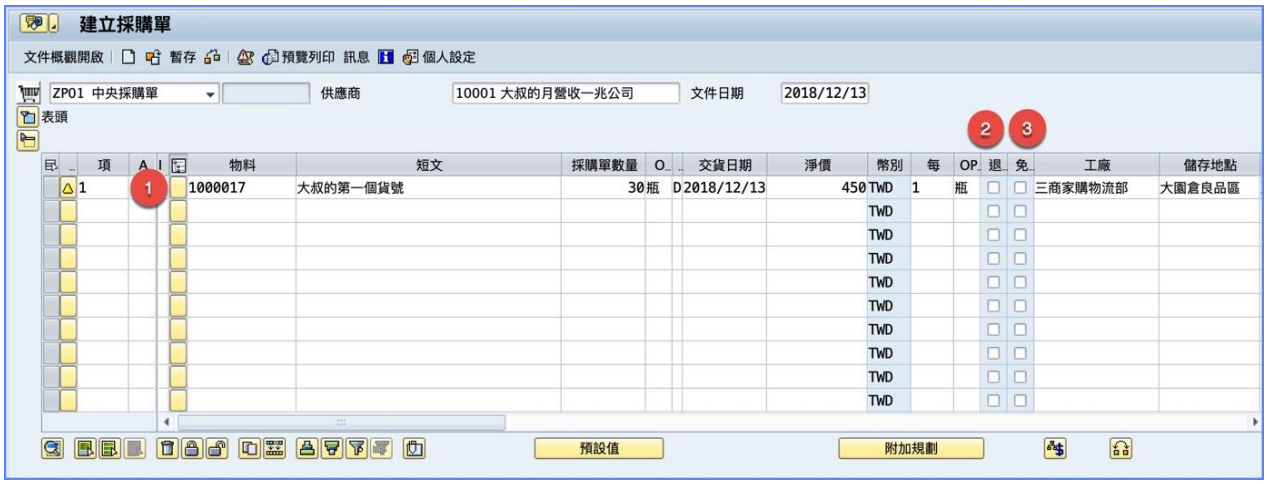
如果覺得每次都要輸入採購組織、採購群組及公司代碼很煩瑣，可以為帳號設定預設值。

➢ 輸入表身資料。如果嫌表頭太佔畫面，可以按 收縮表頭，把表頭縮起來。

① 原則上，表身輸入商品編號（物料）、採購單數量、據點（工廠）即可。物流須填寫儲存地點。

② 如果選擇退貨類的單據類型，退貨旗標會自動打勾。

③ 如果供應商有贈送商品，則在免費旗標打勾，該項次即會變成免費商品。輸入完畢按 enter，系統會自動帶出採購價格、收貨儲存地點等資訊。



➢ 如果要修改採購價格，可以點選畫面下方的項目，選擇「條件」頁籤。

① 「項目」請選擇要更改或查詢的項次。

② 在條件類型「 PMP0 手動價格」輸入新的採購價格。這裡的價格為單價。採購單價格只能改低，不允許改高。更改過後的採購價格，只會存在這張採購單，不會影響其他採購單，也不會更新商品主檔的採購價。退貨單的更改價格則無限制。

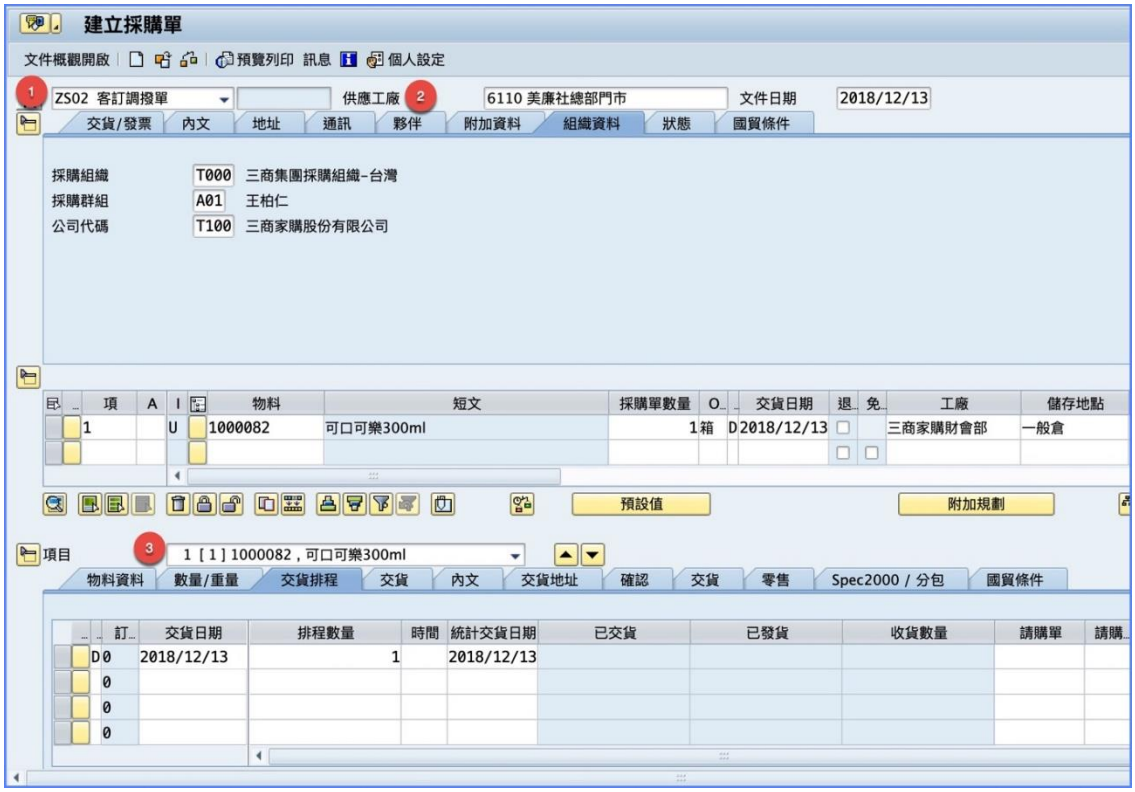
➢ 如果沒有問題，按存檔即可。

➢ 調撥單和採購單很類似，只有下面幾點注意事項，單據流=交貨→發貨→收貨→發票驗收

A. 單據類型：請選擇 ZS 開頭的單據類型。

B. 供應工廠：表頭的供應商會變成供應工廠，也就是發貨的門市、物流或央廚。

C. 沒有採購價格。調撥單沒有採購金額；調撥單的移轉金額，一律是供應工廠發貨當下的商品成本。

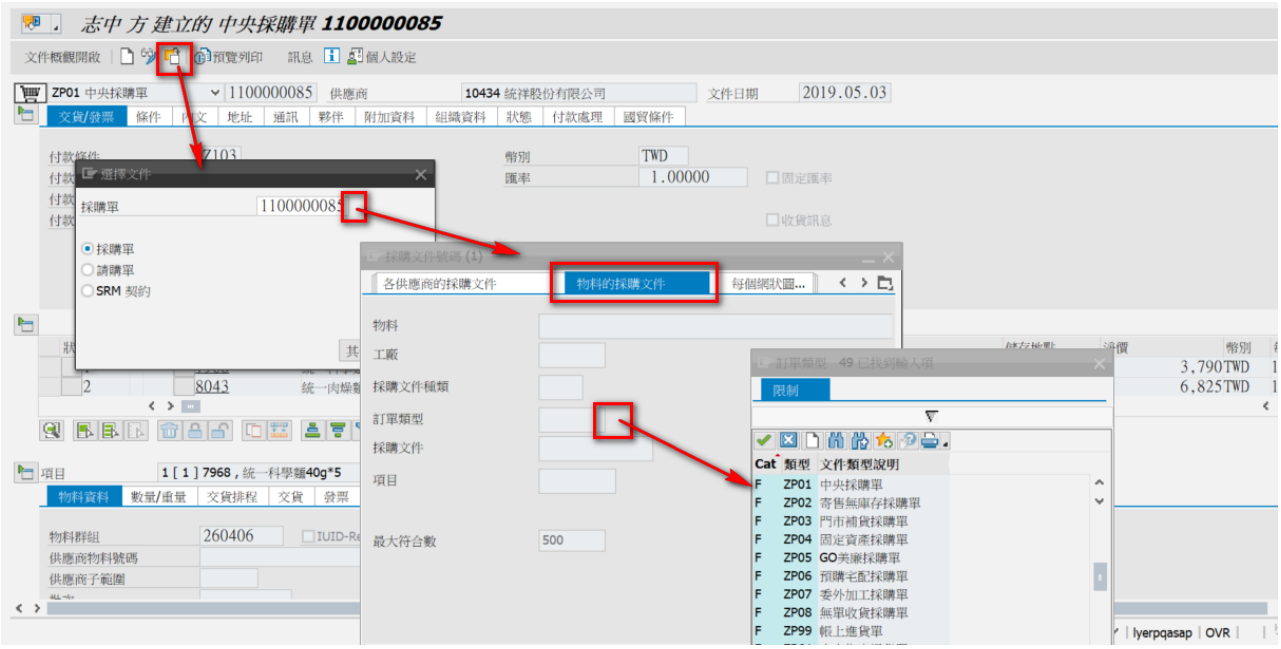


➢ 採購單收貨及退貨單退貨

基本上操作大同小異，注意都是統一使用 MIGO 即可。

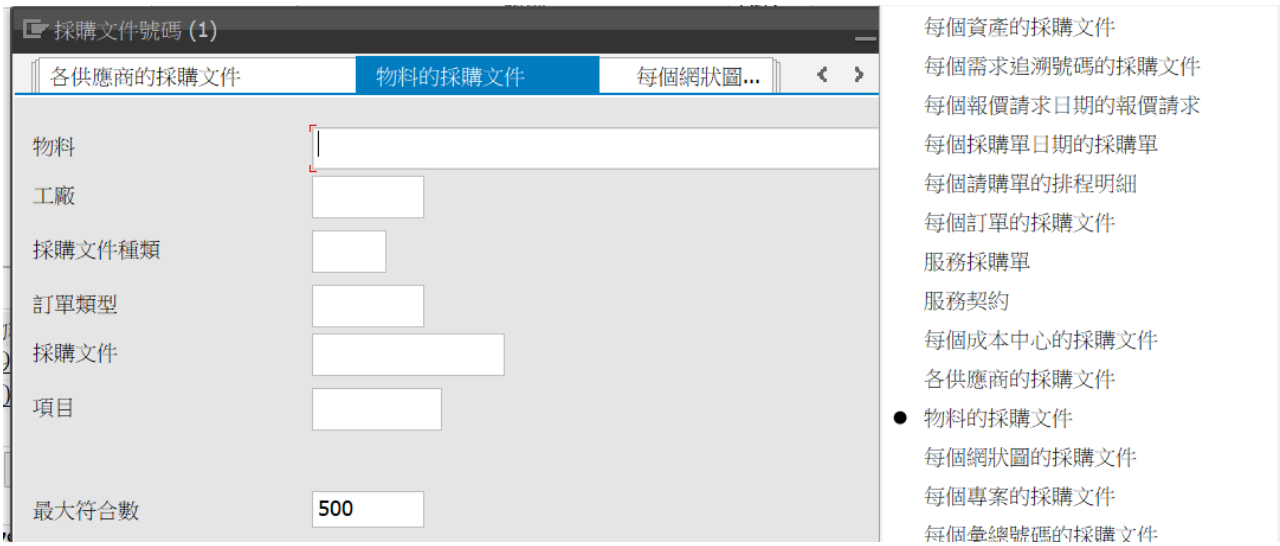
➢ 關於單據的查詢方式，請執行 T-Code=ME23N

按 Shift-F5 其他採購單按鈕後，可以查詢指定的採購文件單號。

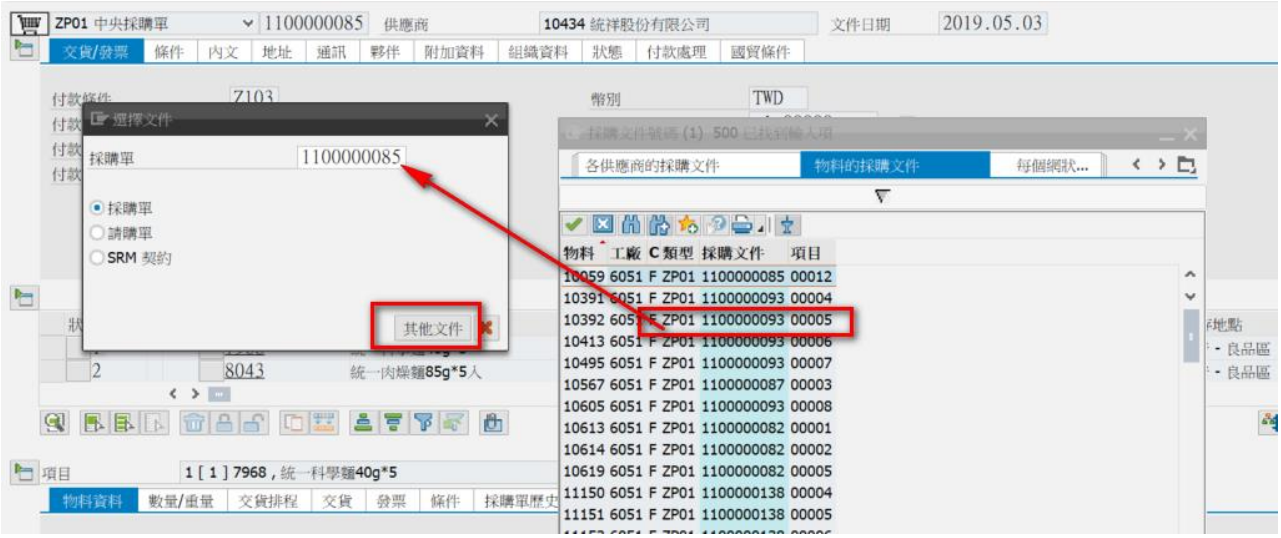


一開始操作這個功能時，最常會卡住的原因，通常是某些查詢頁籤(第三個步驟) 隨便下條件查詢時，常常都不會列表顯示，我猜可能是因為下的條件回傳筆數過多? 例如當我用前三個頁籤，只下單據類別為 [中央採購單] 時，都不會顯示資料，只有當我選擇 [物料的採購文件] 這個頁籤時，才會列表顯示。

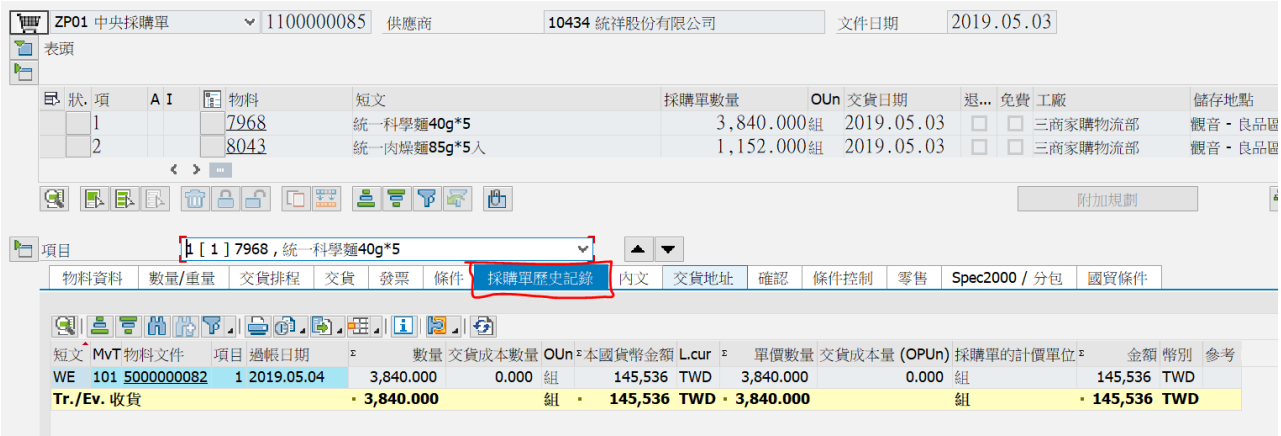
當然這是新手症候群，剛學習任何一套系統，還不熟練時都會有這個問題。只能多問多操作，盡快變成老鳥就不會有這些詭異的神奇操作疑問了。



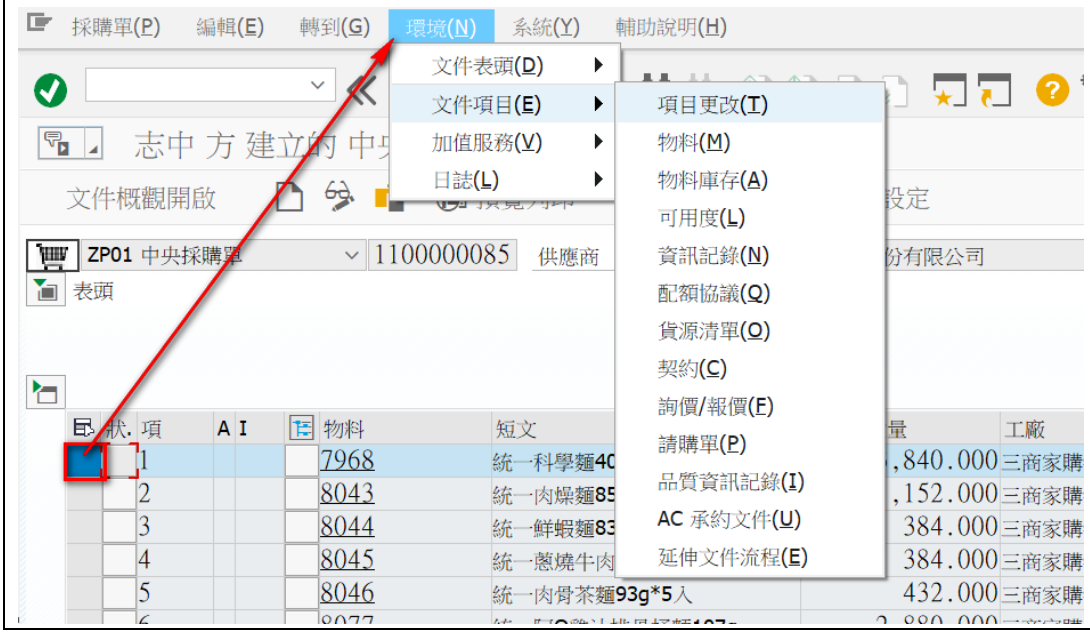
取回要查詢的單號後，按 [其他文件] 即可。



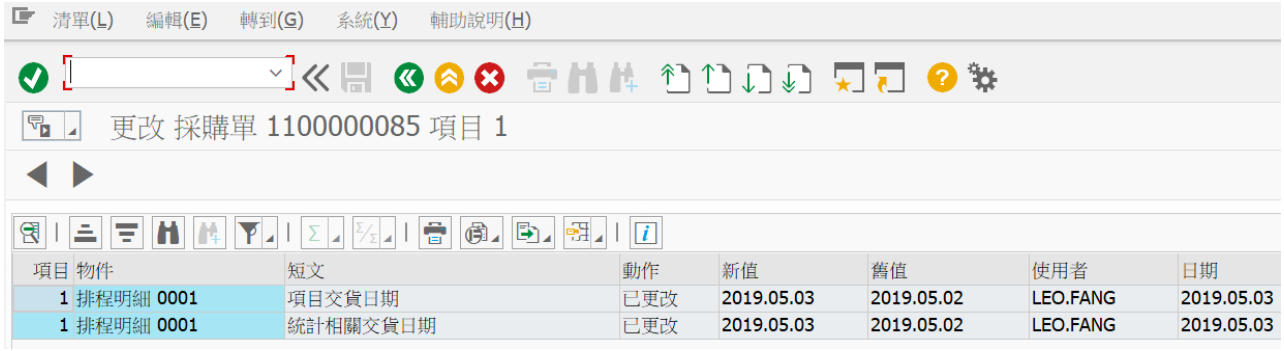
➢ 在 [項目] 中的 [採購單歷史紀錄] 是非常經常使用的查詢功能。詳細的說明會在 Chap05 說明。



➢ 查詢單據表身的異動過程紀錄(就是 Log 明細紀錄)

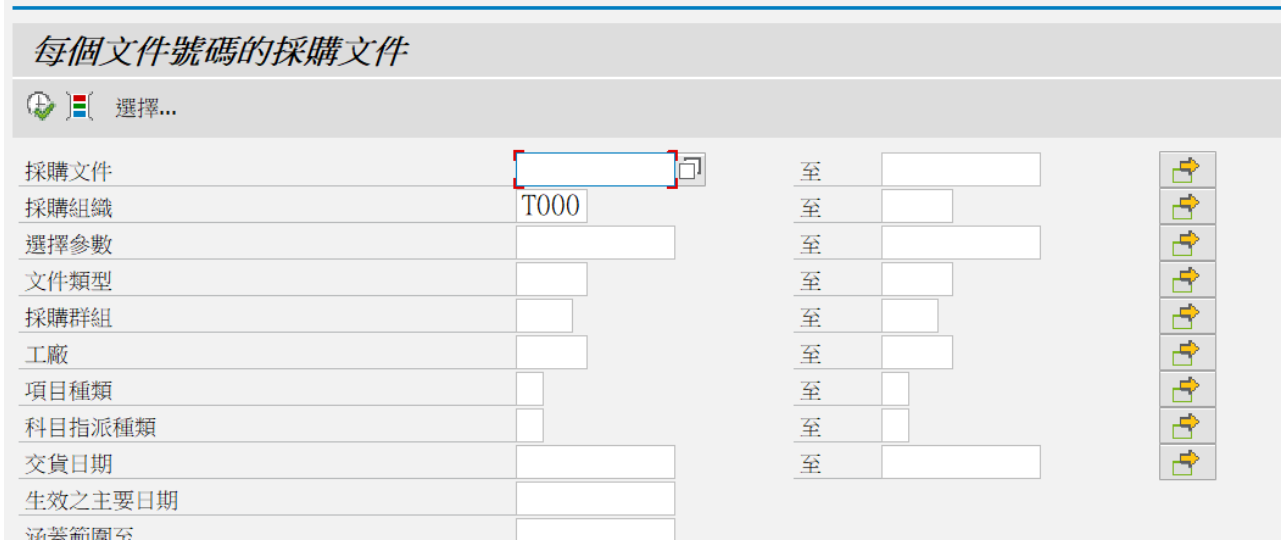


按下 [項目更改] 的 button 後，就會出現如下的 UX，會顯示該張單據的所有異動 Log 紀錄。

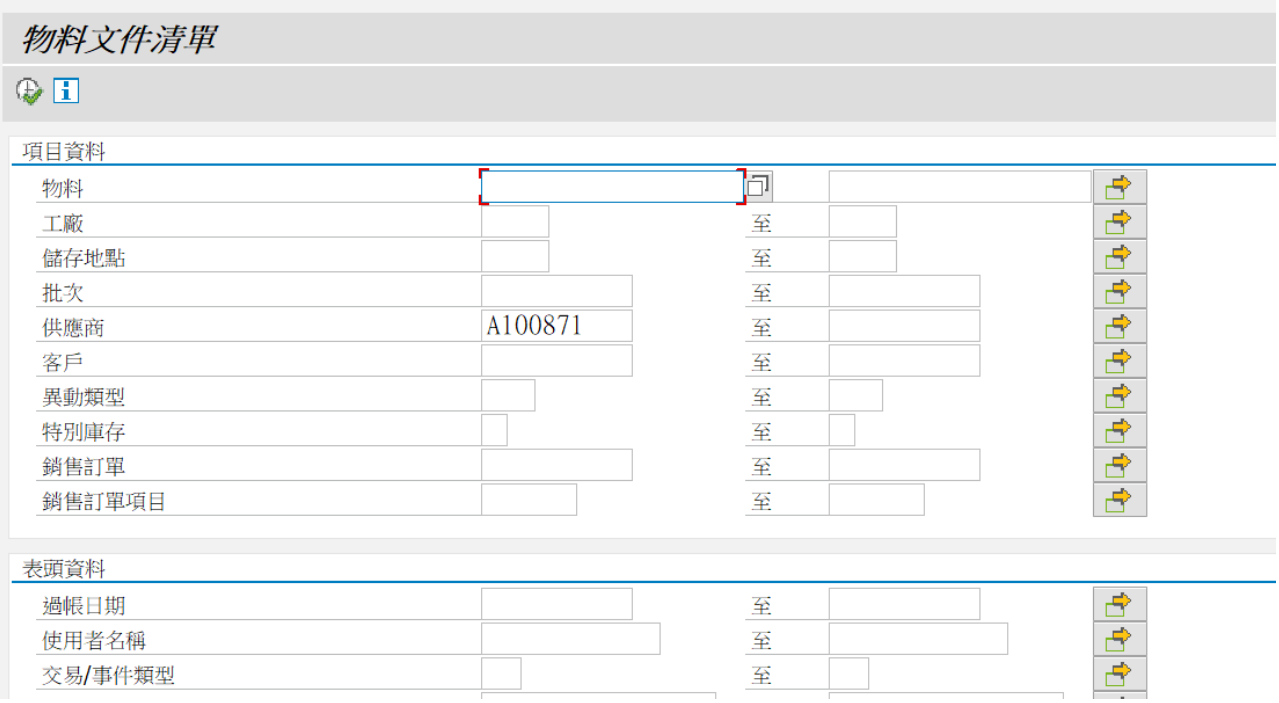


項目	物件	短文	動作	新值	舊值	使用者	日期
1	排程明細 0001	項目交貨日期	已更改	2019.05.03	2019.05.02	LEO.FANG	2019.05.03
1	排程明細 0001	統計相關交貨日期	已更改	2019.05.03	2019.05.02	LEO.FANG	2019.05.03

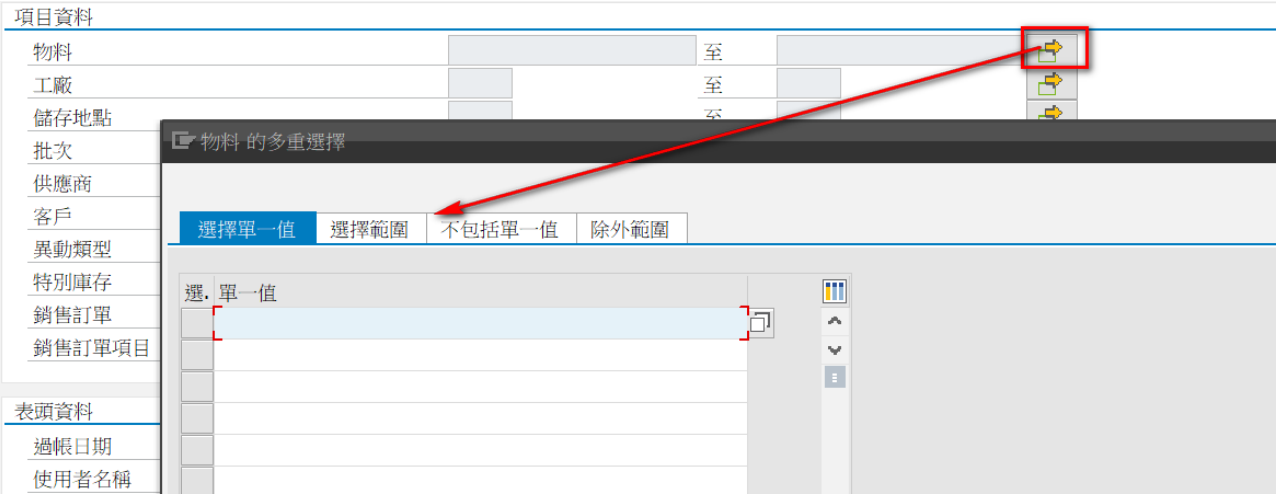
➢ 另外，也常會使用 ME2N 報表查詢，關於報表的查詢，我們統一在後面獨立單元說明。



4	T-Code=MB51 現有庫存報表查詢
	特殊欄位重點說明



- 基本上，SAP 的查詢報表基本上都長同一個模樣，每個查詢條件後，都可以設定多選條件，可以利用四種不同的頁籤設定要查詢的內容，這個通用介面基本上是可以設定出絕大多數的 user 需求。



第五章 MM 一些重要的單據流程及概念說明

本章我想要仔細的探討 SAP MM 模組中採購主流程的相關單據資料流及一些重要的觀念。

在 ERP 系統中，一般的採購流程至少會包含請、採、驗這 3 個步驟，以下分別說明

1. 請：請購或稱採購申請。T-Code=ME51N

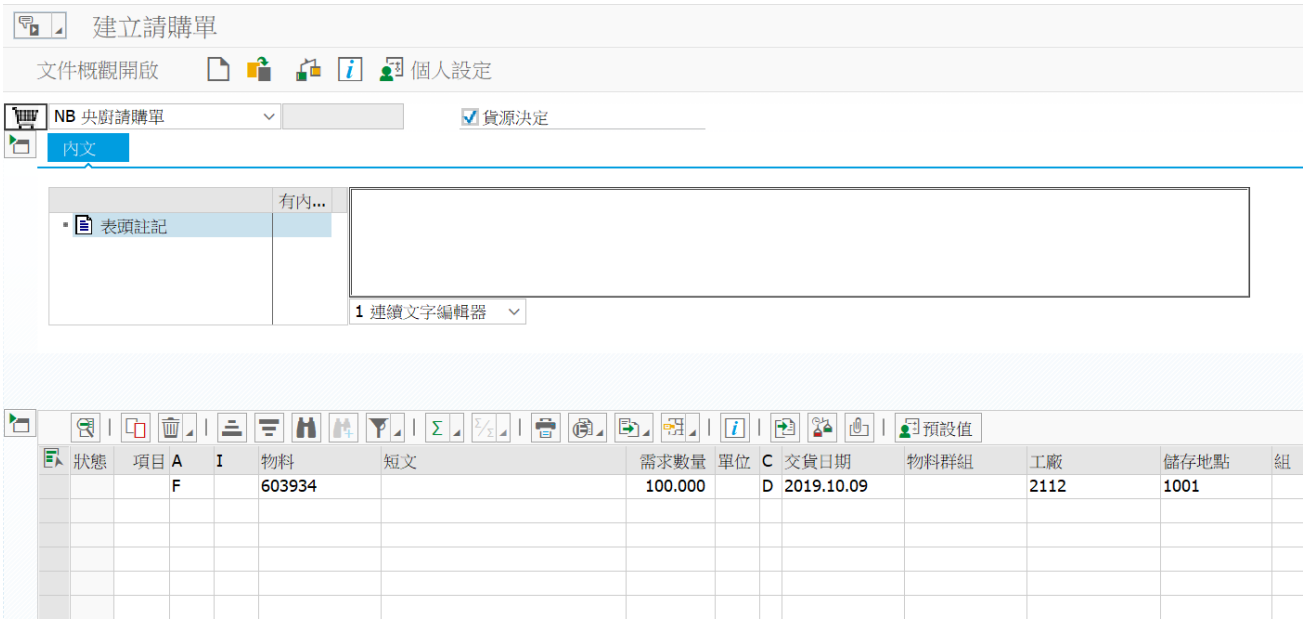
   採購申請基本上是採購部門的願望清單或要求或材料，採購部門在收集公司的所有請購單後管理採購。

   它是一份內部文件，不與供應商共享，需要內部採購組織的批准。

2. 採：採購(採購單位發出 PO)
3. 驗：驗收(倉庫貨物流收貨入庫)

SAP 的核心模組也大致遵循這個邏輯。只是在實作上，有更多的細節要注意。為了避免重複輸入導致的重工或錯誤，系統都會提供轉單功能方便使用者操作。所謂的轉單，事實上就是在一個完整的資料流程中的多張單據，經由程式介面，快速而正確的產生下一個流程單據的作法。

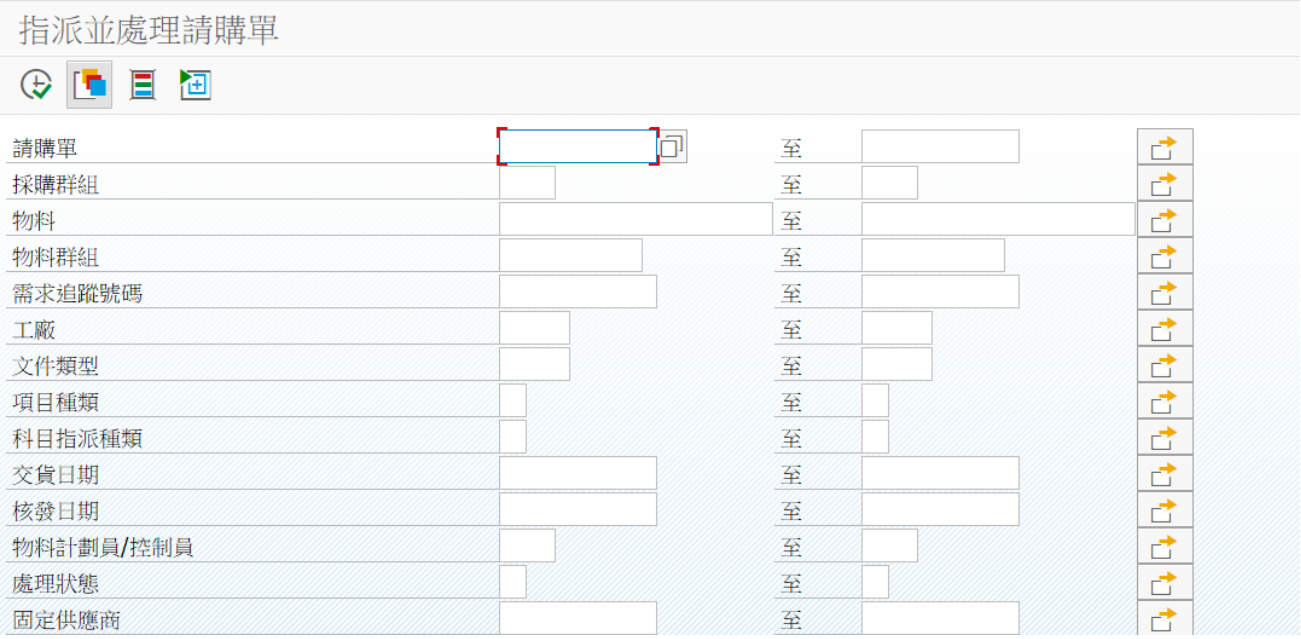
舉例來說，A、B、C 3 個生產單位輸入各自的請購單後，請購單上並沒有指定廠商及單價，而是單純的提出需要的 SKU 及數量(請參考下圖)。



此時，可以透過 MRP 或採購建議書等功能(當然必須先設定相關的採購策略、主供應商、最少採購量等等預先設定)，計算出當前所有請購商品的供應商及採購單價。然後批量的產出多張採購單。

Session 1：在 SAP 中，有 2 個方式可以將 PR(請購單)轉 PO(採購文件)

- 手工轉：ME57 分配貨源（供應商），處理採購申請，由 PR 產生 PO



- 自動轉：1.物料主檔，供應商主檔維護自動產生 PO，PR 指定貨源，ME59N 轉 PO

- 目前在三商家購並沒有使用請購單的功能，所以針對本功能的細項先不深究，先知道大概的流程即可。日後如果有需要，再來補充此部分的說明。
- 可以參考下面 blog 的說明 https://blogs.sap.com/2014/04/09/%E5%85%B3%E4%BA%8Epr%E8%BD%ACpo%E7%9A%84%E6%B3%A8%E6%84%8F%E4%BA%8B%E9%A1%B9/

Session 2：從採購文件(PO)到物料文件(MIGO)

採購文件，包含採購單、退貨單及調撥單三種主要單據類別。而在 SAP 中 MIGO(物料文件)是貨物移動的意思。貨物移動（也稱物料移動或庫存移動）是指一引起庫存改變的事件。

MIGO 類型總共有下列 7 種

A1	收貨
A2	退貨
A3	取消(不明用途)
A4	顯示(不明用途)
A7	發貨
A8	轉讓(不明用途)

SAP 系統允許如下類型的貨物移動：

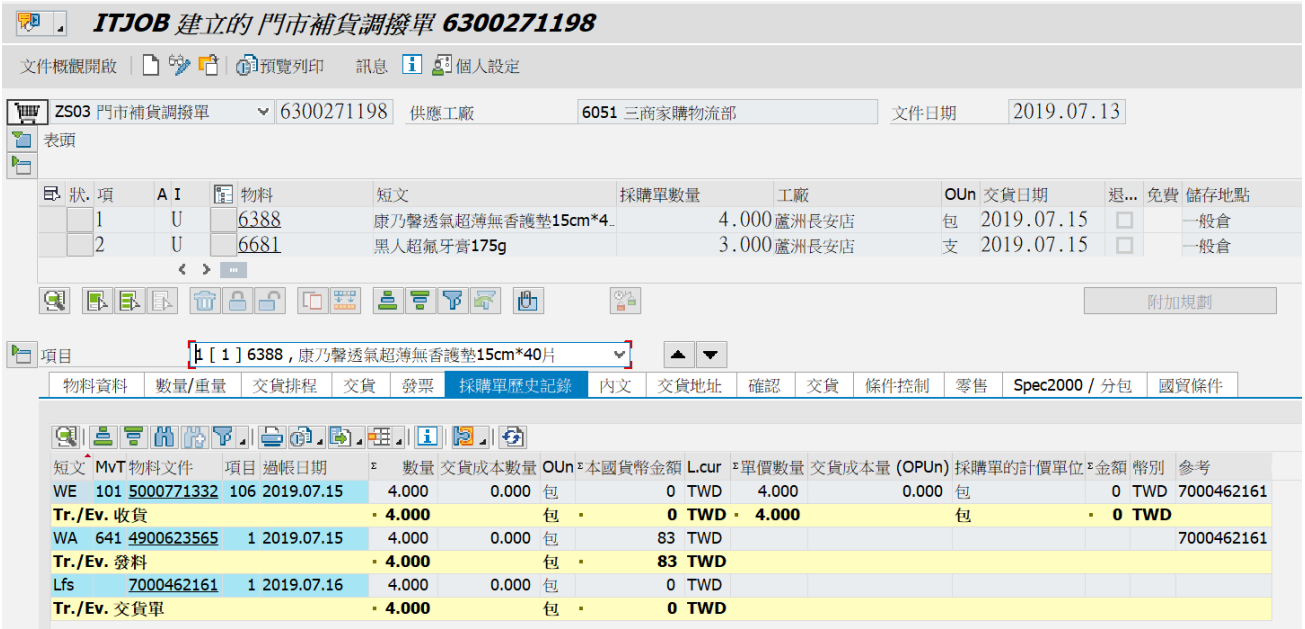
1、收貨（GR）是一貨物移動，通過它記錄來自供應商或來自生產的貨物的接收。收貨將導致倉庫庫存的增加。就是我熟悉的入庫(進貨)作業。

2、發貨（GI）是一貨物移動，通過它記錄物料提取或物料發出，物料消耗，或給供應商的貨物裝運。發貨將導致倉庫庫存的減少。就是我熟悉的出庫(出貨)作業。

採購文件和物料文件，基本上的概念類似 PO 和進貨單(就是轉單作業)，只不過 SAP 包含的範圍更大。它將所有的單據，都分為 Plan & Action，也就是計畫某天某事，和實際上某天執行某事。

單別			
採購文件→採購單	收貨(轉進)	退貨(選項)→應付減項	發票驗證(應付)
→退貨單	退貨		發票驗證(應付減項)
→調撥單	交貨	發貨(641)	收貨(101)

所以採購文件是來源單(發動)，所有物料異動 MIGO 的相關作業，都會回寫，並可以在 [採購單歷史紀錄] 查詢

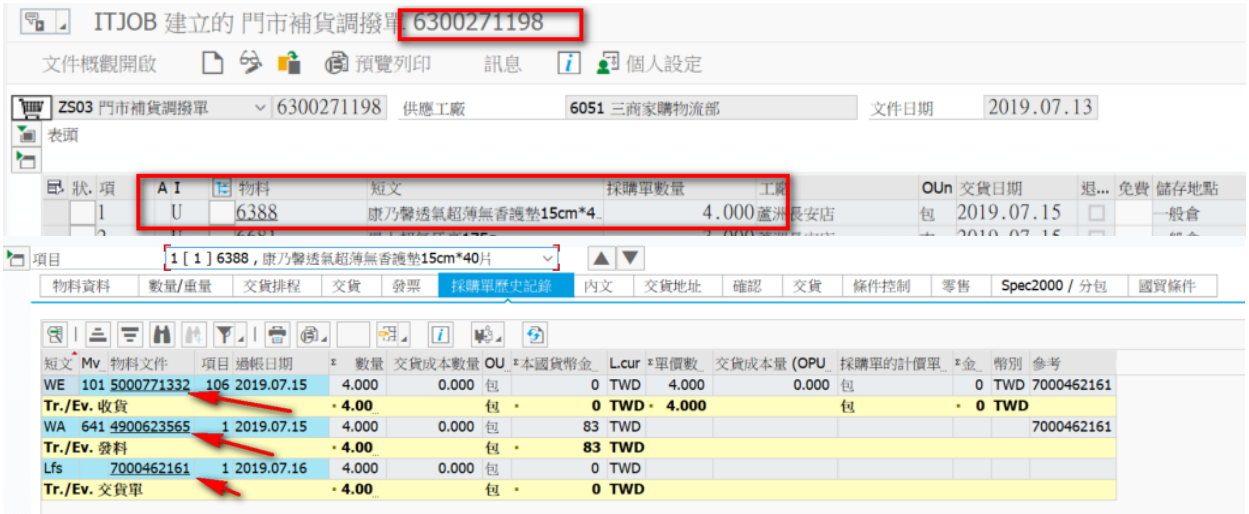


採購文件並不會立刻影響庫存，但是它是正式的對廠商的進貨要求文件，是應付帳款三單驗證的發起單據(企業是收到供應商貨物並成功質檢後才進行發票校驗→Tcode:MIRO，MIRO 要求供應商的送貨單,倉庫的收貨單和供應商發票相匹配才確定應付)。

而物料文件是真正的『貨物移動』類單據，才會真的影響庫存數量，是真的從某某倉拿出來(或入庫)，這是兩件事。所以如果是出貨作業，通常會有類似檢貨作業，或是要再執行另一個 T-Code，才會真正的扣庫，至於應收(付)的產生，就必須透過發票驗證作業後產生。應付也是類似的概念，這和先前小型的 ERP 有很大的不同，不過這才合理，之前我一直有疑惑，出貨單存檔就扣庫存、影響帳款，但是如果我想今天先打明天的出貨單，這樣會有問題，原先想過很多想法，例如覆核才過帳等，看起來都類似 SAP 的解決方案。不過對小企業來說，的確是有點麻煩。

➢ 下面我們用一張採購文件來說明完整的單據流程

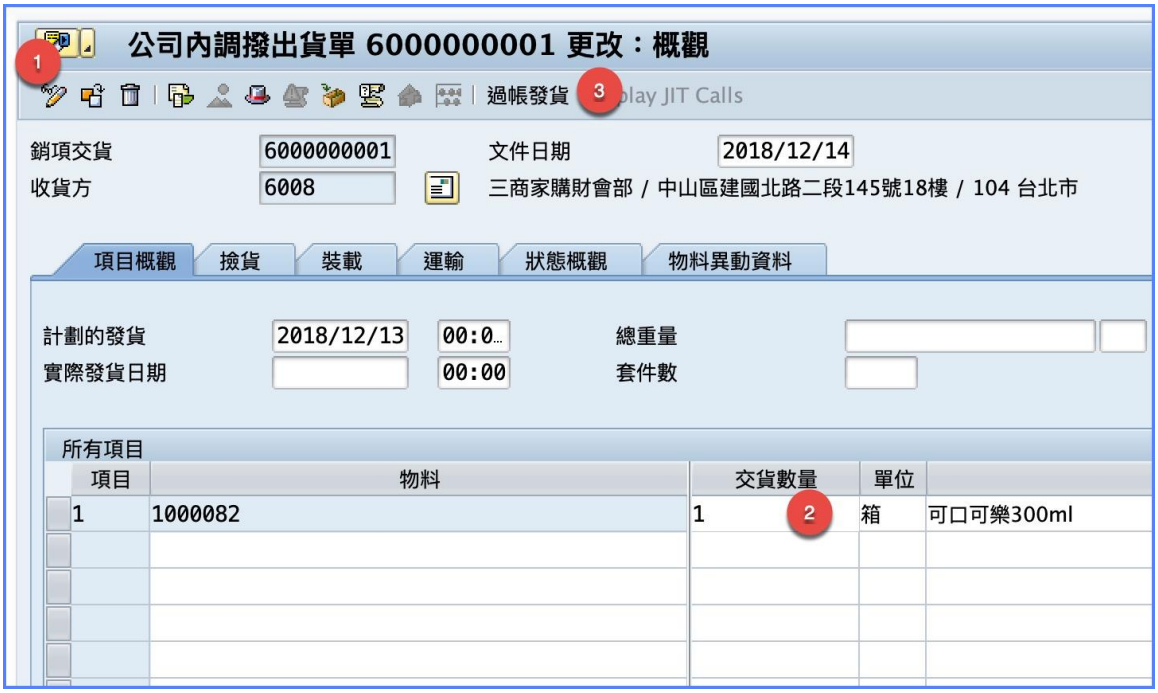
- 以 6300271198 這張採購文件來說，因為單據類型是調撥單，所以完整的流程應該要有交貨、發貨及收貨，請參考下圖，在[採購單歷史紀錄] 中，可以在物料文件一欄，看到 3 張不同單號的物料文件



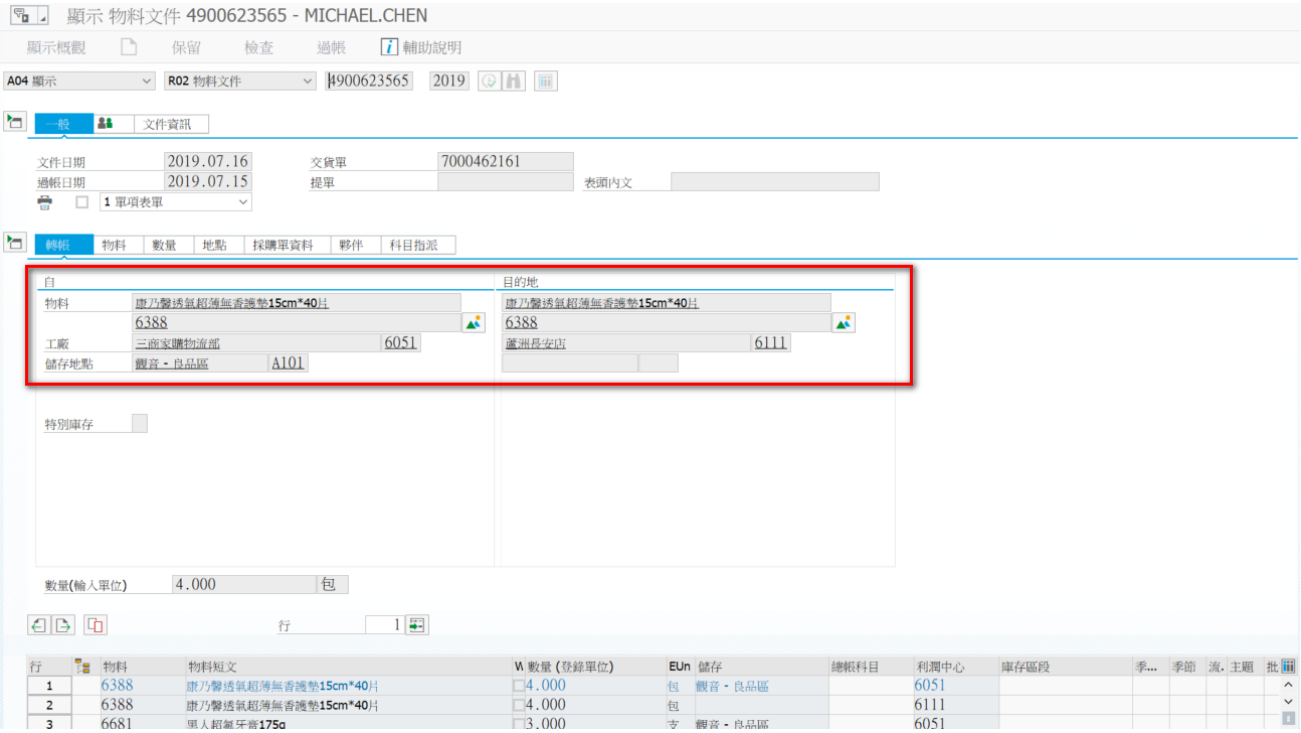
- 從物流到門店，首先，建立調撥出貨單(交貨單)，手工作業時，會使用 T-Code=VL10B 建立，輸入採購單號會自動帶出要出貨的商品內容，本作業不會影響庫存，單純紀錄預計調撥的資訊。產生單號=7000462161 **合理的推測，這個作業應該是由物流回傳(確認)的可出貨的資訊，這樣就合理了(這是我的推測)。**

- 交貨作業執行後的查詢(由 ME23N 歷史紀錄連結)，T-code=VL03N，Table=LIKP

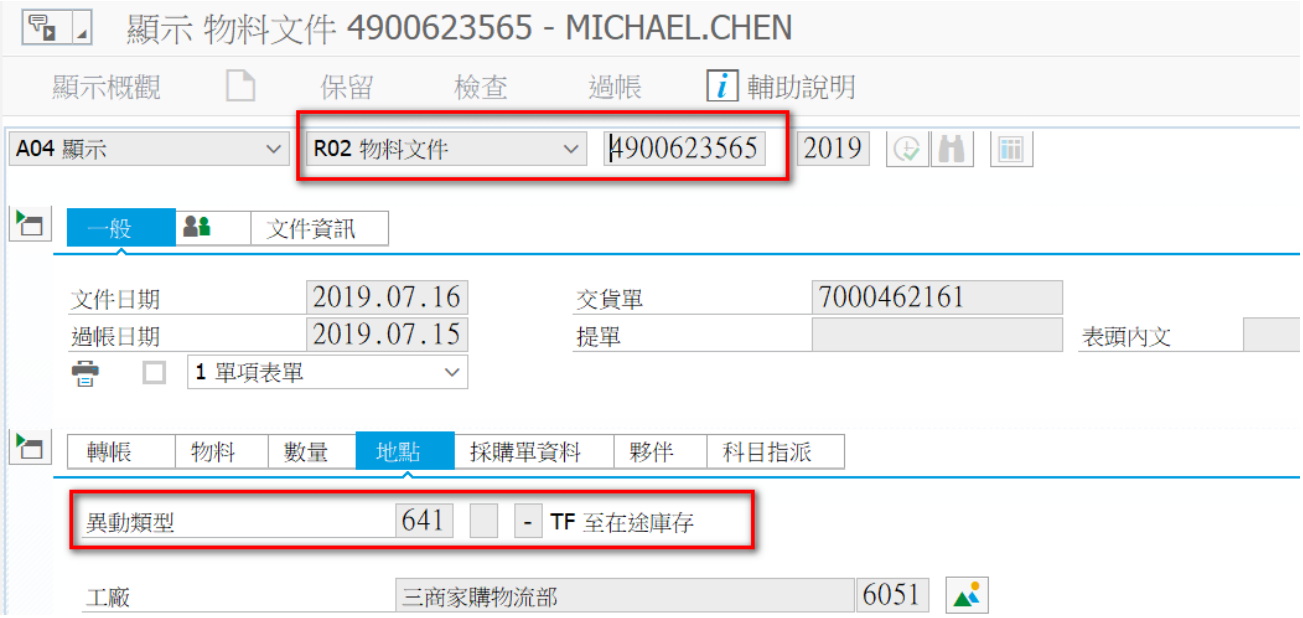
- 再來就是物流實際檢貨裝車後(發貨)，操作流程是先到 ME23N 找到該筆採購單的交貨資訊(調撥出貨單)→點選 [過帳發貨]，這個動作會產生 MIGO 的 641。此時這張單據上的 SKU 庫存會扣掉物流倉的庫存量，然後變成在途量，因為在貨車的運輸途中。



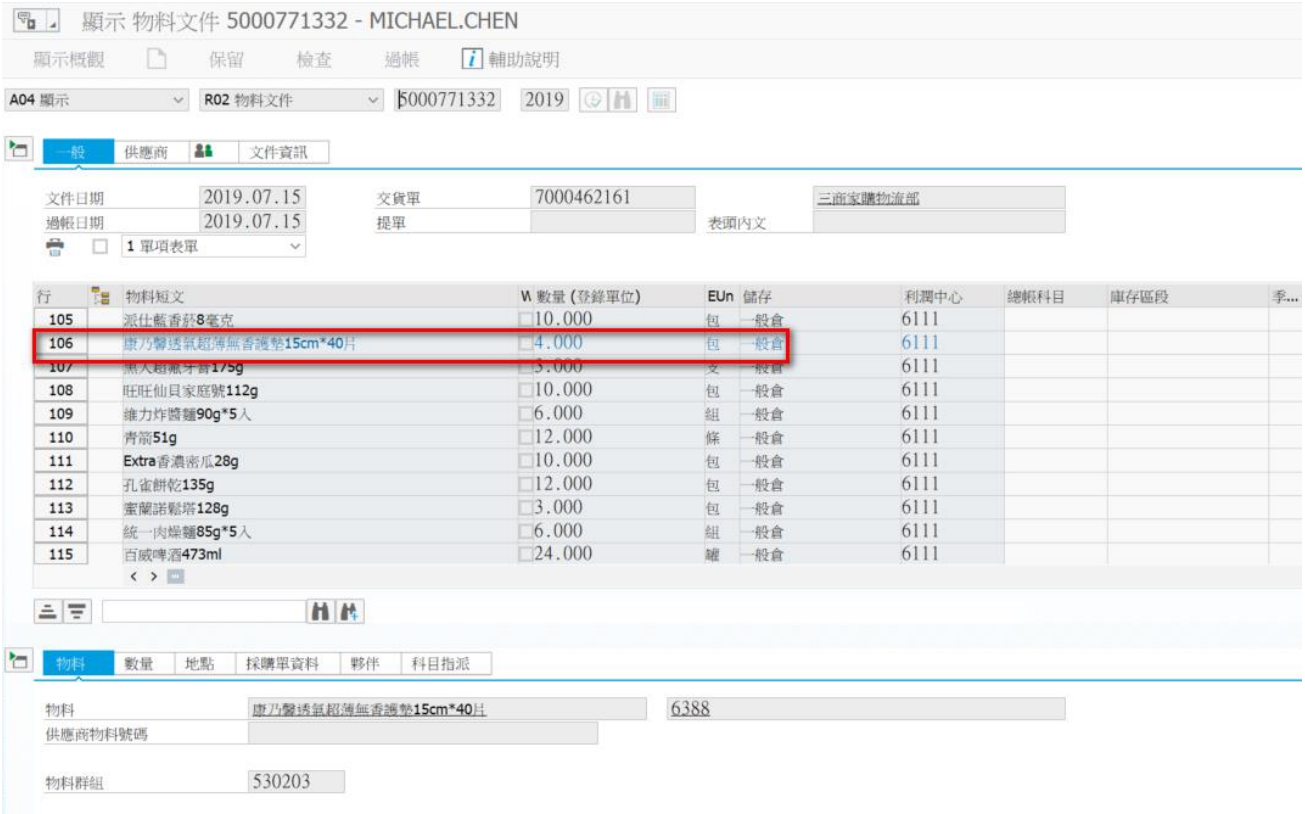
- 物料文件的內容查詢如下，會顯示撥出單位(物流)和收貨單位(門店)。



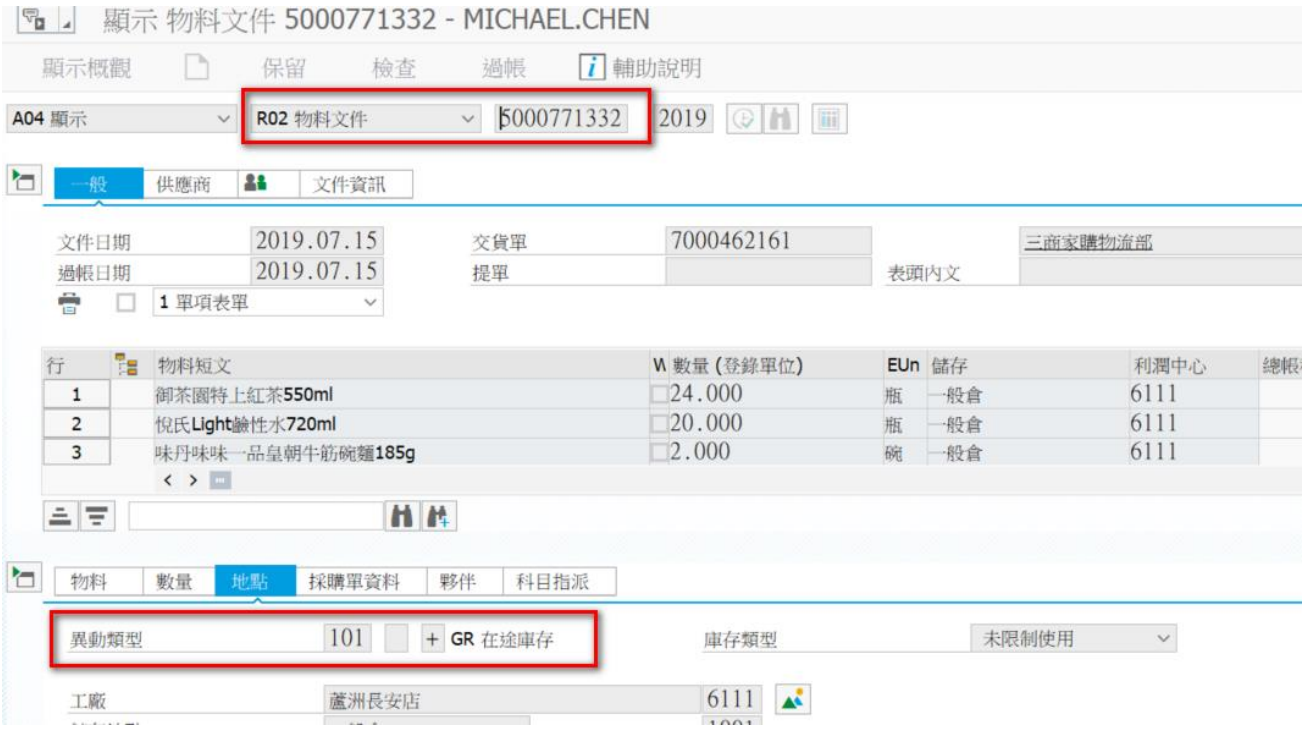
➢ 這是異動類別 641 撥出



- 最後，當門店收到貨，透過 PDA 或其他的作業方式輸入驗收資料，並產生收貨單後，SKU 庫存會由在途，改為門店的庫存。系統會判斷並檢查收貨數量不得大於撥出的數量。



➢ 這個動作會產生 MIGO 的 101。



這樣就完成了整個由物流到門店的所有交易單據的流程。對於交貨(VL10B)這個作業，我原先並不理解它的實際用途。發貨(641)和收貨(10) 都是物料文件 MIGO，這非常合理。但是在採購後→發貨前。要先產生交貨單，目的是?? 是因為物流在收到 PO 後必須先回復(確認)實際的可出貨資訊? 這樣推測還是有可能的。

- **Leo 顧問 mail 回復 Ben 關於物流調撥量差的邏輯和查詢方法(內容值得參考故摘錄之)**

只有物流出貨、門市收貨的調撥單，當物流出貨數量大於門市收貨數量時，差異數會自動彈回物流的量差倉。門市對門市調撥，或門市退貨給物流，都不會啟動自動彈回。

舉例來說，如果物流出 10，但門市只收 9，系統會幫門市補收 1 個，讓系統沒有在途庫存，然後再彈回那 1 個到物流的量差倉。→ Michael Add：如果是這樣，就不應該有關單的 issue 吧??? 因為每一張都會結單

補收的那一張物料文件，在表頭的參考欄位會帶原始交貨單號，並加註 AUTO 字樣，以辨別是系統自動補收的。

過帳日期	數量	交貨成本數量	OU	本國貨幣金	L.cur	單價數	交貨成本量 (OPU	採購單的計價單	金	幣別	參考
2019.07.01	1.000	0.000	瓶	0	TWD	1.000	0.000	瓶	0	TWD	7000364063/AUTO
2019.07.01	23.000	0.000	瓶	0	TWD	23.000	0.000	瓶	0	TWD	7000364063
	24.0		瓶	0	TWD	24.000		瓶	0	TWD	
2019.07.01	24.000	0.000	瓶	154	TWD						7000364063
	24.0		瓶	154	TWD						

為什麼不直接用 642 迴轉掉那 1 個，還要那麼麻煩先收 1 個，再用轉出 1 個？那是因為交貨單不允許部份迴轉，它只能整單迴轉。

自動彈回的異動類型為 301。301 是不需要單據的店對店調撥。當初我們設計只有三個狀況會使用 301 這個異動類型：

- ✧ 自動彈回
- ✧ 物流箱轉出到門市
- ✧ Office mart 的虛擬倉轉出商品到貨架倉

以上這三種異動理論上不會由人工操作，都是介面或程式產生，所以我在教育訓練時也沒教 301 的操作方式。只教 311，311 是物流把庫存從量差倉轉回其他倉庫使用。

4.8 儲存位置間移動商品（調撥差異數量轉回其他倉區，物流使用）.....21

至於為什麼現在系統除了上述三種狀況外，還有別人使用 301？因為上線初期，貴公司 IT 教央廚使用 301 將庫存調撥給門市，或從門市調撥給央廚，而沒按照原先設計的要先開調撥單，再用 641+101。這個我們沒辦法控制，畢竟系統交接之後，怎麼使用就是你們的權利，身為顧問我們只能提出建議。

Sent: Thursday, July 25, 2019 1:24 PM
To: 曹弘照 <vic.tsao@simplemart.com.tw>
Cc: 林雨韻 <Viola.Lin@simplemart.com.tw>; 洪巧盈 <claire.hung@simplemart.com.tw>; 康建智 <leo.kang@simplemart.com.tw>; 林欣怡 <megan.lin@simplemart.com.tw>; Jasmine Chiu <Jasmine.Chiu@nttdata.com>; Kufeng.Lin <Kufeng.Lin@nttdata.com>
Subject: Re: 請增加調撥方式

Dear Vic,

後來想想，我覺得需要再重新說明一次

任何調撥，不管店對店或物流對店，或物流對央廚
都應該要有一張申請單，就是所謂的調撥單
出貨方憑著這張調撥單撿貨出貨
收貨方憑著調撥單和出貨單收貨
這是正常的流程
也就是異動類型641和101

為什麼物流會有一個異動類型301？
301是沒有調撥單的一段性發貨
也就是不需要調撥申請，只要出貨方一做301，收貨方就自動加庫存，不需要驗收
一般來說我們不建議使用這個異動類型
但因為物流出貨有物流箱
門市開調撥單時並不會知道需要多少個物流箱
所以無法在調撥單上輸入物流箱的數量
因此在流程設計上，只允許物流箱或酒箱使用301
其他任何單位的相互調撥，都還是建議走641+101
在內稽內控上也比較不會出問題

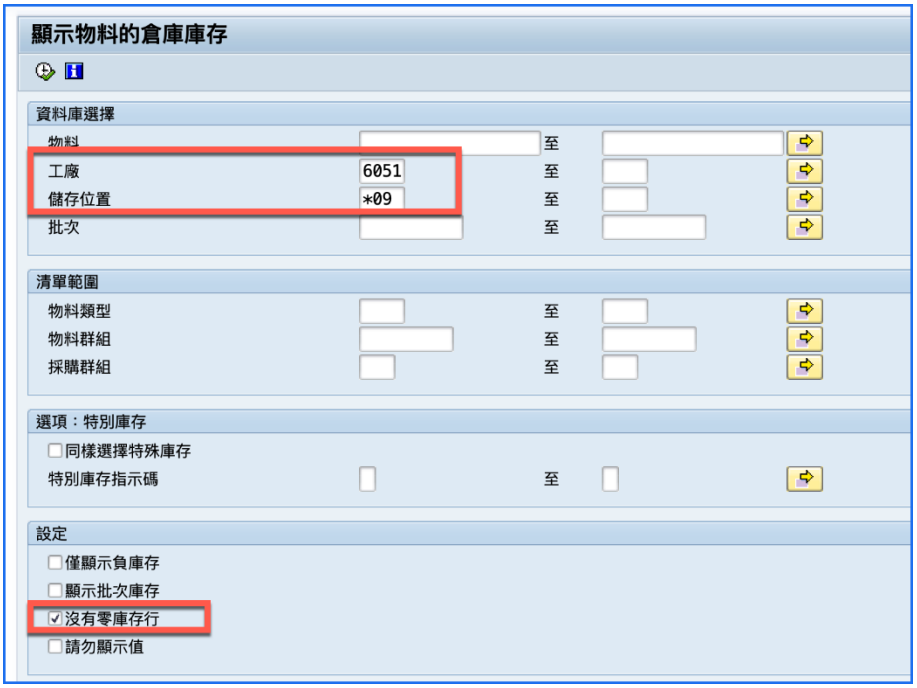
接下來說明 SAP 怎麼查詢調撥量差狀況

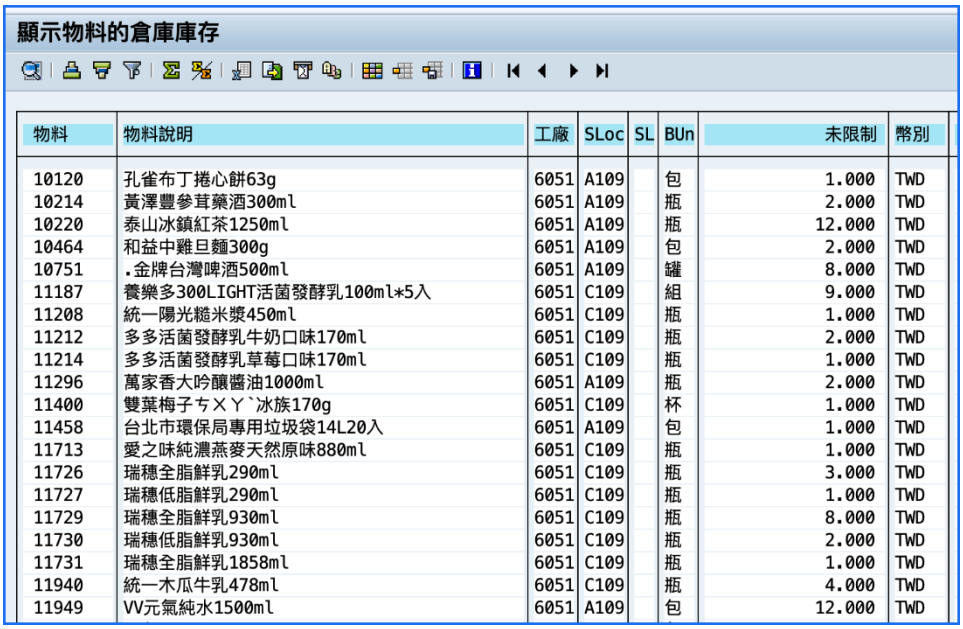
先查詢量差倉有沒有庫存，以及有哪些商品被彈進了量差倉。Tcode: MB52

因為不是每筆調撥都會產生量差，而且差異數量如果處理好了轉回正常倉，量差倉的庫存量就會不見。

所以先確認量差倉目前有哪些商品和數量是重要的。

由於物流的量差倉都是 09 結尾，所以可以直接在儲存位置輸入「*09」，就代表所有量差倉。也可以輸入單一量差倉。





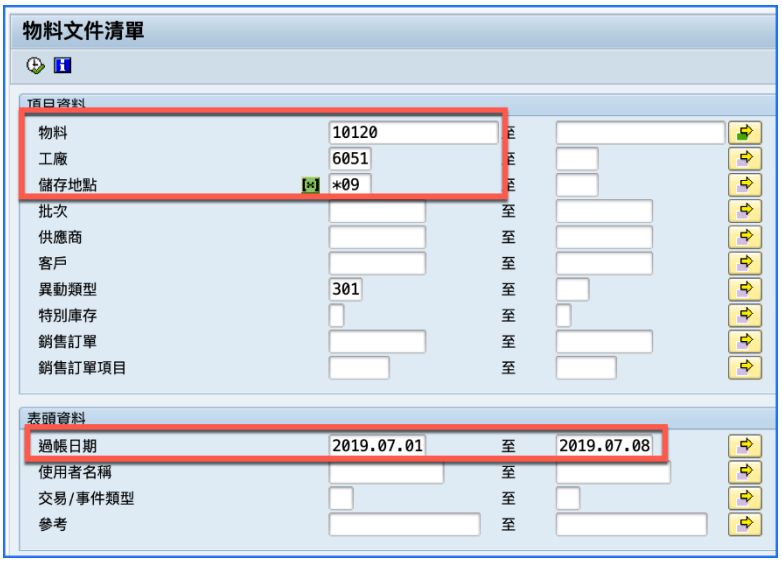
顯示物料的倉庫庫存

物料	物料說明	工廠	SLoc	SL	BUn	未限制	幣別
10120	孔雀布丁捲心餅63g	6051	A109		包	1.000	TWD
10214	黃澤豐參茸藥酒300ml	6051	A109		瓶	2.000	TWD
10220	泰山冰鎮紅茶1250ml	6051	A109		瓶	12.000	TWD
10464	和益中雞旦麵300g	6051	A109		包	2.000	TWD
10751	.金牌台灣啤酒500ml	6051	A109		罐	8.000	TWD
11187	養樂多300LIGHT活菌發酵乳100ml*5入	6051	C109		組	9.000	TWD
11208	統一陽光糙米漿450ml	6051	C109		瓶	1.000	TWD
11212	多多活菌發酵乳牛奶口味170ml	6051	C109		瓶	2.000	TWD
11214	多多活菌發酵乳草莓口味170ml	6051	C109		瓶	1.000	TWD
11296	萬家香大吟釀醬油1000ml	6051	A109		瓶	2.000	TWD
11400	雙葉梅子ㄘㄨㄚ`冰族170g	6051	C109		杯	1.000	TWD
11458	台北市環保局專用垃圾袋14L20入	6051	A109		包	1.000	TWD
11713	愛之味純濃燕麥天然原味880ml	6051	C109		瓶	1.000	TWD
11726	瑞穗全脂鮮乳290ml	6051	C109		瓶	3.000	TWD
11727	瑞穗低脂鮮乳290ml	6051	C109		瓶	1.000	TWD
11729	瑞穗全脂鮮乳930ml	6051	C109		瓶	8.000	TWD
11730	瑞穗低脂鮮乳930ml	6051	C109		瓶	2.000	TWD
11731	瑞穗全脂鮮乳1858ml	6051	C109		瓶	1.000	TWD
11940	統一木瓜牛乳478ml	6051	C109		瓶	4.000	TWD
11949	VV元氣純水1500ml	6051	A109		包	12.000	TWD

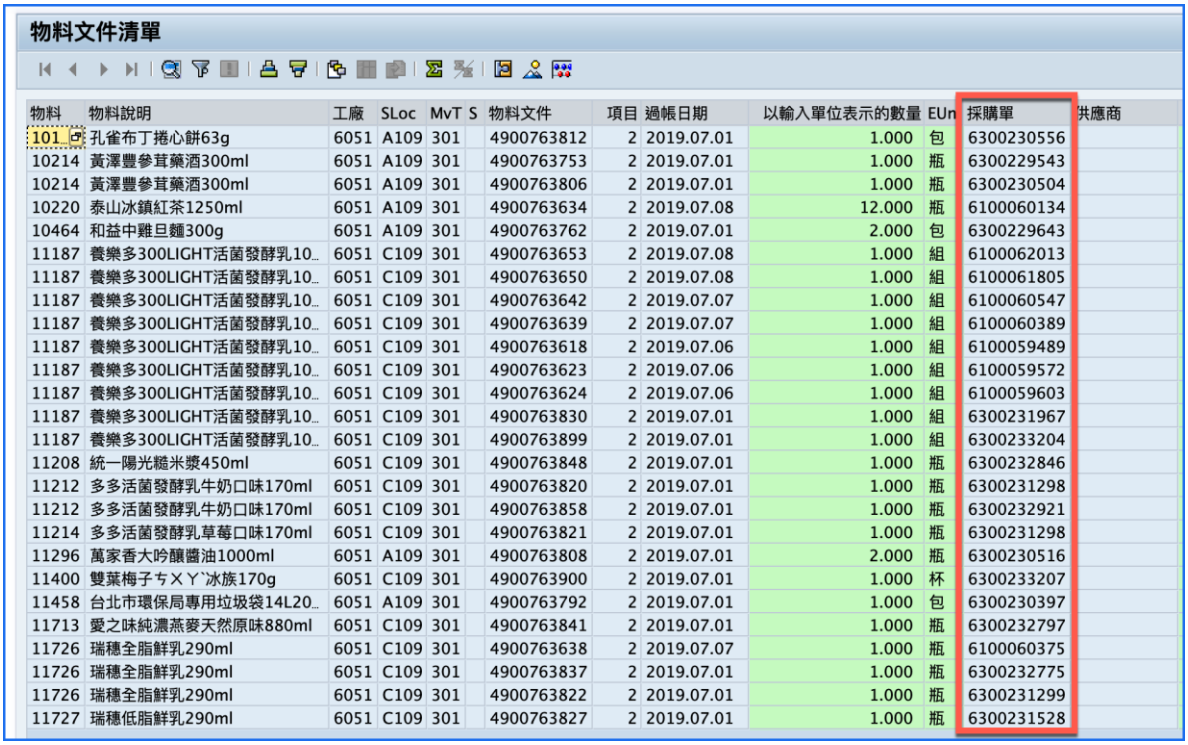
查詢哪些調撥單被彈回差異。Tcode: MB51

可以將 MB52 轉存到電腦，或複製要查詢的貨號清單，貼到 MB51 裡。

請記得輸入過帳日期，通常輸人你上次處理後到今天就可以。



採購單欄位就是被彈回的調撥單號，雙擊可以直接進入 ME23N 查看詳細內容。



物料文件清單

物料	物料說明	工廠	SLoc	MvT	S	物料文件	項目	過帳日期	以輸入單位表示的數量	EUn	採購單	供應商
101	孔雀布丁捲心餅63g	6051	A109	301		4900763812	2	2019.07.01	1.000	包	6300230556	
10214	黃澤豐參茸藥酒300ml	6051	A109	301		4900763753	2	2019.07.01	1.000	瓶	6300229543	
10214	黃澤豐參茸藥酒300ml	6051	A109	301		4900763806	2	2019.07.01	1.000	瓶	6300230504	
10220	泰山冰鎮紅茶1250ml	6051	A109	301		4900763634	2	2019.07.08	12.000	瓶	6100060134	
10464	和益中雞旦麵300g	6051	A109	301		4900763762	2	2019.07.01	2.000	包	6300229643	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763653	2	2019.07.08	1.000	組	6100062013	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763650	2	2019.07.08	1.000	組	6100061805	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763642	2	2019.07.07	1.000	組	6100060547	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763639	2	2019.07.07	1.000	組	6100060389	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763618	2	2019.07.06	1.000	組	6100059489	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763623	2	2019.07.06	1.000	組	6100059572	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763624	2	2019.07.06	1.000	組	6100059603	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763830	2	2019.07.01	1.000	組	6300231967	
11187	養樂多300LIGHT活菌發酵乳10...	6051	C109	301		4900763899	2	2019.07.01	1.000	組	6300233204	
11208	統一陽光糙米漿450ml	6051	C109	301		4900763848	2	2019.07.01	1.000	瓶	6300232846	
11212	多多活菌發酵乳牛奶口味170ml	6051	C109	301		4900763820	2	2019.07.01	1.000	瓶	6300231298	
11212	多多活菌發酵乳牛奶口味170ml	6051	C109	301		4900763858	2	2019.07.01	1.000	瓶	6300232921	
11214	多多活菌發酵乳草莓口味170ml	6051	C109	301		4900763821	2	2019.07.01	1.000	瓶	6300231298	
11296	萬家香大吟釀醬油1000ml	6051	A109	301		4900763808	2	2019.07.01	2.000	瓶	6300230516	
11400	雙葉梅子ㄘㄨㄚˋ冰族170g	6051	C109	301		4900763900	2	2019.07.01	1.000	杯	6300233207	
11458	台北市環保局專用垃圾袋14L20...	6051	A109	301		4900763792	2	2019.07.01	1.000	包	6300230397	
11713	愛之味純濃燕麥天然原味880ml	6051	C109	301		4900763841	2	2019.07.01	1.000	瓶	6300232797	
11726	瑞穗全脂鮮乳290ml	6051	C109	301		4900763638	2	2019.07.07	1.000	瓶	6100060375	
11726	瑞穗全脂鮮乳290ml	6051	C109	301		4900763837	2	2019.07.01	1.000	瓶	6300232775	
11726	瑞穗全脂鮮乳290ml	6051	C109	301		4900763822	2	2019.07.01	1.000	瓶	6300231299	
11727	瑞穗低脂鮮乳290ml	6051	C109	301		4900763827	2	2019.07.01	1.000	瓶	6300231528	

再補充一個不用 642 的原因

就算系統允許交貨單部份迴轉，642 會將差異量迴轉到出貨時的儲存位置，也就是正常倉

這樣會造成物流很難區分正常倉的庫存量，哪一些是真正有存貨的，哪一些是被調撥差異彈回的

這也就是為什麼物流最後決定要設置「調撥量差倉」的原因。

門市也有討論過

後來管控選擇門市不設置調撥量差倉

- 在途存貨(Goods in Transit)

  購入之商品在期末仍未收到，視為在途存貨(Goods in Transit)。商品存貨之認列，必須視交易條件是否完成，且所有權的移轉是否已從賣方移轉買方而定。

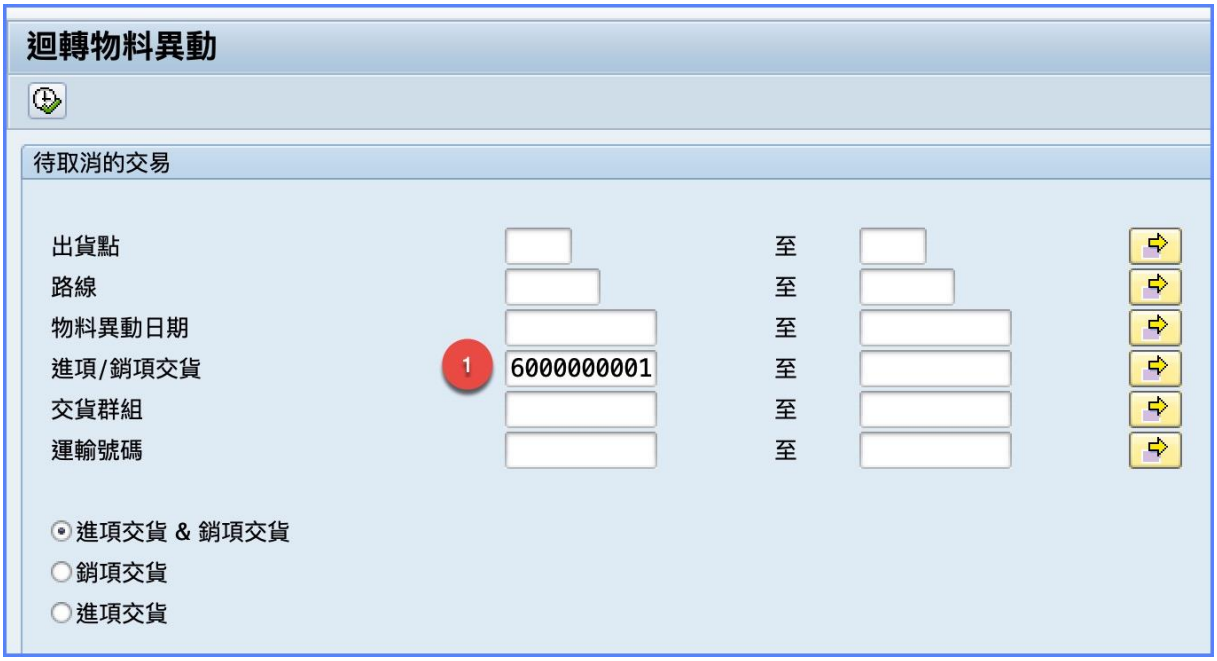
  在途存貨的歸屬權可分為起運點交貨(FOB Shipping Point)與目的地交貨(FOB Destination)。若為起運點交貨，離開賣方處所，所有權就移轉給買方，所以運送途中所遇風險及運費，應由買方負擔，為買方之進貨運費；若為目的地交貨，要送到買方指定地點才算責任終了，所以運送途中所遇風險及運費，應由賣方負擔，為賣方之銷貨運費。

- 迴轉分錄的作法(調撥出貨後，因故必須取消出貨)，T-Code=取消/迴轉 VL09

  若調撥單已收貨，一定要先迴轉收貨，才能做發貨迴轉。

  交貨單只能全數迴轉，不能部份迴轉。

  輸入交貨單號。特別注意：不是輸入調撥單號。



按 [迴轉] 即可執行

- 刪除旗標

  在 SAP 裡面，所謂的刪除有兩種

  一種是所謂的 deletion flag，就是標記刪除（但是 SAP 資料庫檔案還是存在）

  一種是 archiving，就是歸檔（從 SAP 資料庫移除到別的地方）

  差別在 del 還是可以找到該筆資料，只是不能使用。其實這樣對使用上來說，已經 OK。

  不過若是覺得礙眼，想要完全看不見，就一定要做 archiving。

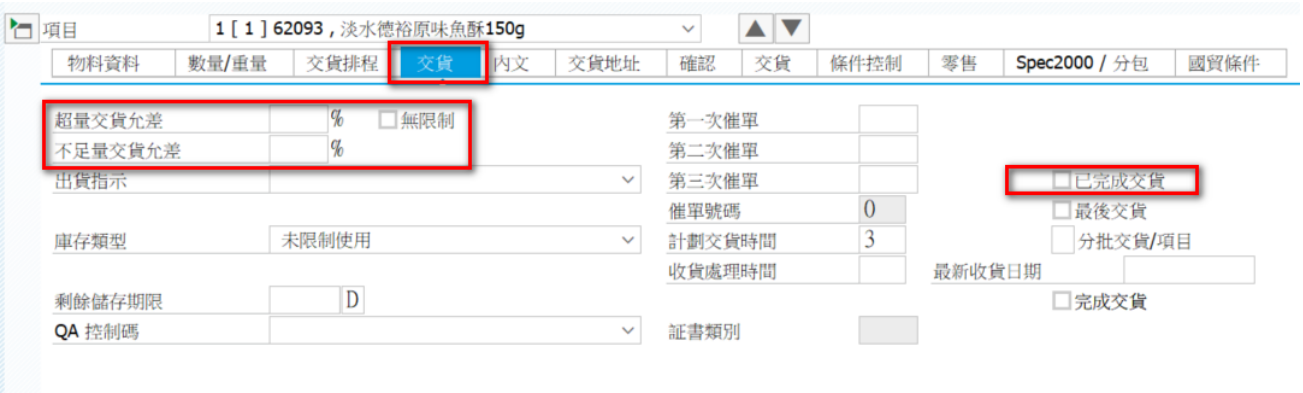
  下圖中的 6041 商品，就是執行刪除後的紀錄。

➢ 另外，SAP 中沒有直接刪除整個採購訂單的功能，只能刪除行項目且只是做了刪除標記。或是利用迴轉分錄對沖掉原單據的內容，這主要是考慮資料的嚴謹性。

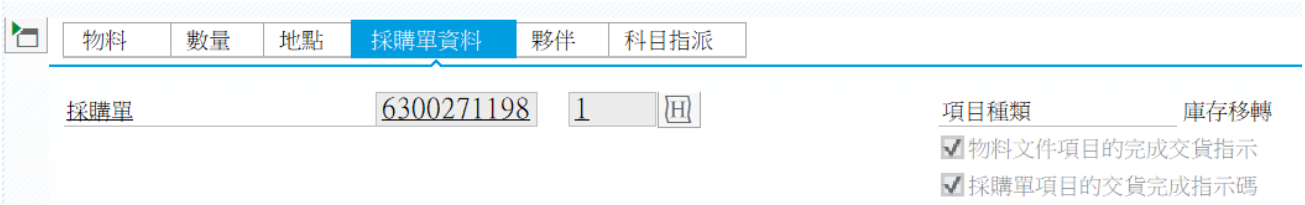
- 關單，如何設定自動關單

https://www.twblogs.net/a/5c1228dcbd9eee5e40bb45b1

採購訂單(ME22N/ME23N)行項目的[交貨] 頁籤有個 [交貨已完成]，如果選擇此標誌,則表示該筆商品的交貨已完成，該採購單不再是未清採購訂單，也就是說該採購行的收貨關閉，不再需要交貨(但仍可繼續交貨)，也不再參與 MRP 運算，如下圖，注意過量交貨限度/交貨不足限度標誌。



- SAP 中未清採購訂單查詢的事務代碼，ME2L，ME2M 都可以查詢。
- 除了在採購文件中的 [交貨] 頁籤設定 [已完成交貨] 外，也可以在 MIGO 收貨時，到 [採購單資料] 頁籤有 2 個 [交貨已完成] 的勾選項，如下圖：



- 如果還有篇幅，也許盤點也可以聊一聊(或單獨一章也可以)

https://zhuanlan.zhihu.com/p/30965245

有幾個 issue 待確定，

1. 是否一定要先 Plan 才能 Action?

也就是說，物料文件能否獨立存在？還是一定要依附在某張採購文件上？以先前的經驗，當然是非必要，也就是可以不需要有採購文件，不過這個尚待驗證。

2. 是否一定要依序執行？也就是一定要先收貨，才能驗收？可以直接執行發票驗證，然後就產生應付帳款？
3. end

第六章 完整的採購流程說明

在第五章中，我們已經將採購流程的常用單據流程說明清楚，本章我們將會針對不同的採購類型，分別說明實際場域的可能應用情境。

Session 0：基本的 SAP 採購流程通常會包括下列單據

請購單(purchase requisition)

詢價單(request for quotation，RFQ)

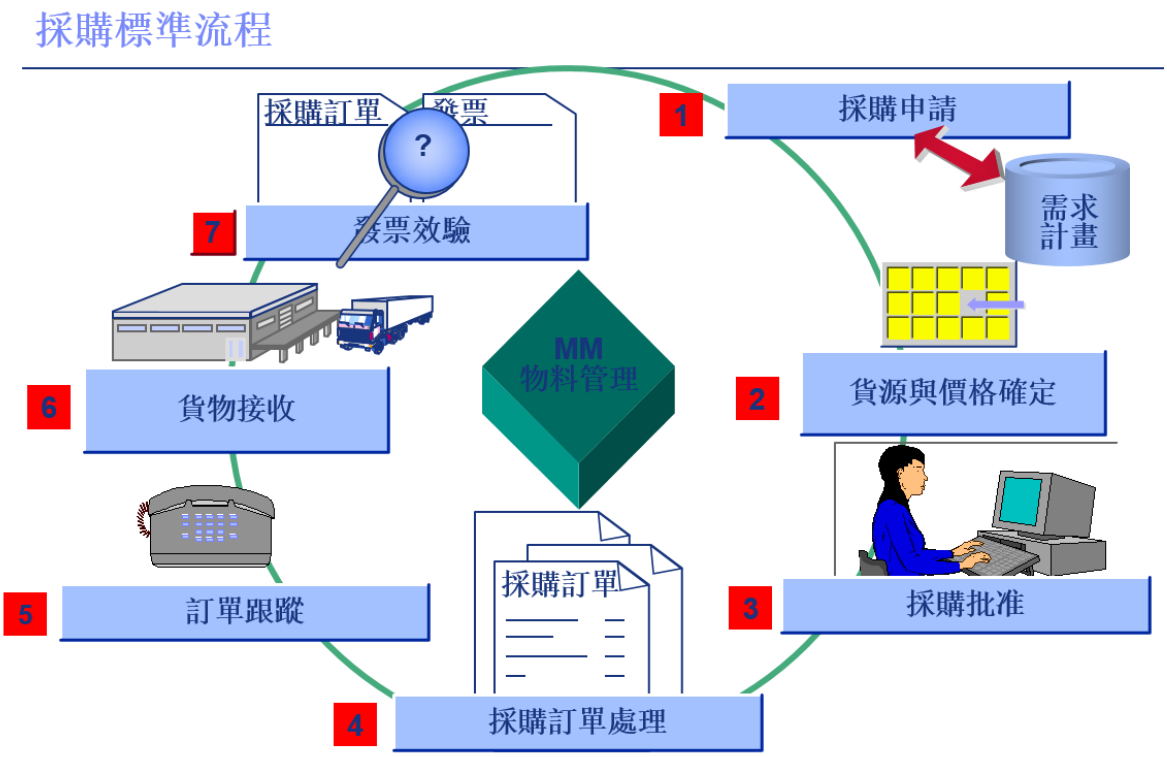
需求計劃(如何將請購單轉為採購單?? 採購建議)

配貨

採購文件：採購單(purchase order，PO) 又稱採購訂單、訂購單

物料文件：到貨單(採購收貨)

進貨發票(vendor invoice)、發票驗證



接下來我們會分別從下列幾種常見的採購流程來討論，並會加上和外圍系統的整合操作

1. 庫存物料的採購流程
2. 採購申請發起的採購流程
3. 退貨流程
4. 倉庫調撥流程
5. 如何查詢庫存狀況

Session 1：庫存物料採購流程

https://www.fenginfo.com/1318.html

此流程是 MM 模組的庫存物料的採購流程，分為三個步驟，如下圖所示：

公司的採購部門對供應商創建採購訂單，確定採購數量，價格則是從採購資訊記錄中獲得，做訂單時確定了收貨工廠但尚未指定庫存地點；

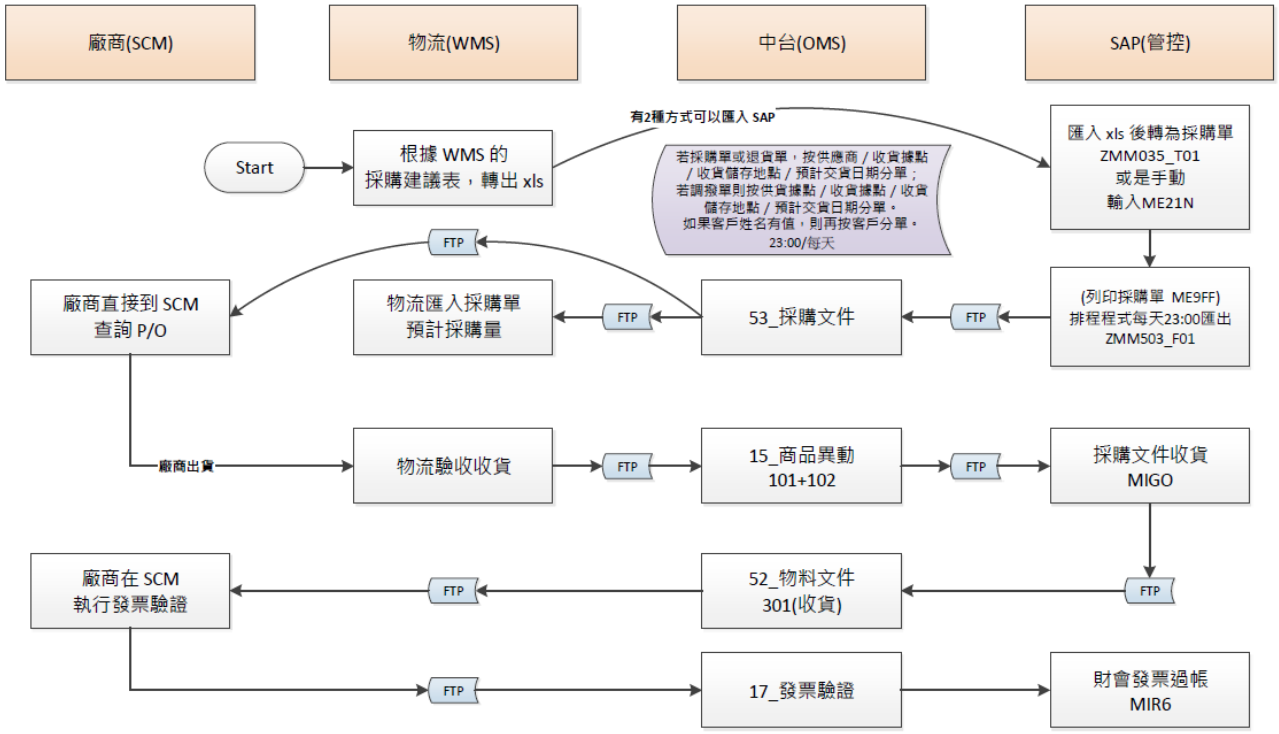
供應商將貨物送到庫房，庫房參考採購訂單對貨物做收貨處理，生成移動類型為 101 的物料憑證，同時生成 GR/IR 會計憑證；

供應商開出發票，財務部門對發票做發票校驗，生成發票憑證，同時生成會計憑證，並增加對供應商的應付款。

此流程不包含審批步驟，做完全部業務流程，庫存的物料增加，對供應商的應付款增加。後續的操作將是財務部門對供應商做付款清帳處理。

此流程相應的單據有：採購訂單、物料文件（同時生成會計憑證）、發票文件（同時生成會計憑證）。在各單據中都可查看到其它相關單據並形成單據流。如下圖所示。

以下是 SAP 和外圍系統的整合流程，下圖是由物流發動的向供應商採購的全流程

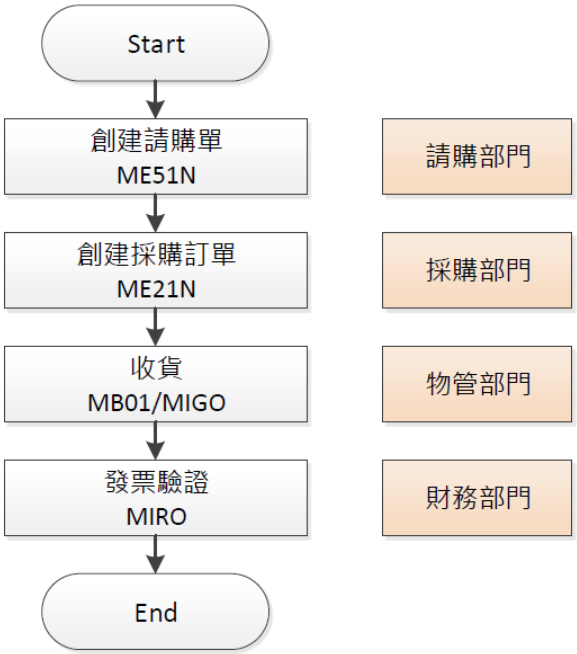


Session 2：採購申請發起的採購流程

https://www.fenginfo.com/1509.html

本流程的物料需求部門並非直接採購部門，當庫存物料數量不足時，需求部門需創建採購申請，採購部門根據申請組織貨源及入庫相關程式。

流程的發起端為採購申請（PR），後接參考創建的採購訂單（PO），之後的步驟與《庫存物料採購流程》相同。具體步驟如下：



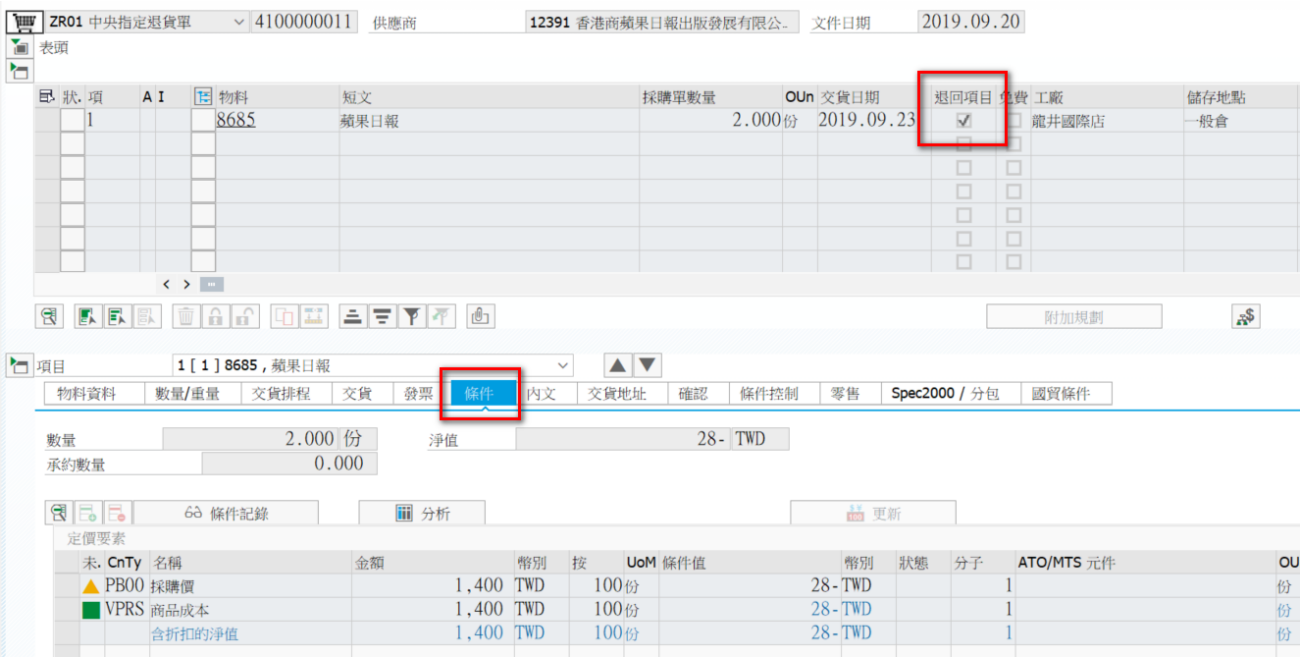
此流程相應的單據有：採購申請、採購訂單、物料憑證（同時生成會計憑證）、發票憑證（同時生成會計憑證）。在各單據中都可查看到其它相關單據並形成單據流。請參考上頁的說明。

Session 3：退貨流程(Return goods)

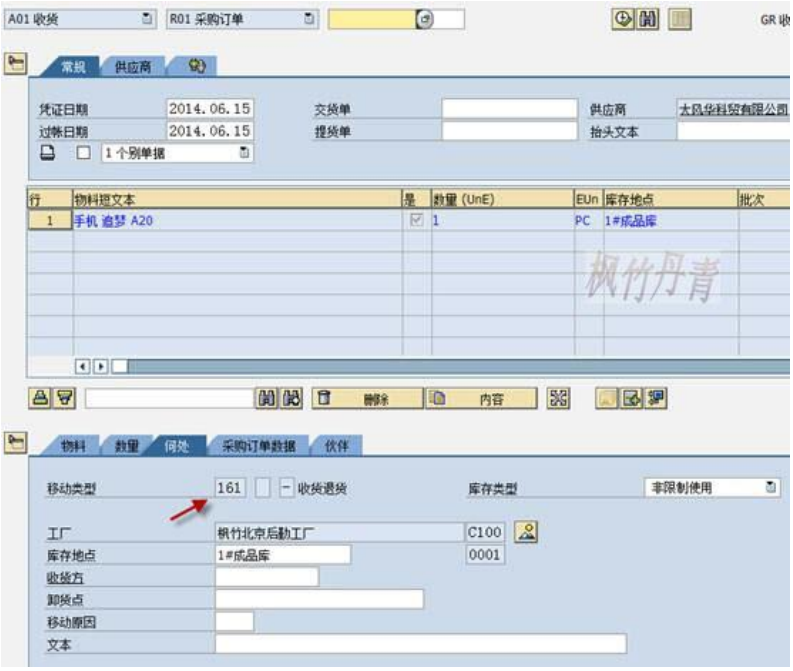
https://www.fenginfo.com/1460.html

當之前採購的貨物已入庫並向供應商結算，由於某種原因（如發現品質問題）需退回給供應商，就可以創建退貨型（Return goods）採購訂單。退貨型採購訂單後續操作也需要收貨，並進行發票校驗。其中收貨的移動類型是 161，發票校驗採用貸方憑證（與發票類作法相反）。

此示例資料及操作與 (**Session 1：庫存物料採購流程)** 類似，細節不再重複，只是表身的 [退回項目] 欄打上選擇標誌（如下圖所示）。價格資料從採購記錄中提取，也可以在專案細節的 [條件] 標籤頁輸入或修改。

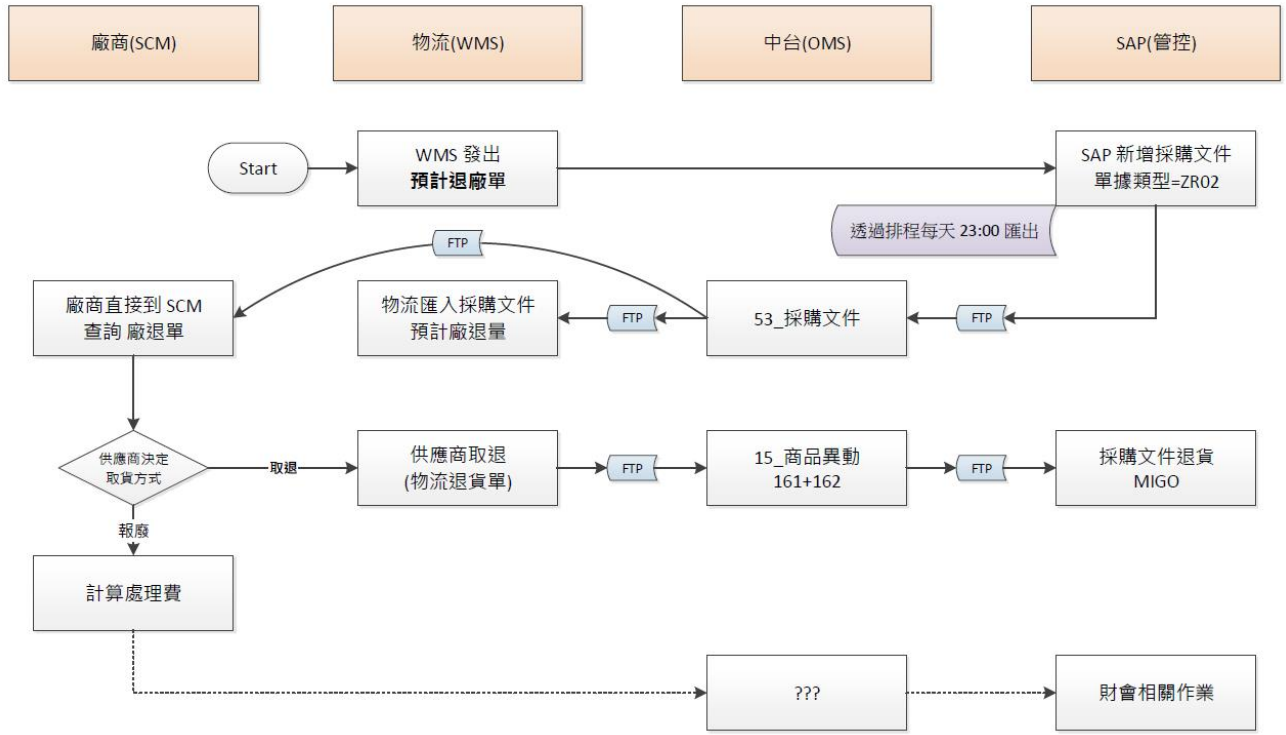


MIGO 或 MB01 收貨操作參照《貨物移動_對採購訂單的 GR 收貨(MvT101)》，只是移動類型 MvT 是 161（收貨退貨，英文 GR returns）（如下圖所示），生成的會計憑證借貸方與正常收貨（移動類型 MvT101）相反。



MIRO 發票校驗的操作參照《發票校驗_參考庫存物料採購訂單》，只是業務處理類型為 [貸方憑證] (如下圖所示英文 Credit Memo)，生成的會計憑證的借貸發與正常發票相反，並扣減對供應商的應付款。

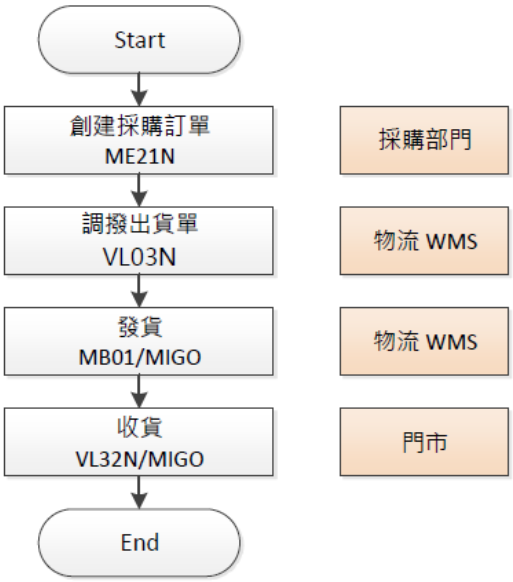
下圖是和外圍系統整合的退貨流程圖



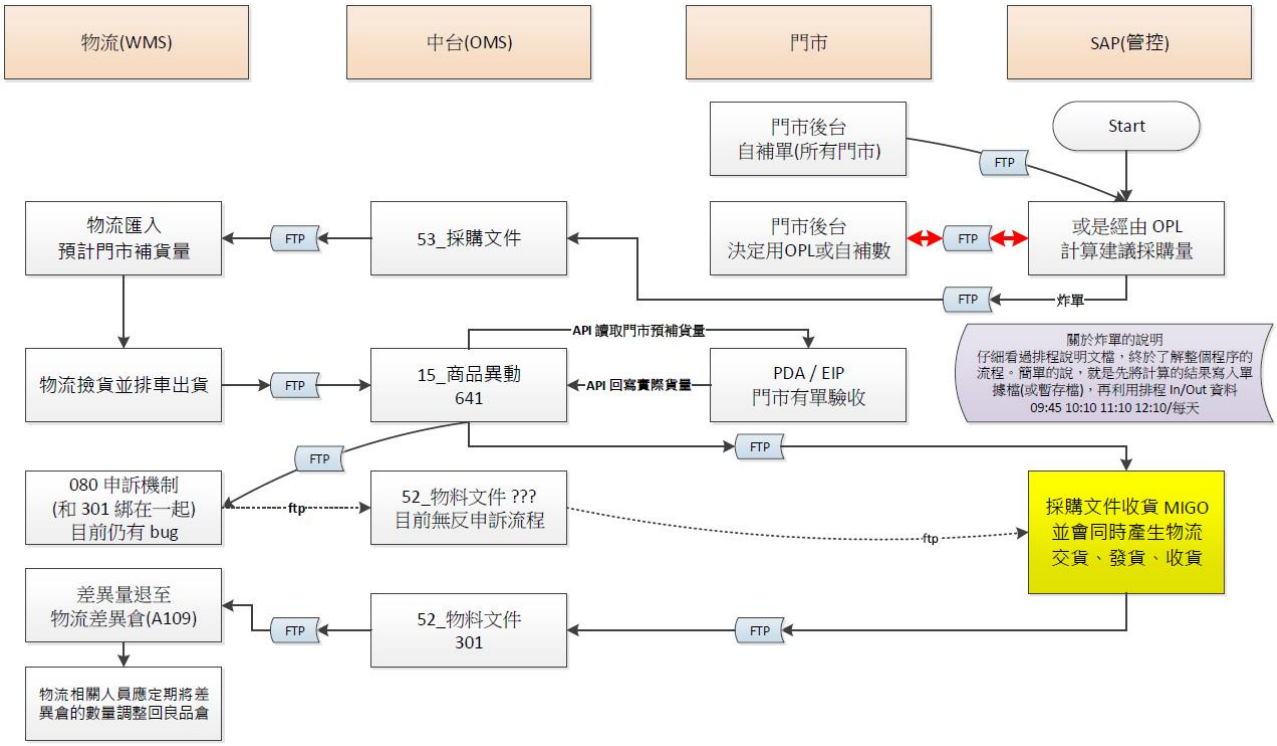
Session 4：倉庫調撥流程

採購文件有 3 種主要的類型，採購單、退貨單及調撥單。本節討論的是物流調撥到門市的流程。第五章中有詳細的說明相關的單據流程，此處不再贅述。

單別			
採購文件→採購單	收貨(轉進)	退貨(選項)→應付減項	發票驗證(應付)
→退貨單	退貨		發票驗證(應付減項)
→調撥單	交貨	發貨(641)	收貨(101)



下圖是和外圍系統整合的物流調撥流程圖



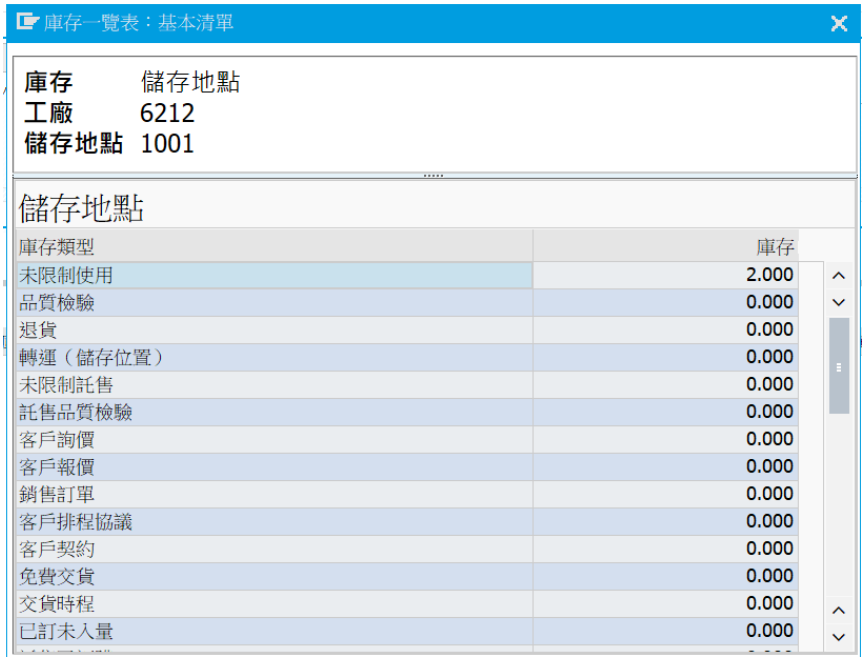
Session 5：如何查詢庫存狀況

https://www.fenginfo.com/1442.html

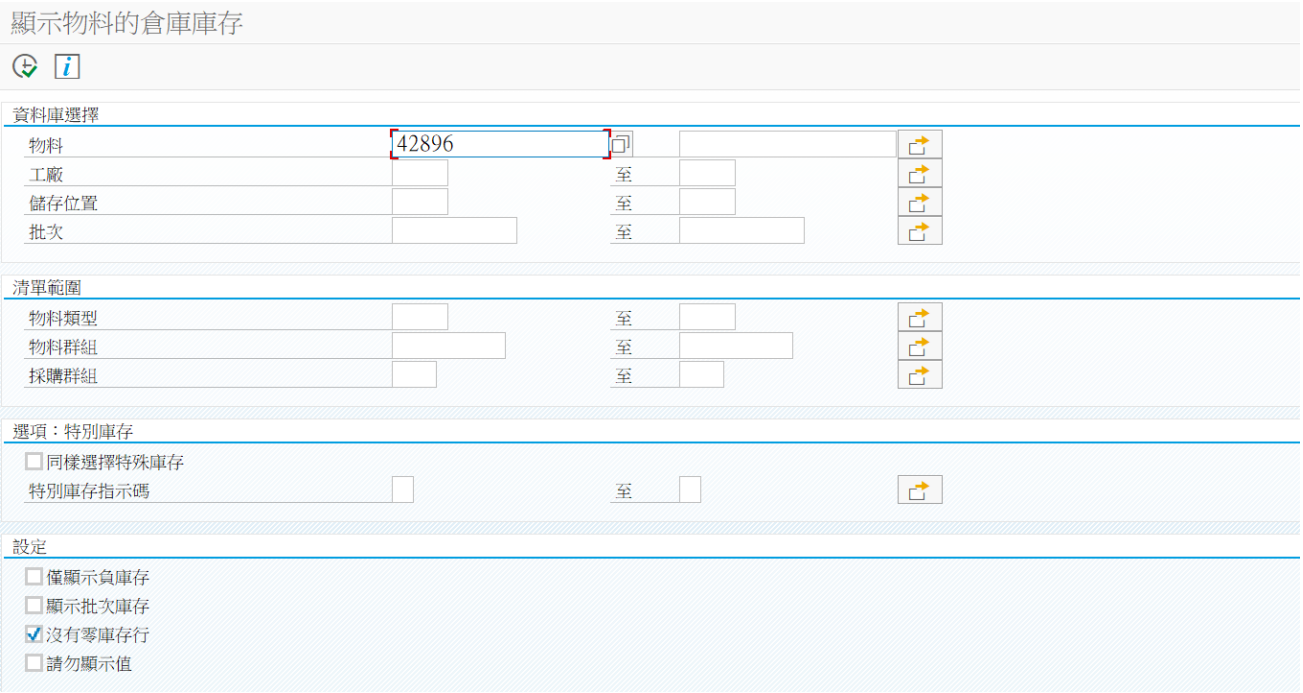
查詢物料(Material)當前的庫存總覽狀況(Stock Overview)，**物料在庫存中不僅只是一個數量資訊，它具有多種形態**，主要有非限制使用（unrestricted use stock）、品質檢驗（stock in quality inspection）、未清採購訂單數（open purchase orders）等。通過查詢可以瞭解後續業務操作的可行性，例如是否可以滿足發貨需求則需查看非限制使用數量。

如果知道物料編號，可用 T-CODE：MMBE 查找指定工廠或庫存地點的庫存狀況，甚至全部工廠及庫存地點的狀況，當然只能查看一種物料的資訊；如果想查找工廠或庫存地點中有哪些物料，則可以使用 MB52，使用 MB52 可以不用知道物料編號。

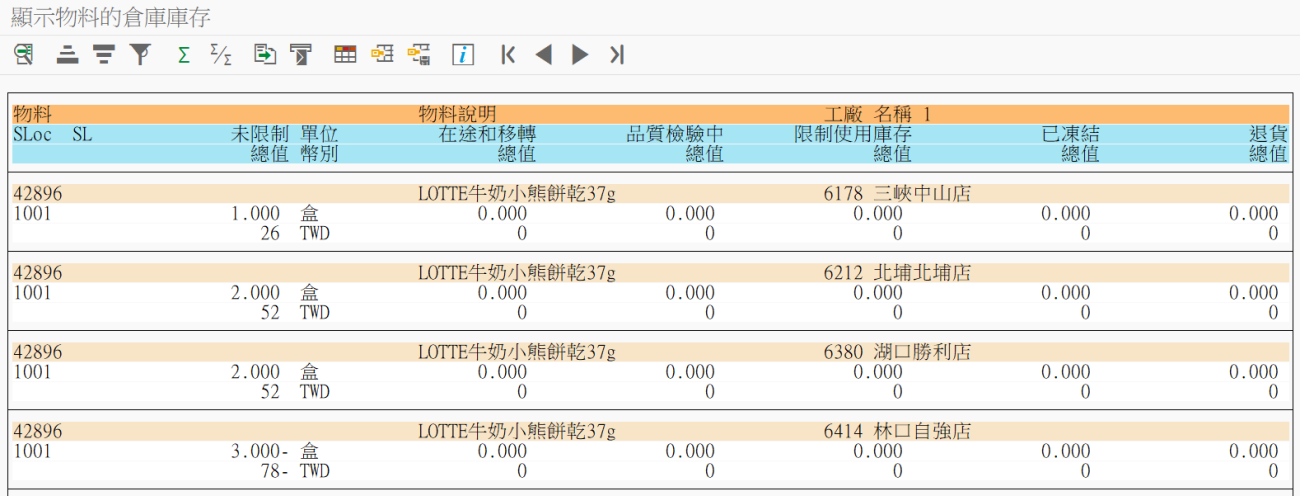
在任一儲位 DbClick，數量的分類(花樣) 多到令人目不暇給，搞不清楚，真是令人迷惑阿。



MB52 報表



顯示查詢內容(零庫存不顯示)



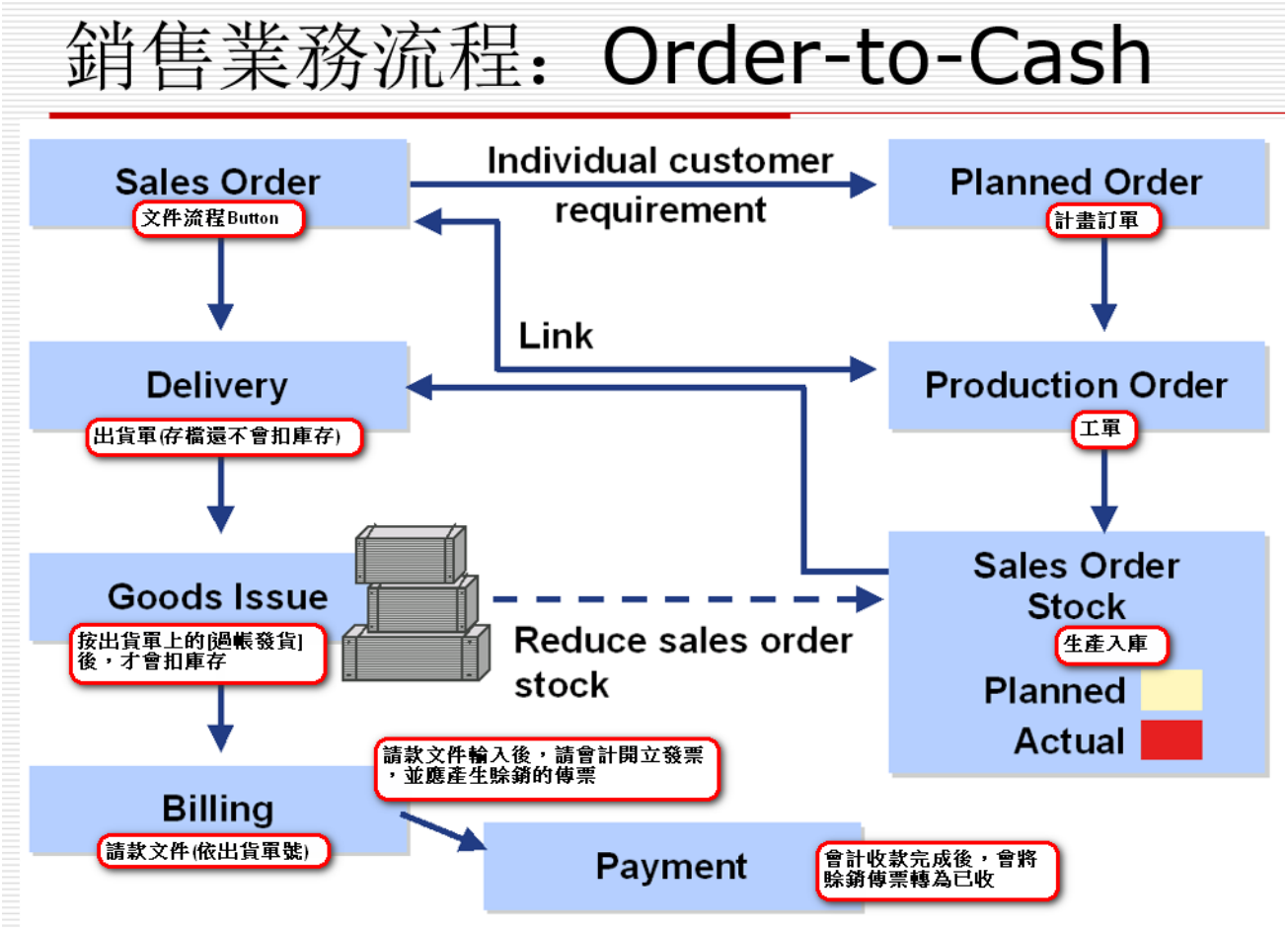
物料			物料說明		工廠 名稱 1		
SLoc SL	未限制 總值	單位 幣別	在途和移轉 總值	品質檢驗中 總值	限制使用庫存 總值	已凍結 總值	退貨 總值
42896			LOTTE牛奶小熊餅乾37g		6178 三峽中山店		
1001	1.000	盒	0.000	0.000	0.000	0.000	0.000
	26	TWD	0	0	0	0	0
42896			LOTTE牛奶小熊餅乾37g		6212 北埔北埔店		
1001	2.000	盒	0.000	0.000	0.000	0.000	0.000
	52	TWD	0	0	0	0	0
42896			LOTTE牛奶小熊餅乾37g		6380 湖口勝利店		
1001	2.000	盒	0.000	0.000	0.000	0.000	0.000
	52	TWD	0	0	0	0	0
42896			LOTTE牛奶小熊餅乾37g		6414 林口自強店		
1001	3.000-	盒	0.000	0.000	0.000	0.000	0.000
	78-	TWD	0	0	0	0	0

第七章 完整的銷售流程說明

銷售訂單管理涵蓋了從訂單到收款的相關的企業流程，組織層次(Organizational Levels)和主檔資料(Master Data) 在前章已有說明。本文主要說明的流程是銷售訂單(Sales Order)、交貨文件(Delivery Document)、過帳發貨、和請款文件(Billing Document)。另外，由於家購是屬於 Retail 零售買賣業，所以大部分的銷售行為都是透過 POS 前端，因為 Row Data 資料太過龐大，沒有必要在 SAP 中記錄明細，所以會透過 POSDM 模組將前端的 POS 資料壓縮後寫入 SAP 銷售訂單。

本文的前面會依 SAP 的標準流程說明 SD 的主要單據的作業流程，主要是參考 Leo 顧問的教學內容及楓竹丹青的 blog，在家購的實際流程中，只有少部分的非 POS 交易會使用，可以先看一次 Leo 的教學視頻(\\10.0.8.14\sap 導入專案\專案公用資料夾\Phase 3 具體實施\3.X 具體實施階段影音檔\MMSD\20190104-KUT_SD_非 POS 交易)。後面則會簡單說明 POSDM 的相關簡介，由於參考資訊不多，詳細說明請參考 NTT 準備的下列 ppt

- POSDM WorkBench_2018_12_13_v01.pptx
- 美廉社_模組功能介紹_POSDM_20180830_V4.pptx



目前查詢銷售數字，Austin 說是查 KE30 及 MCGE(繼續查詢資料流中...)，威錩確定 POSDM 的資料是應該是匯入 VBRP Billing Document 繼續研究...

➢ 主要標準功能總覽

T Code	功能說明
VA21	建立報價單
VA01	建立銷售訂單
VL01N	建立出貨單
VL10A	查詢已訂未出的銷售訂單，並執行發貨過帳作業
VF01	建立請款文件

➢ 適用之文件類型

SN	說明	Quotation	SO	Delivery	Billing
1	團購批發訂單	ZQ01	ZS01	ZD01	ZB01
11	團購批發退貨單	X	ZR01	ZDR1	ZBR1
12	折讓申請	X	ZR02	X	ZBR2
21	POS 銷售	X	X	X	ZPS1
91	迴轉文件	X	X	X	ZZZZ
92	折讓取消	X	X	X	ZZYY

➢ 報價單(Quotation)

- **執行 T-Code=VA21**

① 輸入銷售區域，包含銷售組織、配銷通路、部門（固定 R1）。

② 選擇銷售據點，這裡是指業績入帳的單位。

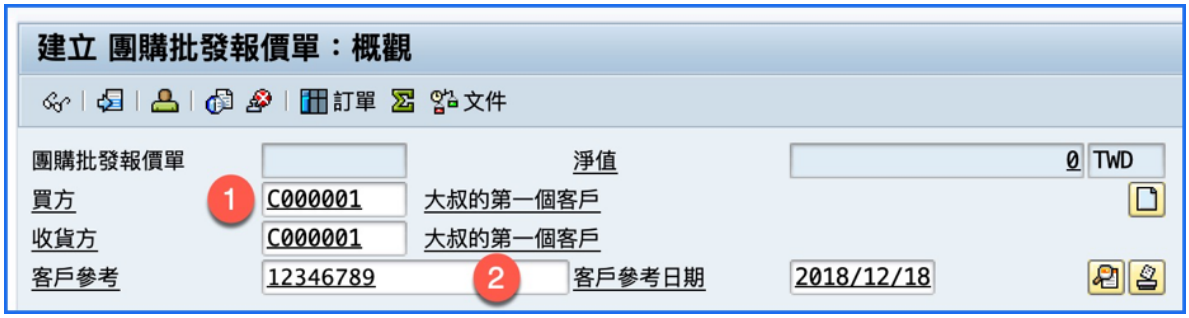
③ 選擇報價類型 ZQ01。

輸入完畢後，按 ✔ 或 enter 開始建立報價單。

- 輸入表頭資料

① 「買方」欄位輸入客戶編號。如果買方與收貨方不同，也可以輸入不同客戶編號。

② 「客戶參考」、「客戶參考日期」可以輸入客戶詢價單號和詢價日期，以利後續查詢。



- 輸入表身資料

① 報價有效期限。

② 商品貨號，按 enter 會帶出品名。

③ 報價數量。

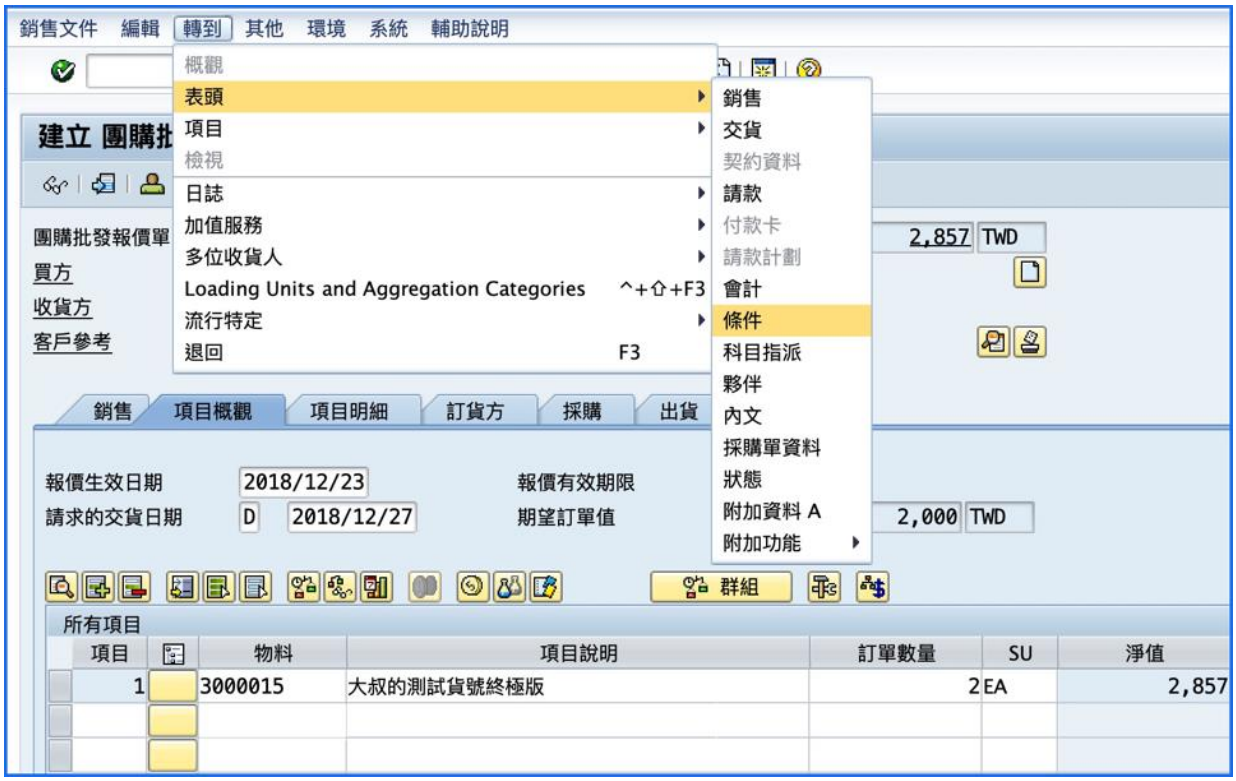
④ 出貨據點。出貨據點可以和初始頁的業績入帳單位不一樣。

- 如果需要修改品項的售價，先選取要修改的項次，再點選① 或②依下圖路徑進入表身「條件」頁。

- 在條件最下面新增一行，選擇①CnTy 條件類型「EDI1 客戶期望價格」，②輸入最終要賣給客戶的金額，按 enter 就會重新計算淨值及稅金。

注意：如果金額輸在「金額」欄位，代表輸入的是單價；若輸在「條件值」則代表總價。

- 如果客戶期待是整筆交易折扣，而不是單一商品折價，可點選 或依下圖路徑進入表頭的「條件」頁修改。



- 在條件最下面新增一行，選擇①CnTy 條件類型，HA00 是折扣（百分比），HB00 是折讓金額。②在「金

額」欄位輸入折扣或折讓，③按啟用鈕 就會重新計算整張的淨值及稅金。輸入完畢後，按 回到前頁。

注意：表頭輸入完畢後，一定要按「啟用」，金額才會重新計算。

- **確認輸入內容無誤後，按存檔即可。**

➢ 銷售訂單（Sales Order）

- **執行 T-Code=VA01**
- 輸入方式和報價單一樣。

① 輸入銷售區域，包含銷售組織、配銷通路、部門（固定 R1）。

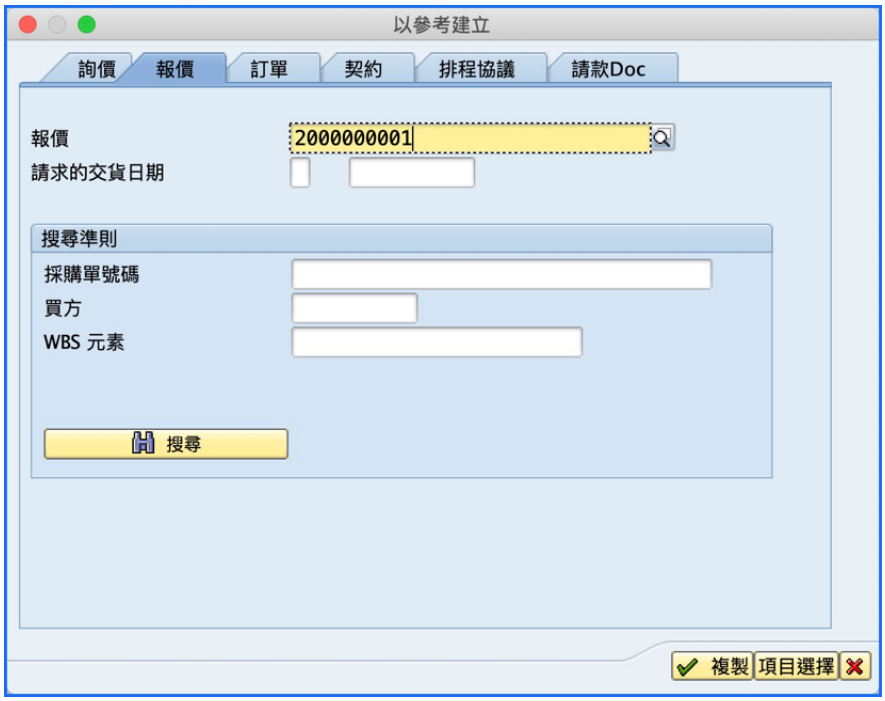
② 選擇銷售據點，這裡是指業績入帳的單位。

③ 選擇銷售訂單類型。

④ 如果須參考報價單或請款文件，可以按 [以參考建立]。部份銷售訂單類型，系統會預設一定要以參考建立，如折讓單一定要參考某一張請款文件。

輸入完畢後，按 或 enter 開始建立銷訂單。

- 如果以參考建立，在跳出「以參考建立」視窗後，選擇參考的文件類型，並輸入文件號碼。如果選擇「複製」，則整張報價單的品項都會被複製到銷售訂單；選擇「項目選擇」則可以選擇複製部份品項和數量。



- 銷售文件建立的方式大致與報價單相同，除了銷售文件不會有報價有效日期，而退貨的銷售文件則須要輸入退貨原因。

  ① 如果銷售時須要提供贈品，在贈品項次將 ItCa 項目類別改成 TANN 免費項目，該項次就會變免費商品，銷售金額為 0。

項目	物料	項目說明	訂單數量	單位	ItCa	淨價	幣別	工廠
1	3000015	大叔的測試貨號終極版	20	EA	TAN	1,429	TWD	8500
2	3000015	大叔的測試貨號終極版	2	EA	TANN	0	TWD	8500

- 如果沒有問題，儲存就完成銷售訂單建立。

https://www.fenginfo.com/1353.html

➢ 出貨(發貨)單

- 說明

當向客戶銷售的訂單（SO）創建完成後，庫房需要執行一系列步驟將貨物發出。所謂外向發貨就是由本埠向客戶發出貨物，一般發出過程需要三個步驟：創建外向發貨單、交貨若干步驟（如揀配、包裝、運輸等）、發貨過帳。本文只描述創建外向發貨單（DN）。

創建外向交貨的事物碼有 VL01N、VL10A 等，兩者的不同點在於前者需要輸入銷售訂單號、裝運點，而後者通過搜索條件查到。操作後創建外向交貨單，可用 VL02N 修改、VL03N 查看。外向交貨單是可以刪除的，可在 VL02N 修改狀態下點擊刪除按鈕進行刪除。

VL01N、VL10A 的操作後都可以揀配及過帳，本文只描述創建交貨單的操作。

- 參考憑證說明

創建外向交貨單參考的憑證是銷售訂單，銷售訂單號(SO)可以通過搜索的方式查到。銷售訂單類型是“TA 標準訂單”（英文 OR），在其類型定義中設定交貨類型為“LF 外向發貨”；銷售訂單的專案類別是“TAN 標準項目”，對應交貨的項目類別與此相同編號和描述（也是 TAN）。

- VL01N 操作，在前台輸入事物碼 VL10A

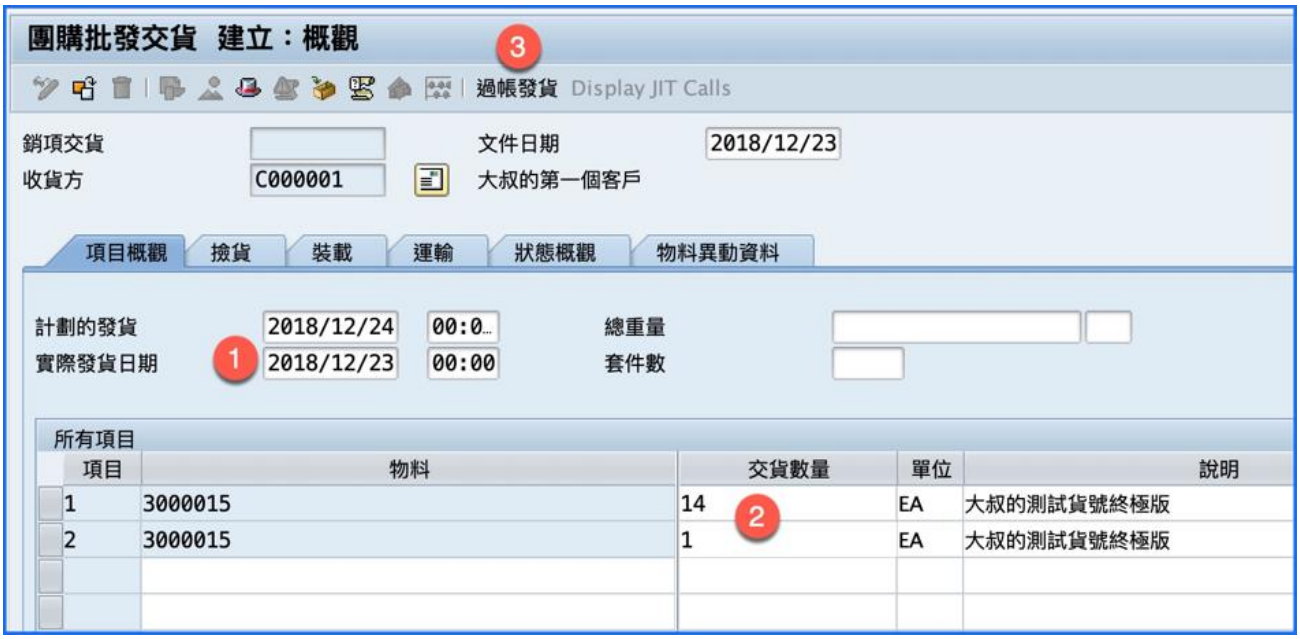
① 如果想要過帳的日期不是今天，可以在「實際發貨日期」填上過帳日期。

② 在「交貨數量」修改實際的揀貨數量。

③ 確認沒問題後，按 過帳發貨 後，便完成交貨及扣庫存。

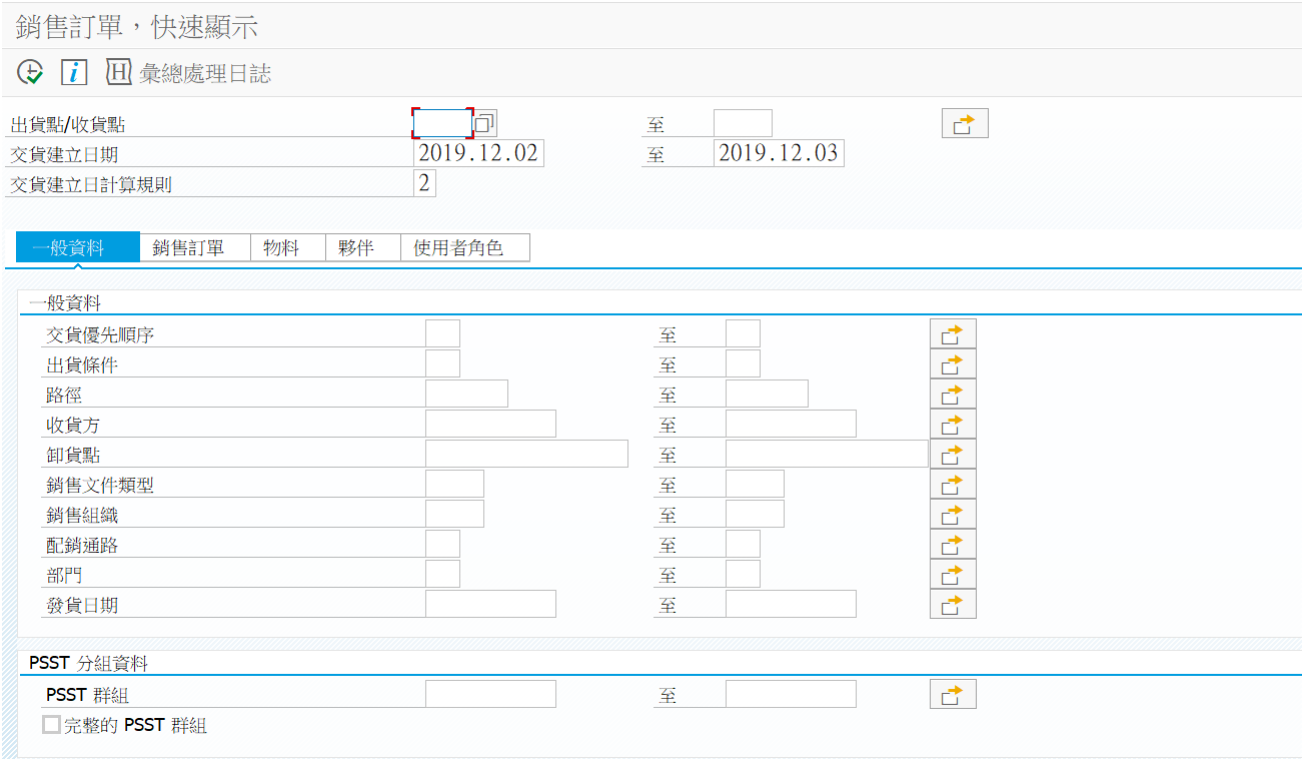
④ 也可以按 ，儲存交貨單。

[存檔] 和 [發貨過帳] 的差別，在於 [存檔] 只儲存這張出貨單，並沒有扣庫存；[發貨過帳] 則儲存出貨單並扣庫存。



此處可以進行交貨若干操作（揀配、包裝、運輸等）及過帳，但這都是後續的操作，將另文描述。按下保存按鈕（ ）保存，視窗下部狀態列顯示外向發貨單號，記下此單據號，按返回按鈕（ ）退出操作。

- VL10A 操作



這個介面是一個搜索條件輸入介面，它將搜索到可以外向交貨的銷售訂單，注意交貨創建日期的範圍，其它條件可根據已知情況縮小查找範圍。如果知道銷售訂單號，可點擊 [銷售訂單] 標籤頁直接輸入訂單號並刪除其它條件。

輸入完成後，點擊執行按鈕（ ）

上圖是搜索到的可以外向交貨的銷售訂單，可以是一個或多個條目，本介面可以看到銷售訂單號是 100000000，發貨日期是 2014.05.19，用滑鼠選中條目，可用前臺對話或背景方式創建交貨單，本文採用背景方式，點擊工具列 [背景] 按鈕。系統操作完成後如圖 5 介面，狀態綠色表明交貨單創建成功，連續點擊返回按鈕 ( ) 退出操作。

本操作已創建但沒有顯示外向交貨單號，可在銷售訂單憑證流中查到，或通過後續操作搜索得到。

https://www.fenginfo.com/1370.html

➢ 發貨過帳（Post Goods Issue）

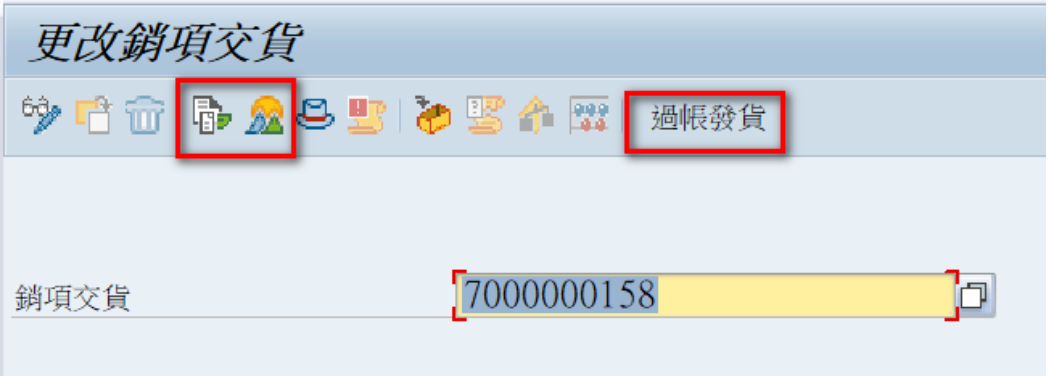
- 說明

交貨單經過揀配、包裝等一系列操作步驟，貨物出庫完成後，需要在 ERP 系統進行發貨過帳(Post Goods Issue) 操作，減少物料帳上的庫存數量，同時生成出庫的物料憑證及相關會計憑證。此項操作很簡單，但支撐的知識點很多。在未發貨過帳時，外向交貨單可用 VL02N 進行修改，並可在修改狀態下進行刪除；如果進行了過帳操作，則既不能進行修改，也不能立即刪除，只有先進行憑證沖銷，才能對交貨單做修改和刪除操作。沖銷操作的事物碼是 VL09。

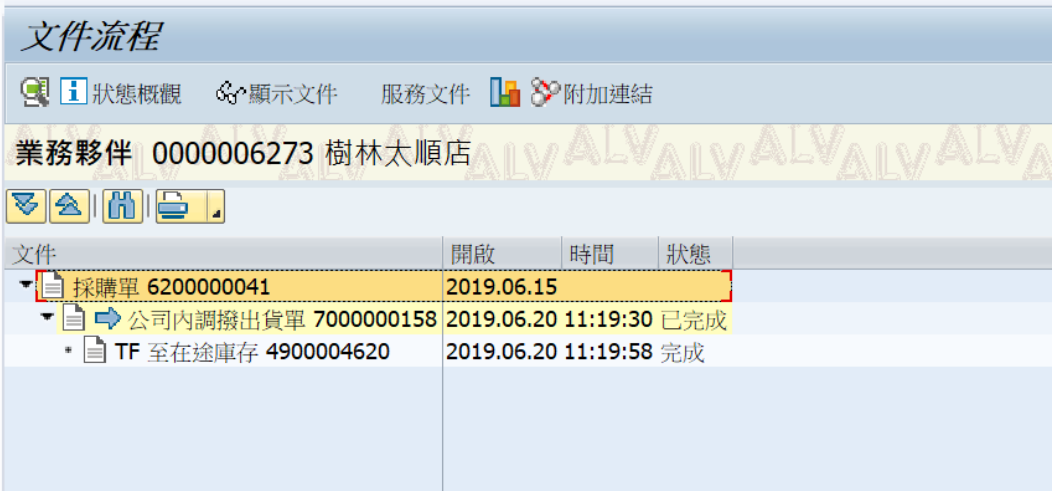
發貨過帳會生成物料憑證，物料憑證的移動類型（MvT）由銷售訂單（SO）專案下的計畫行（Schedule Line）決定。生成交貨單時，銷售訂單計畫行資訊複製到交貨單中，因此在交貨中只能使用指定移動類型而不能進行修改。

- 操作

在前台輸入事物碼 VL02N 執行，如下圖所示輸入外向交貨單號，如不知請通過搜索的方法查詢。如果交貨單已完成揀配、包裝等前述步驟，可直接點擊 [過帳發貨] 按鈕，執行過帳操作。如果再查看相關細節，可按 ENTER 鍵(或概觀 button）繼續查看明細。



文件流程則可以前後追朔相關的表單內容。



下圖是交貨單概覽介面，交貨單由抬頭和多個頁籤組成。

表身細節點擊條目下的細節查看按鈕。

以上查看完成，就可以點擊工具列中”過帳發貨”按鈕（过帐发货），注意不是保存鍵。

https://www.fenginfo.com/1373.html

➢ 發票（Billing Document）依照銷售訂單或出貨單，建立請款文件，立客戶應收帳款

- 說明

當向客戶發出貨物的同時，一般還需要向客戶出具發票（Billing Document），發票是參考交貨單生成，包含確認收款物件、發票金額、稅額等資訊。本操作屬於 SD 模組，操作後只增加對客戶的應收帳款，不涉及 FI 模組應收賬款業務。雖然向客戶出具發票是參考交貨單，但在跨公司銷售、STO 跨公司轉儲等業務中，還會參考交貨單出具公司間發票。

在發票類型定義中，有一個參數“過帳凍結”，如果勾選是兩步法過帳，非勾選則為一步法：

一步法，用事物碼 VF01 創建發票，同時生成會計憑證；

兩步法，用事物碼 VF01 創建發票但不生成憑證，用 VF02 批准生成會計憑證。

採用兩步法過帳，可由業務人員用 VF01 創建發票及 VF02 修改（不給批准許可權），由財務人員檢查無誤後用 VF02 過帳生成會計憑證。

此項操作需要但不限於以下條件：

付款客戶維護了相關公司代碼視圖，參見《客戶主資料_公司代碼視圖》；

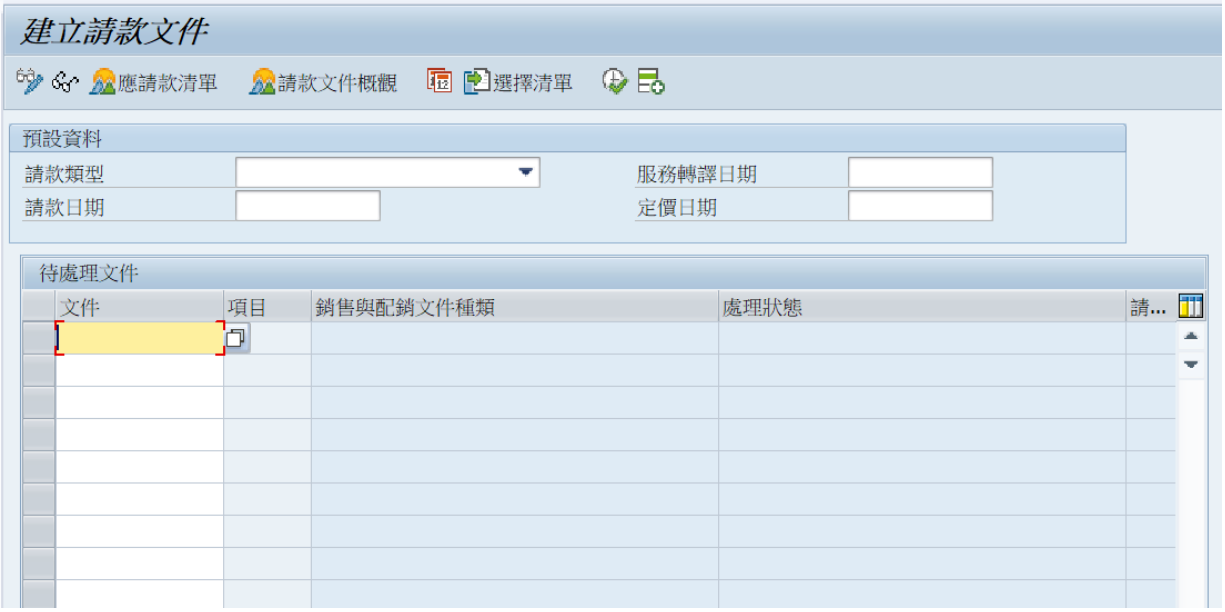
定價過程能夠確認，參見《銷售定價過程的確定》。

發票類型定義參見《定義發票類型(Billing Types)》; 每張發票包含一個或多個條目，每個條目也有專案類別(Item Categories)，但不需要定義，而是從參照憑證複製而來。

由於發票涉及結算價格，自銷售訂單到出具發票會有一定日期間隔，而作為條件類型的價格可能會隨之變化，在做 VF01 出具發票時，默認計價是當前日期，如有業務需要，可手工修改此日期。

- 操作

在前台輸入事物碼 VF01，按 ENTER 進入下圖所示。



輸入參考的憑證單號，可以是銷售訂單號或交貨單號，本例為交貨單號。在此注意抬頭右側的定價日期，此例不輸入值則取定價取當前日期。輸入憑證號後，按 ENTER 鍵進入，如下圖所示。

上圖是出具發票總覽介面，抬頭部分可以看到憑證類型為 [F2 發票]，付款方為 T-C10001，淨價值為 1,200.00 CNY；明細部分可以看到一個出具發票專案的資訊。點擊專案欄（介面中顯示”10”的位置），進入項目(專案)明細。

上圖是項目明細的條件標籤頁，在定價元素中可以看到發票的價格為 600 CNY，在這裡可以手工修改；銷項稅為 17.00% 。頭部顯示此項的發票金額 1,200.00CNY，稅款為 204.00 CNY。

點擊”分析”按鈕（ ）進入圖 4，可以見到採用的定價過程名為 [RVAACN 標準]，這是由銷售範圍（Sales Area）、發票的憑證定價過程、客戶定價過程等的組合確定的，參見《銷售定價過程的確定》。

項目金額確認後，點擊”出具發票項”按鈕（出具发票项）返回總覽介面。

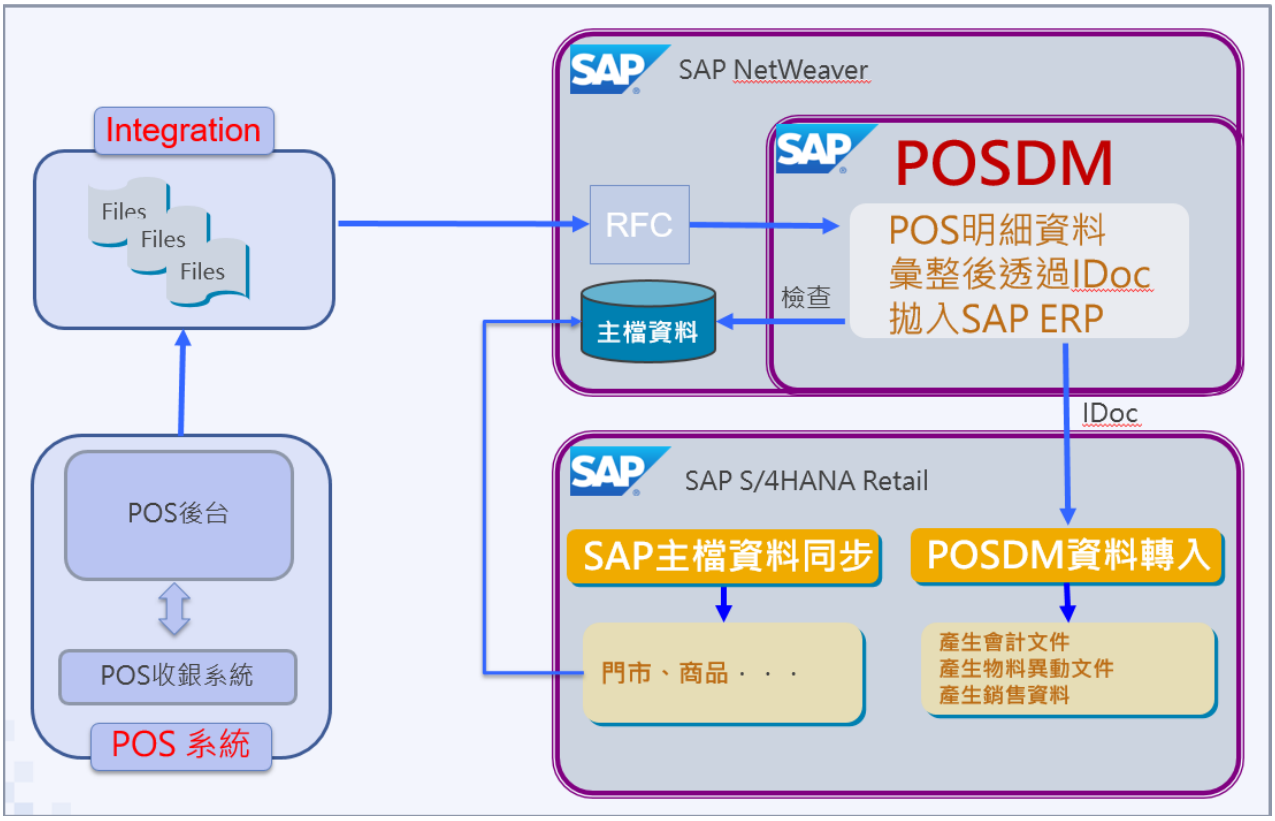
在總覽介面，點擊抬頭按鈕（）或功能表”轉到→抬頭→抬頭”，進入抬頭詳細介面如圖 5 所示。

抬頭的資訊是以數個標籤頁顯示的，圖 5 顯示的是條件標籤頁，可以看到各發票項目相加後的結果。查看清楚後，點擊保存按鈕，生成發票憑證，狀態標顯示發票號（xxxx 號），手工記下此憑證號，以便後續操作參照使用。

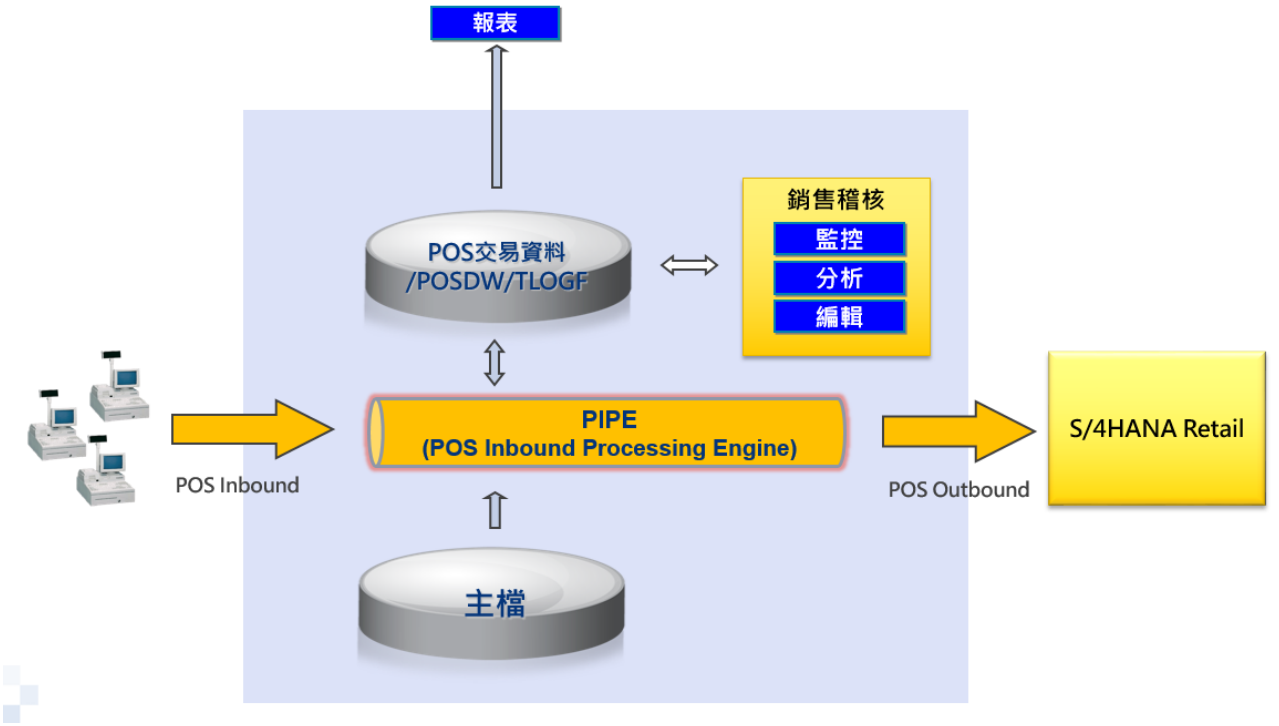
https://www.itsfun.com.tw/SAP+POS+DM/wiki-7167844-7624824

➢ POSDM

- SAP S/4HANA Retail 與外部業務系統資料交換



- POSDM 架構



- SAP POS DM 由內向接口，外向接口和 PIPE 組成。POS DM 收集、校驗以及處理銷售、POS 控制數據和日結記錄。POS DM 將校驗後的 POS 信息一方面傳遞給 SAP BI 進行報表分析，另一方面傳遞給 ERP 進行其它業務處理。國內 SAP 人才的稀缺，直接制約著 SAP 的發展，自 2008 年，由 51SAP 教育中心發起的以項目實戰為核心的 SAP 培訓受到國內市場的廣泛關注，伴隨著時間的發展，51SAP 已在全國成立了 20 個教學培訓中心，國內的 SAP 人才市場需求也得到進一步的緩解。

簡介

POS 數據是通過零售商門店收銀機和其它銷售終端收集的數據。這些數據首先需要通過數據校驗，然後傳輸到後續系統進行分析和跟蹤。POS DM 是收集、校驗以及傳輸 POS 數據的場所和工具。

處理狀態

SAP POS DM 由。POS DM 收集、校驗以及處理銷售、POS 控制數據和日結記錄。POS DM 將校驗後的 POS 信息一方面傳遞給 SAP BI 進行報表分析，另一方面傳遞給 ERP 進行其它業務處理。
PIPE(POS Inbound Processing Engine) 是 SAP POS DM 解決方案的一部分，用以審計和處理 POS 數據。PIPE 還提供了一個狀態監控操作台來管理和控制 POS 數據的處理狀態。
內向接口是一個被 SAP XI 調用的函式模組，用來收集來自於門店產生的符合 ARTS 標準的數據檔案。ARTS (Association for Retail Technology Standards) 零售行業技術標準組織是由零售商發起的會員組織。該組織專注於建立開放的環境以供零售商和技術提供商共同制定零售行業技術標準。
外向接口是一個 SAP 程式，將經過 POS DM 校驗和處理的數據傳輸到 ERP 進行處理。
POS DM 所用到的主數據是從 SAP ERP 抽取到 SAP BI 的主數據。

https://wiki.scn.sap.com/wiki/pages/viewpage.action?pageId=113246846

The SAP for Retail solution portfolio supports with a cost effective, comprehensive Pos data management solution that enables "trickle-feed" processing. Cash register sales data from individual store locations can be transmitted to back-office systems on a regular basis throughout the business day. This application is called POS DM.

With POS DM you

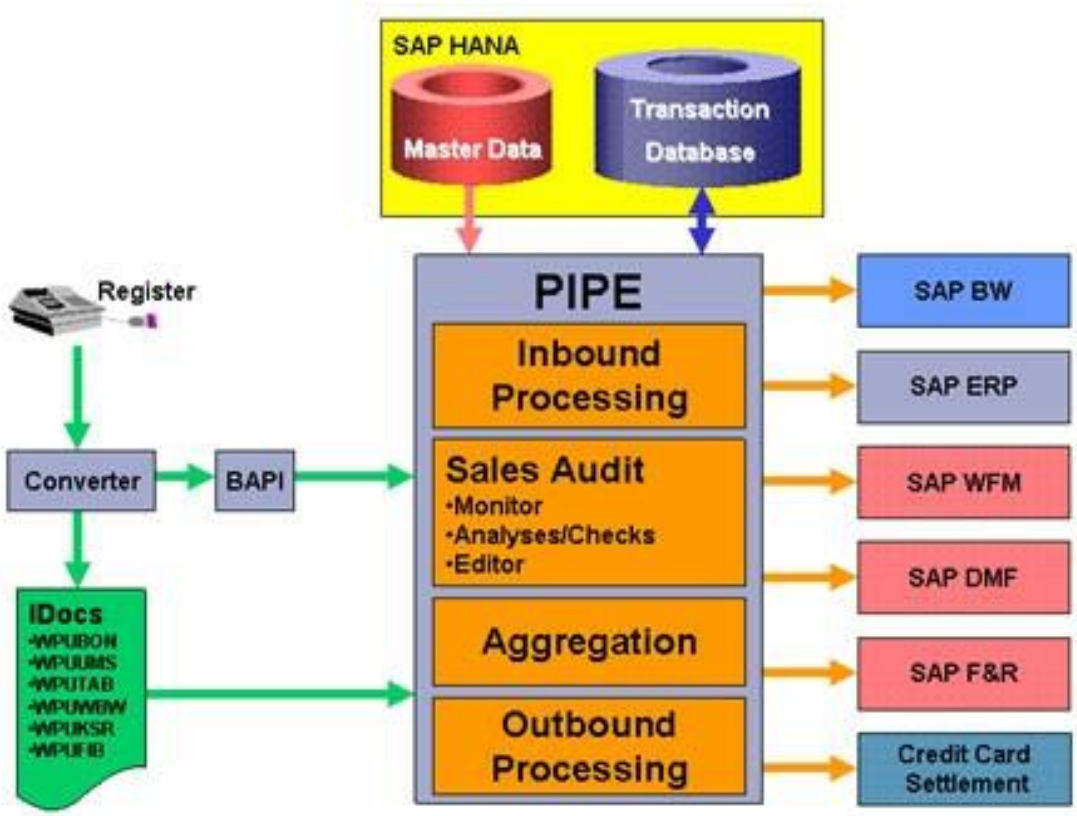
- can cleanse and audit your Pont-of-Sales transactions
- can see which products sell the strongest in all of its regions
- gauge price sensitivity across different markets or different areas within the market
- evaluate the effectiveness of promotions
- compare and contrasts how products sell when they are promoted versus being sold at everyday prices

https://www.stechies.com/pos-dm-pointofsale-data-management/

Definition -What is SAP POS-DM ?

Full form or **SAP PoSDM** stands for **(Point of Sale Data Management)**, the integrated tool of **SAP Point of Sale (POS) Data Management** is used to receive, review, process and analyze transactional data from business stores. It leads to positive improvements in decision making, at every life stage of a business enterprise. All information pertaining to business stores is available with the tools and integrated features of **SAP applications**, thereby making it essential for relevant departments to look for experts in the fields of SAP Point of Sale (POS) Data Management and other allied areas of concern.

With this system in place, data has only one entry point to pass through and allows users to get access to enriched and defined information, before the integration or compilation of any kind of data. The integral parts of SAP POS DM include POS Inbound Processing Engine and Analytics, along with diverse sub parts as POS Analytics and POS Analytics data supply.



- 跟銷售相關的一些 Table 架構

https://www.se80.co.uk/saptables/v/vbrp/vbrp.htm#relations

- Sales Document 銷售文件相關

  ADMIX-Help structure for reading via ADK in LDB(SD)

    VBUK-Sales Document: Header Status and Admin. Data

      VBRK-Billing Document: Header Data

        NAST-Message status

        VBFA-Sales Document Flow

        VBRP-Billing Document: Item Data

        KONV-Conditions (Transaction Data)

        VBPA-Sales Document: Partner

        SADR-Address Management: Company data

        VBSSSK-Table Generated for View VBSSSK

        EIKP-Foreign Trade: Export/Import Header Data

        EIPO-Foreign Trade: Export/Import: Item Data

        STXH-STXD SAPscript text file header

        STXL-STXD SAPscript text file lines

        VBRL-Sales Document: Invoice List

- 客戶主檔

  KNVK-Customer Master Contact Partner

    KNA1-General Data in Customer Master

      KNVV-Customer Master Sales Data

      KNVP-Customer Master Partner Functions

      VBAP-Sales Document: Item Data

      LIPS-SD document: Delivery: Item data

      VBRP-Billing Document: Item Data

      VBKA-Sales Activities

- 請款文件(會計會根據此文件再另行開立發票)

  VBRK-Billing Document: Header Data

    VBUK-Sales Doc.: Header Status and Administrative Data

    VBPA-Sales Document: Partner

    VBRP-Billing Document: Item Data

      VBFA-Sales Document Flow

      VBFAVG-Sales document: Cumulative records,followed by doc

      VBPAPO-Item partner of sales document (for LDB)

      KONV-Conditions (Transaction Data)

-

第八章 關於期初開帳

https://answers.sap.com/questions/6555230/sap%c3%83%c2%a4%c3%85%c2%a1%c3%83%c2%a7%c3%82%c2%ba%c3%83%c2%a6%c3%85%e2%80%9c%c3%83%c2%bf%c3%83%c2%a5%c3%8b%e2%80%a0%c3%83%c2%a5%c3%8b%e2%80%a0%c3%83%c2%a5%c3%83%c2%a6%c3%83%c2%a6%c3%83%c2%a5%c3%83%c2%a4%c3%82%c2%bd%c3%83%c2%a5%c3%82%c2%bc%c3%83%c2%a5.html

https://zhuanlan.zhihu.com/p/31348951

模組	資料內容	其他說明
	確認開帳日期	例如：2019-01-01 開帳 期初資料指 2018-12-31 的期末值
FI	資產科目(實帳戶)餘額 **總帳可以使用 F-02 批導所有的科目餘額。**	整理到開帳日前一天各科目的期末數字，如現金、各銀行帳戶的存款餘額等
	期初固定資產及餘額 **主數據用 TCODE:AS91 導入**	目前在有效期內的固定資產明細及扣除折舊後的殘值
	銀行往來帳戶餘額	銀行的帳戶明細及餘額
	客戶、廠商及商品的餘額，必須和 MM 建立內容一致	同上
MM	客戶資料及應收帳款餘額	往來客戶明細資料及帳款餘額
	廠商資料及應付帳款餘額	同上
	商品資料及期初庫存量(分倉)、庫存成本	同上，可由盤點得出正確數量，成本如無正確數字，也必須輸入近似值
	歷史交易紀錄(僅供查詢開帳日前的重要交易資料)→非必要	非必要，如果需要歷史交易明細供查詢的話。非標準功能，通常客製處理

期初資料導入有多種方式,一般常用的有如下幾種.

1) LSMW

2) SECATT

3) ABAP PROGRAM (USE BAPI)

4) ABAP PROGRAM (USE BDC)

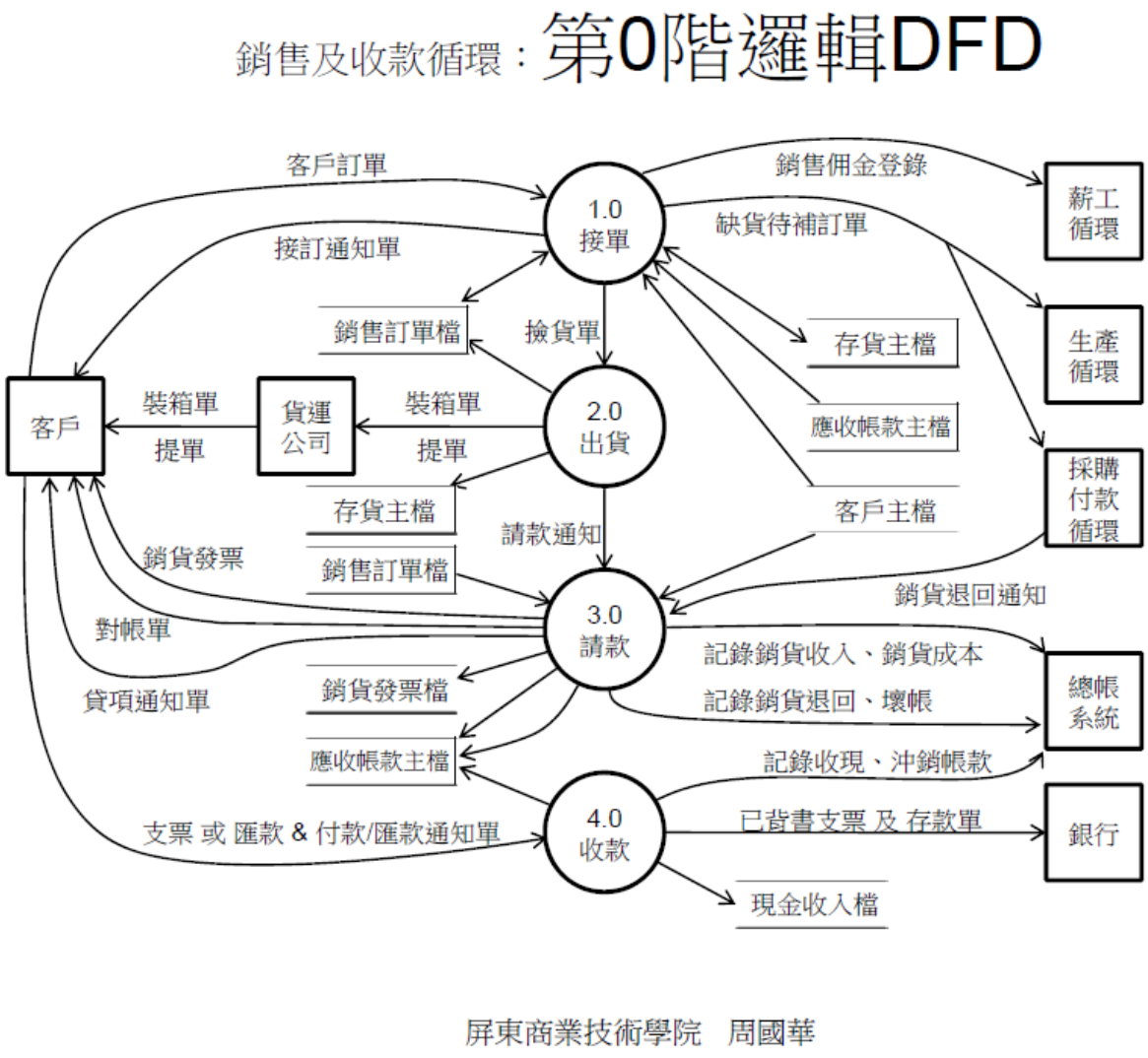
https://answers.sap.com/questions/3094610/what-is-catt-in-sap.html

一般來說，初始資料都可以用 CATT (Computer Aided Test Tool) 錄製相關的操作，然後在交易碼 SCAT 執行該 Task 上載特定格式的資料。客戶主檔，供應商主檔，總帳主檔，固定資產主檔等都可以用 CATT 上載的。

Tips：由於目前還沒研究過 FICO 的任何功能，所以 FICO 的開帳說明，都是直接引用相關 blog 的說明)。以下僅先就 MMSD 的期初開帳相關內容說明。

屏東商業技術學院的周國華教授，整理如下收款循環：
http://www.ais.nptu.edu.tw/ais/982%20materials/SalesCollection_20100211.pdf

從以下這張圖可以發現早期對 ERP 系統我一直存在一個誤解，這個誤解的根源是來自正航早期的進銷存系統，由於應收付帳款的原始憑證來自於進銷存的進出貨單，所以應收付帳款功能一直被放在進銷存的子功能中。但是實際上對帳及沖帳都是會計部門的權責，所以後來我修正為將應收付帳款獨立成一個子系統。看來這個修正是合理的，只是對 SAP 而言，因為整合度更高，在 MMSD 中，我完全看不到沖款的作業流程，看來應該都是在 FI 模組中處理。



Step 29-34 是沖帳作業的說明，但是看不出沖帳時，是針對單一單據沖帳? 還是針對客戶餘額沖帳? 也就是總額沖帳或單一單據沖帳?

這個部分也是屬於 FI 的功能，稍後如果有時間深入研究再補充相關的說明。

3.0
- 23. 通知總帳系統，記錄銷貨收入及銷貨成本。
- 24. 定期郵寄對帳單給客戶。
- 25. 收到採購付款循環(驗收部門)銷貨退回通知。
- 26. 寄貸項通知單給客戶。
- 27. 更新應收帳款主檔，沖銷應收客戶明細帳。
- 28. 通知總帳系統，記錄銷貨退回、壞帳等事項。

4.0
- 29. 收到客戶郵寄支票及付款通知單(remittance advice, RA)。
- 30. 確認支票金額與付款通知單上之金額相符。
  - ➢ 例外處理：支票金額與RA不符，通知客戶補正。
- 31. 立即在支票上背書，並列印存款單，將支票及存款單繳存銀行。
- 32. 將收現資料(支票號碼、金額、客戶編號、銷貨發票號碼等)輸入現金收入檔。
- 33. 更新應收帳款主檔，根據付款通知單沖銷應收客戶明細帳。
- 34. 通知總帳系統，記錄現金收入及沖銷應收帳款。

註：上述「例外處理」部分，並未繪製在第0階邏輯DFD內。

http://blog.sina.com.cn/s/blog_4c01b76501013isi.html

FICO 期初數據

（一）資料整理及導入

提供前月結帳完畢後的完整的資產負債表。

財務期初資料導入順序：

（1）維護尚未存在的供應商/客戶主資料，注意相同供應商/客戶可統一使用同一編號；

（2）固定資產、無形資產主資料及其明細導入，檢查無誤後過帳到總帳；

（3）應收/應付明細未清專案導入，要求進行帳齡的劃分；

（4） 一般總帳未清項和總帳科目餘額導入；

注釋：(未清項：指一個帳戶的項目可被該帳戶的其他項目結清或核銷掉。在清帳或核銷過程中涉及到的項目的合計金額必定為零。因此該帳戶的餘額總是等於未清掉的項目的合計金額。)

（5） 財務對帳，如果存在差異，財務手工調帳。

（6） 財務關帳

（7）資料導入結束後將期初導入的科目 9999999997、9999999998 和 9999999999 在公司代碼下凍結

（二）一般總帳科目及貨幣資金餘額

1. 一般總帳科目

提供前月結帳日的所有總帳科目餘額，按目前 SAP 系統所設置的總帳科目餘額單列

手工錄入會計憑證分錄為：

借或貸：一般總帳科目

貸或借：9999999999

2. 現金、銀行存款

a. 現金

提供前月結帳日的科目餘額

b. 銀行存款

區分帳戶分別提供前月結帳日的科目餘額

錄入會計憑證分錄為：

借：現金/銀行存款

貸：9999999999

（三）票據科目餘額

1．一般總帳科目：應收票據-銀行承兌/貼現在途

spacerun: yes"> “應收票據-銀行承兌”、“應收票據-貼現在途”是一般總帳科目管理，不是未清項明細管理，提供前月結帳日的總帳科目餘額，

錄入會計憑證分錄為：
借：應收票據
貸：9999999999
2. 總帳未清項明細管理：應收票據-商業承兌、應付票據-銀行承兌/商業承兌
在 SAP 中的總帳科目中，該科目為明細管理，必須按明細專案單列並區分借貸方單列；提供客戶(或供應商)應收/應付票據的特別總帳標識，應收/應付票據的出票人、受票人名稱、票據到期日等資訊；必須保證明細總額與總帳科目餘額一致

導入分錄為：
借：應收票據-商業承兌
貸：9999999999
或 spacerun: yes"> 借：9999999999
spacerun: yes"> 貸：應付票據-商業/銀行承兌

（四）存貨
1. 存貨資料導入的步驟：
（1）在導庫存數量之前，必須先按 sd 提供的物料編碼將標準價格導入系統，導入方式為 CATT，維護會計視圖 1 中的標準價格、評估類及成本視圖 1 中的物料源
（2）庫存導入的數量是盤點的實物庫存數量，由 sd 模組導入。產生的庫存計價金額為：庫存數量 × 標準價格，憑證分錄為：
借：庫存商品科目
貸：9999999998
（3）對於實物庫存與財務庫存的差異，應列明差異明細，並分類處理差異原因（例如系統上線前已收貨未收發票及已收發票未收貨的），通過上線期初產成品未達科目記錄差異值，待上線後逐步清理。
（4）對於標準成本與帳面加權平均成本的價格差異，一般記入產成品成本差異科目，在後期的業務中逐步參與分攤。
（5）上線後的未達清理，應通知配置人員把相應的清理庫存科目與清理移動類型（901、902）配置連接
2.後期正常未達差異的一般處理：
(1)上線前實物已入庫財務未收發票，取得發票時，財務直接記帳，分錄為：
借：庫存商品-期初上線未達
貸：xx 供應商/客戶
（2）上線前實物未入庫財務已收發票，真正實物入庫時，SD 做 901 或 902 移庫，產生財務憑證：
借：庫存商品-xx 型號 （正數或負數）
貸：庫存商品-期初上線未達 （正數或負數）
（五）往來賬明細未清項
1．供應商明細未清項
系統上線前應盡可能清理掉往來掛帳！保證最少的未清項導入系統！！！

將實行供應商往來管理的應付款包括其他應付款按供應商（供應商編碼）及其統馭總帳科目未清項明細一一列出（區分借貸方），並注明該筆業務的應付款到期日的起始基準日期（區分賬齡）。對於在 SAP 系統中還未有供應商主資料的（如，其他應付款供應商或員工等）必須先在系統裡建立相應的主資料

特別注意：SAP 中將個人員工作為供應商管理，對於員工主資料的維護必須與 HR 人力資源管理的人事編碼相一致，保證集團內的同一個人的個人往來主資料必須唯一。

a) 供應商應付未清項明細
b) 供應商預付未清項明細, 同時提供供應商預付款的特別總帳標識
導入分錄為：
借：9999999999
貸：XXXX 供應商

2.客戶明細未清項

系統上線前應盡可能清理掉往來掛帳！保證最少的未清項導入系統！！！

將實行客戶往來管理的應收款包括其他應收款按客戶（客戶編碼）及其統馭總帳科目未清項明細一一列出（區分借貸方），並注明該筆業務的應收款到期日的起始基準日期。對於在 SAP 系統中還未有客戶主資料的（如，其他應收款客戶等）必須先在系統裡建立相應的主資料
a) 客戶應收未清項明細
b) 客戶預收未清項明細,同時提供客戶預收款的特別總帳標識
導入分錄為：
借：XXXX 客戶
貸：9999999999
（六）固定資產餘額
系統上線前應盡可能清理掉已報廢的資產，保證帳實一致，明細帳總帳一致，保證最少的可用資產導入系統！！！

（1）固定資產和無形資產在期初資料導入時主資料與期初餘額同時導入，事務代碼為 AS91，更改使用 AS92；應將本年度購入和以前年度購入分別導入

（2）檢查固定資產的重點在資產的資本化日期、有效使用年限、至前一年度 12 月 31 日累計已提折舊及本年度 1 月-上線前一月累計折舊金額。然後按 SAP 的折舊計算原理測算一下月折舊額與以前的月折舊差異，如果差異較大，則需要重新測算倒推已使用年限，差異較小可不作調整，差異值將在上線當月進行調整並計入當月的期間損益。

SAP 理論計提月折舊={原值*（1-殘值率）-至前一年度 12 月 31 日累計已提折舊-至前一年度 12 月 31 日累計減值準備}/{（有效使用年限-至前一年度 12 月 31 日已提折舊年限）*12}

（3）將固定資產明細導入系統後，經核對無誤，再將科目餘額轉帳到總帳；通過事務代碼 OASV，將固定資產、累計折舊按資產類別錄入總額
會計分錄為：
借：固定資產
貸：累計折舊
貸：9999999997

（4）將固定資產明細與總帳科目進行核對，通過事務代碼 ABST、ABST2，檢查餘額為 0 則表示正確，否則進行更改
另注：假如上線時間為 1 月 1 號，導入資產時間為 12 月 31 日，此時首先要將資產的新財政年度打開 AJRW，資產計入新的年度即不需要作餘額結轉 AJAB。

這個時間還要注意一點：除了上面的導入資產時間之外還有一個導入期間的設置。都在 SPRO-FI-AA-ASSET DATA TRANSFER-PARAMETERS FOR DATA TRANSFER-DATA SPECIFICATIONS 裡頭，順序應該是配置期間（OAYC）->配置日期->AJAW 開新的資產年度->導入資料。 remark by: Immortalwill

（七）CO 上前線準備
（1） 期初成本要素維護：
（2） 成本中心設置：
（3） 成本物件層次結構設置
（4）創建生產成本主資料：成本收集器和成本物件層次
（5） 生產性成本中心計畫：
（6） 非生產性成本中心計畫

費用計畫：按生產性成本中心及作業類型（事務代碼：KP06），與生產性成本中心的費用計畫雷同。

https://www.fenginfo.com/1138.html

SAP 系統上線，通常需要進行初始化庫存操作，將目前庫存物料存入系統，包括數量和計價金額兩部分資料。特定 T-CODE 是 MB1C；MIGO 也可以進行操作。操作後生成移動類型為 561 的物料憑證，如果物料在工廠價值更新（詳見《維護物料類型的數量、價值更新》)，將同時生成會計憑證。生成的物料憑證可通過《查看物料憑證(Material Document)》查看詳細資訊。

如果需要生成會計憑證，則至少需要記以下總帳科目：

BSX（存貨記帳）：記庫存物料相關科目；

GBB（庫存記帳的沖銷輸入）：key 為 BSA，記初始化餘額相關科目；

PRD（成本價格差額）：key 為 PRA，記價格差異科目，此為可選項。

在操作中需注意本位幣金額的正確輸入。

除常規初始化，特殊庫存的初始化還有以下幾類：

MvT 561 E，對銷售訂單的初始化；

MvT 561 K，對供應商的寄售，物料所有權歸供應商，但寄存在本埠庫存地點；

MvT 561 O，供應商外協庫存的初始化；

MvT 561 W，對客戶的寄集，物料所有權歸本埠工廠，但寄存在客戶，操作時無需輸入庫存地點。

以上內容是說明 SAP 系統的內部作業流程，實際上由於品項眾多，一般都是執行批次轉入。

http://www.sapdoc.cn/sap/theme/1151.html

no	step	description	t-code
1	總帳期初	使用 f-02 直接錄入，期間選擇上線前一個會計期間。以下科目不需要導入： （1）應收科目。 （2）應付科目 （3）資產科目 （4）庫存科目	f-02
2	應收期初	使用 f-02 直接錄入，期間選擇上線前一個會計期間。 如果需要帳齡分不同日期導入。	f-02
3	應付期初	使用 f-02 直接錄入，期間選擇上線前一個會計期間。 如果需要帳齡分不同日期導入。	f-02
4	資產期初	取消資產科目統馭設置後，使用 as91 導入。會計憑證用 oasv 導入。通過事務代碼 abst、abst2，檢查餘額為 0 則表示正確	as91、oasv
5	存貨期初	使用移動類型 561 入庫，相應的存貨類科目憑證自動產生。 上線後的未達清理，應通知配置人員把相應的清理庫存科目與清理移動類型（如果901、902）配置好。	mb1c
6	未結採購訂單	將未結的採購訂單導入系統。 （1）未完成採購訂單的處理。 從舊系統下載未完成收貨的採購訂單，交各採購員進行核對確認。將這部分採購訂單資料錄入 sap 系統。 （2）採購訂單部分收貨，未進行發票校驗。 與供應商進行協商，重新將部分收貨的採購訂單分為已收貨與未收貨兩張獨立採購訂單： 已收貨部分的處理方法見 未收貨部分的處理方法見“未完成採購訂單的處理” （3）採購訂單已完全交貨，但未進行發票校驗。 當收到供應商發票時，採購員要確認該發票的物料正式入庫是否為上線日以前的（不含上線日），如果為上線前的收貨，可以直接交財務部處理（發票上標注為期初的收貨，系統無採購訂單）；財務直接作入當月的應付帳。 期初庫存導入： 借：庫存 貸：科目 99999999 財務根據舊系統的供應商暫估記帳： 借：科目 99999999 貸：應付暫估 供應商發票的處理：	

		借：應付暫估及進項稅 貸：應付	
7	未結生產訂單	兩種方式： 1.全部還原到材料，在錄入生產訂單後按照已領料補做領料。 2.做一筆憑證： 借：生產成本-訂單 貸：期初科目 在上線前做一次在製品結算，產生憑證： 借：在製品 貸：生產成本轉出-在製品 這一步驟的的作用在於給系統標記一個在製品，在下月入庫時可以紅沖這部分金額。	
8	未結銷售訂單		

說明：

1.導入資料後需要核對以下內容：

（1）資產各分類同總帳各科目是否一致。

（2）存貨金額是否同總帳科目是否一致。

（3）所有的期初導入科目餘額是否為零。

（4）資產負債表是否同原系統一致。

（5）如果需要以前月份的損益表，核對損益表同原系統是否一致。

2.結清期初導入科目餘額。

3.啟用資產科目的統馭科目設置。

4.假如上線時間為 1 月 1 號，導入資產時間為 12 月 31 日，此時首先要將資產的新財政年度打開 ajrw，資產計入新的年度即不需要作餘額結轉 ajab。

5.後期正常未達差異的一般處理：

(1)上線前實物已入庫財務未收發票，取得發票時，財務直接記帳，分錄為：

借：庫存商品-期初上線未達

貸：xx 供應商/客戶

(2)上線前實物未入庫財務已收發票，真正實物入庫時，sd 做 901 或 902 移庫，產生財務憑證：

借：庫存商品-xx 型號（正數或負數）

貸：庫存商品-期初上線未達（正數或負數）

6.本說明基於 sap ecc6。

第九章 SAP 成本計算有哪些方式

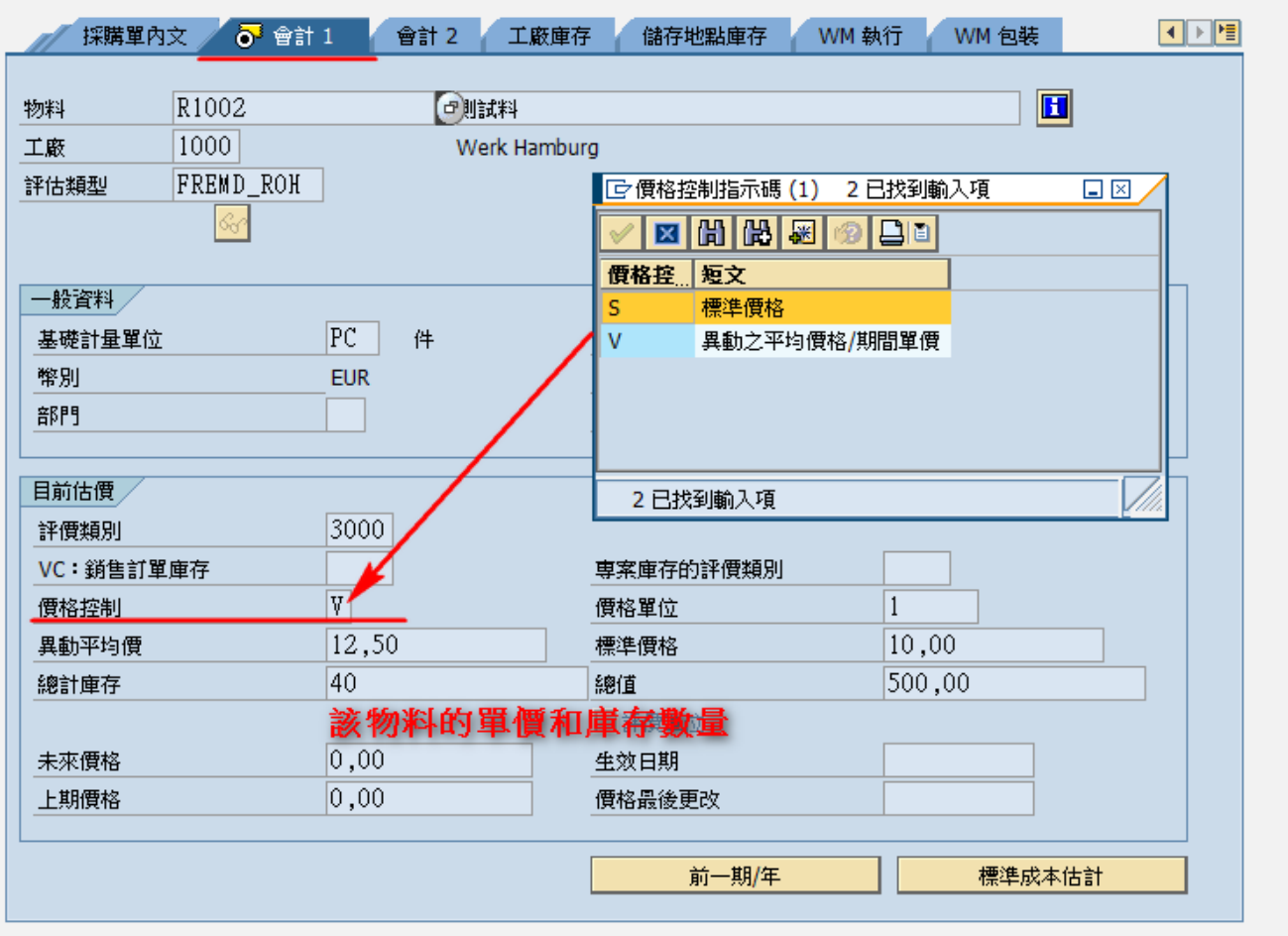
https://saperp.pixnet.net/blog/post/6227796
https://kknews.cc/zh-tw/news/e32ge6z.html

本文主要以 SAP 為例，論述在 ERP 系統的物料移動過程中，移動平均法和標準價格法這兩種存貨計價方法的應用。

企業的各種存貨，由於是分散購入或分批生產形成，同一次領用或發出的存貨，其單位成本可能有所不同。為了合理的核算低值易耗品和包裝物以外的存貨發出和領用的價值，就需要選擇一定的計價方法，對企業發出或領用的存貨進行計價，使企業的生產成本和產品銷售成本有統一的計價標準。我國企業所得稅法和會計制度提供了幾種可供選擇的會計制度，如先進先出法、後進先出法、移動平均法、計劃成本法、個別計價法等計價方法。本文主要以 SAP 為例，論述在 ERP 系統的物料移動過程中，移動平均法和標準價格法這兩種存貨計價方法的應用。

移動平均法能夠使管理當局及時瞭解存貨的結存情況，而且計算出的平均單位成本以及發出和結存的存貨成本比較客觀；標準成本法的按標準價格計價存貨，可以簡化會計處理工作，有利於考核部門業績。但二者在收發貨比較頻繁的企業，都需要進行大量的計算工作。ERP 系統強大的數據處理功能使這方面已經不成問題，在 ERP 系統運作中，許多模塊的多數功能都涉及到物料的移動及計價，系統將他們有機的整合在一起，並生成物料憑證和會計憑證。物料憑證反映每種物料的移動數量、方向時間及操作員；而財務憑證反映的是每種物料數量和價格的變化。根據這些憑證 可以跟蹤和分析每一筆物料的移動變化記錄。

ERP 系統中，存貨的計價方法主要控制在物料主數據內。物料主數據（Material Master Data)包括基本數據部分(Basic Data)、採購部分（Purchasing)、MRP 部分、會計部分(Accounting)及成本部分(Costing)。在建立每一種物料時，會計部分可設定選用移動平均價格(Moving Average Price)還是用計劃/標準價格(Standard Price)計價。選擇相應的方法後，如果該物料已經有歷史記錄，在會計部分還可以顯示該物料的單價和庫存。



移動平均法在系統收貨及收發票中的計價過程：

- 當採購主管在 SAP 中建立採購訂單時，首先要估計（Accrual）每一種物料的價格、運費及雜費。系統根據採購數量計算出總價值。
- 收貨時，根據實際收貨數量錄入系統，系統依據採購訂單價格計算出採購物料的總價值。下面舉例說明這一過程。

收貨過程

上期庫存數量為:100PC,移動平均價為 2.庫存金額為 2*100=200.

本期 PO 收貨為:100PC ,單價為:2.4,移動平均價為 2.2.庫存金額為:2.2*200=440.

對就 PO 收貨會計分錄:借:原材料 240 貸:暫估應付 240

- 在物料憑證中反映為原料庫物料增加。
- 在會計憑證中一方面反映存貨中該種物料的數量及金額增加，另一方面反映暫估應付賬款的增加。暫估應付款（GR/IR）是資產負債表中的可抵消科目。
- 移動平均價格被重新計算。

計算過程為：

New M.A.P=(100x2.00+100x2.40)/(100+100)=2.20

- 當收到供應商提供的發票後，首先檢查發票中的物料數量與系統中收貨的數量是否一致。確認無誤後錄入系統。

收發票過程

發票數量:100,單價:2.2.

會計分錄:借:暫估應付 240 貸:應付帳款 220, 貸:原材料 20

●系統會重新計算存貨總價值。如果估計的價格大於發票價格，其差額系統會依據訂單沖減庫存中物料的價值；反之，該差異會被分配到庫存中的物料價值中。

●暫估應付賬款科目被抵消和清除。

●依據發票金額，系統將該筆貨款傳送到應付賬款模塊。

●依據發票價格，移動平均價格重新被計算。

計算過程為：

New M.A.P=(100x2.00+100x2.20)/(100+100)=2.10

標準價格法在系統收貨及收發票中的計價過程：

例子同上

收貨過程

上期庫存數量為:100PC,材料價格為 2.庫存金額為 2*100=200.

本期 PO 收貨為:100PC ,單價為:2.4

對就 PO 收貨會計分錄:借:原材料 200 借:材料價格差異 40 貸:暫估應付 240

●收貨的原材料價格依照標準價格計算。

●暫估應付賬款依據採購訂單價格計算。

●訂單價格與標準價格的差異轉入材料價格差異科目。也就是說材料的實際價格=標準價格+材料價格差異。

●材料價格差異有兩種處理方法。一種是在當期已發出材料和期末結存材料之間進行分配，屬於已消耗材料應分配的差異，應從該科目中轉到相關科目。另一種是直接轉入當期損益。

收發票過程

發票數量:100,單價:2.2.

●此步操作對庫存材料沒有直接影響。

●暫估應付賬款科目被沖銷。

●依據發票價格記應付賬款科目。

●按照發票價格重新計算材料價格差異。

會計分錄:借:暫估應付 240 貸:應付帳款 220, 貸:材料價格差異 20

移動平均法在生產過程中的計價

在生產過程中，從原材料開始加工到成品完成，按照工藝流程和生產步驟，會存在多種在產品形態，不同的形

態的在產品消耗的活勞動和物耗勞動的價值構成了其生產成本。SAP 系統中，生產不同的在產品主要依據生產訂單，生產訂單通過 MRP 運行產生，系統根據標準物料清單，計算出生產該訂單的產品所耗的標準材料數 量，和標準的工時。當該訂單完成時通過一個"確認功能 "(Confirmation)，執行該功能後，系統增加庫存中的新產品，減少所耗材料的庫存。同時生成物料憑證和會計憑證。

從會計憑證看，這也是一個生產成本計算分配的過程。但此時的新產品的庫存價格並不是真正的移動平均價格，需要通過"調整功能"（Settlement）,將新產品的價格調整為移動平均價格。該功能不產生物料憑證，僅產生會計憑證，也就是說僅對庫存價格和生產成本計算有影響。一般來說，一個生產訂單依靠一道工序，當每一個訂單被完成後，在系統中必須執行這兩個功能，而且次序不能弄錯。另外，一定要按生產訂單的流程順序來執行，否則計算的半成品和相應產成品的價格是錯誤的，從管理角度講，影響客戶獲利能力的正確分析；從財務角度講，導致產品銷售成本錯誤，產品邊際貢獻錯誤，利潤總額，所得稅，淨利潤等一系列的數據錯誤。

下面舉例說明這些過程：

- 生產訂單 10000 號要求生產產品 Dummy 100 個。根據物料清單 100 個 Dummy 消耗 100 個 A，50 個 B。直接人工和製造費用暫不考慮。

  庫存餘額
  產成品 Dummy 庫存 100,單價 2.2,材料 A 庫存數量:150,價格:1 .材料 B 庫存數量:150,價格:0.8.

- 訂單完成確認過程（Confirmation)
  庫存餘額
  生產發料 A 100PC,B 50PC.會計分錄為:
  借:生產成本:140 貸:材料-A 100 貸:材料-B 40
  產成品 Dummy 完工入庫存 100PC
  借:產成品 220 貸:生產成本 220
  ● Dummy 材料的移動平均價沒有變化，庫存數量增加 100 個，金額增加了為 100*2.20=220.00。
  ●該移動平均價為存貨增加前價格 2.20 元。
  ●原材料 A 和 B 的庫存數量減少。
  ●變動成本科目產生差異為 80.00。

- 價格差異調整過程(Settlement)
  庫存餘額
  借:生產成本 80 貸:產成本 80
  ●Dummy 的移動平均價發生變化。該物料的庫存價值被沖減 80 元
  其移動平均價計算過程為：
  New M.A.P=（庫存原值+新增價值-與所耗材料價值差異）/期初庫存量+新增庫存價值
  =（220+220-80）/100+100=1.80
  也就是說：

New M.A.P（庫存原值+所耗材料價值）/期初庫存量+新增庫存價值

=（220+100*1.00+50*0.80）/100+100=1.80

●變動成本科目為零。

●此步驟對庫存材料數量沒有影響，僅調整價值。

標準成本法在生產過程中的計價比較容易理解。原材料按照標準價格發貨到生產訂單，這時原材料減少，變動成本增加；在生產訂單結束確認後，產生的產成品按標準價格計價，同時變動成本減少；原材料標準價格與產成品之間的標準價格差異均被轉到當期的損益中。

移動平均價格和標準價格計價的主要區別是，移動平均價主要通過當前收貨價格計算得出，基本接近於的市場價格；但建立採購訂單時如果估計價格與實際價格相比過高或過低，在收到發票前會造成系統存貨價格不准。而標準價格是一種計劃價格，標準價格與實際價格的差異不再分配到存貨中，直接轉到當期損益。如果標準價格制定合理，通過分析該差異，可以對採購部門和生產部門進行考核，促使其降低採購成本，節約支出。但標準價值計算的存貨不能準確反映當時市場價格。

存貨的計價方法不同，對企業的財務狀況、經營成果和現金流量會產生不同的影響。所以採用移動平均價格還是標準價格對存貨計價，應該結合企業的具體實際情況，合理的選擇計價方法。

第 10 章 ABAP 入門基本要點

➢ ABAP 的一些基本注意事項：

https://blog.csdn.net/szlaptop/article/details/8465103
https://blog.csdn.net/weixin_38500922/article/details/96423835
https://noahchou.wordpress.com/2011/11/17/abap%e7%a8%8b%e5%bc%8f%e5%9f%ba%e6%9c%ac%e8%aa%9e%e6%b3%95%e5%92%8c%e6%a0%bc%e5%bc%8f/

https://www.itread01.com/content/1548803704.html

在第一章中，我們有簡單的說明 ABAP 在 SAP 的地位及重要性，如果想要邁進 ABAP 的大門，你至少需要了解下面幾個方面，特別說明，ABAP 是完整的開發工具，本文只是簡單的介紹一些觀念重點，並不是短期快速入門的內容：

一、關於符號和字母,記住下面這幾點，你程式的小錯誤基本減少 99%

- 語句結束用"."。
- 沒有雙引號只有單引號。Error Exp: WRITE "AAA".
- 一些保留字元需要單引號括起來，比如.等。Error Exp: a = 1.7
- 運算子前後有空格，也就是=/+等這些前後一定要有空格。Error Exp: a=b
- 雖然 abap 不區分大小寫，但是這個也是一個傳說.....，你還是儘量全部大寫！特別是功能碼、欄位名等。
- 整行註釋用 *,語句註釋用 "，多行批量註釋與反註釋: ctrl+, ctrl+.
- 自己寫的程式(含新增的 Table、View 等)，只能以 z 和 y 開頭
- / 表示換行

二、基本的語法語句: 字元操作、邏輯迴圈....簡單說明如下

宣告語句：TYPES、DATA、PARAMETERS、TABLES

基本操作：WRITE、MOVE

字元操作：CONCATTENATE、SPLIT、CONDENSE、SEARCH、REPLACE、TRANSLATE

邏輯迴圈：IF、WHILE、CASE、DO (迴圈完全中止 exit,條件終止 check +exit ,迴圈內中止 continue)

呼叫語句：PERFORM, CALL, SUBMIT, LEAVE TO

函式定義： FORM/ENDFORM, FUNCTION/ENDFUNCTION, MODULE/ENDMODULE

比較運算：=、<>.....

三、資料型別：

常用的 C（文字）、I（整數）P 和 F（小數）、N（可把數值作為文字）、X （16 進位制）、D（日期）

四、SAP 的開發大多數情況都是訪問資料，所以你需要清楚瞭解一下內表(internal Table)：

關於 Internal Table 請參考上一節的詳細說明(非常重要)。

五、熟悉運用以下的系統定義的常用變量

SY-SUBRC=0 表示找到記錄，sy-dbcnt 表示找到的記錄數量。

SY-DBLNT: 被處理過的記錄的筆數

SY-UNAME: 當前使用者登入 SAP 的 USERNAME

SY-DATUM: 當前系統日期

SY-UZEIT: 當前系統時間

SY-TCODE: 當前執行程式的 Transaction code

SY-INDEX : 當前 LOOP 迴圈過的次數

SY-TABIX: 當前處理的是 internal table 的第幾筆

SY-TMAXL: Internal table 的總筆數

SY-MANDT: SAP 的 Client ID

SPACE: 空字串

六、SAP 的二次開發大多是操作資料，而那些資料都是存放在資料庫裡面，ABAP 語法有透過 Open SQL 直接整合常用的 SQL 語法，方便開發人員存取資料內容，Open SQL 的語法近似標準 SQL，但功能較陽春，Open SQL 的優點是不必針對不同的資料庫撰寫不同的 SQL 語法，它會自動將 Open SQL 轉為各 DB 能執行的語法，也因為如此，所以提供的功能當然就會較陽春，不過根據我的自學過程看來，忍一忍還是可用，況且，ABAP 還是提供 EXEC 語法，還是可以針對特定資料庫直接下命令。

Open SQL 稍後會有專章說明。

ABAP 開發內容主要為報表、螢幕、表單、增強、接口(轉分享 SAP 癡貨的 ABAP 學習地圖)

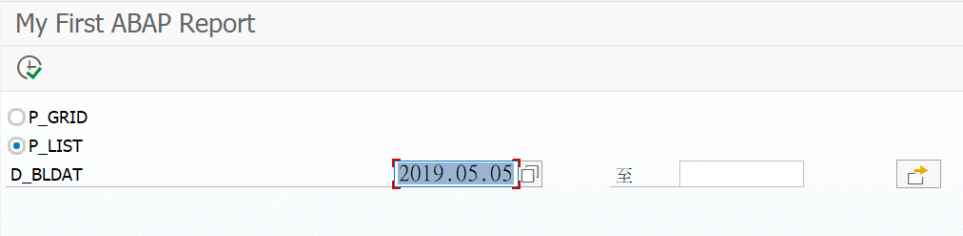
- 報表：輸出用戶指定需求的數據信息。像在財務工作中經常使用的現金流量表、資產負債表、損益表。這裡主要需要掌握數據結構定義、資料庫操作、內表操作、ALV（輸出顯示用）等。
- 螢幕：輸入輸出數據，實現與用戶的互動。主要需要掌握螢幕流原理。
- 表單：將實際工作中需要列印的表格在 SAP 裡面進行實現。主要需要掌握 Smartform。
- 增強：SAP 提供的標準功能不能夠完全覆蓋每家企業的實際業務。SAP 針對這一現象，對它的標準功能預留了一些接口，我們通過這些接口技術來實現企業獨特的業務需求。這裡主要需要掌握出口、BADI、增強點技術。
- 接口：實現 SAP 間、SAP 與其他軟體間的一種數據交互。如汽車生產廠商向零部件生產商進行訂貨。這裡主要是 RFC、IDOC、Webservice 等。

我個人認為，看範例學程式是最有效率的，所以接下來，就由淺入深的多支範例程式來開始 ABAP 的學習之旅

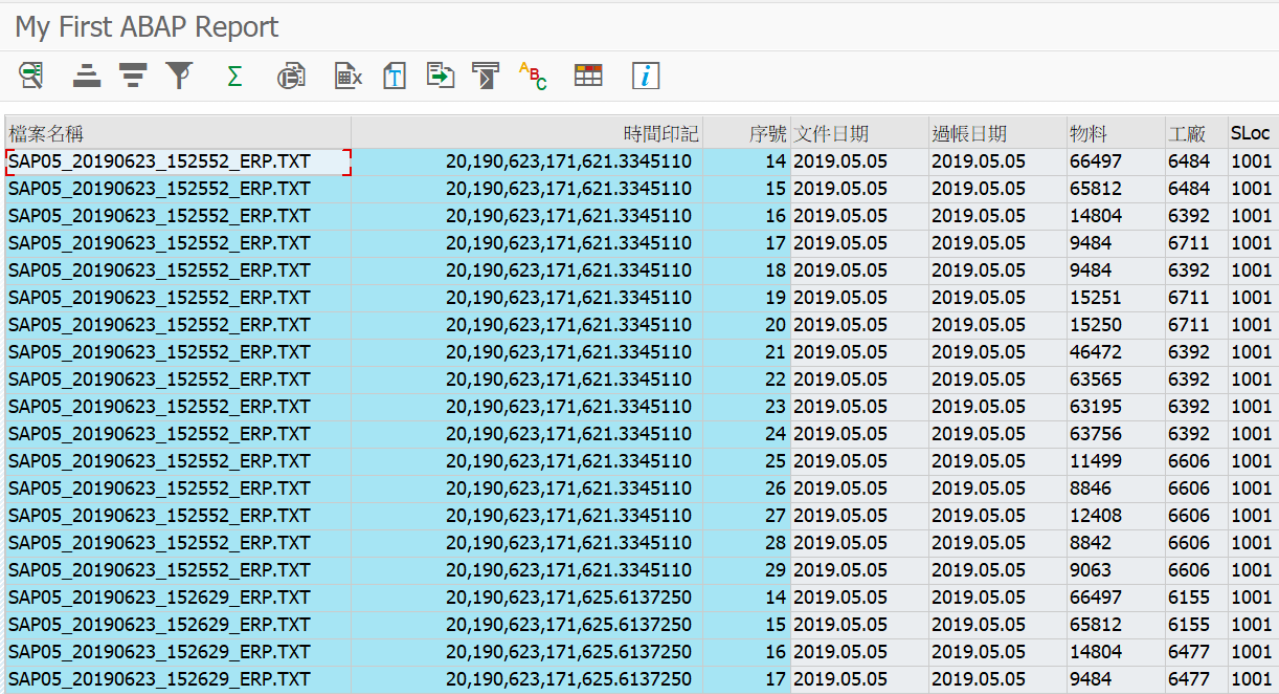
以下是第一支代表性的報表程式，可以讓有程式設計基礎的開發人員能一眼就明瞭 ABAP 的基本架構。

```
*&---------------------------------------------------------------------*
*& Report ZMC001
*&---------------------------------------------------------------------*
REPORT ZMC001.
TYPE-POOLS: slis.

DATA itab TYPE TABLE OF zmm011_t01 WITH HEADER LINE.

PARAMETERS:
  p_grid RADIOBUTTON GROUP rg1 DEFAULT 'X',
  p_list RADIOBUTTON GROUP rg1.

SELECT-OPTIONS:
  d_BLDAT FOR itab-BLDAT.

START-OF-SELECTION.
  SELECT * INTO TABLE @itab FROM zmm011_t01 UP TO 100 ROWS
    where BLDAT in @d_BLDAT.

CASE 'X'.
  WHEN p_grid.
    CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
  exporting
    I_STRUCTURE_NAME        =     'zmm011_t01'   " Internal output table structure
    T_OUTTAB                =     itab.          " Table with data to be displayed
  WHEN p_list.
    CALL FUNCTION 'REUSE_ALV_LIST_DISPLAY'
  exporting
    I_STRUCTURE_NAME        =     'zmm011_t01'   " Internal output table structure
    T_OUTTAB                =     itab.          " Table with data to be displayed
  WHEN OTHERS.
ENDCASE.
```

前頁的程式按 F8 後執行的畫面

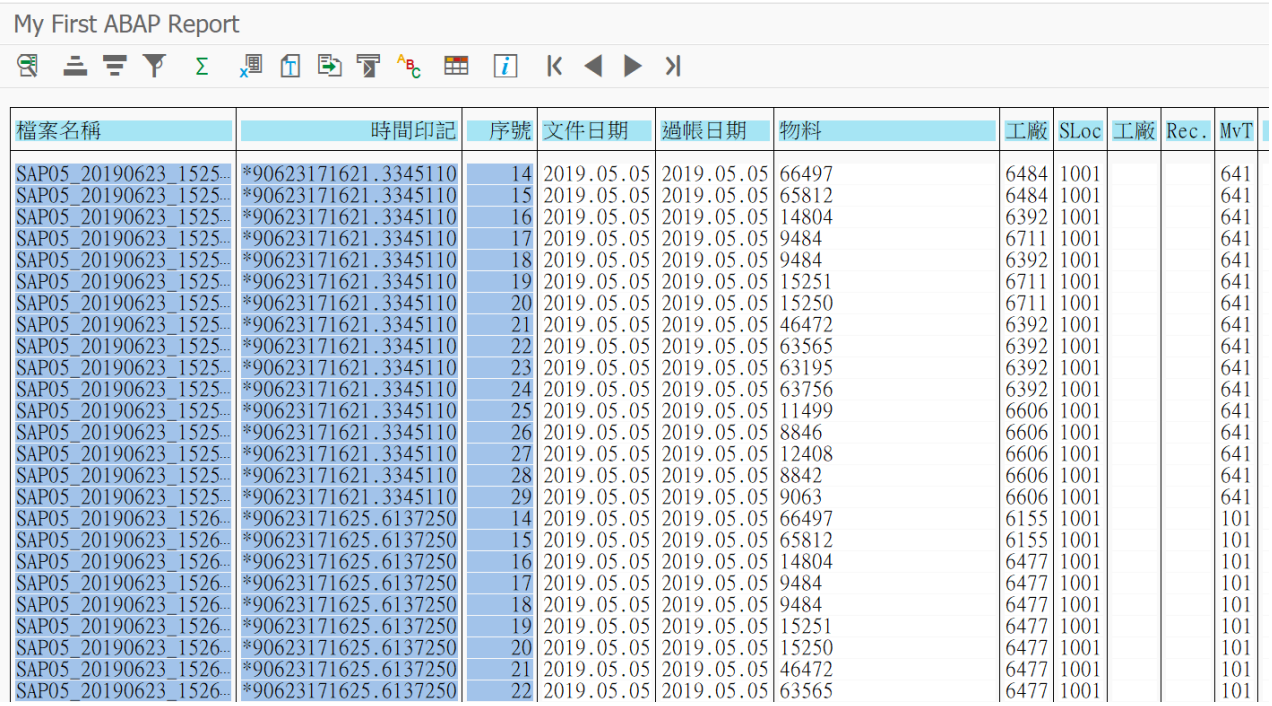


選 P_GRID 後的報表執行畫面(日期輸入 20190505)



檔案名稱	時間印記	序號	文件日期	過帳日期	物料	工廠	SLoc
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	14	2019.05.05	2019.05.05	66497	6484	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	15	2019.05.05	2019.05.05	65812	6484	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	16	2019.05.05	2019.05.05	14804	6392	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	17	2019.05.05	2019.05.05	9484	6711	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	18	2019.05.05	2019.05.05	9484	6392	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	19	2019.05.05	2019.05.05	15251	6711	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	20	2019.05.05	2019.05.05	15250	6711	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	21	2019.05.05	2019.05.05	46472	6392	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	22	2019.05.05	2019.05.05	63565	6392	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	23	2019.05.05	2019.05.05	63195	6392	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	24	2019.05.05	2019.05.05	63756	6392	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	25	2019.05.05	2019.05.05	11499	6606	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	26	2019.05.05	2019.05.05	8846	6606	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	27	2019.05.05	2019.05.05	12408	6606	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	28	2019.05.05	2019.05.05	8842	6606	1001
SAP05_20190623_152552_ERP.TXT	20,190,623,171,621.3345110	29	2019.05.05	2019.05.05	9063	6606	1001
SAP05_20190623_152629_ERP.TXT	20,190,623,171,625.6137250	14	2019.05.05	2019.05.05	66497	6155	1001
SAP05_20190623_152629_ERP.TXT	20,190,623,171,625.6137250	15	2019.05.05	2019.05.05	65812	6155	1001
SAP05_20190623_152629_ERP.TXT	20,190,623,171,625.6137250	16	2019.05.05	2019.05.05	14804	6477	1001
SAP05_20190623_152629_ERP.TXT	20,190,623,171,625.6137250	17	2019.05.05	2019.05.05	9484	6477	1001

選 P_LIST 後的報表執行畫面(日期輸入 20190505)



檔案名稱	時間印記	序號	文件日期	過帳日期	物料	工廠	SLoc	工廠	Rec.	MvT
SAP05_20190623_1525...	*90623171621.3345110	14	2019.05.05	2019.05.05	66497	6484	1001			641
SAP05_20190623_1525...	*90623171621.3345110	15	2019.05.05	2019.05.05	65812	6484	1001			641
SAP05_20190623_1525...	*90623171621.3345110	16	2019.05.05	2019.05.05	14804	6392	1001			641
SAP05_20190623_1525...	*90623171621.3345110	17	2019.05.05	2019.05.05	9484	6711	1001			641
SAP05_20190623_1525...	*90623171621.3345110	18	2019.05.05	2019.05.05	9484	6392	1001			641
SAP05_20190623_1525...	*90623171621.3345110	19	2019.05.05	2019.05.05	15251	6711	1001			641
SAP05_20190623_1525...	*90623171621.3345110	20	2019.05.05	2019.05.05	15250	6711	1001			641
SAP05_20190623_1525...	*90623171621.3345110	21	2019.05.05	2019.05.05	46472	6392	1001			641
SAP05_20190623_1525...	*90623171621.3345110	22	2019.05.05	2019.05.05	63565	6392	1001			641
SAP05_20190623_1525...	*90623171621.3345110	23	2019.05.05	2019.05.05	63195	6392	1001			641
SAP05_20190623_1525...	*90623171621.3345110	24	2019.05.05	2019.05.05	63756	6392	1001			641
SAP05_20190623_1525...	*90623171621.3345110	25	2019.05.05	2019.05.05	11499	6606	1001			641
SAP05_20190623_1525...	*90623171621.3345110	26	2019.05.05	2019.05.05	8846	6606	1001			641
SAP05_20190623_1525...	*90623171621.3345110	27	2019.05.05	2019.05.05	12408	6606	1001			641
SAP05_20190623_1525...	*90623171621.3345110	28	2019.05.05	2019.05.05	8842	6606	1001			641
SAP05_20190623_1525...	*90623171621.3345110	29	2019.05.05	2019.05.05	9063	6606	1001			641
SAP05_20190623_1526...	*90623171625.6137250	14	2019.05.05	2019.05.05	66497	6155	1001			101
SAP05_20190623_1526...	*90623171625.6137250	15	2019.05.05	2019.05.05	65812	6155	1001			101
SAP05_20190623_1526...	*90623171625.6137250	16	2019.05.05	2019.05.05	14804	6477	1001			101
SAP05_20190623_1526...	*90623171625.6137250	17	2019.05.05	2019.05.05	9484	6477	1001			101
SAP05_20190623_1526...	*90623171625.6137250	18	2019.05.05	2019.05.05	9484	6477	1001			101
SAP05_20190623_1526...	*90623171625.6137250	19	2019.05.05	2019.05.05	15251	6477	1001			101
SAP05_20190623_1526...	*90623171625.6137250	20	2019.05.05	2019.05.05	15250	6477	1001			101
SAP05_20190623_1526...	*90623171625.6137250	21	2019.05.05	2019.05.05	46472	6477	1001			101
SAP05_20190623_1526...	*90623171625.6137250	22	2019.05.05	2019.05.05	63565	6477	1001			101

有程式開發經驗的童鞋，應該一眼就理解上面這支程式的架構寫法

- Line1-Line3 程式的註解，在 ABAP 中，行開始處輸入星號 (*) 或是要將某行的一部分變為注解，則在注解之前輸入雙引號 (") 即可。ABAP 句尾固定是用句號當作是一行指令的結束，習慣就好。
- Line 4 是本報表的名稱，一定要是 REPORT 開頭
- Line 7 宣告一個 DataSet (內表，InternalTable)
- Line9-Line11 報表的輸入條件參數宣告(單純的 text)
- Line13-Line14 SELECT-OPTIONS 報表變數宣告，支援開窗查詢功能
- Line16 START-OF-SELECTION ABAP 程式的開始執行點，類似 Java 的 main()
- Line17-Line18 取得報表要顯示的資料，並寫到 DataSet(內表) 中
- Line20-Line32 是一個 case when 判斷程式，根據 PARAMETER 的輸入，決定要執行哪一段程式
- Line22-Line25 呼叫 ABAP 內建的 FunctionModules，執行報表的顯示。類似將取得的 DataSet 資料，綁定到 ABAP 的 DataGrid 顯示元件。
- Line27-Line30 同上

通過上面這支簡單的範例說明，就可以了解要如何完成一張報表的基本架構。除了宣告的變數較特殊外，基本上和傳統程式差異不大。因為 ABAP 是一個完全獨立的開發語言，自成一個開發體系，所以要深入了解，就必須繼續研讀相關的文件(例如下面的這些 Function Modules 要如何使用?) 並多看相關的範例程式碼。除了報表外，ABAP 也可以開發表單程式，但這是 another story 了。

Contents

209 REUSE_ALV_FIELDCATALOG_MERGE 276
210 REUSE_ALV_GRID_DISPLAY 277
211 REUSE_ALV_GRID_LAYOUT_INFO_GET 282
212 REUSE_ALV_GRID_LAYOUT_INFO_SET 283
213 REUSE_ALV_HIERSEQ_LIST_DISPLAY 283
214 REUSE_ALV_LIST_DISPLAY 286
215 REUSE_ALV_LIST_LAYOUT_INFO_GET 286
216 REUSE_ALV_LIST_LAYOUT_INFO_SET 289
217 REUSE_ALV_POPUP_TO_SELECT 289
218 REUSE_ALV_VARIANT_DEFAULT_GET 291
219 REUSE_ALV_VARIANT_EXISTENCE 292
220 REUSE_ALV_VARIANT_F4 293
221 REUSE_ALV_VARIANT_SELECT 294
222 RPY_DYNPRO_READ 296
223 RPY_TRANSACTION_READ 297
224 RS_COVERPAGE_SELECTIONS 298
225 RS_CREATE_VARIANT 299
226 RS_EXTERNAL_SELSCREEN_STATUS 300
227 RS_FUNCTION_POOL_CONTENTS 301
228 RS_REFRESH_FROM_SELECTOPTIONS 301
229 RS_REPORTSELECTIONS_INFO 302
230 RS_SELECTIONSCREEN_READ 303
231 RS_SET_SELSCREEN_STATUS 303
232 RS_SELECTIONSCREEN_UPDATE 304
233 RS_SUBMIT_INFO 305
234 RS_SUPPORT_SELECTIONS 306
235 RS_TOOL_ACCESS 306

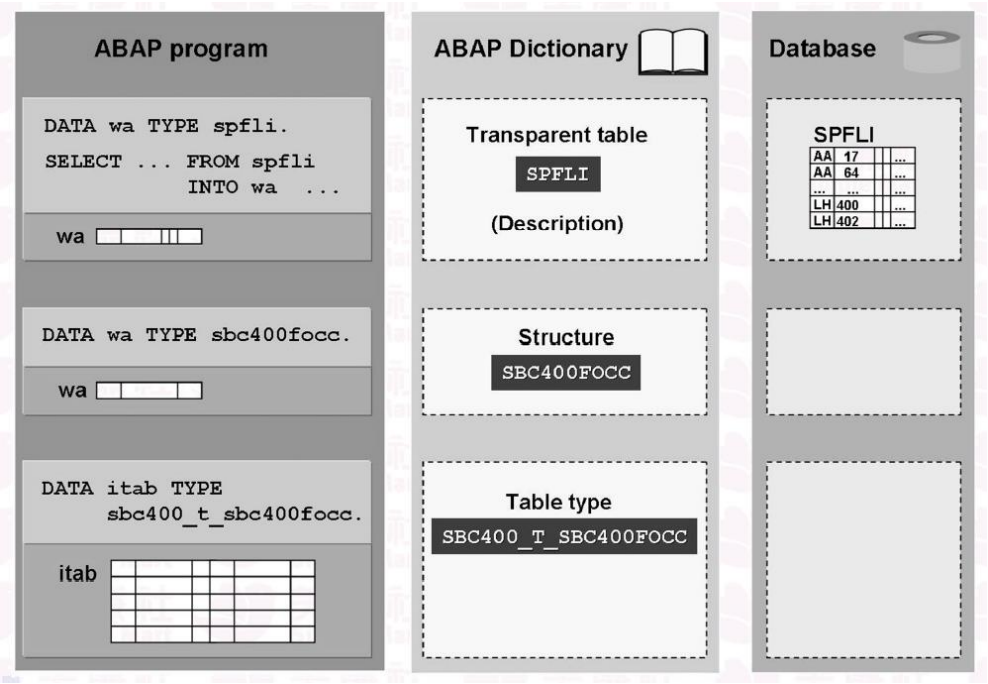
➢ 關於 Internal Table 下面有一個 Sample 簡單的說明其觀念

Internal Table 是一種 Data Structure(數據結構)，類似於其它語言中的 Structure，Internal Table 與資料結構體基本類似，它同樣是程式執行中被創建的一個存儲空間，它是一個可包含多條記錄的資料表，它可以由幾個不同類型的欄位(field)組成，用來表示具有不同屬性的某一事物，單獨一筆數據表示某個事物，多筆數據表示具有相同屬性的多個事物。

Internal Tables 共有以下 3 種類型。

- Standard
- Sorted
- Hashed

	Index table		Hashed table
Table kind	Standard table	Sorted table	Hashed table
Index access	V	V	
Key access	V	V	V
Uniqueness of key	Non-unique	Unique Non-unique	Unique
Use in	Primarily index access	Primarily key access	Only key access
Change key field	V	X	X



```
1   PROGRAM demo_internal_table.
2
3   TYPES: BEGIN OF mytext,
4                 number TYPE i,
5                 name(10) TYPE c,
6             END OF mytext.
7
8   TYPES mytab TYPE STANDARD TABLE OF mytext WITH DEFAULT KEY.
9   DATA text TYPE mytext.
10  DATA itab TYPE mytab.
11  text-number = 1. text-name = 'John'.
12  APPEND text TO itab.
13  text-number = 2. text-name = 'Paul'.
14  APPEND text TO itab.
15  text-number = 3. text-name = 'Ringo'.
16  APPEND text TO itab.
17
18  LOOP AT itab INTO text.
19     WRITE: / text-number,text-name.
20  ENDLOOP.
```

第 3、6 行 使用 BEGIN-END 宣告一個資料結構，結構體名稱為：mytext，該結構體有 number,name 共 2 個欄位

第 8 行 使用 TYPES 指令建立一個 Internal Table ；類型為 Standard ；參考的資料結構體為 mytext ；通過關鍵字：DEFAULT KEY 指定預設字段為 key 列，對於 Standard tables，預設的 key 列是由所有 CHAR-LIKE 字段組成的(C,N,D,T,X,STIRNG,XSTRING)

第 9 行 使用 DATA 宣告變數，名稱為：text、類型引用自行定義資料類型：mytext

第 10 行 使用 DATA 宣告變數，名稱為：itab、類型引用自行定義資料類型：mytab

第 11、16 行 將資料賦值給 text(結構體)後再透過 APPEND 指令將 text(結構體)的資料寫入 itab(Internal Table)

第 18、20 行 透過 LOOP 指令將 itab(Internal Table)內容印出至螢幕。

一開始看第一支範例程式時，看到 Line7

DATA itab TYPE TABLE OF zmm011_t01 WITH HEADER LINE.

可能會有一個誤解，會誤以為如果要顯示報表的資料，就要先在 se11 中定義 DataElement 才行。

但是透過上面的程式，對 Internel Table 就更理解並非如此，只需在程式中用 TYPES 定義一個 Structure，然後用 DATA 定義 TYPE OF TABLE 就可以在 Report 中使用 Internal Table，這就符合一般程式中可以動態建立 Client 端的暫存表(Internal Table) 用來暫存計算並顯示相關資料，並綁定元件顯示的標準做法。

http://www.sapjx.com/alv-fieldcat-and-layout.html

```
*&---------------------------------------------------------------------*
*& Report ZMC002
*&---------------------------------------------------------------------*
*& 改寫用自定義的 Types 取代 DataElement
*& 並練習寫 FORM (function)
*&---------------------------------------------------------------------*
REPORT ZMC002.

TYPE-POOLS: slis.

TYPES: BEGIN OF sumReport,
        "BILLDATE TYPE d,
        BLDAT TYPE d,
        BWART(10) TYPE c,
        WERKS(10) TYPE c,
        QTY TYPE i,
        RECNO TYPE i,
      END OF sumReport.

TYPES tpl_tab TYPE STANDARD TABLE OF sumReport WITH DEFAULT KEY.
DATA itab TYPE tpl_tab WITH HEADER LINE.
DATA wa_data LIKE LINE OF itab.

DATA: wa_fieldcat TYPE slis_fieldcat_alv ,    " 相当于工作区
      i_fieldcat  TYPE slis_t_fieldcat_alv ,  " 存放列名的列表
      i_layout    TYPE slis_layout_alv .      " 负责整个 ALV 的全局属性

i_layout-zebra              = 'X' .         " 使 ALV 界面呈现颜色交替
"i_layout-colwidth_optimize     = 'X' .         " 自动优化列宽
"i_layout-detail_initial_lines  = 'X' .         " 是否出现细节屏幕
"i_layout-detail_titlebar       = '详细内容' .  " 细节屏幕标题

PARAMETERS:
  p_grid RADIOBUTTON GROUP rg1 DEFAULT 'X',
  p_list RADIOBUTTON GROUP rg1,
  p_write RADIOBUTTON GROUP rg1.

```

```
SELECT-OPTIONS:
  d_BLDAT FOR itab-BLDAT.

START-OF-SELECTION.
  "設定 alv 參數
  PERFORM frm_set_fieldcat  USING:
  'ITAB'  'BLDAT'   '文件日期' '日期'  'X',
  'ITAB'  'BWART'   '異動類別' '異動類別' '',
  'ITAB'  'WERKS'   '倉庫代號' '倉庫' '',
  'ITAB'  'QTY'   '數量合計' '數量' '',
  'ITAB'  'RECNO'   '總筆數' '總筆數' ''.

  "取得資料
  SELECT BLDAT,BWART,WERKS,sum( MENGE ) AS QTY,count( * ) AS RECNO INTO TABLE @itab
    FROM  zmm011_t01
    "UP TO 1000 ROWS
    group by BLDAT,BWART,WERKS
    HAVING BLDAT in @d_BLDAT
   ORDER BY BLDAT,BWART,WERKS.

CASE 'X'.
  WHEN p_grid.
    CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
  exporting
   " I_STRUCTURE_NAME =     'TPL_TAB'   " Internal output table structure name
   is_layout         = i_layout          " 屬性內表
   it_fieldcat       = i_fieldcat[]      " 列明內表
  tables
    T_OUTTAB                  =     itab.          " Table with data to be displayed
  WHEN p_list.
    CALL FUNCTION 'REUSE_ALV_LIST_DISPLAY'
  exporting
    "I_STRUCTURE_NAME =     'tpl_tab'  " Internal output table structure name
    is_layout         = i_layout          " 屬性內表
    it_fieldcat       = i_fieldcat[]      " 列明內表
  tables
    T_OUTTAB                  =     itab .          " Table with data to be displayed
  WHEN p_write.
```

```
  LOOP AT itab INTO wa_data.
    WRITE:/ wa_data-BLDAT, wa_data-BWART,wa_data-WERKS,wa_data-QTY,wa_data-RECNO.
  ENDLOOP.
 WHEN OTHERS.
ENDCASE.

*&---------------------------------------------------------------------*
*&      Form  frm_set_fieldcat
*&---------------------------------------------------------------------*
*      text
*----------------------------------------------------------------------*
FORM frm_set_fieldcat
 USING  l_tabname l_fieldname l_titlename_l l_titlename_s l_fix_column.

 CLEAR  wa_fieldcat.
 wa_fieldcat-tabname     = l_tabname.
 wa_fieldcat-fieldname   = l_fieldname.        " 需要输出的内表的字段名
 wa_fieldcat-seltext_l   = l_titlename_l. " 字段的描述-长字段标签
 wa_fieldcat-seltext_m   = l_titlename_l. " 字段描述-中字段标签
 wa_fieldcat-seltext_s   = l_titlename_s. " 字段描述-短字段标签
 wa_fieldcat-fix_column  = l_fix_column.          " 是否是固定列
 APPEND wa_fieldcat TO i_fieldcat.

ENDFORM.                    " frm_set_fieldcat
```

再來一支範例，列出指定 Table 的檔案結構(如 ZMM*)

```
*&---------------------------------------------------------------------*
*& Report ZMC003
*&---------------------------------------------------------------------*
*& Table Schema From DD03L and DD02T
*&---------------------------------------------------------------------*
REPORT ZMC003.

TYPE-POOLS: slis.

TYPES: BEGIN OF sumReport,
       TABNAME(20) TYPE c,
       TABDESC(100) TYPE c,
```

```
13          FIELDNAME(20) TYPE c,
14          ROLLNAME(20) TYPE c,
15          KEYFLAG(1) TYPE c,
16          INTTYPE(2) TYPE c,
17          INTLEN(10) TYPE c,
18        END OF sumReport.
19
20  TYPES tpl_tab TYPE STANDARD TABLE OF sumReport WITH DEFAULT KEY.
21  DATA itab TYPE tpl_tab WITH HEADER LINE.
22  DATA itab_new TYPE tpl_tab WITH HEADER LINE.
23  DATA wa_data LIKE LINE OF itab.
24  DATA git_dfies TYPE STANDARD TABLE OF dfies WITH HEADER LINE.
25  DATA lc_desc(40) TYPE c.
26
27  DATA: wa_fieldcat TYPE slis_fieldcat_alv ,    " 相当于工作区
28        i_fieldcat  TYPE slis_t_fieldcat_alv ,  " 存放列名的列表
29        i_layout    TYPE slis_layout_alv .      " 负责整个 ALV 的全局属性
30
31  i_layout-zebra                = 'X' .         " 使 ALV 界面呈现颜色交替
32  "i_layout-colwidth_optimize     = 'X' .         " 自动优化列宽
33  "i_layout-detail_initial_lines   = 'X' .         " 是否出现细节屏幕
34  "i_layout-detail_titlebar        = '详细内容' .  " 细节屏幕标题
35
36  SELECTION-SCREEN  BEGIN  OF  BLOCK  blk1 WITH  FRAME  TITLE  text-001 .
37  PARAMETERS:
38    p_grid RADIOBUTTON GROUP rg1 DEFAULT 'X',
39    p_list RADIOBUTTON GROUP rg1,
40    p_write RADIOBUTTON GROUP rg1.
41  SELECTION-SCREEN  END    OF  BLOCK  blk1.
42
43  SELECT-OPTIONS:
44    c_tbl FOR itab-TABNAME.
45
46  START-OF-SELECTION.
47    "設定 alv 參數
48    PERFORM frm_set_fieldcat  USING:
49    'ITAB'  'TABNAME'   'TableName' 'Table'  'X',
50    'ITAB'  'TABDESC'   'TableDesc' 'Desc' '',
```

```
  'ITAB'  'FIELDNAME'   'FieldName' 'Field' '',
  'ITAB'  'ROLLNAME'   'RollName' 'Roll' '',
  'ITAB'  'KEYFLAG'   'PK' 'PK' '',
  'ITAB'  'INTTYPE'   'Type' 'Type' '',
  'ITAB'  'INTLEN'   'Len' 'Len' ''.

  "取得資料  CAST (10.5 AS INTEGER)
  SELECT DD03L~TABNAME,DD02T~DDTEXT,DD03L~FIELDNAME,DD03L~ROLLNAME,
      DD03L~KEYFLAG,DD03L~INTTYPE,DD03L~INTLEN INTO TABLE @itab
    FROM  DD03L LEFT JOIN DD02T ON DD03L~TABNAME = DD02T~TABNAME
      AND DD02T~DDLANGUAGE = 'M'   "M=ZF SAP
    WHERE DD03L~TABNAME in @c_tbl
    ORDER BY DD03L~TABNAME,DD03L~POSITION.

  " 修改 field desc
  LOOP AT itab INTO wa_data.
    PERFORM frm_get_fielddesc USING wa_data-TABNAME wa_data-FIELDNAME.

    itab_new-TABNAME = wa_data-TABNAME.
    itab_new-TABDESC = wa_data-TABDESC.
    itab_new-FIELDNAME = wa_data-FIELDNAME.
    itab_new-ROLLNAME = lc_desc.
    itab_new-KEYFLAG = wa_data-KEYFLAG.
    itab_new-INTTYPE = wa_data-INTTYPE.
    itab_new-INTLEN = wa_data-INTLEN.

    APPEND itab_new.
  ENDLOOP.

CASE 'X'.
  WHEN p_grid.
    CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
  exporting
   " I_STRUCTURE_NAME =     'TPL_TAB'   " Internal output table structure name
   is_layout          = i_layout          " 属性内表
   it_fieldcat        = i_fieldcat[]      " 列明内表
  tables
    T_OUTTAB                  =      itab_new.          " Table with data to be displayed
```

```
89    WHEN p_list.
90      CALL FUNCTION 'REUSE_ALV_LIST_DISPLAY'
91    exporting
92      "I_STRUCTURE_NAME =     'tpl_tab'  " Internal output table structure name
93      is_layout          = i_layout          " 属性内表
94      it_fieldcat        = i_fieldcat[]      " 列明内表
95    tables
96      T_OUTTAB                   =      itab .          " Table with data to be displayed
97    WHEN p_write.
98      LOOP AT itab INTO wa_data.
99        WRITE:/ wa_data-TABNAME, wa_data-FIELDNAME,wa_data-ROLLNAME,
100           wa_data-INTTYPE,wa_data-INTLEN.
101     ENDLOOP.
102   WHEN OTHERS.
103 ENDCASE.
104
105 *&---------------------------------------------------------------------*
106 *&      Form  frm_set_fieldcat
107 *&---------------------------------------------------------------------*
108 FORM frm_set_fieldcat
109   USING  l_tabname l_fieldname l_titlename_l l_titlename_s l_fix_column.
110
111   CLEAR  wa_fieldcat.
112   wa_fieldcat-tabname     = l_tabname.
113   wa_fieldcat-fieldname   = l_fieldname.        " 需要输出的内表的字段名
114   wa_fieldcat-seltext_l   = l_titlename_l. " 字段的描述-长字段标签
115   wa_fieldcat-seltext_m   = l_titlename_l. " 字段描述-中字段标签
116   wa_fieldcat-seltext_s   = l_titlename_s. " 字段描述-短字段标签
117   wa_fieldcat-fix_column  = l_fix_column.          " 是否是固定列
118   APPEND wa_fieldcat TO i_fieldcat.
119
120 ENDFORM.                    " frm_set_fieldcat
121
122 *&---------------------------------------------------------------------*
123 *&      Form  frm_set_fieldcat
124 *&---------------------------------------------------------------------*
125 *     這個 issue 撞牆快一天, 只因為引數的型別錯誤, 重點在 line 120-121
126 *---------------------------------------------------------------------*
```

```
FORM frm_get_fielddesc USING  lc_table lc_field.
  DATA ld_tabname   TYPE DDOBJNAME.
  DATA ld_field TYPE DFIES-FIELDNAME.

  ld_tabname = lc_table.
  ld_field = lc_field.

CALL FUNCTION 'DDIF_FIELDINFO_GET'
  EXPORTING
    tabname        = ld_tabname
    fieldname      = ld_field
    langu          = sy-langu
  TABLES
    dfies_tab      = git_dfies[]
  EXCEPTIONS
    not_found      = 1
    internal_error = 2
    OTHERS         = 3.

  IF sy-subrc = 0.
    "Implement suitable error handling here
    LOOP AT git_dfies.
      lc_desc = git_dfies-scrtext_l.
    ENDLOOP.
  ENDIF.
ENDFORM.
```

呼叫 DDIF_FIELDINFO_GET 這支程式時，我足足被卡住一天，一直鬼打牆，報如下的錯誤

ABAP 執行時期發生錯誤

除錯

Category	ABAP programming error
Runtime Errors	CALL_FUNCTION_CONFLICT_TYPE
Except.	CX_SY_DYN_CALL_ILLEGAL_TYPE
ABAP Program	ZMC003
Application Component	Not assigned
Date and Time	2019.11.29 09:15:29

Short Text

Type conflict during a function module call

當我終於搞清楚原因，回頭再看這個錯誤訊息，覺得他講得很清楚

➔ Type conflict during a function module call

就是型別錯誤，但是，一開始我怎麼都沒找到我的型別哪裡錯了?

我一開始的寫法如下

呼叫端	PERFORM frm_get_fielddesc USING wa_data-TABNAME wa_data-FIELDNAME.
程式	FORM frm_get_fielddesc USING lc_table lc_field. CALL FUNCTION 'DDIF_FIELDINFO_GET' EXPORTING tabname = lc_table fieldname = lc_field langu = sy-langu TABLES dfies_tab = git_dfies[]

我怎麼看都沒問題啊，但是就是錯，一直說是型別錯誤，後來 Google 半天，

https://www.erpgreat.com/abap/how-to-get-the-field-descriptions-of-a-table.htm 改為

```
FORM frm_get_fielddesc USING  lc_table lc_field.
  DATA ld_tabname   TYPE DDOBJNAME.
  DATA ld_field TYPE DFIES-FIELDNAME.

  ld_tabname = lc_table.
  ld_field = lc_field.

  CALL FUNCTION 'DDIF_FIELDINFO_GET'
  EXPORTING
    tabname        = ld_tabname
    fieldname      = ld_field
```

真的是型別問題，以後寫程式要看清楚註解

```
CALL FUNCTION 'DDIF_FIELDINFO_GET'
  exporting
    TABNAME        =                  " Name of the Table (of the Type) for which Information is Req
*   FIELDNAME      = ' '              " Use Parameter LFIELDNAME Instead
*   LANGU          = SY-LANGU         " Language of the Texts
*   LFIELDNAME     = ' '              " If Filled, only Field with this Long Name
*  tables
*   DFIES_TAB      =                  " Field List if Necessary
*   FIXED_VALUES   =                  " Description of Domain Fixed Values
```

再新增一支 Excel 匯入的範例程式

```
*&---------------------------------------------------------------------*
*& Report ZFD006_U02
*&---------------------------------------------------------------------*
*& Description: 特約加盟管控記錄匯入作業
*& Func.Spec  : 三商家購_外掛程式規格書_SP_FI-056~062 加盟分潤系統_擴充報表.doc
*& Author     : Michael Chen
*&=====================================================================
=====*
*& Modification Log - History
*&=====================================================================
=====*
*& Date        Modifier       Request_No     Description
*&---------------------------------------------------------------------*
*& 2012001/06   Michael Chen      DEVK900132     程式建立
*&---------------------------------------------------------------------*
REPORT ZFD006_U02.

TYPE-POOLS: slis.
TYPE-POOLS truxs.

TABLES : ZFD005_T04.

DATA:
     wa_fieldcat TYPE slis_fieldcat_alv ,    " 相当于工作区
     i_fieldcat  TYPE slis_t_fieldcat_alv ,  " 存放列名的列表
     i_layout    TYPE slis_layout_alv .      " 负责整个 ALV 的全局属性

i_layout-zebra                = 'X' .         " 使 ALV 界面呈现颜色交替

TYPES:
  BEGIN OF t_tab,
    BUKRS(4)  type c,
    WERKS(8)  type c,
    ZBUPER(6) type  c,
    ZCTRL_SCORE type I,
    ZCTRL_MEMO(60) type c,
  END OF t_tab.
```

```
DATA :
  t_upload TYPE STANDARD TABLE OF t_tab,
  wa_upload TYPE t_tab,
  it_type TYPE truxs_t_text_data.

DATA:
      recno type I,
      ok_no type I,
      err_no type I.

DATA: ok_c(10) TYPE c, err_c(10) TYPE c.
DATA: lv_result(200) TYPE c.

* Selection screen
SELECTION-SCREEN  BEGIN  OF  BLOCK  blk1 WITH  FRAME  TITLE  text-001 .
PARAMETERS:
  p_view RADIOBUTTON GROUP rg1 DEFAULT 'X',
  p_save RADIOBUTTON GROUP rg1.
SELECTION-SCREEN  END    OF  BLOCK  blk1.

SELECTION-SCREEN  BEGIN  OF  BLOCK  blk2 WITH  FRAME  TITLE  text-002 .
  PARAMETERS p_file TYPE rlgrap-filename DEFAULT 'D:\Data\ZFD005_T04.xlsx'.
SELECTION-SCREEN  END    OF  BLOCK  blk2.

AT SELECTION-SCREEN ON VALUE-REQUEST FOR p_file.
  CALL FUNCTION 'F4_FILENAME'
  EXPORTING
    field_name = 'P_FILE'
  IMPORTING
    file_name = p_file.

START-OF-SELECTION.
* Uploading the data in the file into internal table
  CALL FUNCTION 'TEXT_CONVERT_XLS_TO_SAP'
  EXPORTING
* I_FIELD_SEPERATOR =
* I_LINE_HEADER = 'X'
```

```
  i_tab_raw_data = it_type
  i_filename = p_file
  TABLES
  i_tab_converted_data = t_upload[]
  EXCEPTIONS
  conversion_failed = 1
  OTHERS = 2.

  IF sy-subrc NE 0.
    MESSAGE ID sy-msgid
    TYPE sy-msgty
    NUMBER sy-msgno
    WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
  ENDIF.

  CASE 'X'.
  WHEN p_view.
    PERFORM frm_view.
  WHEN p_save.
    PERFORM frm_save.
  WHEN OTHERS.
  ENDCASE.

END-OF-SELECTION.

*&---------------------------------------------------------------------*
*&      Form  frm_view
*&---------------------------------------------------------------------*
FORM frm_view.
  "設定 alv 參數
  PERFORM frm_set_fieldcat  USING:
  't_upload'  'BUKRS'   '公司代碼' '公司代碼'  'X',
  't_upload'  'WERKS'   '工廠' '工廠' '',
  't_upload'  'ZBUPER'   '年月(YYYYMM)' '年月' '',
  't_upload'  'ZCTRL_SCORE'   '點檢分數' '點檢分數' '',
  't_upload'  'ZCTRL_MEMO'   '缺失紀錄' '缺失紀錄' ''.

  "顯示 excel 內容
```

```
 CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
 exporting
  " I_STRUCTURE_NAME =     'TPL_TAB'   " Internal output table structure name
  is_layout          = i_layout          " 屬性內表
  it_fieldcat        = i_fieldcat[]      " 列明內表
 tables
   T_OUTTAB                  =      t_upload.          " Table with data to be displayed
ENDFORM.

*&---------------------------------------------------------------------*
*&      Form  frm_view
*&---------------------------------------------------------------------*
*& 目前版本沒有防錯機制, 應該要加入檢查及是否已存在的判斷
*&---------------------------------------------------------------------*
FORM frm_save.
* Uploading the data into the database table
 ok_no = 0.
 err_no = 0.
 LOOP AT T_UPLOAD INTO WA_UPLOAD.
   "判斷資料是否已存在, 若已存在 skip
   select count( * ) into @recno from ZFD005_T04 where BUKRS = @WA_UPLOAD-BUKRS and WERKS
= @WA_UPLOAD-WERKS  and ZBUPER = @WA_UPLOAD-ZBUPER.

   if recno = 0.
     ZFD005_T04-BUKRS = WA_UPLOAD-BUKRS.
     ZFD005_T04-WERKS = WA_UPLOAD-WERKS.
     ZFD005_T04-ZBUPER = WA_UPLOAD-ZBUPER.
     ZFD005_T04-ZCTRL_SCORE = WA_UPLOAD-ZCTRL_SCORE.
     ZFD005_T04-ZCTRL_MEMO = WA_UPLOAD-ZCTRL_MEMO.

     MODIFY ZFD005_T04.
     ok_no = ok_no + 1.
   else.
     err_no = err_no + 1.
   endif.
 ENDLOOP.

 ok_c = ok_no.
```

```
  err_c = err_no.
  CONCATENATE '資料寫入成功筆數:' ok_c '，失敗筆數:' err_c INTO lv_result.
  CALL FUNCTION 'POPUP_TO_DISPLAY_TEXT'
    exporting
      TEXTLINE1    =  lv_result.
ENDFORM.

*&---------------------------------------------------------------------*
*&      Form  frm_set_fieldcat
*&---------------------------------------------------------------------*
*      text
*----------------------------------------------------------------------*
FORM frm_set_fieldcat
  USING  l_tabname l_fieldname l_titlename_l l_titlename_s l_fix_column.

  CLEAR  wa_fieldcat.
  wa_fieldcat-tabname     = l_tabname.
  wa_fieldcat-fieldname   = l_fieldname.        " 需要输出的内表的字段名
  wa_fieldcat-seltext_l   = l_titlename_l. " 字段的描述-长字段标签
  wa_fieldcat-seltext_m   = l_titlename_l. " 字段描述-中字段标签
  wa_fieldcat-seltext_s   = l_titlename_s. " 字段描述-短字段标签
  wa_fieldcat-fix_column  = l_fix_column.          " 是否是固定列
  APPEND wa_fieldcat TO i_fieldcat.

ENDFORM.                    " frm_set_fieldcat
```

Session 2：ABAP 資料表_事物碼(T-Code)資訊

https://www.fenginfo.com/472.html

Session 2.1：說明

SAP 的事物碼（Transaction Code，簡寫 T-CODE）相關資訊存放在三張表（Table）中：

- 資料表 TSTC 存放 T-CODE 的關聯程式、螢幕資訊
- 資料表 TSTCP 存放相關參數，如變式名稱及賦值等
- 資料表 TSTCT 存放語言文本，為多種語言翻譯服務

以上各表均可用 SE11、SE16N 查詢結構及資料。

T-CODE 表的資訊是 cross-client 級別的，一套 ERP 系統下各 Client 之間 T-CODE 的數量及相關資訊均相同。

Session 2.2：表說明

1．資料表 TSTC

中文名稱：SAP 事務代碼

類型：轉帳表，全部欄位見下

欄位名	主鍵	類型	說明
TCODE	Y	CHAR(20)	事務代碼
PGMNA		CHAR(40)	程式名稱
DYPNO		NUMC(4)	螢幕號碼
MENUE		CHAR(5)	SYST 的 CHAR05 資料元素
CINFO		RAW(1)	SYST 之 HEX01 資料元素
ARBGB		CHAR(20)	消息識別

2．資料表 TSTCP，中文名稱：事務參數

欄位名	主鍵	類型	說明
TCODE	Y	CHAR(20)	事務代碼
PARAM		CHAR(254)	事務的參數

3．資料表 TSTCT

中文名稱：事務代碼文本

類型：轉帳表，全部欄位見下

欄位名	主鍵	類型	說明
SPRSL	Y	LANG(1)	語言代碼
TCODE	Y	CHAR(20)	事務代碼
TTEXT		CHAR(36)	事務說明

Session 3：查看事物碼相關的資料物件

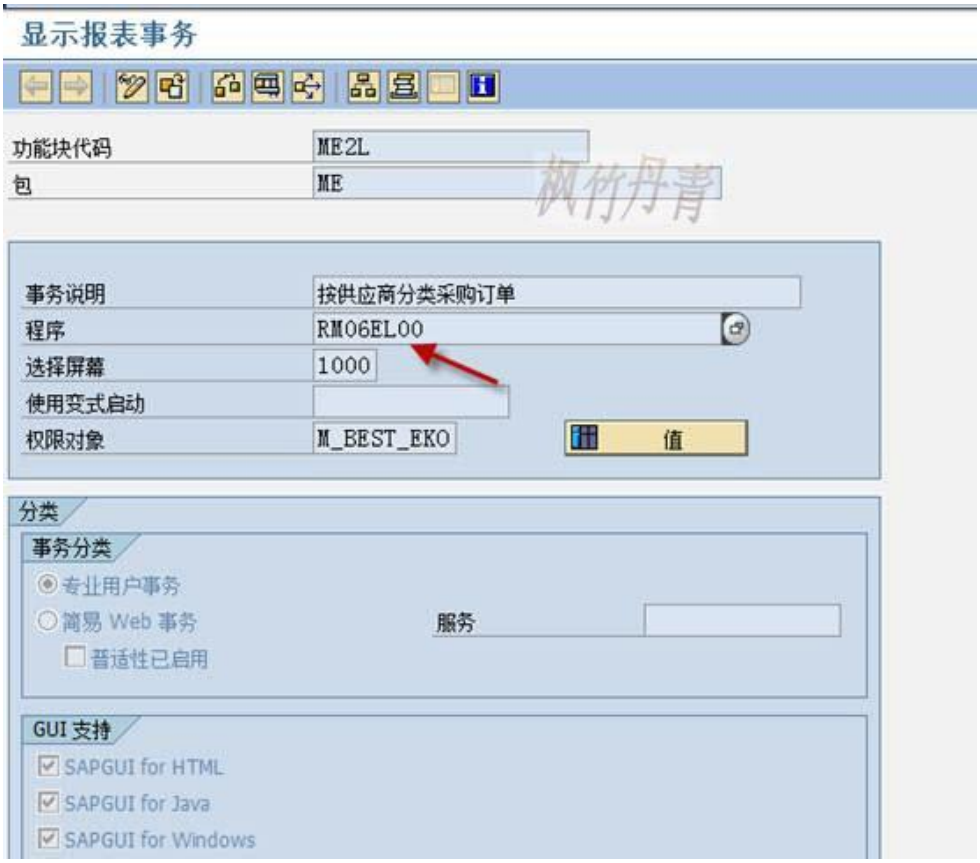
https://www.fenginfo.com/2747.html

隨著對業務的深入，作為諮詢顧問很多時候需要知道業務涉及的資料表，如果想知道某個事物碼涉及的資料表名稱，可先用 SE93 進行查看，大致分為兩種情況：

- 事物碼是綁定的程式，查看程式的涉及物件就可找到涉及的資料表。
- 事物碼是綁定的視圖或視圖族，查看表結構可以看到涉及的資料表。

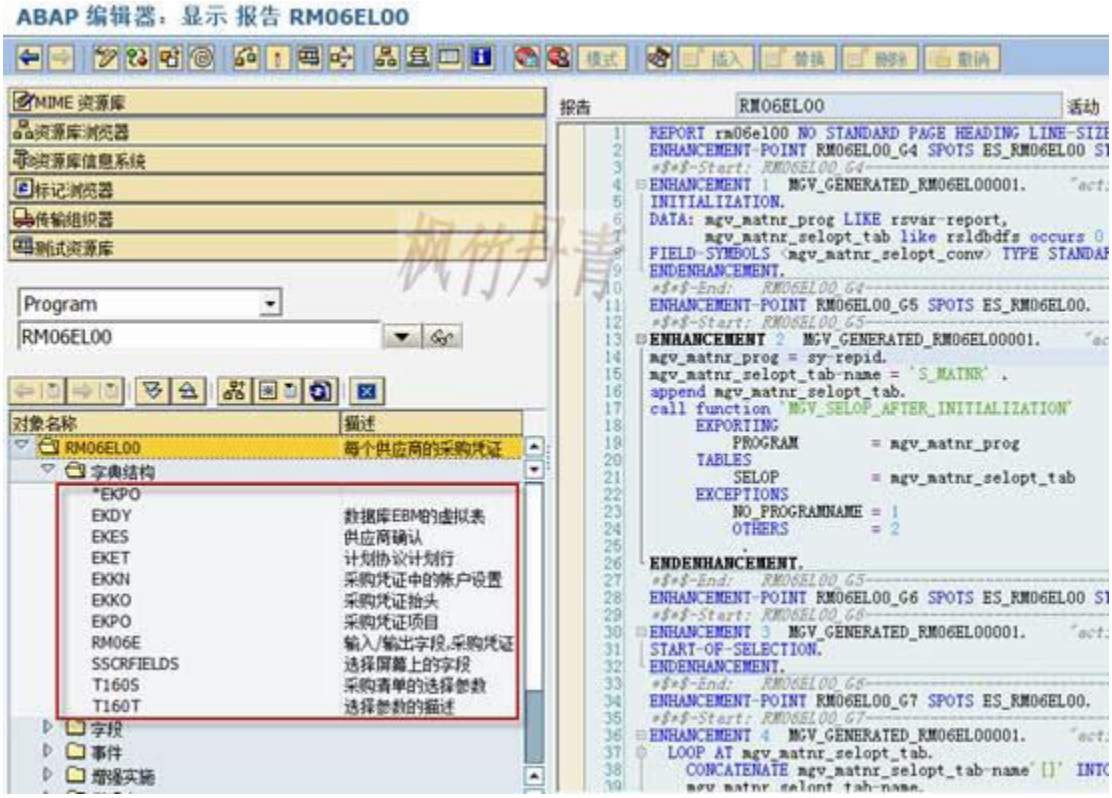
以查看 ME2L 為例，前臺執行 SE93，進入後如圖 1 所示，輸入事物碼 ME2L，並點擊“顯示”按鈕。

進入後顯示事物碼對應的程式資訊，如下圖所示，注意這裡的程式條目值“RM06EL00”和選擇螢幕條目值“1000”，兩個條目都可以點擊進入 ABAP 編輯介面，前者程式也可用 SE38 進入後查看，後者螢幕也可用 SE51 進入後查看。



以查看程式為例，如圖 3 所示顯示程式的代碼，在工具列點擊“顯示物件清單”按鈕（ ）。

介面的右邊出現導向條形功能表，在“資源庫流覽器”條形功能表下，有此程式的關聯物件，其中在“字典結構”下可以看到程式涉及到的資料表。



對於配置類的基於 SM30 維護視圖、SM34 維護視圖所生成的事物碼，如 OX18(為工廠分配公司代碼)，用 SE93 進入後介面如圖 5 所示，其事務為 SM30，底部缺省值條目“VIEWNAME”的值為“V_T001K_Assign”,這就是維護視圖的名稱。

事务代码	OX18
包	MBOC
事务文本	工厂->公司码

枫竹丹青

此项的缺省值

事务 SM30

☑ 跳过初始屏幕

已弃用：使用事务的缺省值

屏幕 0

从模块池

分类

☑ 继承 GUI 属性

事务分类

◉ 专业用户事务

○ 简易 Web 事务　　服务

☐ 普适性已启用

GUI 支持

☑ SAPGUI for HTML

☑ SAPGUI for Java

☑ SAPGUI for Windows

缺省值

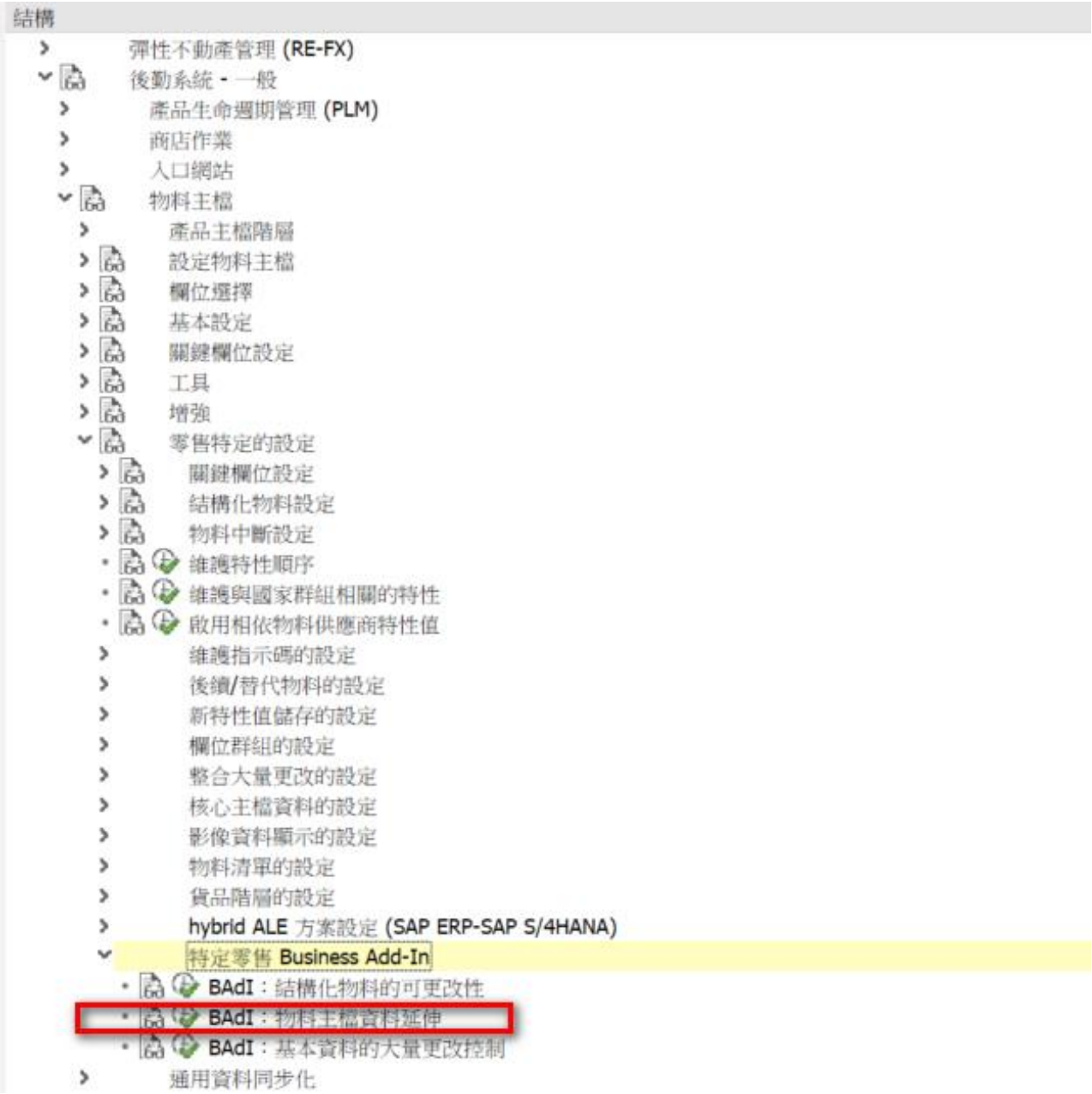
屏幕字段名称	值
VIEWNAME	V_T001K_Assign
UPDATE	X

前台執行 SE11 查看視圖 V_T001K_Assign 的結構，在“表/連接條目”標籤頁下，左側有多個資料表，這就是配置涉及的表名，其中第一個“T001K”是資料維護的主表，其它表則是輔助表，很多輔助表只是用來顯示附加的描述資訊。SM30 維護視圖也涉及程式，相關資訊參見《自訂 SM30 維護視圖》、《為維護視圖創建事物碼(Transaction code)》。

圖 6 SE11 查看視圖資訊_表/連接條件標籤頁

關於表單的 Enhanse 介面的設定

後勤系統-一般-物料主檔-零售特定的設定-特定零售 Business Add-In 下有 3 個 Bapi



所有物料主檔的相關的擴充程式，都是在此建立修改。此處為進階內容，稍有了解即可。

Business Add-In Builder: Display Implementation ZARTICLE_REFSITE_LOG

Definition Documenta 紀錄文件

Implementation Name	ZARTICLE_REFSITE_LOG 啟用中
Implementation Short Text	BADi for Article Master Data
Definition Name	BADI_ARTICLE_REF_RT
Runtime Behavior	Implementation will be called

屬性 介面

Interface name	IF_EX_BADI_ARTICLE_REF_RT
Name of Implementing Class	ZCL_IM_ARTICLE_REFSITE_LOG

方法	Implementati...	說明
ADD_INACTIVE_FIELDS	ABAP ABAP_	Add Inactive Fields In Template Handling
REFERENZ_BEFORE	ABAP ABAP_	Default Data For Template Handling: Time 'Be
REFERENZ	ABAP ABAP_	Default Data For Template Handling: Time 'No
REFERENZ_AFTER	ABAP ABAP_	Default Data For Template Handling: Time 'Aft
TRANSPORT_TO_UPDATE_TASK	ABAP ABAP_	Transfering Customer Data To Posting Module
IMPORT_AT_UPDATE_TASK	ABAP ABAP_	Import Customer Data In Update Module

Default implementation class	
Example implementation class	

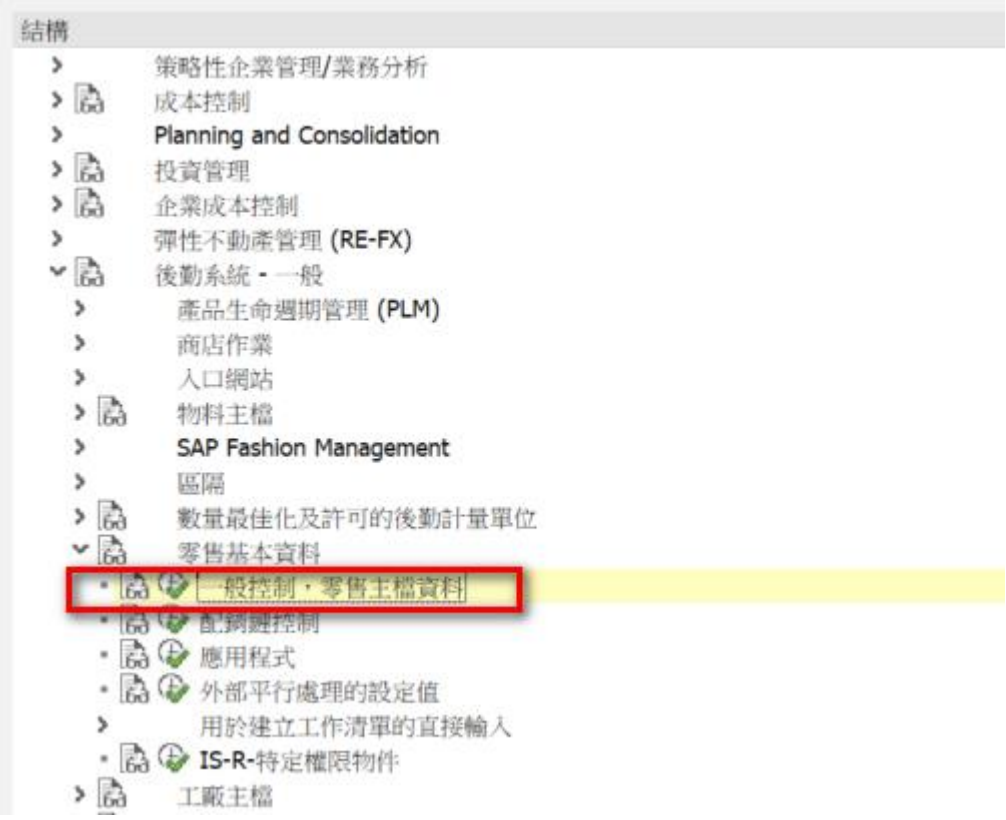
類別建立器： 類別 ZCL_IM_ARTICLE_REFSITE_LOG 顯示

式樣 好的印表機 簽名 Public Section Prot

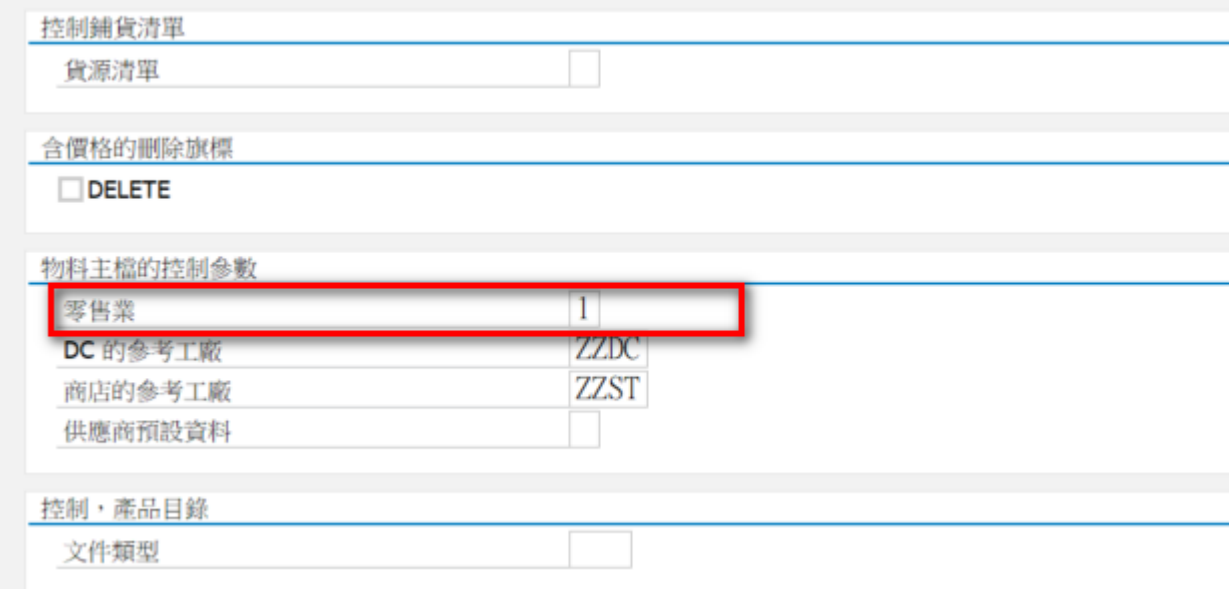
方法 IF_EX_BADI_ARTICLE_REF_RT~REFERENZ_BEFORE 啟用

```
METHOD IF_EX_BADI_ARTICLE_REF_RT~REFERENZ_BEFORE.

  IF SY-TCODE = 'MM41' OR SY-TCODE = 'MM42'.
   IF BADI_RMMG1-MTART NE 'Z005' AND BADI_RMMG1-MTART NE 'Z006'.
     BADI_MBEW-PEINH = '100'.
     BADI_MBEW-HKMAT = 'X'.
     BADI_MBEW-EKALR = 'X'.
     BADI_MARC-LOSGR = '100'.
     BADI_MARC-AWSLS = '000001'.
     BADI_MARC-KAUTB = 'X'.
     IF BADI_MARC-LGFSB IS INITIAL.
       BADI_MARC-LGFSB = '1001'.
     ENDIF.
     BADI_MARC-MTVFP = 'KP'.
     BADI_MARA-SPART = 'R1'.
     IF BADI_MARC-WERKS = 'ZZST'.
       BADI_MARC-DISMM = 'ND'.
       BADI_MARC-XMCNG = 'X'.
     ELSEIF BADI_MARC-WERKS = 'ZZDC'.
       IF BADI_MARC-DISMM IS INITIAL.
         BADI_MARC-DISMM = 'PD'.
         BADI_MARC-DISLS = 'TB'.
         BADI_MARC-RGEKZ = '1'.
         BADI_MARC-DISPO = '200'.
```

接下來繼續設定



啟用零售業的相關元件



DC 的參考工廠和商店的參考工廠，等於是範本的建立，當要建立物流(或央廚)、門店時，會依據此處設定的代號執行複製 Site 的範本功能。不建議用一個既有以實際存在的 Site 為範本(雖然也可以)。

➢ 關於 Change Request(CR) 及 SE10 及 STMS 的說明

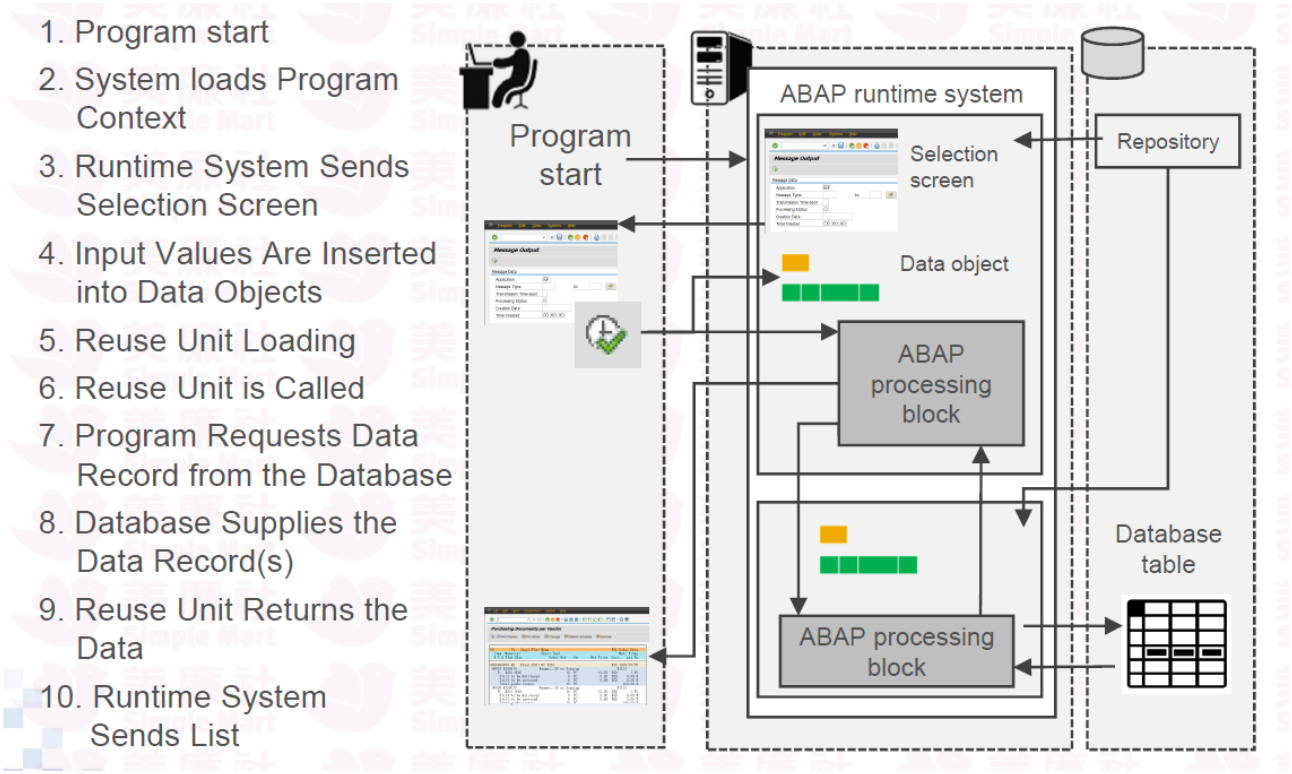
開發程式時，不管使用哪一個 Development Tools，都一定有一個議題要特別關注，那就是版本管理。尤其是多人同時開發，且又使用共用 Library 底層的情況。SAP ABAP 是一種類似直譯式的程式語言，他的 Source Code 推測應該是存放在 DataBase 中，他是透過 Change Request(以下簡稱 CR) 來做版本管理，並透過 DEV→QAS→PROD 的環境建置及一組 T-Code，就能完整的執行包括新版本的測試及發佈等作業。以下簡單的說明主要的操作及概念。

完整的說明，可以參考 Michael 顧問的四天上課課程(含講義)，內容說明非常清楚。

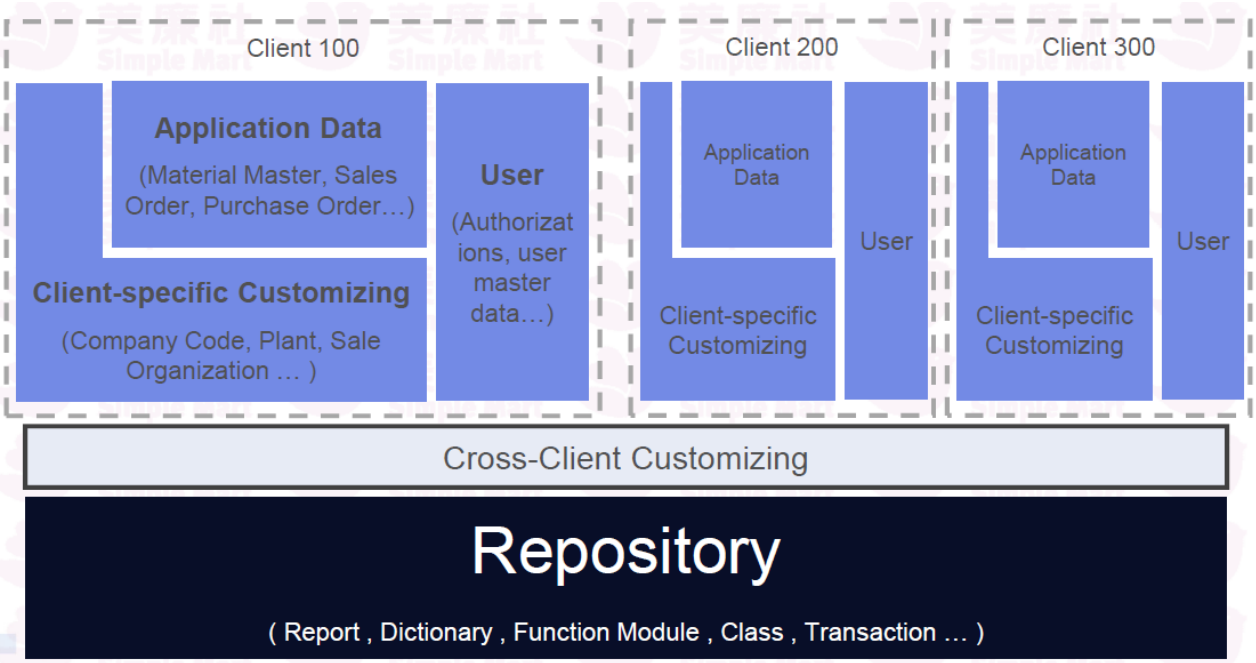
\\10.0.8.14\sap 導入專案\專案公用資料夾\Phase 1 專案準備\1.X 影音檔\ABAP\

\\10.0.8.14\sap 導入專案\專案公用資料夾\Phase 1 專案準備\1.4 SAP ERP 功能教育訓練\ABAP

下圖摘自講義，是 ABAP 執行的完整流程

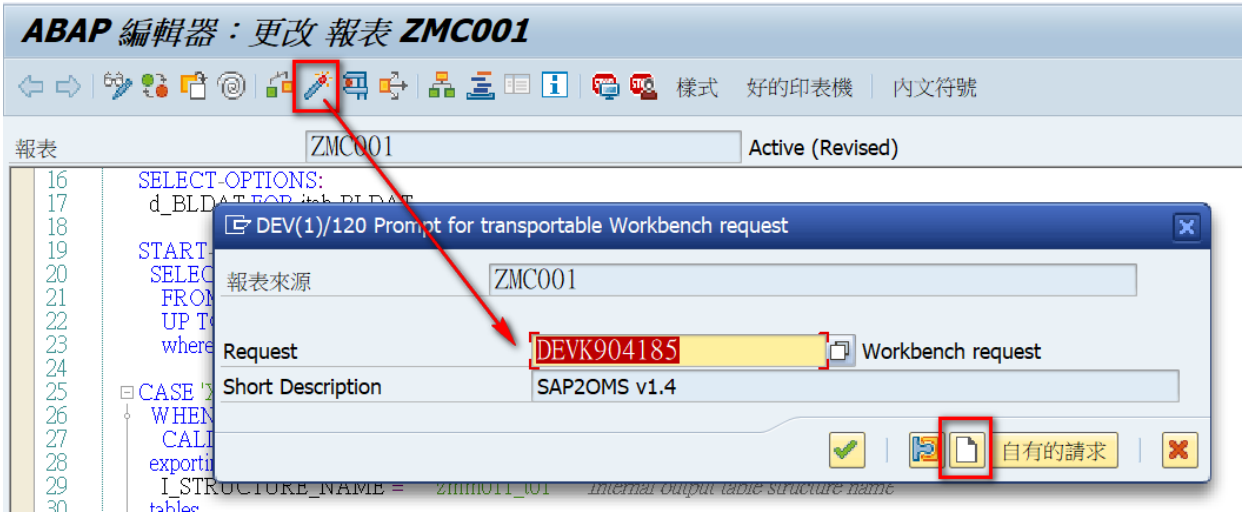


- Data Structure Of SAP ERP System

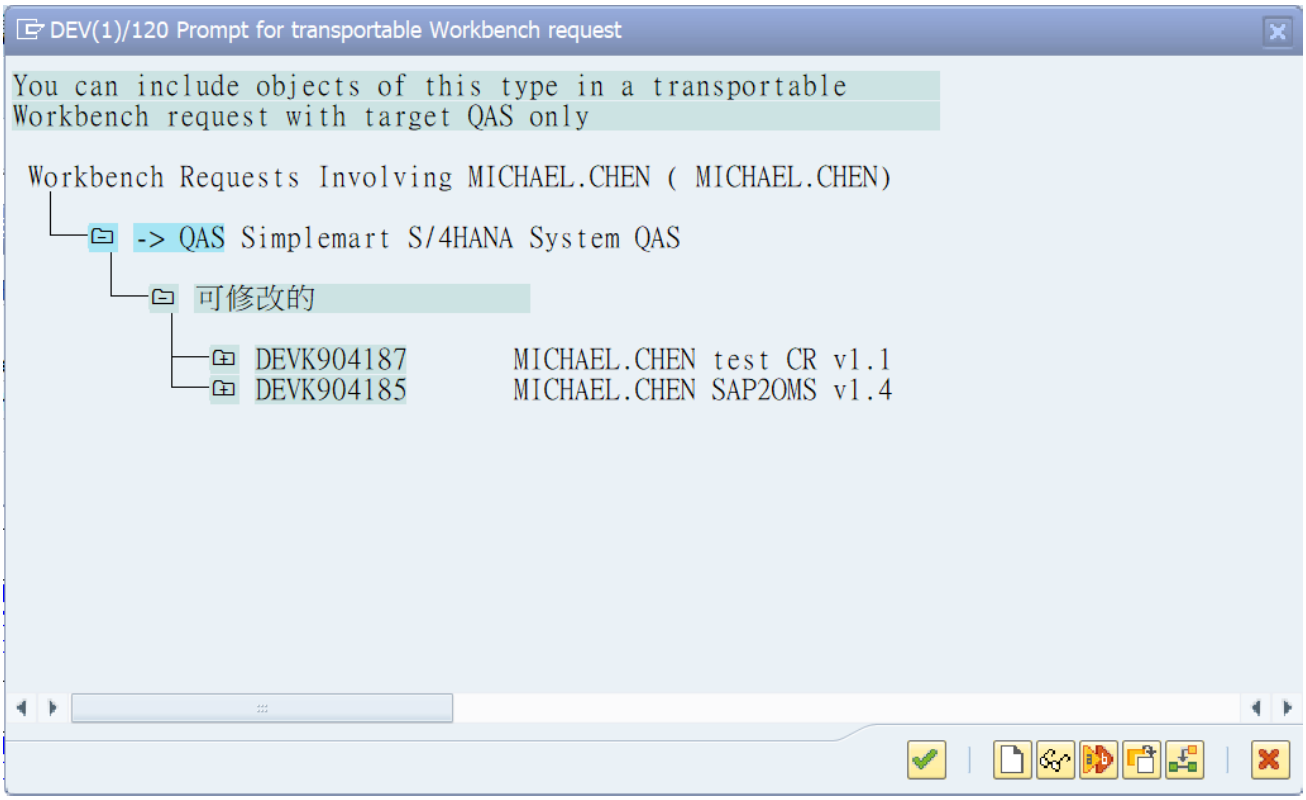


- 相關常用的 T-Code(開發相關)

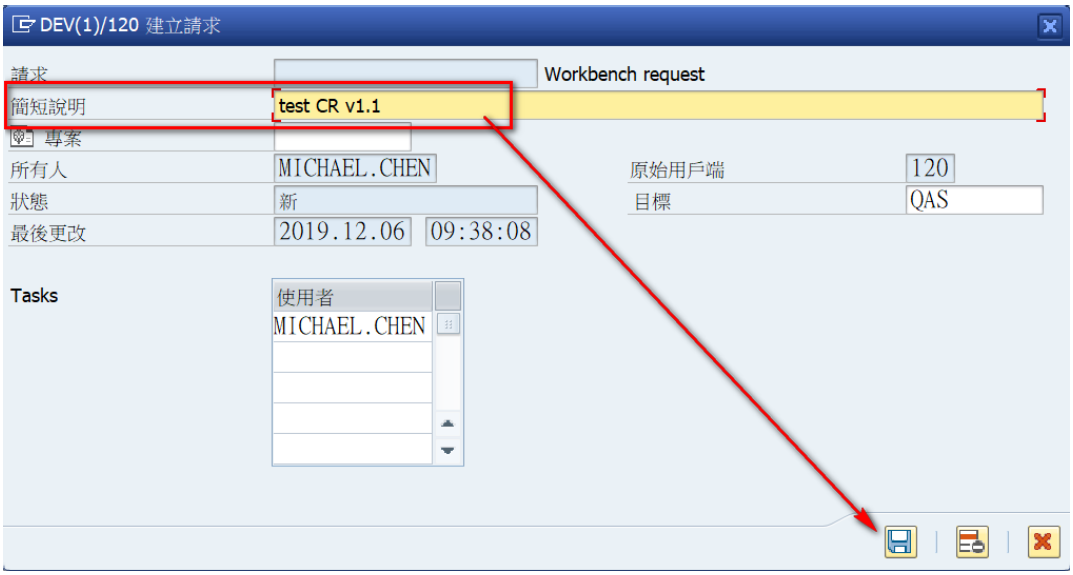
- Change Request
  - ✓ 當新增程式，或是修改程式時，系統會談出下列視窗，請選擇一個 CR Code，如果是新的版號，請按右下角的新增。

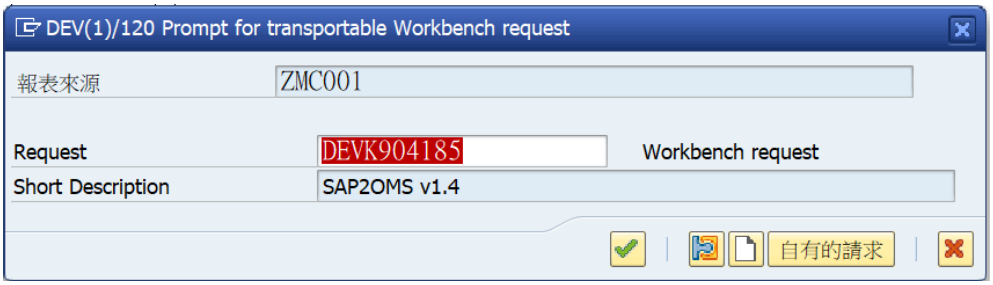


  - ✓ 也可查詢目前帳號還有哪些進行中的 CR



✓ 輸入說明後按存檔，會自動產生一個新的 CR Code(版本代號)

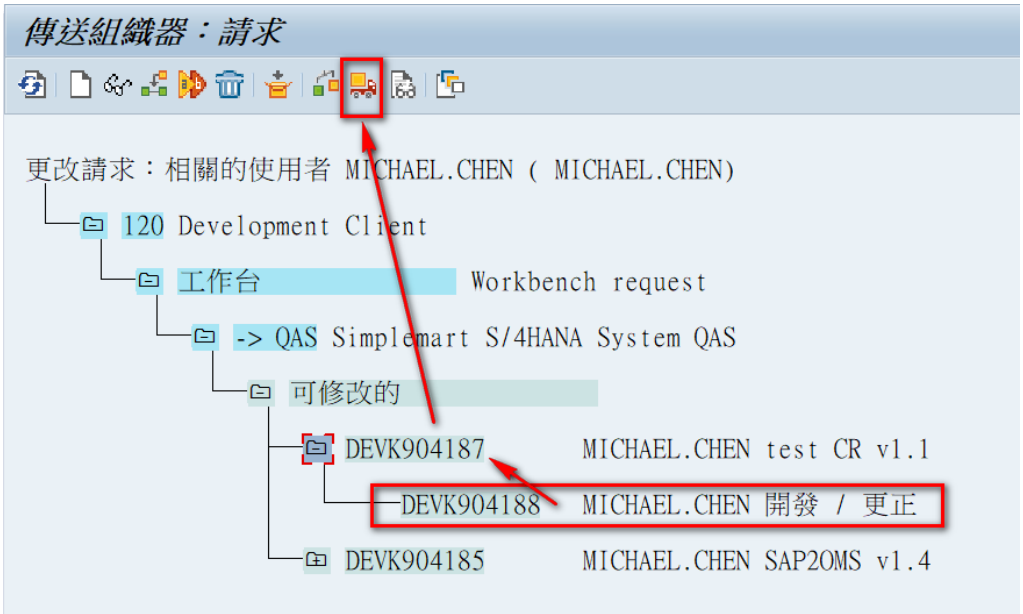




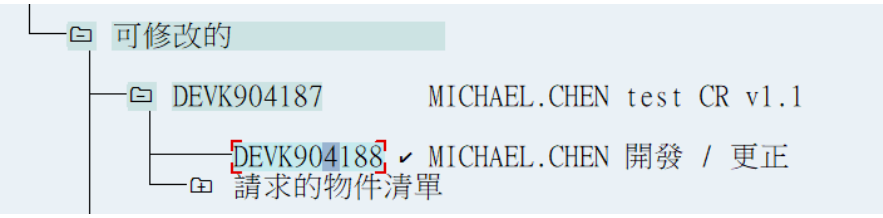
✓ 上面這個作業類似版控軟體的 Check Out 作業，只要在本程式沒有執行 SE10 的發佈 [貨車 Transfer] 功能前(類似 Check In)，可以持續修改程式。

- **SE10**

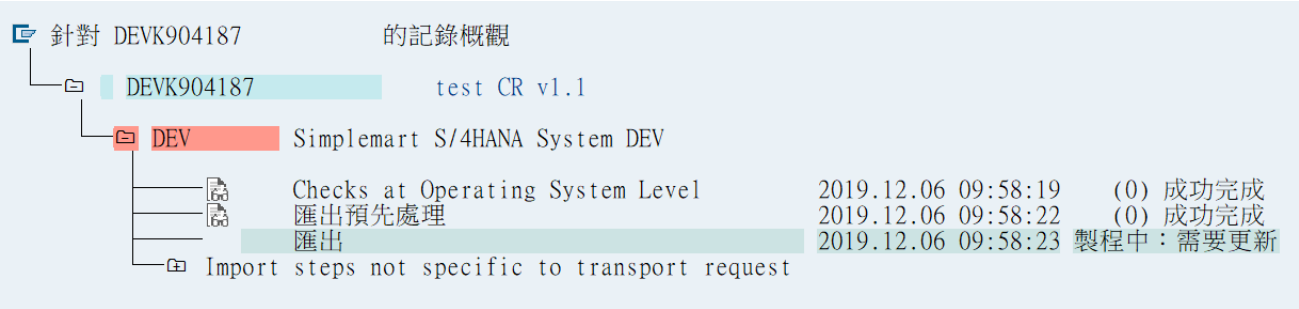
✓ **執行 T-Code SE10，準備將程式由 DEV 發佈到 QAS，此時是要先登入 DEV (執行匯出) 注意原則，一定必須要由下往上執行 Transfer(核發)，先點選 DEVK904188 按 [核發]，再點選 DEVK904187 再按 [核發]。**



✓ 當執行 4188 的核發後，會出現下列訊息

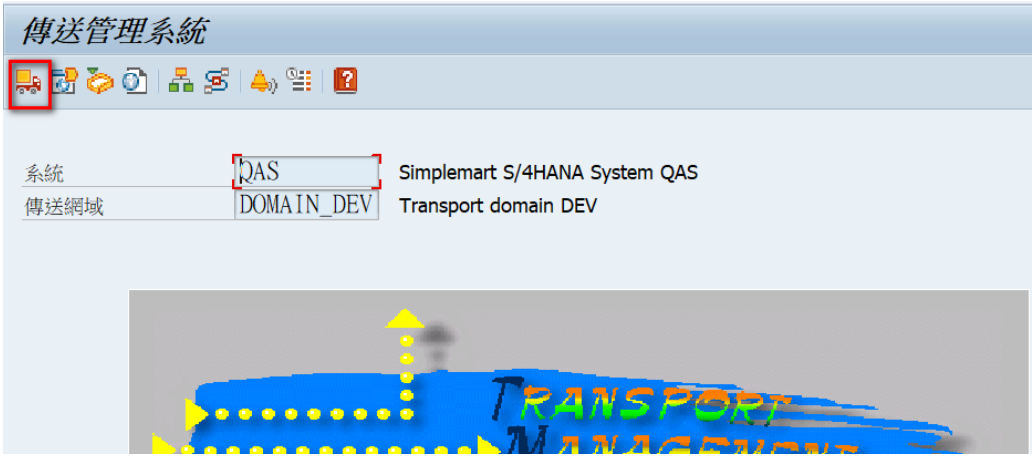


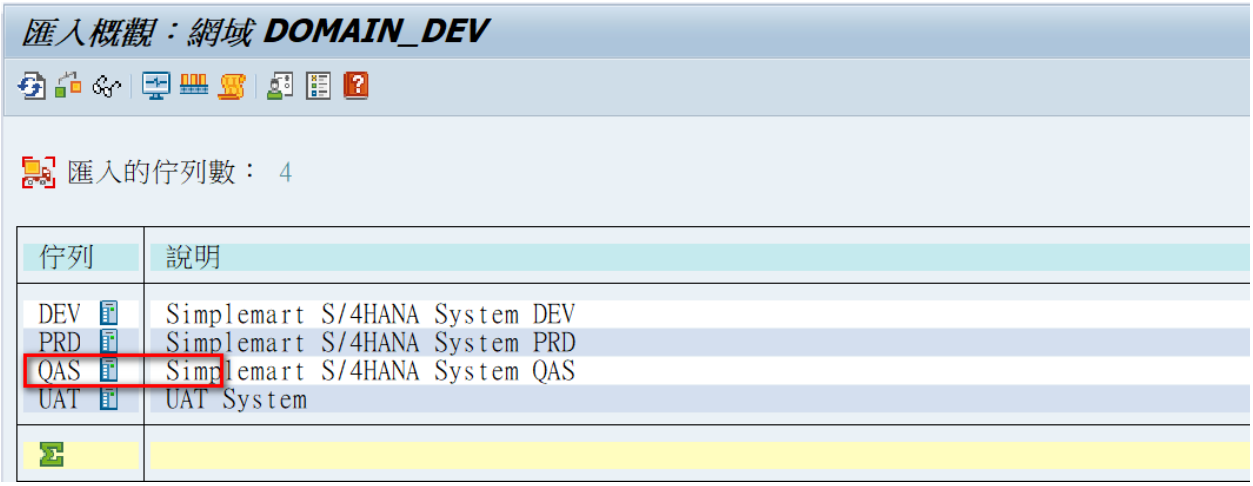
✓ 當 4187 核發後(全部程式)，出現下列的訊息



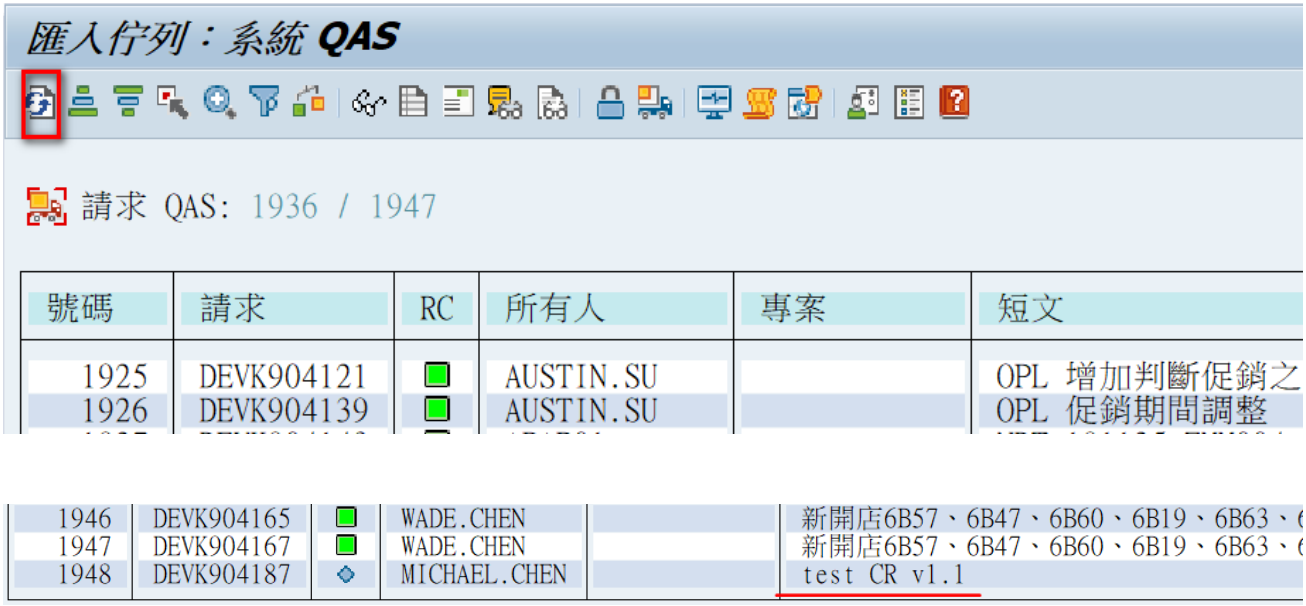
- STMS

✓ 接下來，請重新登入 QAS，準備執行匯入作業，先按下貨車 button，再選 QAS

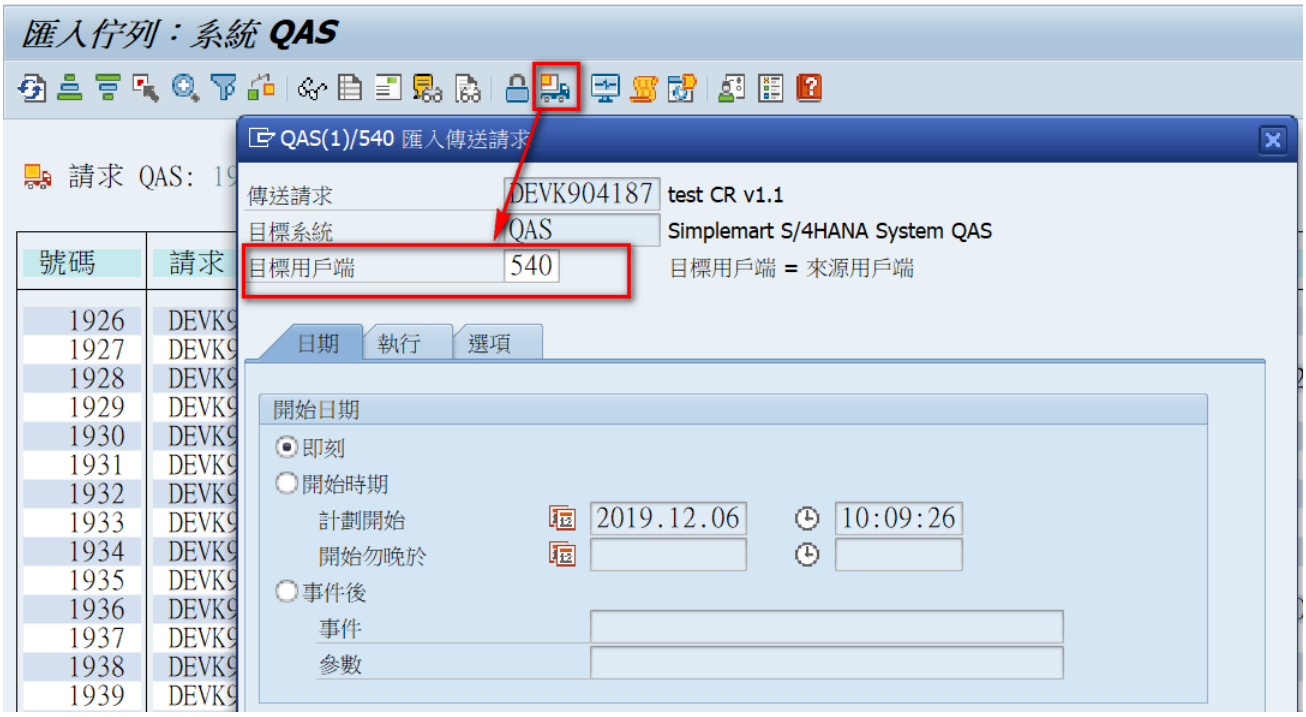




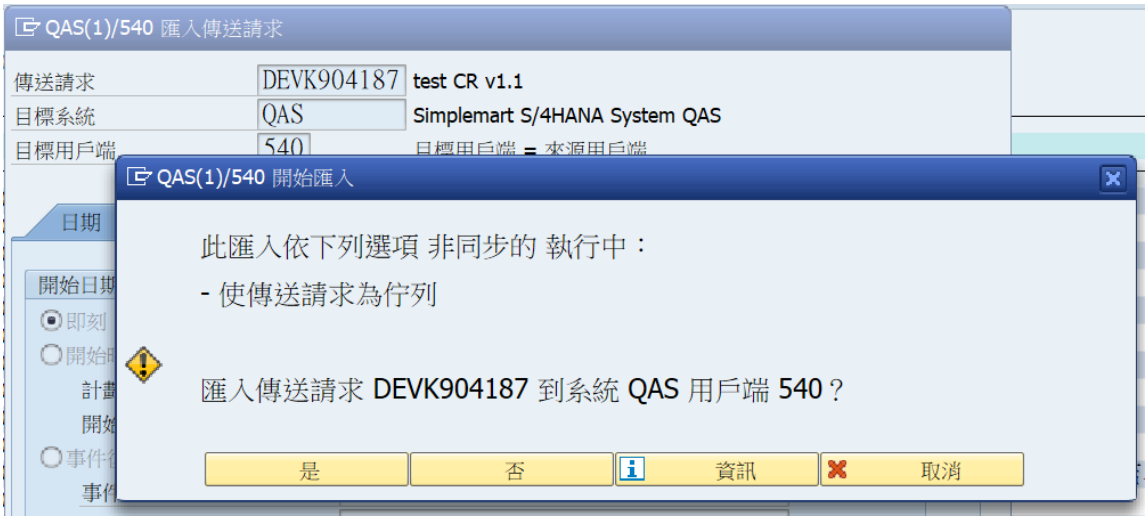
✓ 先按 Refresh，才會看到剛才由 DEV 傳過來的 CR

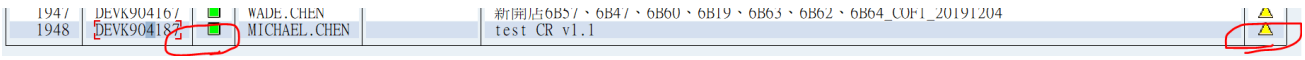


✓ 執行匯入，請務必要輸入目標用戶端的內容，我的 QAS 是 540(Client)。



✓ 按確定，會出現提示，再按 [是]，稍後等狀態變為綠燈即完成





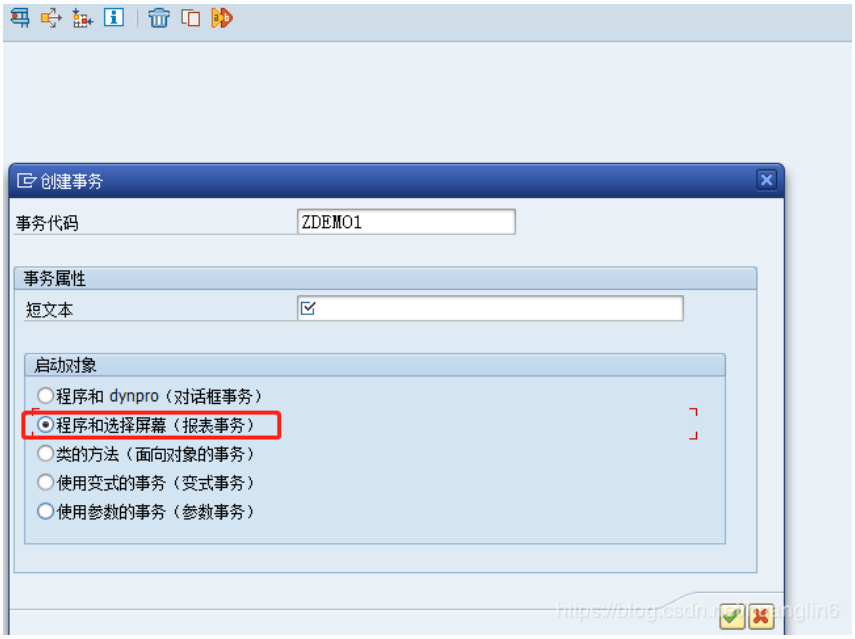
✓ **曾發生的 issue：因為我一開始開發沒有搞清楚應該先 DEV→QAS→PROD 的更新順序，而是直接在 QAS 開發，所以是先有 QAS 的程式，後有 QAS 的程式(相同程式名稱)，當我執行上面的作業時，會報錯(紅燈) 訊息看不太懂。後來就直接更新程式名稱，重新再來一次，就全部正常。**

✓ **T-Code 閃退問題** https://linhuang.blog.csdn.net/article/details/101083774

原因：TCODE 配置錯了！

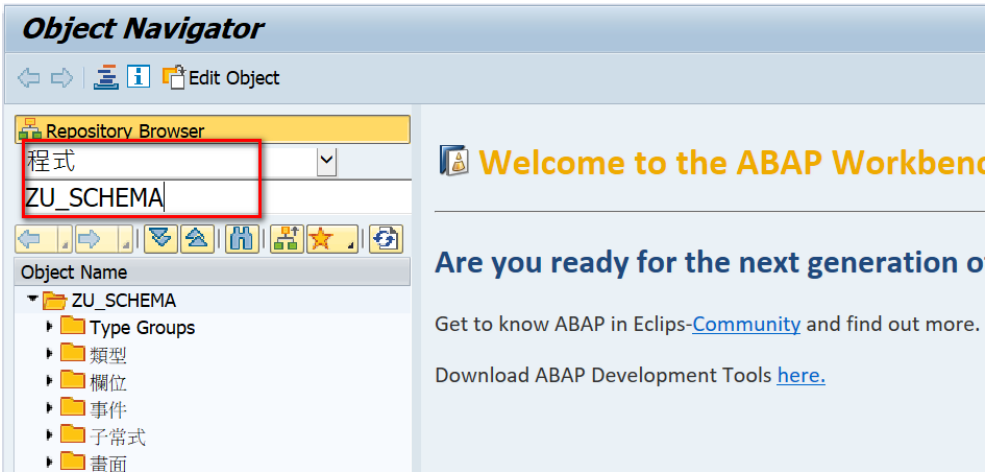
如圖，正常我們創建 TCODE，一般的 report 選擇的是第二個，預設螢幕是自動生成的 1000 螢幕，而 dialog 的 TCODE 選第一個，螢幕號給自建螢幕(非 1000)，而這邊他創建時候選的第一個，還分配了 1000 螢幕。。。So

解決辦法：重新創建個 TCODE：選第二個，問題解決。

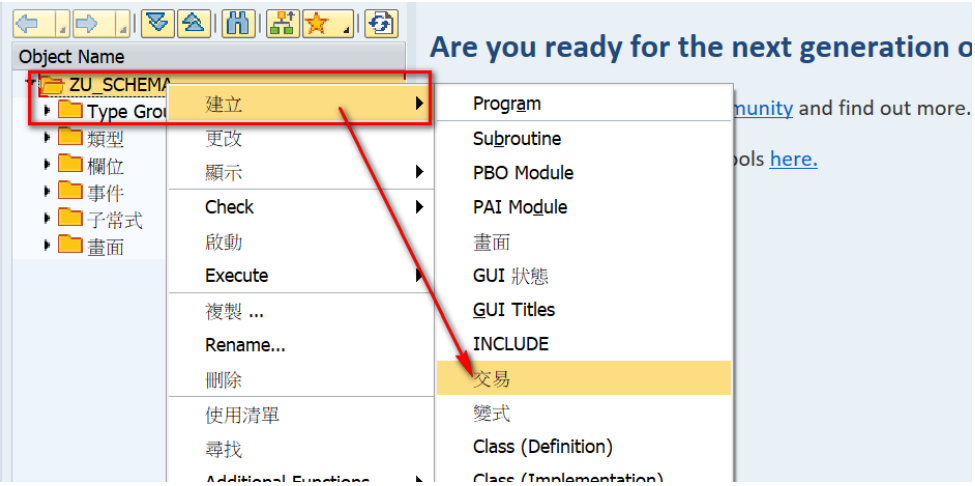


➢ 使用 SE80 指定程式的 T-Code

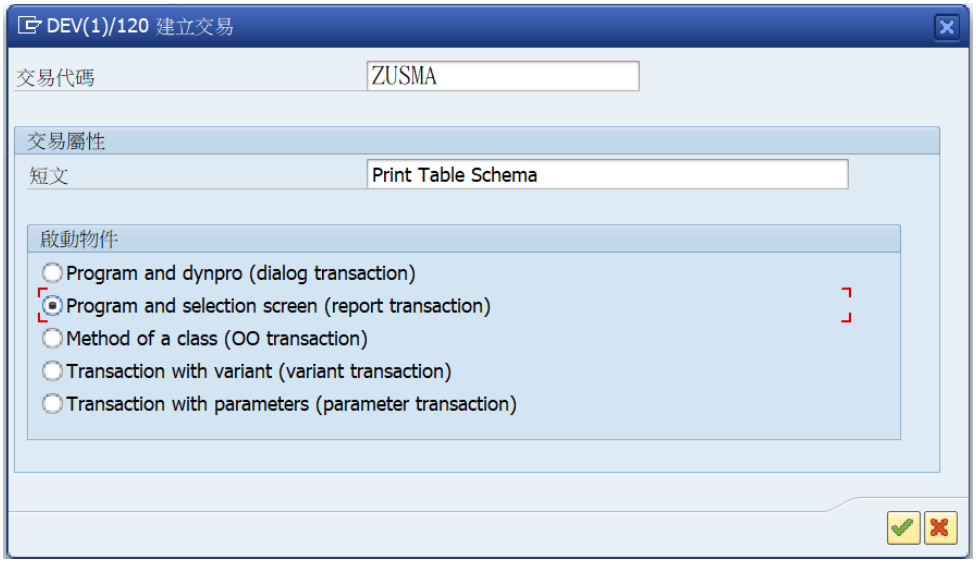
✓ 選擇 [程式] 及程式名稱(ZU_SCHEMA)



✓ 在程式名稱上按右鍵-建立-交易

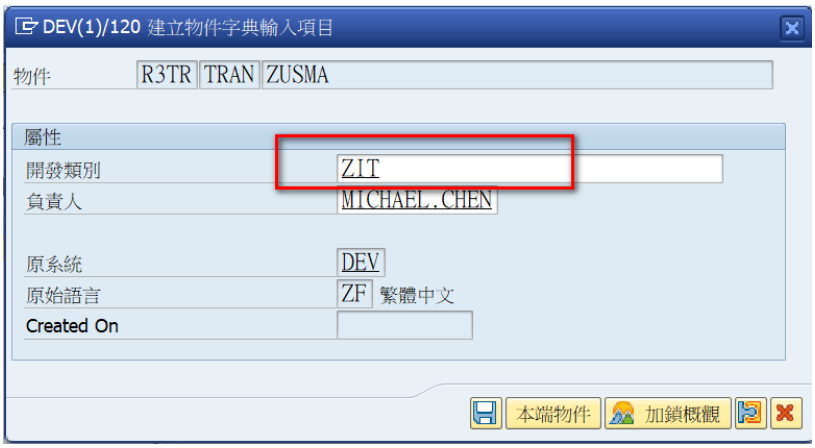


✓ 輸入下列資訊

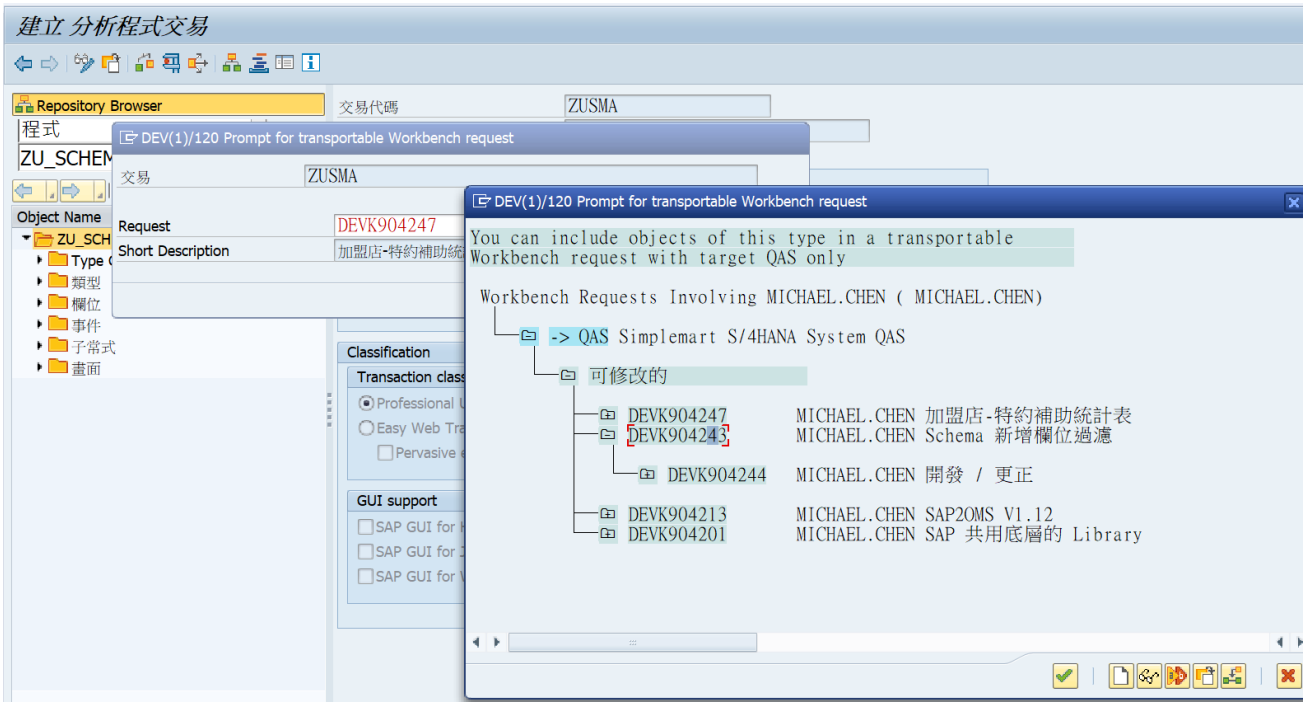


✓ 輸入程式名稱後按存檔

✓ 再輸入開發類別(Package)，可以開窗查詢自行選擇所屬的 Package



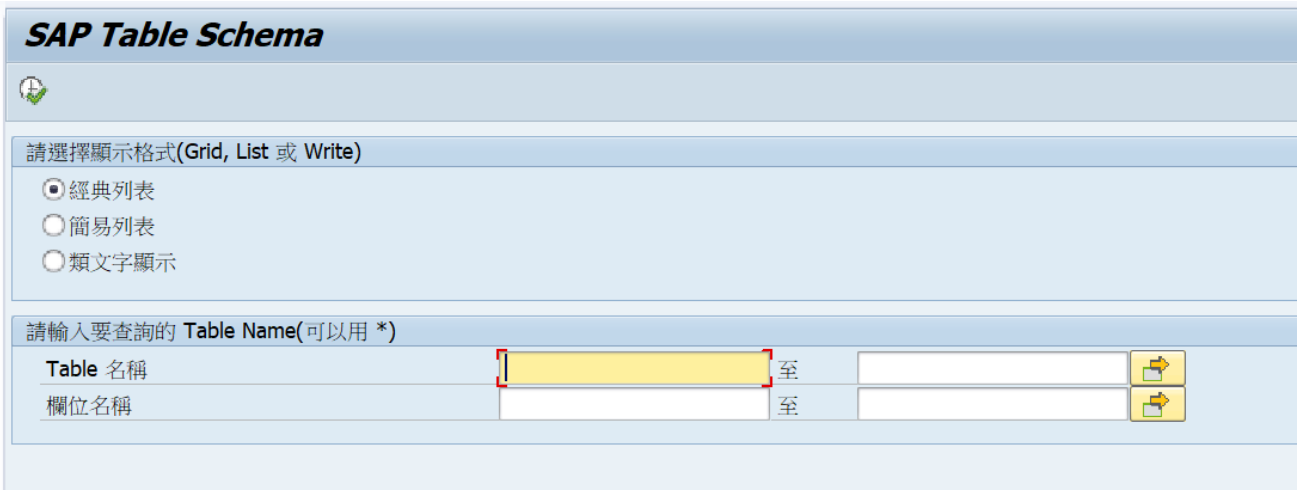
✓ 新增或選擇正確的 CR 代號，然後確定即可。



✓ 螢幕左下角會顯示操作正確的訊息

✓ 輸入 ZUSMA 這個 T-Code

✓ 即可正確的執行



- **與 SAP 資料整合的常用方式**

IDOC、BAPI、RFC 差別

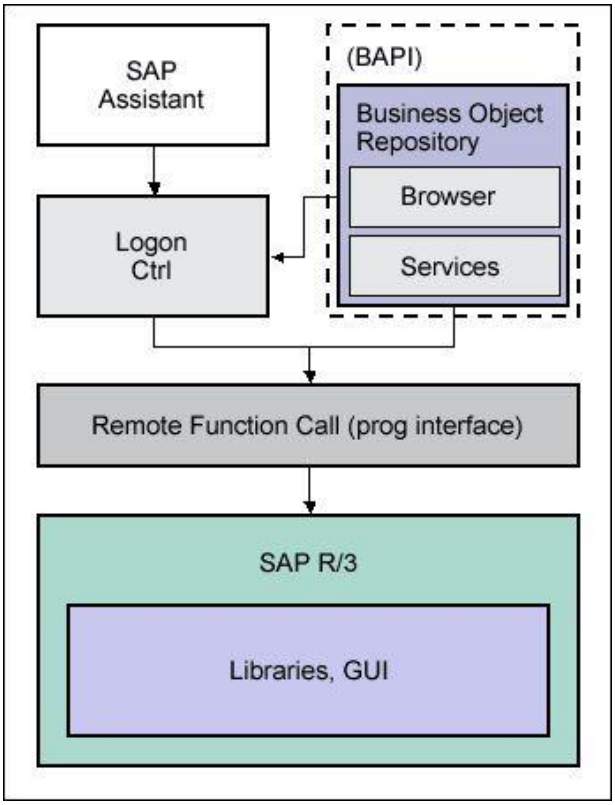
https://www.itread01.com/content/1546822276.html
https://www.stechies.com/difference-between-rfc-bapi/

- SAP 的資料整合方式主要有 RFC、IDOC、BAPI 三種。
  - RFC 分二種：1 正常的本地 RFC。2 遠端 RFC(外部系統調用)。
    SE37 寫出來的叫 function,其中可以遠端呼叫的叫 RFC，Remote-enabled Function Call，abap 語法和輸入輸出引數就會有一些限制。
  - BAPI（SAP 自帶業務函數--封裝好函數）。
    RFC 和 BAPI 都可以理解成一個函數。它有輸入、輸出，改變，排除 參數，表(既可以輸入也輸出)。一般都用在同步、線上的即時作業。
  - IDOC 是基於 sap 自己的類似 xml 格式的文件資料交換的方式。因為是文件格式，所以一般是非及時、非同步的。
- 三種方式的差異
  - RFC 是面向過程的，呼叫簡單直接；
  - BAPI 是面向物件的，有屬性、有方法、有事件，更加複雜和豐富，更能反映 SAP 的業務應用，而 BAPI 方法的構造是基於 RFC 的，你也可以認為 BAPI 封裝了 RFC，RFC 在應用時更為靈活。
  - IDOC 是 SAP 標準的檔案交換格式，SAP 已經有了大量的 Function Module 來處理和傳遞 IDOC，特別是對於要和其它系統交換資料時，配合一些系統如 biztalk server，IDOC 會顯得非常的方便，開發的工作量也是最小的(不過目前我還沒有實際的開發經驗，所以只能先聽聽)。

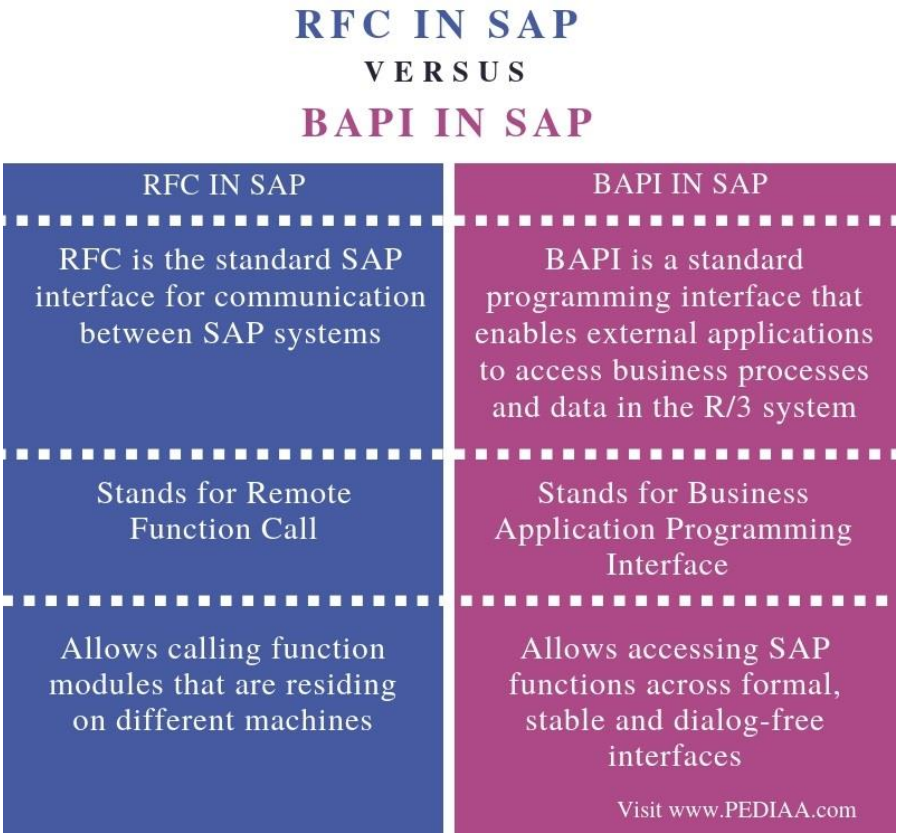
- 應該說 RFC 是其它內容的基礎，它是一個 Function module，可以被遠端呼叫。而 BAPI 本身就是一個 RFC，但它被作為 BO 的 Interface，作用更進一步，除了 BAPI 文件中提到的內容外，還可以作為 ALE/IDOC 的開發基礎。
  ALE 是 R/3 系統之間的應用層資料交換，至於用什麼，就看具體配置了，比如可以用 IDOC，同步 / 非同步 BAPI，甚至用 EDI。非 SAP 系統無法用 ALE 來實現。
  至於資料傳輸的方式，可以是 IDOC（底層是用 RFC 來實際的），也可以是 EDI，所以說 IDOC / EDI 實際上是資料的載體。

以上是關於 RFC、BAPI 及 IDOC 的簡單說明，基本上在 ABAP 程式中，RFC 當然是一切程式庫(Library)的基礎。我們先了解在 SAP 中有提供這些呼叫的方式，先由熟悉使用現有的程式為開始，等到更熟悉相關的技術後，再深入研究。這個部分屬於較進階的範圍，本文先不多做說明。可以在 ABAP 的介紹章節中，參考範例程式的寫法。

- BAPI 由外部呼叫 SAP 的流程



- RFC VS BAPI



第 11 章 SAP OpenSQL

http://www.sapjx.com/sap-open-sql.html

- 在 ABAP 程式中有兩種方式訪問資料庫：Open SQL 和 Native SQL

- Open SQL 是完全整合到 ABAP 語言中的 Standard SQL 的子集。它通過資料庫介面將 Open SQL 轉換成 Standard SQL，所以它可以用相同的方式訪問不同的資料庫系統，然後由該介面把 SAP 的 Open SQL 自動轉換為相應資料庫的特定 SQL 語句。

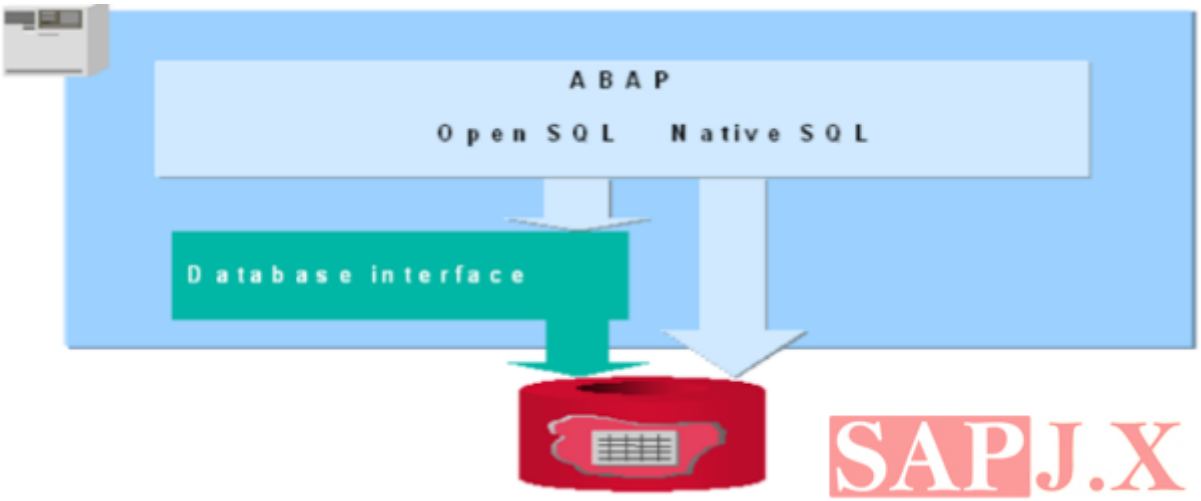
  Open SQL 包含 Standard SQL 的資料操作語言 Data Manipulation Language (DML)部分，允許你讀取和修改資料。

  Data Definition Language (DDL) 和 Data Control Language (DCL) 部分使用 ABAP Dictionary 處理。

  OPEN SQL 語句不是標準 SQL 語句，是 ABAP/4 語言(聽說有要朝 ANSI-92 標準靠攏, 希望能早日達成)

  通過 Open SQL 可以 insert、update、delete、modify table 的內容，但是不能 create and delete table。

- Native SQL 語法不會使用資料庫介面進行轉換，它直接訪問資料庫，所以一個 SQL 只能訪問一個指定資料庫，且具有一定的安全風險，對 SAP 執行效率也會有較大影響，所以一般不推薦使用。
  - ✓ Native SQL 和 ABAP 只是部分的和 ABAP 整合，和使用的 DATABASE 相關。
  - ✓ Native SQL 語句則是標準的 SQL 語句(該 DATABASE 的 SQL 語法)
  - ✓ Native SQL 語句是不經過 compiler，它直接針對 Databases 操作。
  - ✓ Native SQL 可以連結任何外部 Databases
  - ✓ 透過 Native SQL 可以 create and delete table



- Open SQL 語句

Open SQL 包含的 DML 語法有 4 種：SELECT, INSERT, UPDATE, DELETE, MODIFY

SELECT： 查詢語法

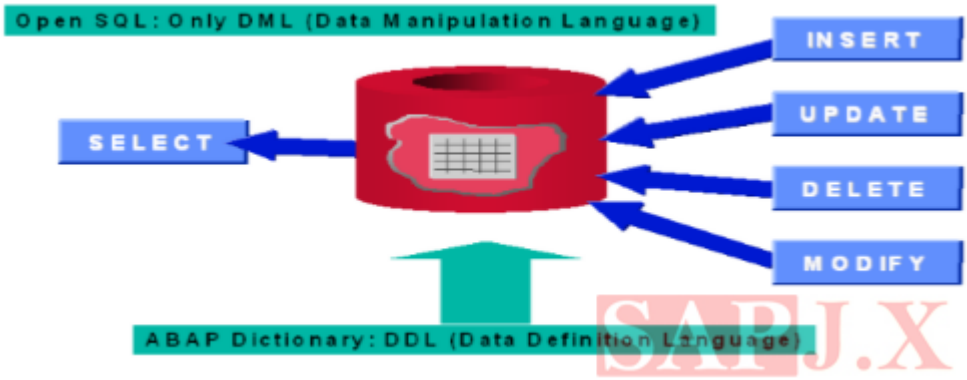
INSERT： 插入資料語法

UPDATE： 更新資料語法

DELETE：　刪除資料語法

MODIFY：　修改資料語法，相當於 INSERT + UPDATE

修改資料前會先在資料庫查詢是否存在該記錄，如果存在則修改，不存在則插入新的資料



Open SQL 執行成功後，會返回程式碼 SY-SUBRC = 0, 如果失敗 SY-SUBRC <> 0

除 SELECT 以外的命令，其他都會涉及到資料的修改，要謹慎使用。

在系統標準表中，我們只允許用 SELECT 命令去查詢資料，其他命令語句只能用於自建表(Zxx)，

如若要修改系統標準表，只能通過系統標準程式去操作。

在基礎階段，通常是配合 ABAP 寫客製報表，通常是使用 SELECT 語法取得資料，然後 into 到內表(Internal Table)，有些 Open SQL 無法做到的功能，通常會在 Internal Table 的 Modify 來達成報表需求，因為報表顯示的資料來源通常是處理後的 Internal Table。

➢ SELECT

這絕對是最常被使用的 SQL 語法，包含如下內容：

SELECT: 需要查詢資料庫指定表的那些欄位

INTO: 查詢的結果儲存在哪裡

FROM: 從哪些表查詢資料

WHERE: 指定查詢條件

GROUP BY: 以哪些欄位進行分組

ORDER BY: 以哪些欄位進行排序.

範例 1：

```
SELECT BLDAT,BWART,WERKS,sum( MENGE ) AS QTY,count( * ) AS RECNO INTO TABLE @itab
        FROM   zmm011_t01
        "UP TO 1000 ROWS
        group by BLDAT,BWART,WERKS,FILENAME
        HAVING BLDAT in @d_BLDAT and FILENAME like '%_WMS%'
        ORDER BY BLDAT,BWART,WERKS.
```

範例 2：

```
Select A~BEDAT AS BLDAT,A~BSART,B~WERKS,sum( B~MENGE ) AS QTY,count( * ) AS RECNO INTO
TABLE @itab
        From ZMM013_T01 AS A Left Join ZMM013_T02 AS B ON A~IHREZ = B~IHREZ
        group by A~BEDAT,A~BSART,B~WERKS
        HAVING A~BEDAT in @d_BLDAT
        ORDER BY A~BEDAT,A~BSART,B~WERKS.
```

- SELECT SINGLE 命令

SELECT SINGLE 命令允許你查詢一條記錄 ,為了確保你查詢的記錄是唯一的，你必須在你的 WHERE 子句指定所有 KEY 值，如若查詢的記錄不止一條，系統會返回程式碼 SY-SUBRC = 8，查詢結果為空；

```
SELECT SINGLE <F1> <F2>   FROM <dbtab>
INTO <work area>
INTO (<f1>, <f2>, <f3> ...   )
INTO      CORRESPONDING      FIELDS      OF <work area>
WHERE   <Key1> <op>   AND <Key2> <op> ...
```

在 INTO 中使用 CORRESPONDING FIELDS OF，系統只填充與你的 Work Area 相同名稱的欄位值；
執行成功返回程式碼 SY-SUBRC = 0,如果失敗 SY-SUBRC <> 0。

範例 3：TABLES: spfli.

```
DATA   wa_spfli LIKE spfli.
SELECT   SINGLE Carrid connid airpfrom airpto
FROM spfli INTO CORRESPONDINGFIELDS OF wa_spfli
WHERE carrid ='UA'ANDconnid ='3504'.
```

- SELECT 查詢多條記錄

SELECT 查詢多條記錄有兩種方式:
1.使用 SELECT ... ENDSELECT 命令，進行迴圈處理；
2.一次性讀取多筆記錄到內表中，在 INTO 使用 TABLE 附加字(個人建議寫法)

第一種方式語法格式：

```
SELECT <F1> ... <Fn>   FROM <dbtab>
INTO <work area>
INTO (<f1>, <f2>, <f3> ...   )
INTO CORRESPONDING FIELDS OF <work area>
WHERE   <tabfield> <operator> <value>.
ENDSELECT.
```

第二種方式語法格式：

SELECT <f1> ... <fn> FROM <dbtab>

INTO TABLE <itab>

WHERE ...

範例 4-1：

```
DATA tab_spfli TYPE STANDARD TABLE OF spfli.
DATA wa_spfli  LIKE spfli.
SELECT carrid connid airpfrom airpto
FROM spfli INTO wa_spfli
WHERE carrid ='LH' ANDconnid ='400'.
ENDSELECT.
```

範例 4-2：

```
SELECT carrid connid airpfrom airpto
FROM spfli
INTO CORRESPONDING TABLE OF tab_spfli
WHERE carrid ='LH' AND connid ='400'.
```

注意：使用 SELECT... ENDSELECT 語法中不能加入 SINGLE、INTO CORRESPONDING FIELDS 、TABLE 等關鍵字，中間可以使用 CHECK 語法來判斷查詢值。

- SELECT 附加選項(ABAP 專用的功能)

SINGLE [FOR UPDATE]: 使用它會產生一個排他鎖，其他程式不能查詢此筆記錄。

範例 5：

```
* 星號表示查詢所有欄位
SELECT SINGLE FOR UPDATE *
FROM SCARR
WHERE currcode ='USD'.
WRITE : /  scarr-CARRID.
```

DISTINCT：查詢非重複記錄，需要使用 ORDER BY 排序

- SELECT 標準函式

在使用 SELECT 語句時，可以直接使用 Open SQL 提供的標準函式進行相關操作，常用的標準函式如下：

1.COUNT()： 統計查詢記錄總數

2.SUM(): 彙總某查詢數字欄位數量

3.AVG()：計算某查詢數字欄位平均值

4.MAX()：查詢表中記錄最大值

5.MIN():查詢表中記錄最小值

使用這些函式時，如果要包含其他欄位，必須使用 GROUP BY 命令**(請參考範例 2)**

- FROM 子句
  - ✓ FROM dbtab [AS alias] - AS 附加字，用於對 Table 另起別名
  - ✓ JOIN 聯合查詢

    JOIN 的聯合查詢有兩種， INNER JOIN 和 LEFT/RIGHT [OUTER] JOIN

    通過 ON 語句關聯 2 個或 2 個以上的表，且必須至少指定一個連線條件；在多個連線條件時，可以使用 AND；ON 左右的條件只能使用 = (EQ) 進行連線。

範例 6：

```
SELECT SINGLE B~BUKRS
FROM BSEGASB
INNER JOIN T001 AS TONB~BUKRS = T~BUKRS.
```

其中，INNER JOIN 與 LEFT/RIGHT JOIN 的區別：

INNER JOIN 的 ON 條件下，兩個表資料必須完全匹配；

LEFT JOIN 的 ON 條件下，左邊的表需滿足，右邊的表不用完全匹配。

  - ✓ BYPASSING BUFFER - 繞過 Application Buffer，直接讀取資料庫(不常用)。
  - ✓ UP TO n ROWS - 查詢 n 筆記錄，限制讀取資料的筆數。**(請參考範例 1)**
- WHERE 子句

WHERE 關鍵字後，可帶子句包括：

1. <field> OP g

OP 包括：=, EQ, <>, NE, <, LT, >, GT, <, LE, >=, GE, <=, LE 等關係操作符

2. <field> [NOT] BETWEEN g1 AND g2： 包含 g1 和 g2

範例 7：

```
SELECT carrid INTOTABLE itab
FROM scar WHERE carrid BETWEEN 'AA'  AND 'AZ'.
LOOP AT ITAB.
WRITE: / ITAB-CARRID.
ENDLOOP.
```

3. <field> [NOT] LIKE g ：通過萬用字元('_' , ' %')模糊查詢

範例 8：

```
SELECT * FROM mara
WHERE matnr LIKE '101-% '.
*   WHERE matnr LIKE '101-___ '.
WRITE : / mara-matnr.
ENDSELECT.
```

這裡強烈建議，儘量避免使用 LIKE 語法作為條件篩選，因為模糊查詢的資料量多數時候會非常龐大；
這樣會直接消耗系統資源，影響系統性能。

4. <field> [NOT] IN (g1, ..., gn) : 多個組合作為篩選條件
5. <field> [NOT] IN itab ： 使用 Range table 作為篩選條件
6. <field> IS [NOT] NULL ：篩選某欄位是否為空

由於內表可以臨時儲存多條資料，而 Open SQL 允許將內表資料作為查詢條件，故可以通過語句參照內表作為條件查詢。

相當於使用 INNER JOIN 連線兩個表一樣，然後在資料量龐大的時候，會比 INNER JOIN 的查詢快捷。
兩者各有優缺點，視具體情況而定。

範例 9： * INNER JOIN 用法：

```
SELECT DISTINCT knb1~bukrs t001~butxt
FROM knb1 INNER JOIN t001 ON knb1~bukrs = t001~bukrs
INTO CORRESPONDING FIELDSOF TABLE tb_bukrs
WHERE kunnr IN rn_kunnr.
```

範例 10： * FOR ALL ENTRIES IN 用法：

```
SELECT DISTINCT bukrs
INTO CORRESPONDING FIELDSOF TABLE gt_knb1
FROM knb1.
IF gt_knb1[] IS NOT INITIAL.
SELECT DISTINCT butxt FROM t001
INTO CORRESPONDING FIELDSOF TABLE gt_too1
FOR ALLENTRIES IN gt_knb1
WHER Ebukrs = gt_knb1-bukrs.
END IF.
```

注意事項：

- ✓ 使用該語句，對於最後得到的結果集，系統會自動刪除重複行。
  因此如果你要保留重複行記錄時，記得在 SELECT 語句中新增足夠鍵值專案（有必要時，增加全部鍵值專案），以保證結果集中所需重複專案不會被刪除。（例如選取支付金額時，支付事件可能不同，但金額可能相同，此時一定要注意，以避免錯誤刪除結果記錄。）
- ✓ 後面使用的內部表 itab 如果為空，系統將當前 CLIENT 下所有記錄選出。
  因此為避免無意義的檢索，在使用該語句前一定要判斷內部表 itab 是否為空，為空時不執行包含該語句的資料庫檢索處理。
- ✓ 由於 itab-f 實際上是作為佔位符被替換，所以內部表 itab 中不要包含 HEADER 行（專案標識名稱行），以免造成混淆，檢索出錯。
- ✓ 內部表 itab 中作為條件替換用專案的型別和長度，一定要和檢索資料庫中對應的專案相同，否則編譯不能通過。
- ✓ 對於內部表 itab 中 作為條件替換用專案，不能使用 LIKE,BETWEEN,IN 比較操作符。
- ✓ 使用該語句時，ORDER BY 語句和 HAVING 語句將不能使用。
- ✓ 使用該語句時，除 COUNT(*)以外的所有標準合計函式（MAX,MIN,AVG,SUM）都不能使用。

➢ INSERT 語句

INSERT 語法結構：

INSERT INTO <dbtab> VALUES wa.

INSERT <dbtab> FROM TABLE itab.

範例 11：

```
TABLES scustom.
scustom-mandt        ='002'.
scustom-id           ='12400177'.
scustom-name         ='Robinson'.
scustom-postcode     ='69542'.
scustom-city         ='Heidelberg'.
scustom-custtype     ='P'.
scustom-discount     ='003'.
scustom-telephone    ='01234/56789'.
INSERT INTO scustom CLIENT SPECIFIEDVALUES scustom.
```

CLIENT SPECIFIED: 是否指定 CLIENT

如果指定 CLIENT，則只插入指定 CLIENT 的相關資料；

如果用在 WHERE 語句，則是抽取指定 CLIENT 的相關資料；如若沒有指定 CLIENT，則抽取所有 CLIENT 的資料。

範例 2：

```
TABLES SPFLI.
DATA ITAB LIKE SPFLIOCCURS10 WITH HEADER LINE.
ITAB-CARRID = 'UA'. ITAB-CONNID = '0011'. ITAB-CITYFROM = ..
APPEND ITAB.
ITAB-CARRID = 'LH'. ITAB-CONNID = '1245'. ITAB-CITYFROM = ..
APPEND ITAB.
ITAB-CARRID = 'AA'. ITAB-CONNID = '4574'. ITAB-CITYFROM = ..
APPEND ITAB.
................
INSERT SPFLI FROM TABLEITAB ACCEPTINGDUPLICATEKEYS.
IF SY-SUBRC = 0.
.....
ELSEIF-SUBRC = 4.
.....
ENDIF.
```

➢ UPDATE 語句

1. UPDATE <dbtab> SET f1 ... fn [WHERE sql_cond] .
2. UPDATE <dbtab> FROM TABLE itab [WHERE sql_cond] .

範例 12：

```
UPDATE scustom SET: DISCOUNT    ='003',
TELEPHONE ='0621/444444'
WHERE ID ='00017777'.
TABLES scustom.

DATA wa LIKE scustom.
wa-id              = '12400177'.
wa-telephone    = '06201/44889'.
UPDATE scustom FROM wa.
```

➢ MODIFY 語句

1.MODIFY <dbtab>. （相當於 INSERT 和 UPDATE）

2.MODIFY <dbtab> FROM TABLE itab.

範例 13：

```
TABLES scustom.
scustom-id          ='12400177'.
scustom-name        ='Robinson'.
scustom-postcode    ='69542'.
scustom-city        ='Heidelberg'.
scustom-custtype    ='P'.
scustom-discount    ='003'.
scustom-telephone   ='06201/44889'.
MODIFY scustom.
```

➢ DELETE 語句

1. DELETE FROM <dbtab> WHERE cond.

2. DELETE <dbtab> FROM TABLE itab.

Delete 最好有條件限定，謹慎使用，避免誤刪資料

範例 14：

```
DELETE FROM SBOOK
WHERE CARRID ='LH' ANDCONNID ='0400' AND  FLDATE ='19950228'.
```

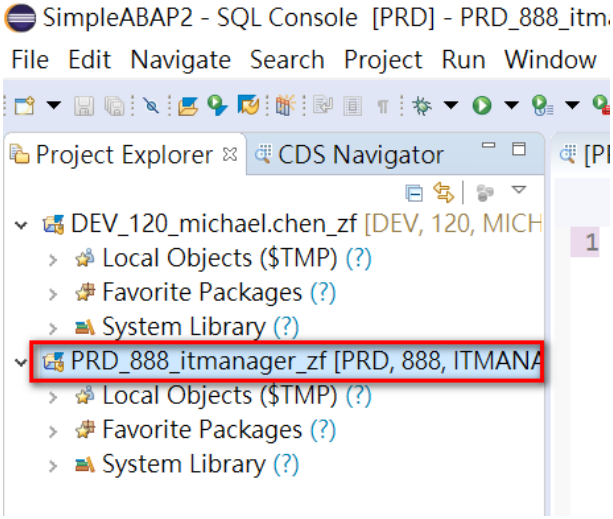
➢ 最後補充一個如何下任何 Open SQL 語法，即可查詢 SAP 資料庫內容的 SQL Explorer 方式

http://www.voidcn.com/article/p-ahspensi-brx.html

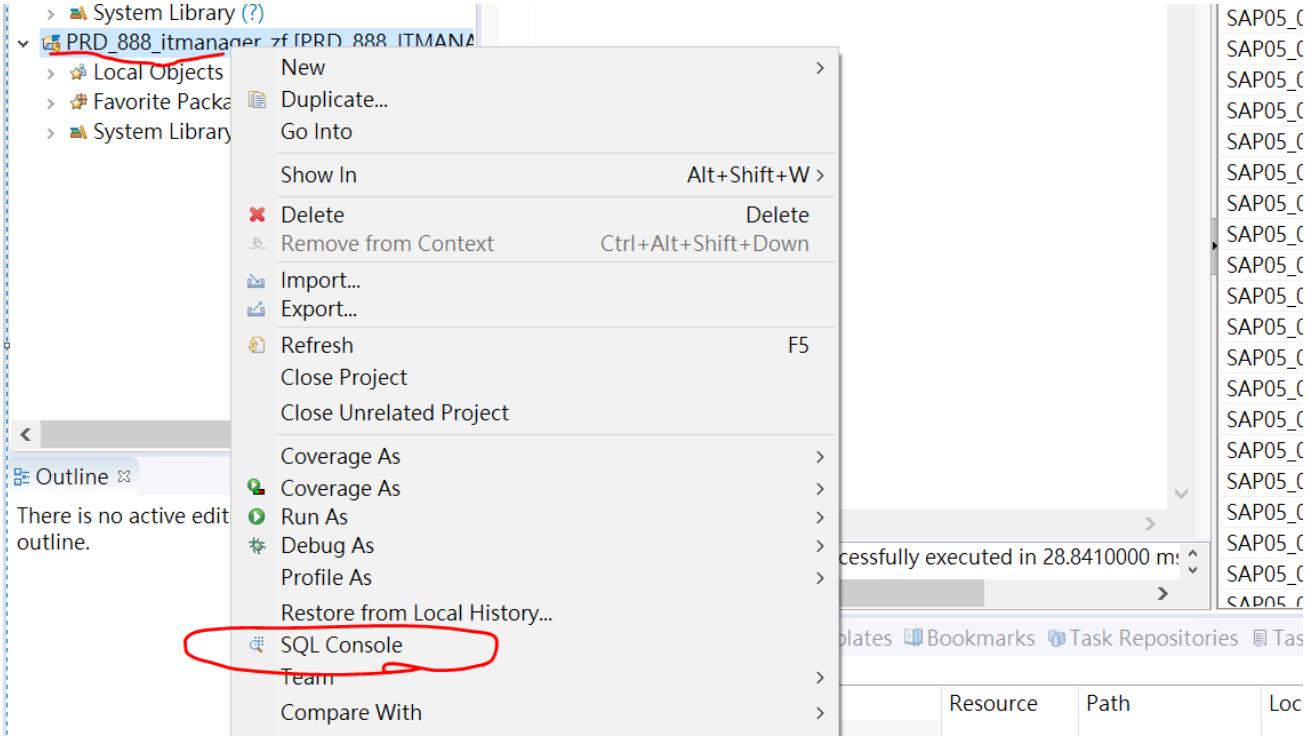
一開始花了一點時間想找現成的 abap 程式，但是要嘛是商業版本，不然就是版本太舊，要嘛下載要付費。後來我找到上面的 blog，他提到 Eclipse 有現成的工具可以使用，半信半疑的情況下，一試之下驚為天人，果然是好物一枚，故紀錄分享於下

安裝 Eclipe 可參見顧問教學：\\10.0.8.14\sap 導入專案\專案公用資料夾\Phase 1 專案準備\1.X 影音檔\ABAP\20180918-ABAP DAY2\錄製_2018_09_18_15_21_29_858.mp4

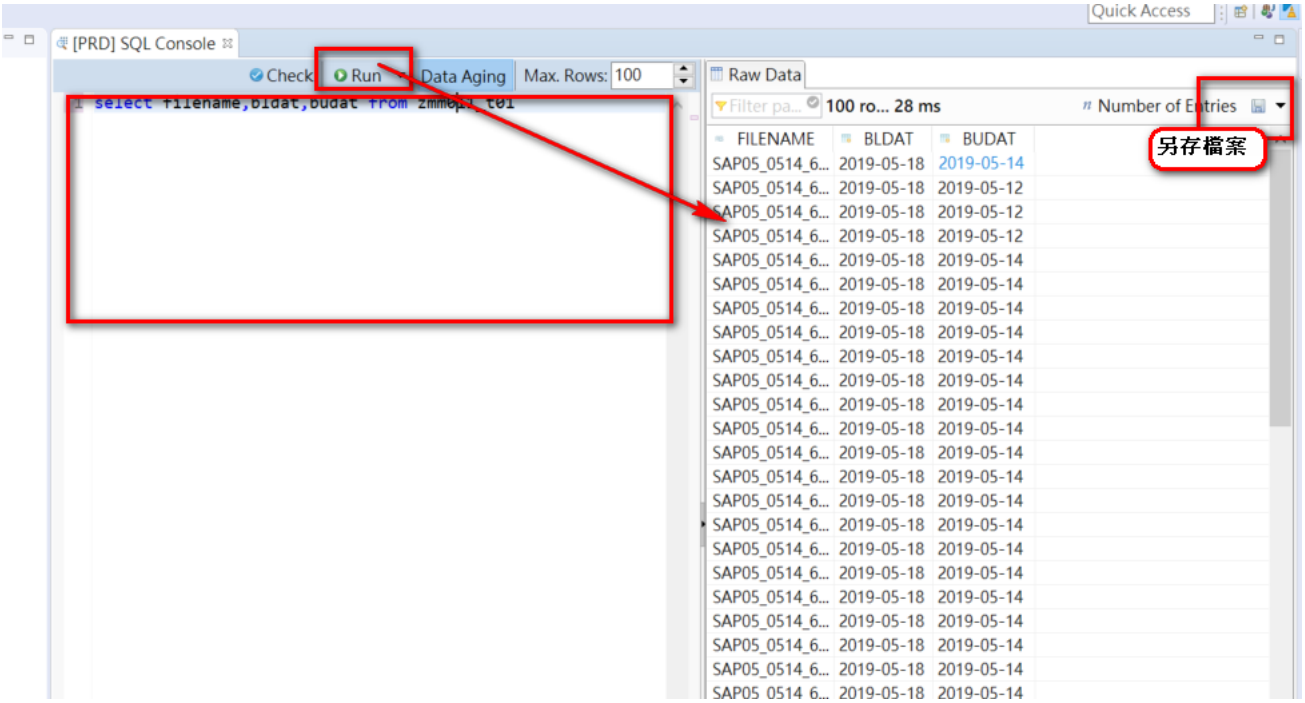
A. 請先安裝好 Eclipse，並建立連線到 SAP



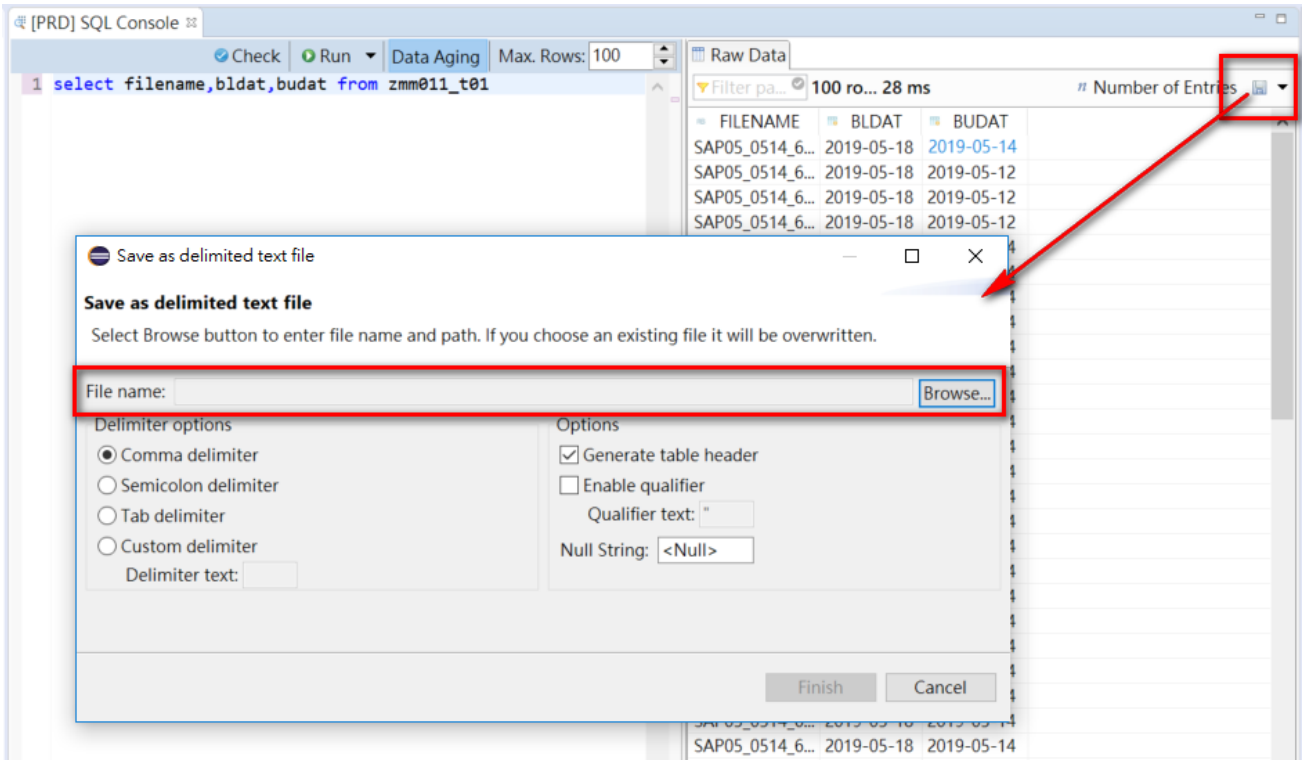
B. 在指定連線的 ABAP 專案案右鍵，選擇 [SQL Console]



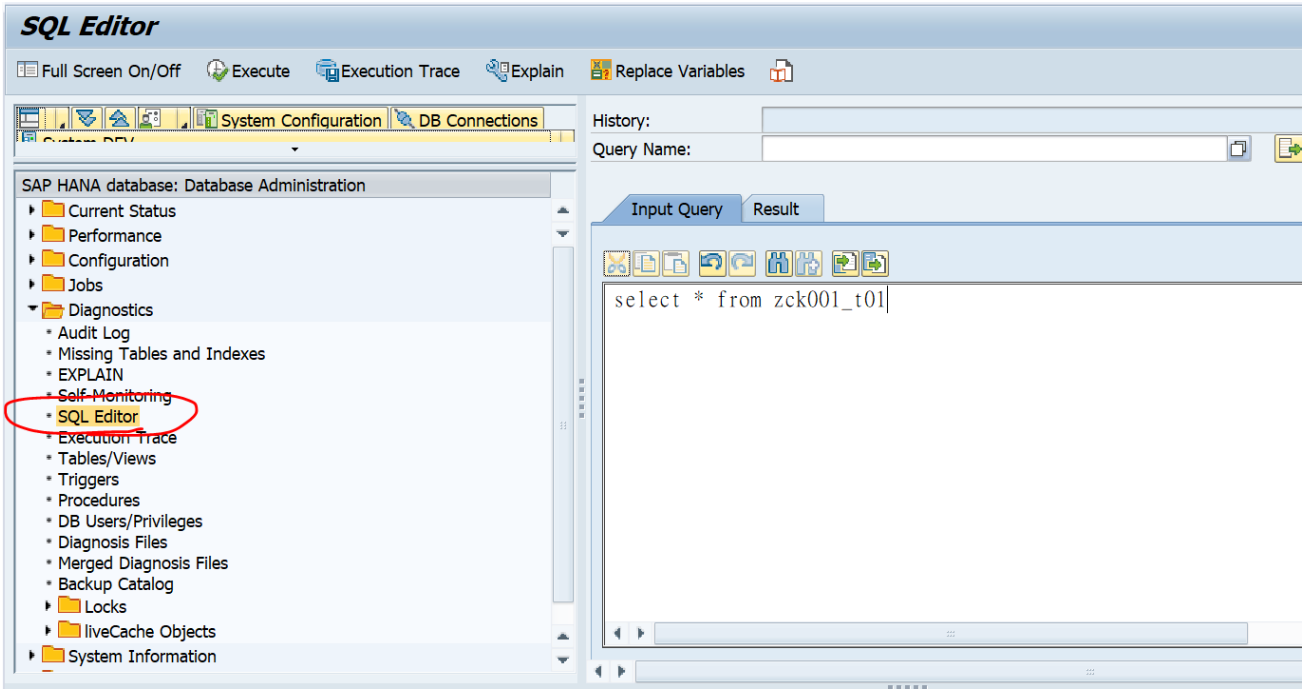
C. 就可以執行 Open SQL 語法

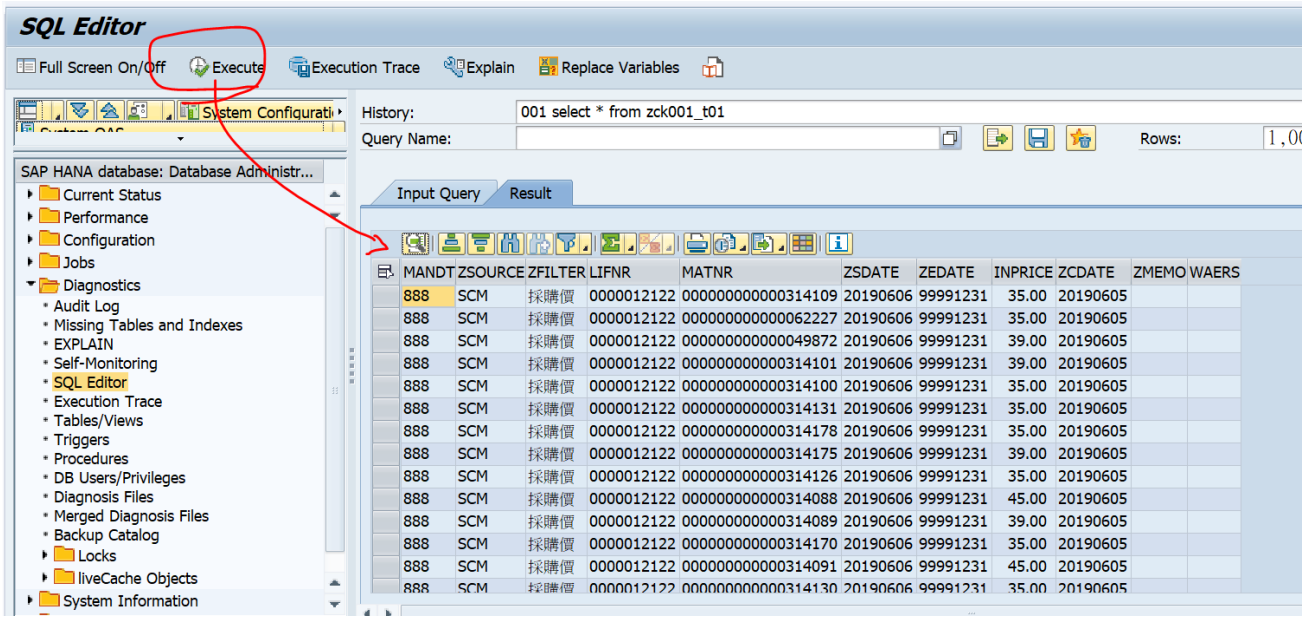


D. 查詢結果可以另存新檔



➢ 補充說明，上 Spancer 課時有提到，事實上 T-Code=DB02 本來就有類似的功能，不過 SQL 是 Native SQL(HANA)，不是 Open SQL。





➢ end

第 12 章 用 SM30 撰寫單檔維護程式

https://www.fenginfo.com/2277.html
http://www.baidusap.com/abap/6230

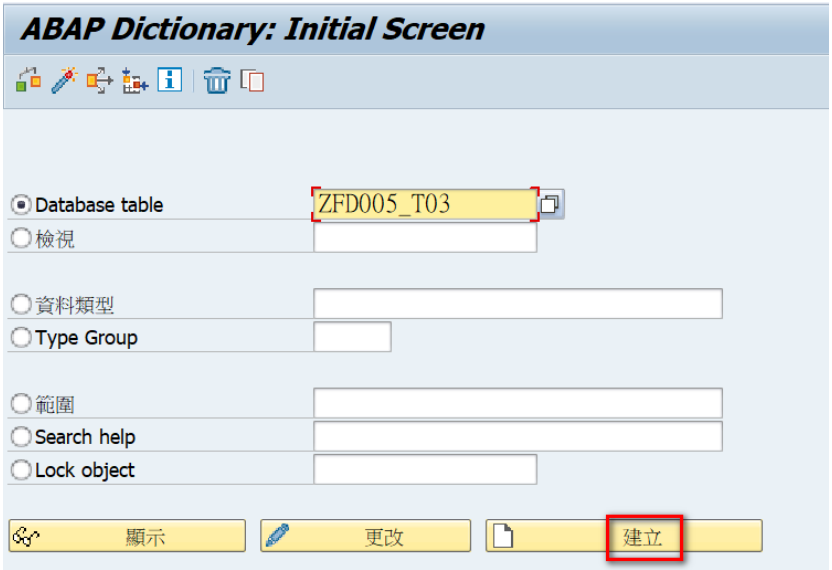
當有開發客製程式的需求時，SAP 提供了 ABAP，可以讓工程師自行設計程式，來解決客戶端所提的各式功能需求。但是要熟悉 ABAP 畢竟不是短期內可以精通的，如果我們有幾支單純的單檔程式需要維護(包含新增、刪除、修改)，有沒有甚麼方法可以透過簡單的設定，就可以快速的定義(開發)一支簡單的單檔維護程式? 原本以為這只是一個美好的願望，沒想到 SAP 真的就提供 SM30 可以做到。而且自訂的視圖(View) 也可通過 SE93 分配獨立的事物碼，這實在是太神奇了。

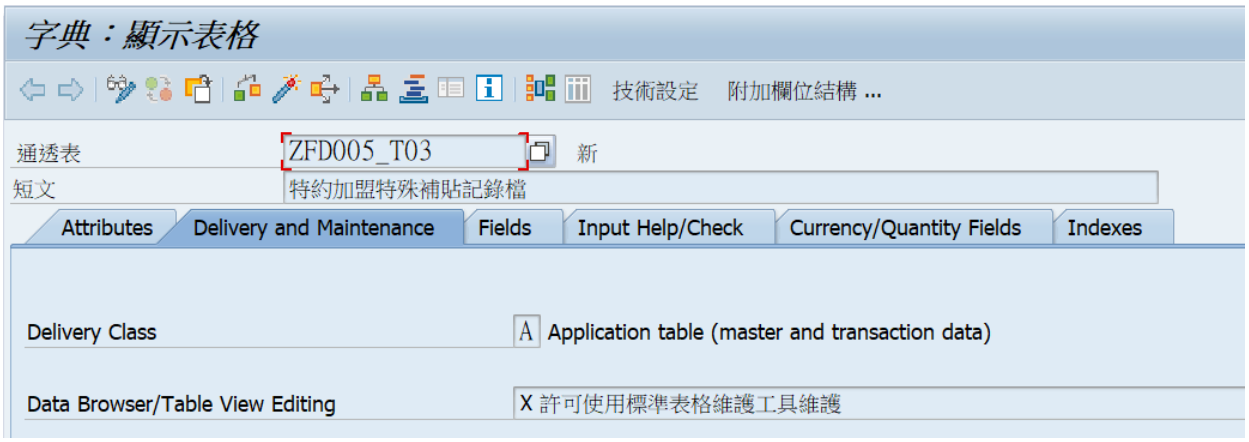
自訂維護視圖包含兩大步驟：

✓ 定義維護視圖的表結構；

✓ 生成表維護程式。

➢ 本文假設是要對自行新增的 Table 建立維護功能，先用 SE11 建立 DataBae Table=ZFD005_T03





✓ 切換到 Fields 頁籤，開始逐項輸入欄位，這個地方一開始會有點卡，主要是下列知識點要先掌握，否則會不曉得到底哪裡弄錯。請注意，SAP 不允許直接對 DB 層級的操作，也就是我們以前會習慣直接連到 DataBase 下 Create Table xxx 或 Update / Delete 等指令，基本上在 SAP 是不被允許的。
請習慣使用 SE11 及 SE16N 等 T-Code 操作 DB，Eclipse 有開發 SQL Console 功能，但也僅限於方便查詢(請參見第 10 章的說明)。

✧ 首先，請先搞懂 SAP 中的 View-Tables-Data Element 及 Domain 的關係。

- ◆ View 可以是多 Tables 的 Join 聯集(當然也可以是單一 Table)
- ◆ Table 等同於我們熟悉的 DataBase Table，是實際存放資料的主體
- ◆ Data Element(以下簡稱 DE) 資料庫字典，通常每個欄位都有自己的 DE，不同表但相同欄名可以設為同一個 DE，別的系統通常稱為 DD→ Data Directory，是相同的意義。
  請注意：欄位名稱不一定要和 DE 名稱相同(當然設為相同是常見的作法)，如果不同，通常兩者的名稱還是會有關聯性，方便操作記憶，如果要自定義 DE，必須以 Z 開頭。
- ◆ 至於 Domail(域)，則是相同性質的 DE，可以歸屬到同一個 Domain，方便設定，如多個欄位(不同欄名，都是統計類的數值欄位，可以歸屬於同一個 Domain，方便設定，稍後會有實例說明)。

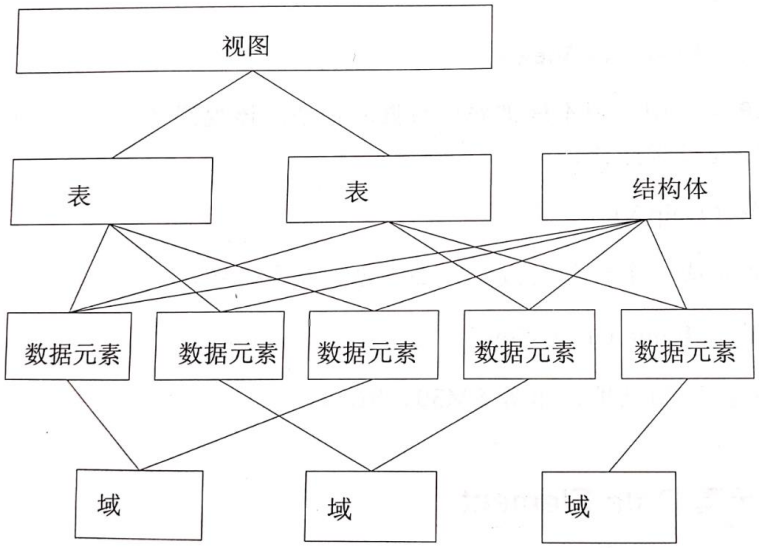
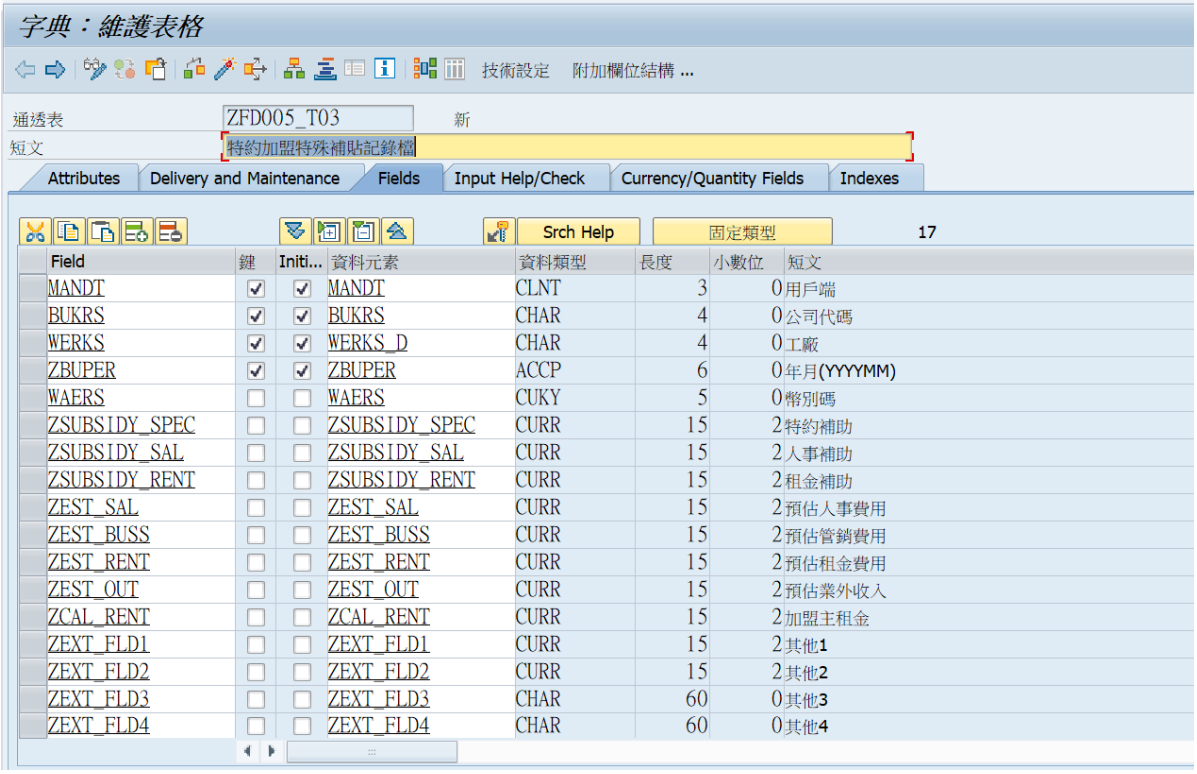


图 5-2 视图、表、结构体、数据元素和域的关系

✧ 以下我們會以新增如下 Table 為例說明如何新增 Table、設定 DE、使用 Domain

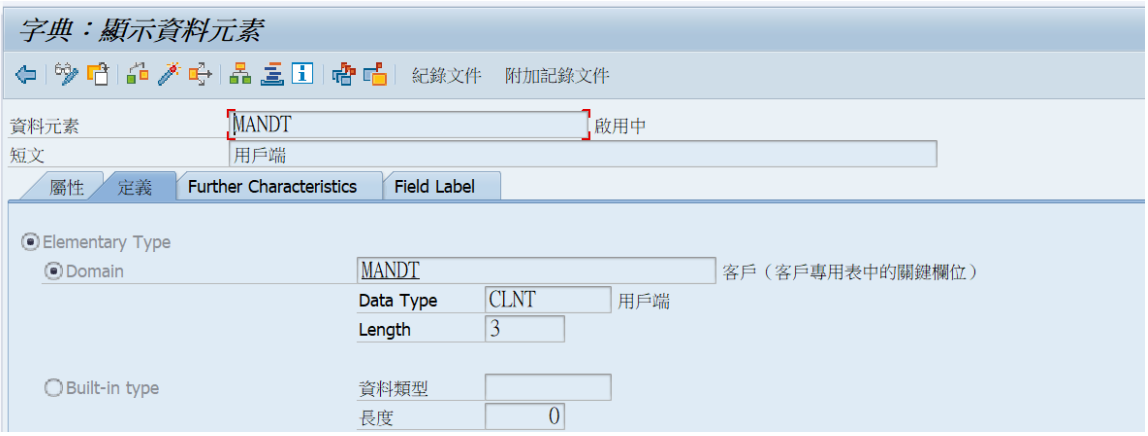
新增 Table ZFD005_T03 特約加盟特殊補貼記錄檔				
MANDT	用戶端	X	C	000006
BUKRS	公司代碼	X	C	000008
WERKS	工廠	X	C	000008
ZBUPER	年月(YYYYMM)	X	N	000012
WAERS	幣別		C	000010
ZSUBSIDY_SPEC	特約補助		P	000008
ZSUBSIDY_SAL	人事補助		P	000008
ZSUBSIDY_RENT	租金補助		P	000008

ZEST_SAL	預估人事費用	P	000008
ZEST_BUSS	預估管銷費用	P	000008
ZEST_RENT	預估租金費用	P	000008
ZEST_OUT	預估業外收入	P	000008
ZCAL_RENT	加盟主租金	P	000008
ZEXT_FLD1	其他 1	P	000008
ZEXT_FLD2	其他 2	P	000008
ZEXT_FLD3	其他 3	C	000100
ZEXT_FLD4	其他 4	C	000100



✧ 以下我們舉幾個欄位的新增來說明應注意事項

◆ MANDT 是 SAP 的 Client，就是 Login 時的 DEV 120/PROD 888 代號，通常都會固定使用此欄位，欄位名稱和 DE 相同，在 DE 輸入 MANDT 後會自動帶出後方的欄位，也可以在 MANDT 欄位 DbClick 可以開啟 DE 的設定。

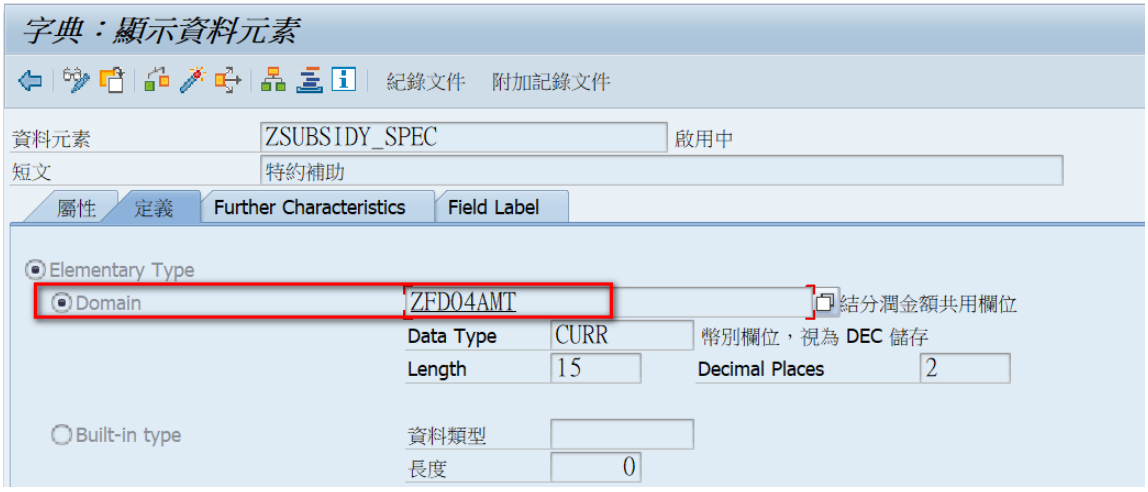


◆ WERKS 要稍微注意一下，如果 DE 輸入 WERKS 會出現 STRU(Structure)，請改為 WERKS_D。

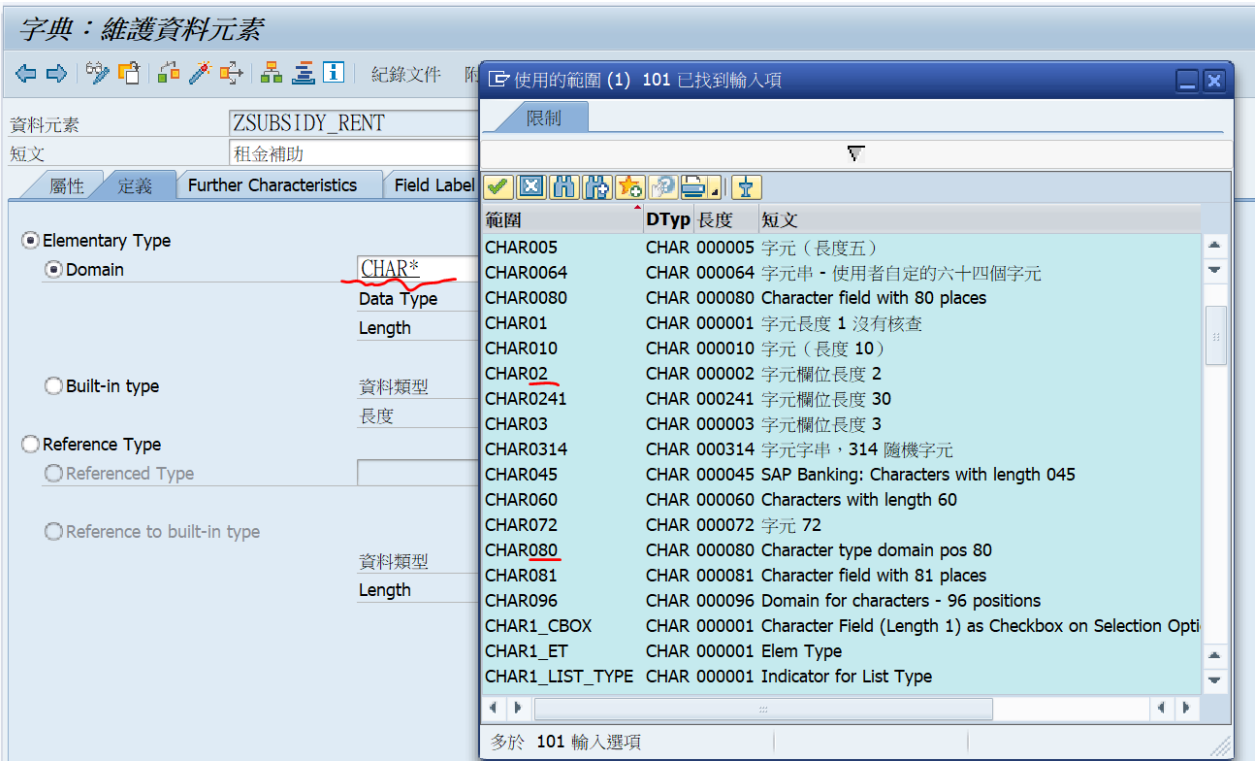
BUKRS	☑	☑	BUKRS	CHAR	4	0 公司代碼
WERKS	☑	☑	WERKS	STRU	0	0 國家契約工廠表（中央同意）
ZBUPER	☑	☑	ZBUPER	ACCP	6	0 年月(YYYYMM)

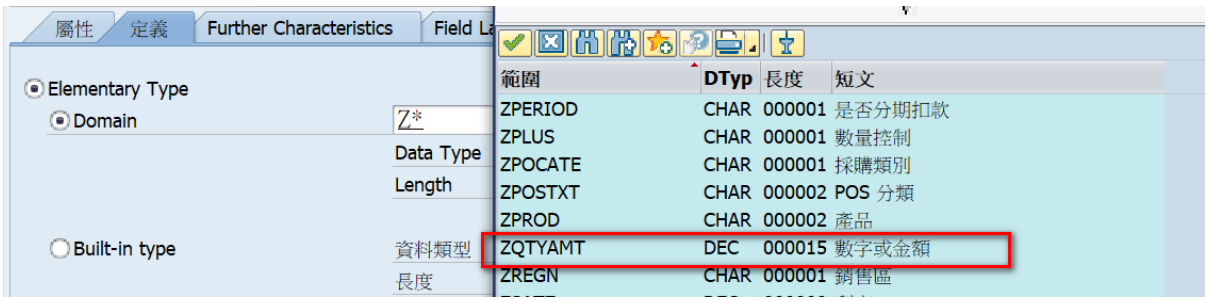
BUKRS	☑	☑	BUKRS	CHAR	4	0 公司代碼
WERKS	☑	☑	WERKS_D	CHAR	4	0 工廠
ZBUPER	☑	☑	ZBUPER	ACCP	6	0 年月(YYYYMM)

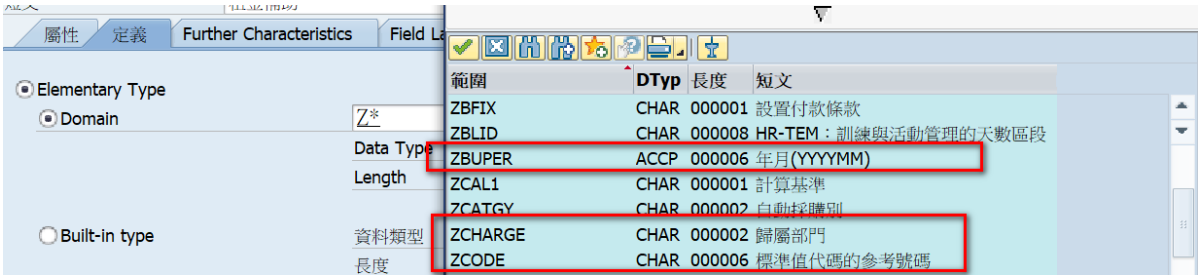
◆ ZSUBSIDY_SPEC：欄位名稱可以任意取名，不一定要 Z 開頭，但是 DE 一定要 Z 開頭，因為都是數值欄位。所以設定為 ZFD04AMT 這個 Domain。



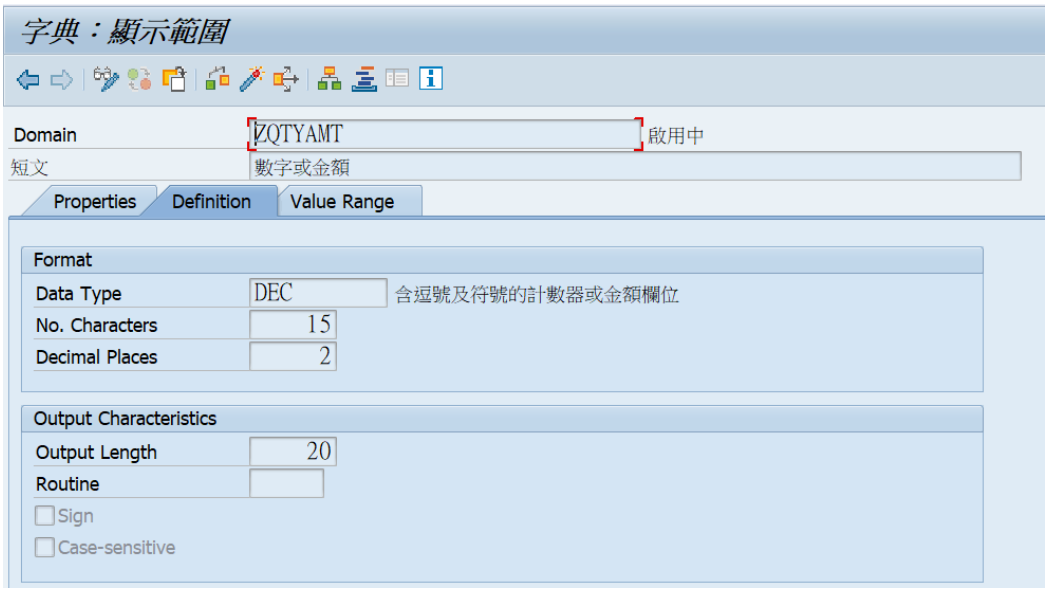
◆ 目前有設定如下的 domain，可參考後自行新增。Domain 是通用彙總的概念，目的是有效率的建立 DE。



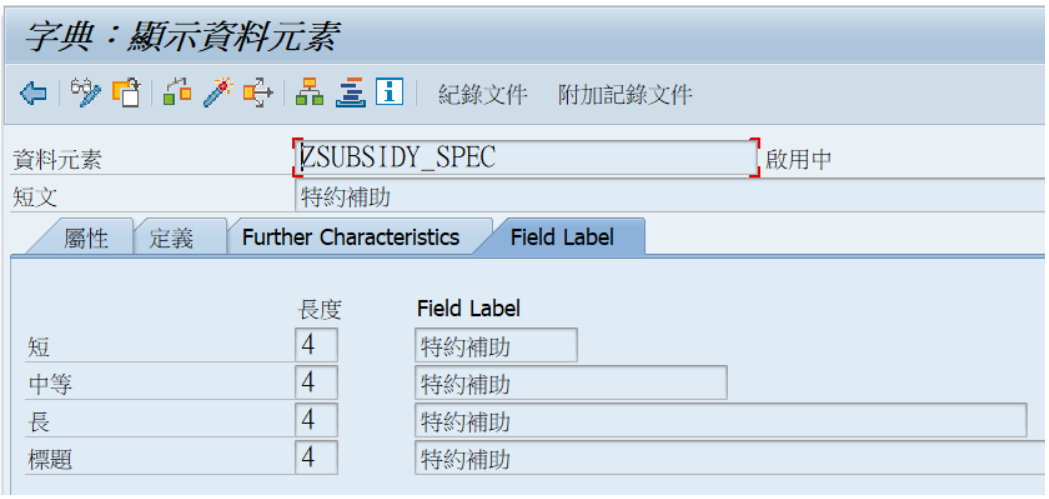




Domain 的設定畫面如下：



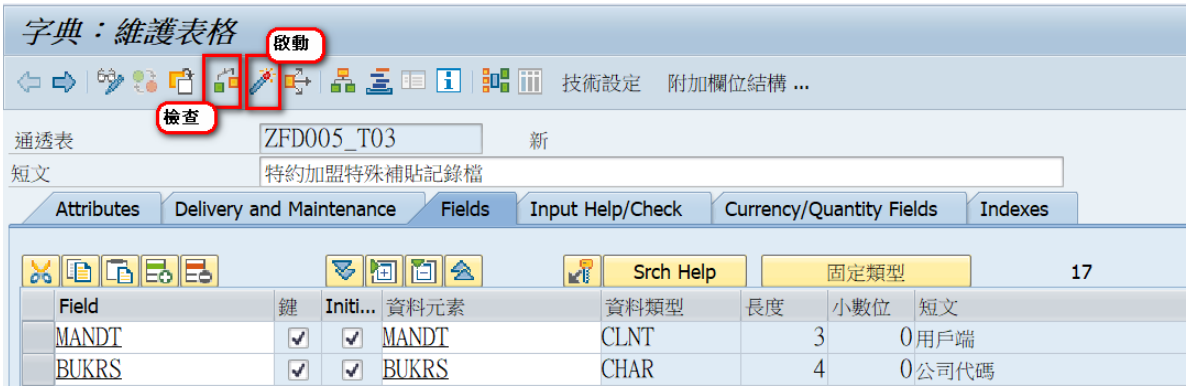
◆ 設完 Domain 後，請切換到 Field Label 頁籤，設定抬頭，設定分為 4 種情境寬度設定。



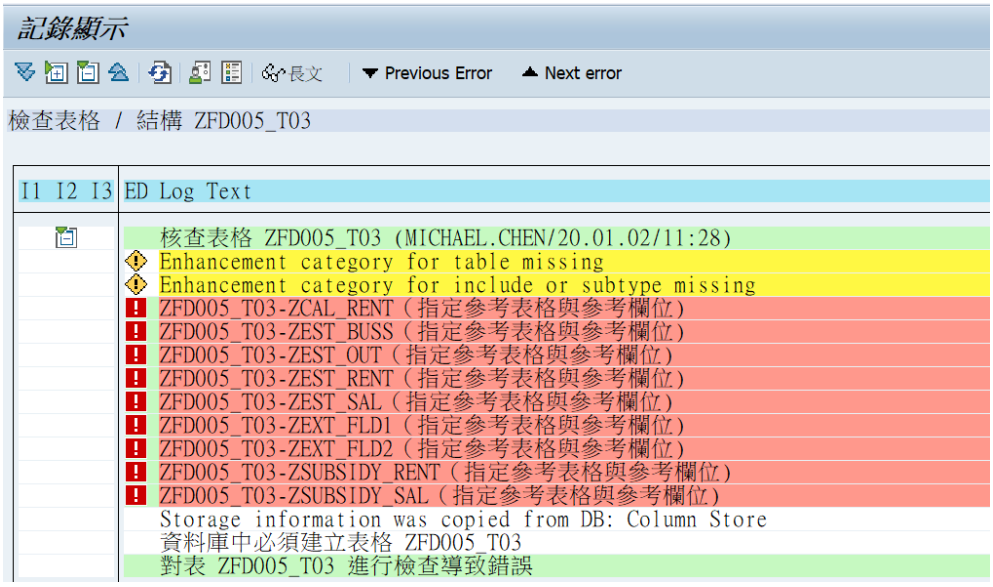
◆ 設定完每個欄位後，先按檢查，然後務必要記得要按仙女棒，設為啟動(Active)。

◆ 重複上面的過程，逐一設定每一個欄位後即可。

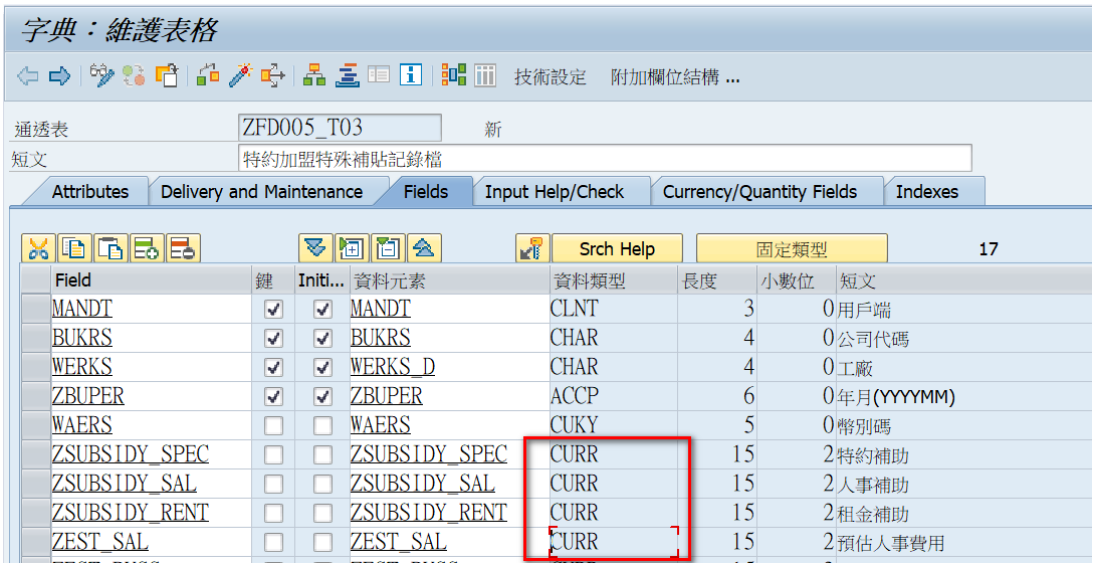
✧ 全部欄位都新增完畢後，請回到 se11 主頁面，一樣要執行檢查和啟用。



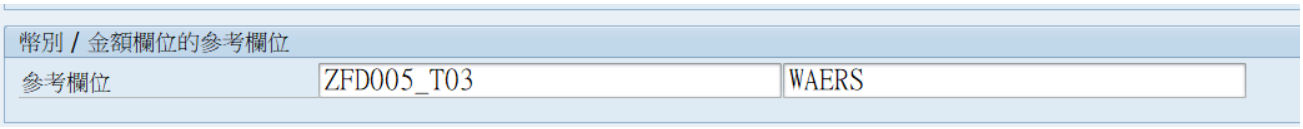
◆ 會報如下的錯誤，都是數值欄位，錯誤原因都是(指定參考表格與參考欄位)



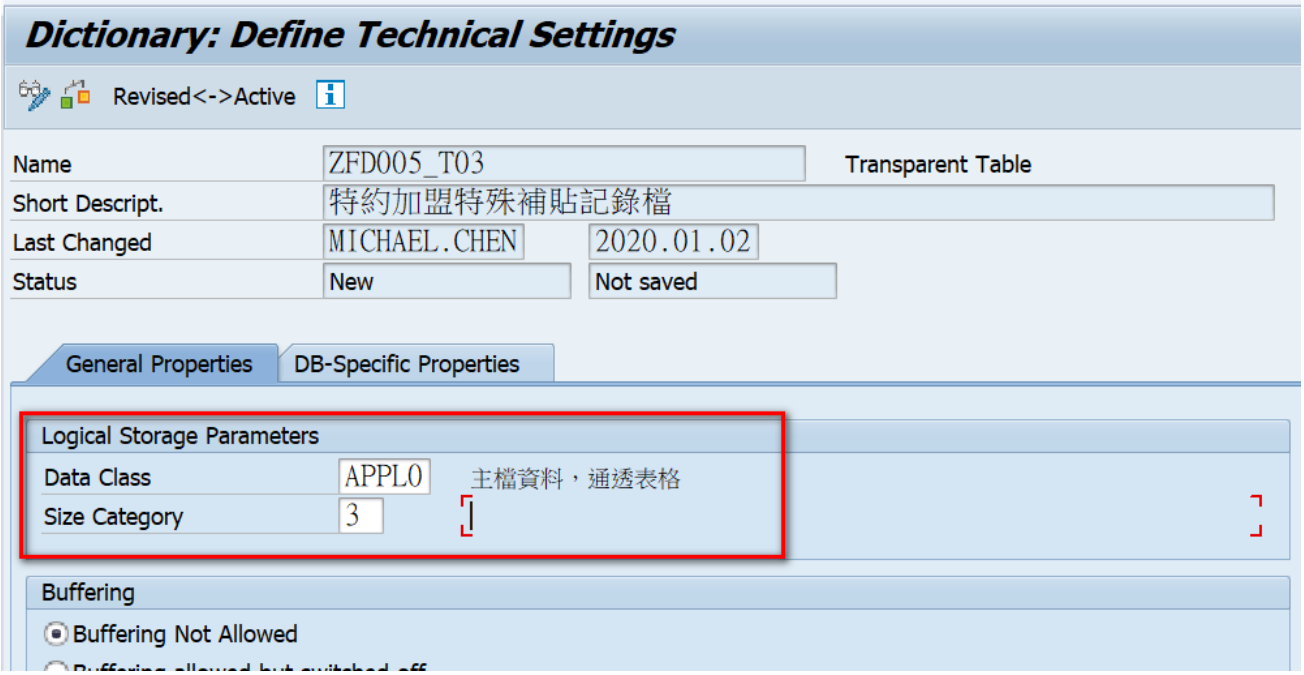
◆ 原因如錯誤說明，有一個地方沒有設定好，請在 CURR 上 DbClick



◆ 請輸入 ZFD005_T03 及 WAERS 即可(每個數值欄位逐一輸入)



◆ 檢查無誤後，再按啟用(仙女棒)，輸入下列資訊(可能有警告訊息，無妨)



◆ 除了上面這個錯誤，我還曾經碰過一個神奇的 bug，系統一直發無建立 Index 的 Error Message，最後才發現，欄位的順序是有關係的。如果改欄位是 PK，就一定要全部排在前面，我原先的錯誤是把 INPRICE 這個欄位放在 ZSDATE 前，怎麼存的報錯。

Field	鍵	Initi...	資料元素	資料類型	長度	小數位	短文
MANDT	☑	☑	MANDT	CLNT	3	0	用戶端
ZSOURCE	☑	☑	ZSOURCE	CHAR	10	0	資料來源
ZFILTER	☑	☑	ZFITER	CHAR	10	0	條件類型
LIFNR	☑	☑	LIFNR	CHAR	10	0	供應商或貸方的帳號
MATNR	☑	☑	MATNR	CHAR	40	0	物料號碼
INPRICE	☐	☐	ZINPRICE	CURR	15	2	價格
ZEDATE	☑	☑	ZEDATE	DATS	8	0	生效日期
ZCDATE	☐	☐	ZCDATE	DATS	8	0	建立日期
ZMEMO	☐	☐	ZEXT_FLD3	CHAR	60	0	其他3
WAERS	☐	☐	WAERS	CUKY	5	0	幣別碼

檢查表格 / 結構 ZCK001_T01

I1 I2 I3	ED Log Text
	核查表格 ZCK001_T01 (MICHAEL.CHEN/20.02.26/10:49)
	Enhancement category for table missing
	Enhancement category for include or subtype missing
	已定義鍵值；欄位 ZEDATE 不能在鍵值中
	Old key field ZSDATE has been deleted
	表格 ZCK001_T01 已修改（已刪除或建立新的）
	對表 ZCK001_T01 進行檢查導致錯誤

把 ZINPRICE 調整位置，就 OK。

通透表 ZCK001_T01 啟用中

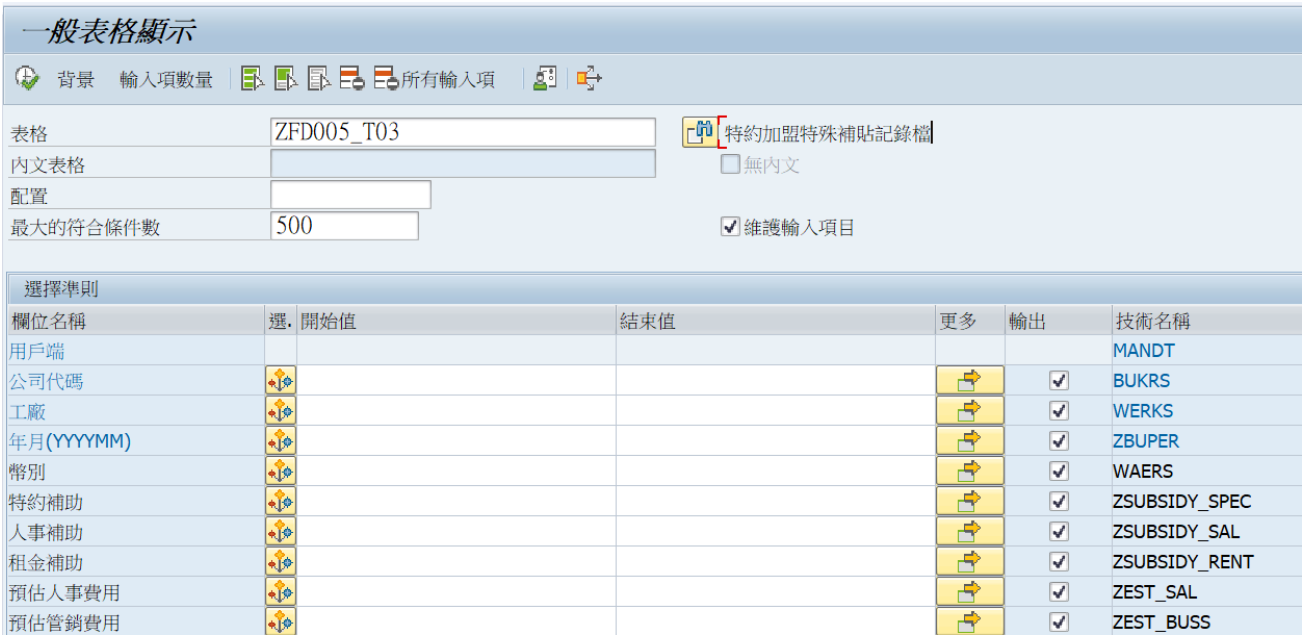
短文 促進價檢查表

Attributes | Delivery and Maintenance | Fields | Input Help/Check | Currency/Quantity Fields | Indexes

Srch Help | 固定類型 | 11

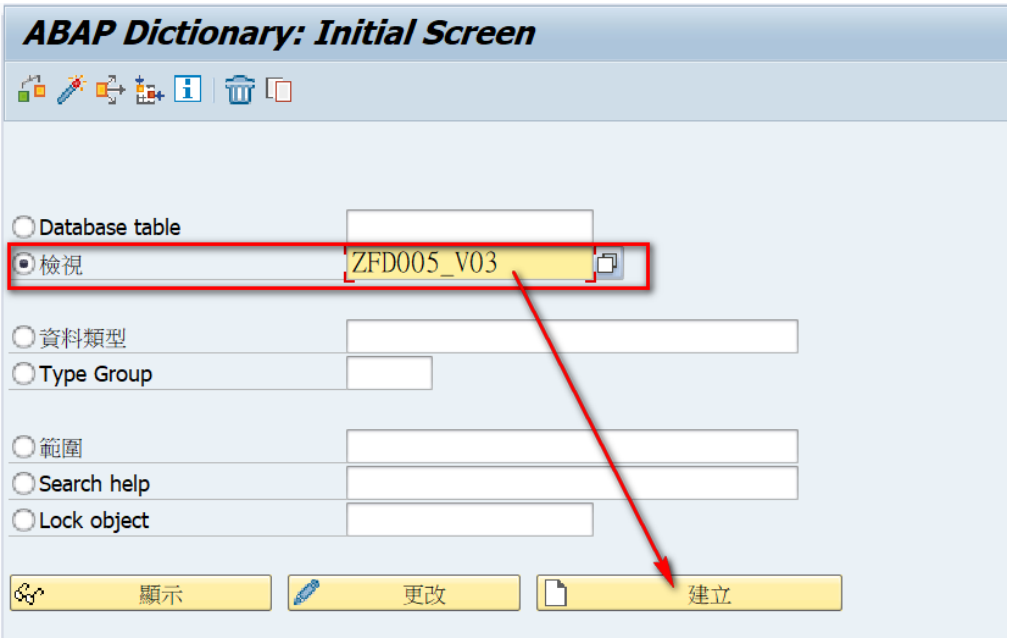
Field	鍵	Initi...	資料元素	資料類型	長度	小數位	短文
MANDT	☑	☑	MANDT	CLNT	3	0	用戶端
ZSOURCE	☑	☑	ZSOURCE	CHAR	10	0	資料來源
ZFILTER	☑	☑	ZFITER	CHAR	10	0	條件類型
LIFNR	☑	☑	LIFNR	CHAR	10	0	供應商或貸方的帳號
MATNR	☑	☑	MATNR	CHAR	40	0	物料號碼
ZSDATE	☑	☑	ZSDATE	DATS	8	0	促銷起日
ZEDATE	☑	☑	ZEDATE	DATS	8	0	生效日期
INPRICE	☐	☐	ZINPRICE	CURR	15	2	價格
ZCDATE	☐	☐	ZCDATE	DATS	8	0	建立日期
ZMEMO	☐	☐	ZEXT_FLD3	CHAR	60	0	其他3
WAERS	☐	☐	WAERS	CUKY	5	0	幣別碼

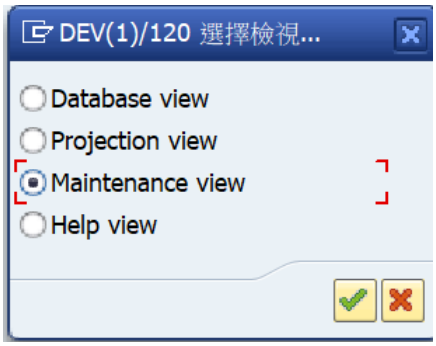
✧ 使用 se16n 查看是否建立成功。



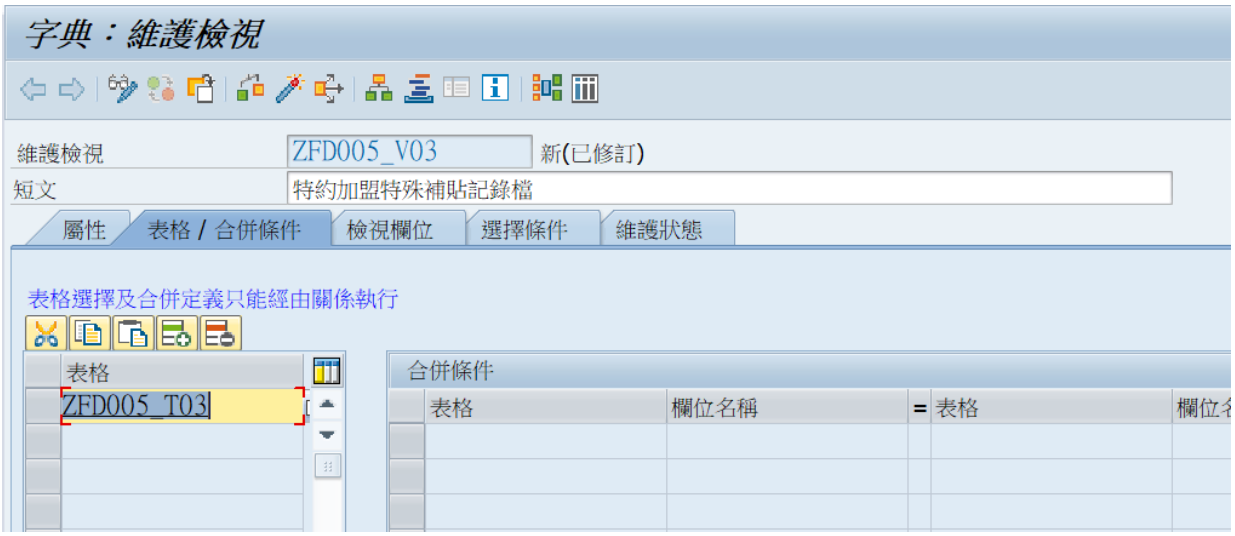
➢ 建立好新增的 ZTable 後，接下來請依下列步驟設定

✓ 使用 SE11 來定義 [維護視圖] 的表結構

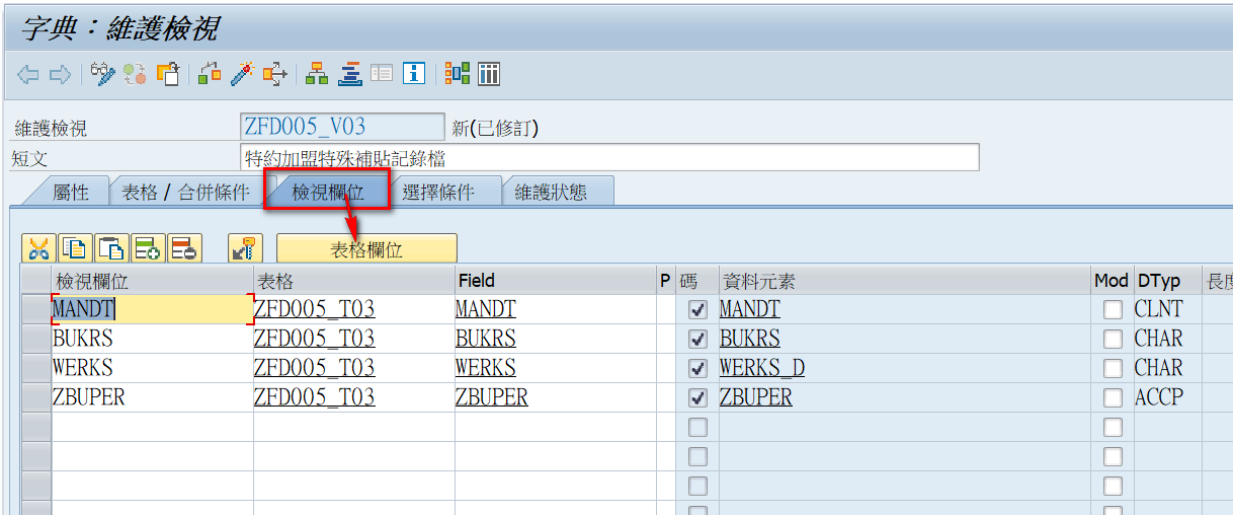


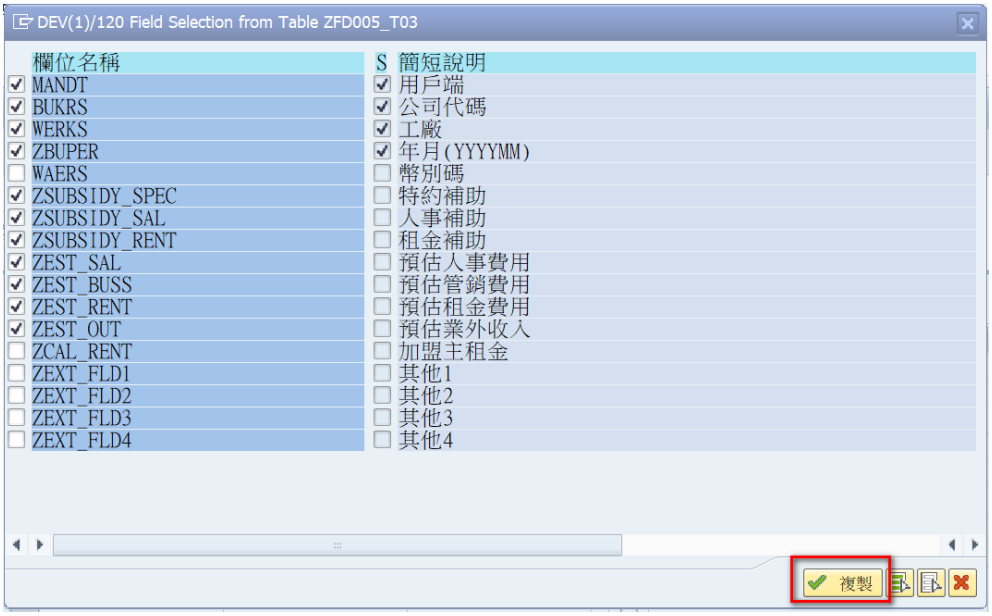


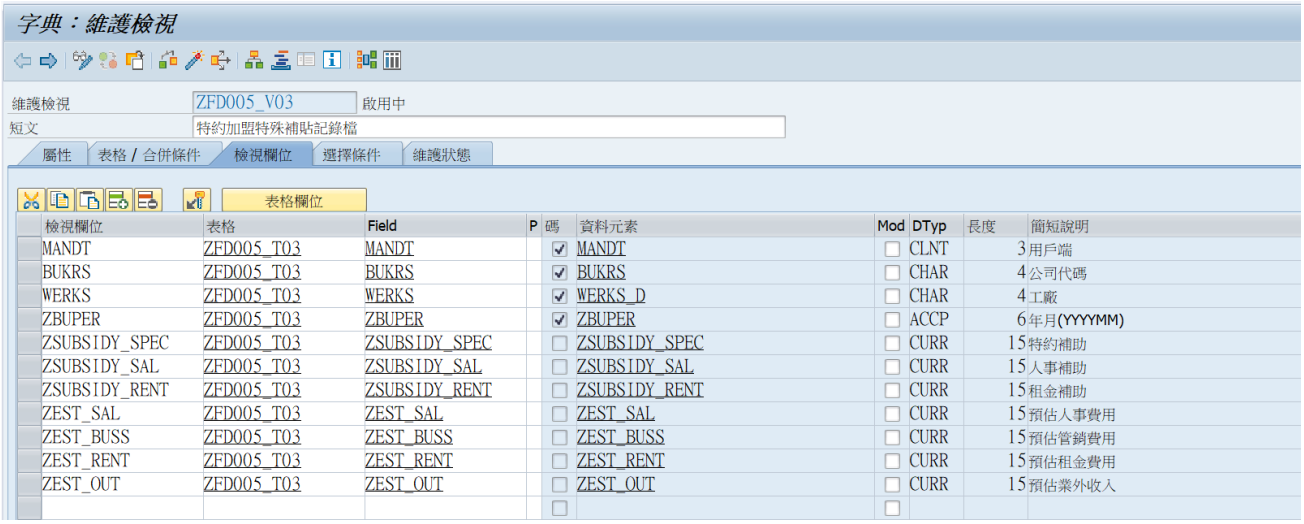
✧ 請輸入短文及表格名稱(如下)

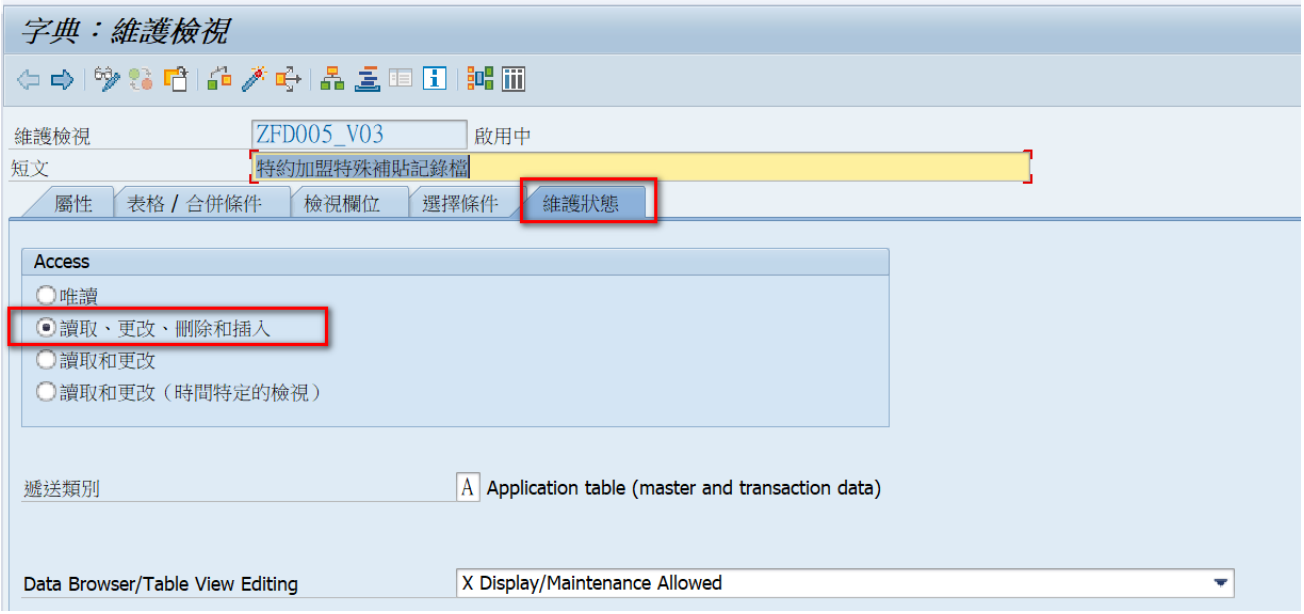


✧ 切換到[檢視欄位]頁籤，但是只看到 4 個欄位，請按一下[表格欄位]，選擇要維護的欄位，後面 4 個預留欄位請先不要選擇，其餘都勾選。

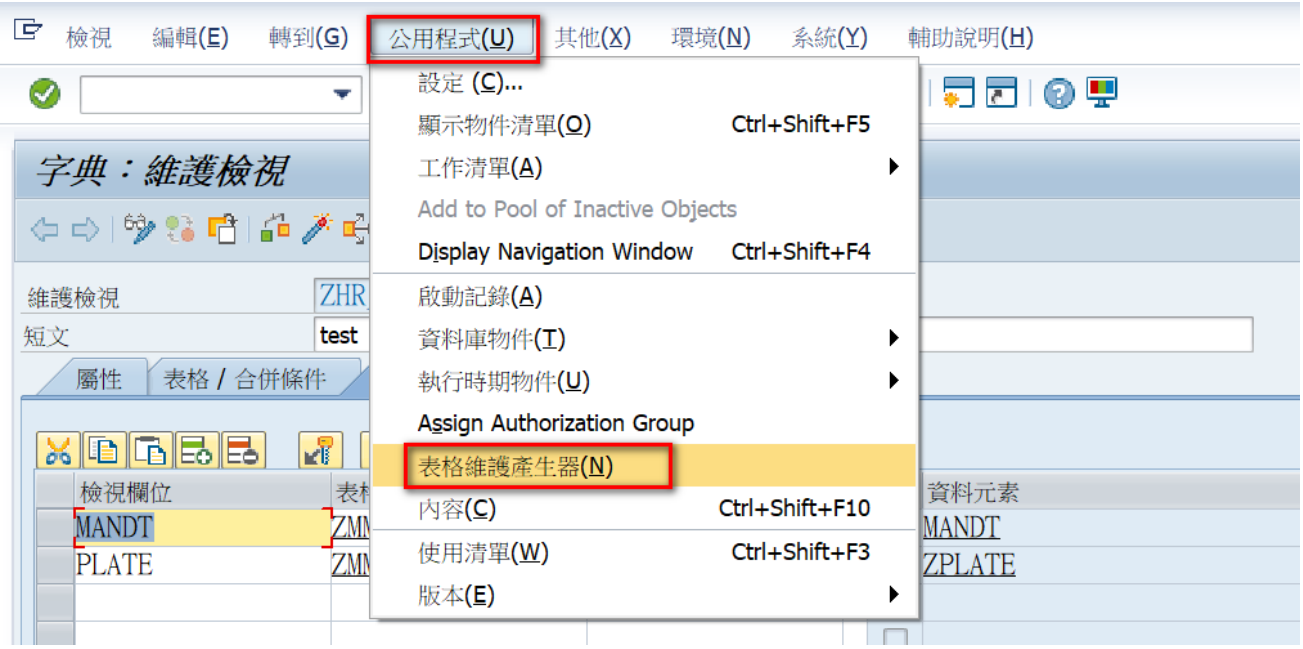






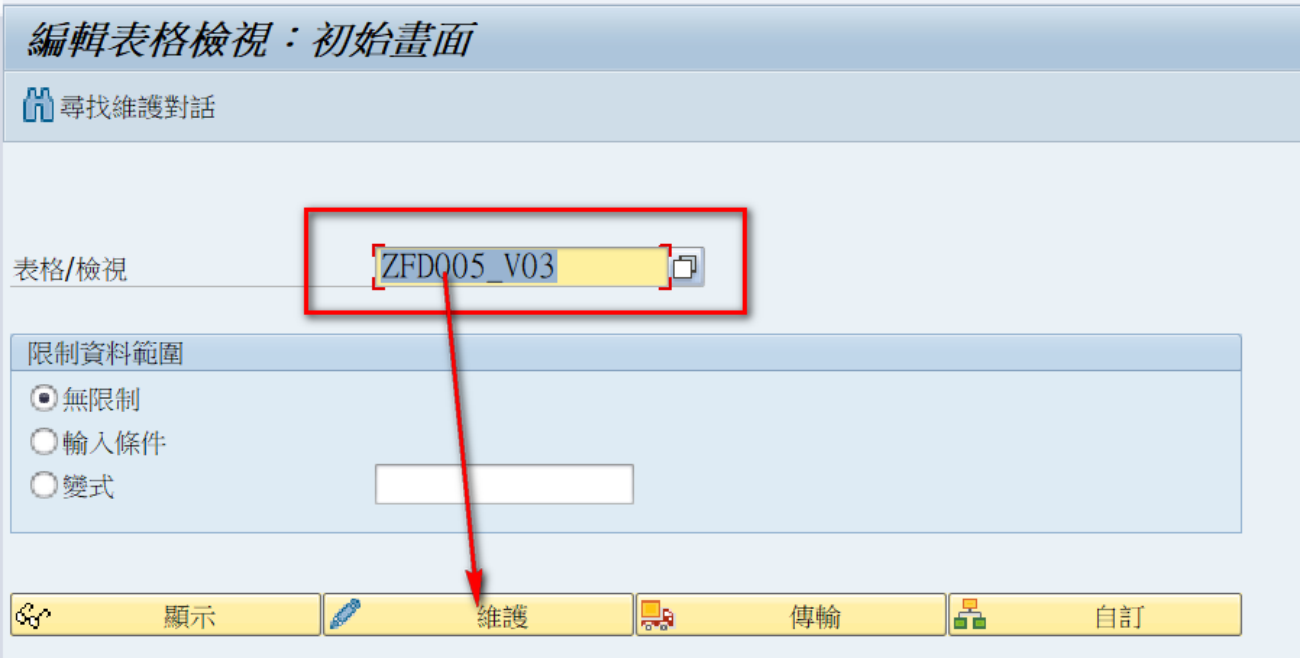


➢ 使用公用程式(Utilitys)→表格維護產生器

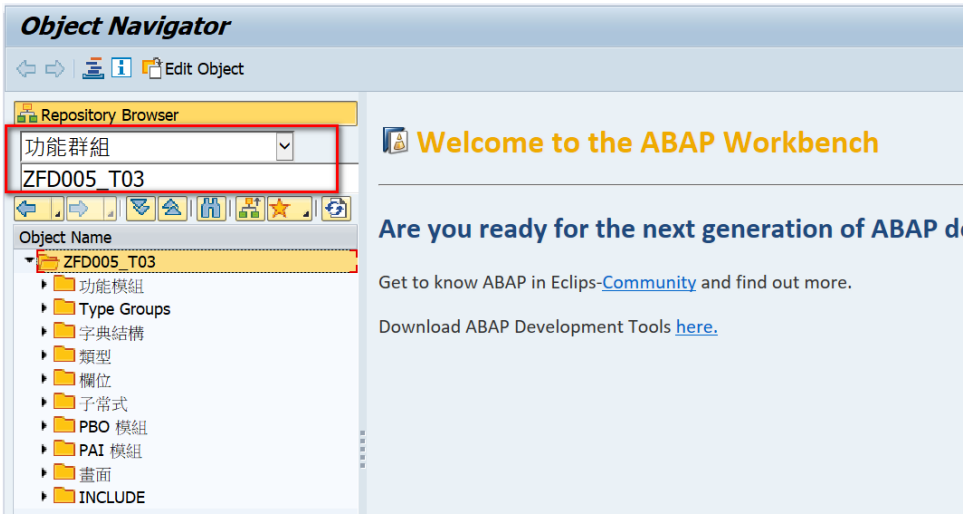


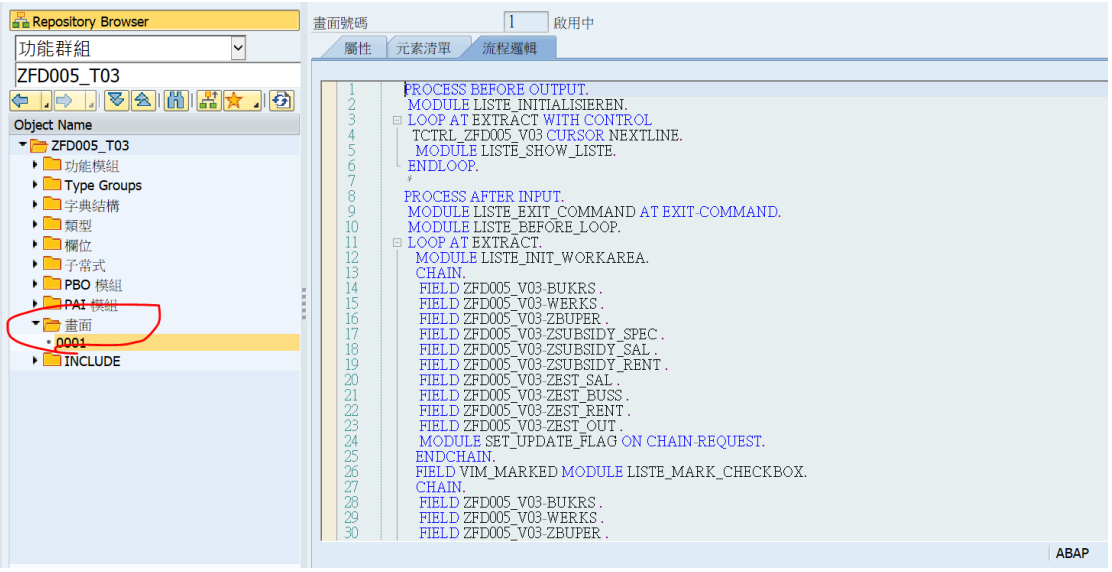
➢ 輸入下面內容後，按左上角的[建立] button，存檔時可能有一些警告或錯誤訊息，請依提示逐項排除即可。

➢ 接下來請進入 SM30 並輸入 ZFD005_V03，按[維護]，再按[新輸入項]，即可開始編輯

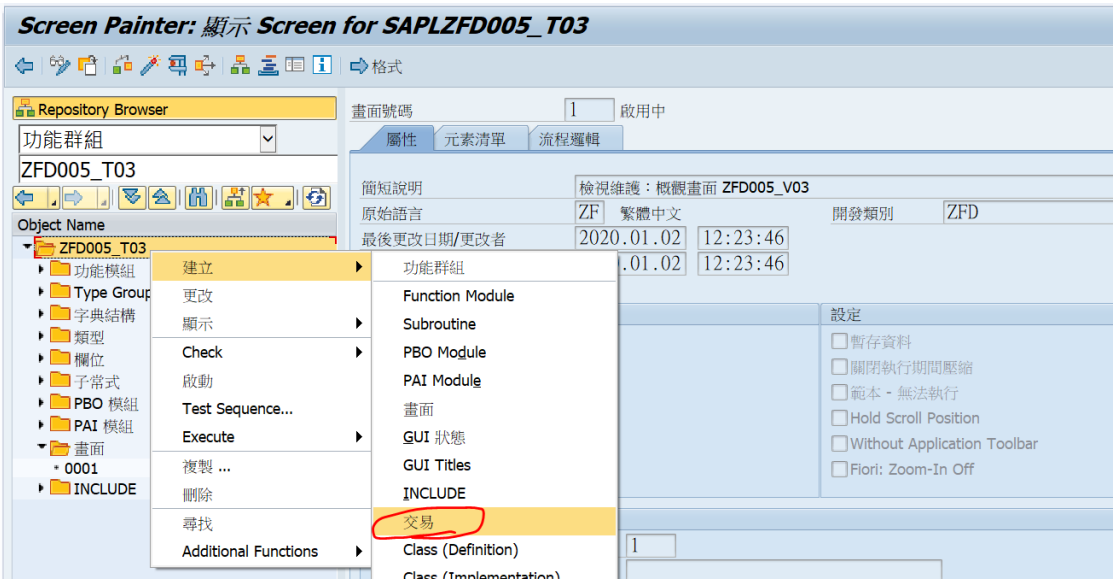


➢ 設定完後，執行 SM80，查看剛產生的程式。理論上可以在此修改程式內容，但目前先都使用預設，等日後進階時再試試如何擴充功能。



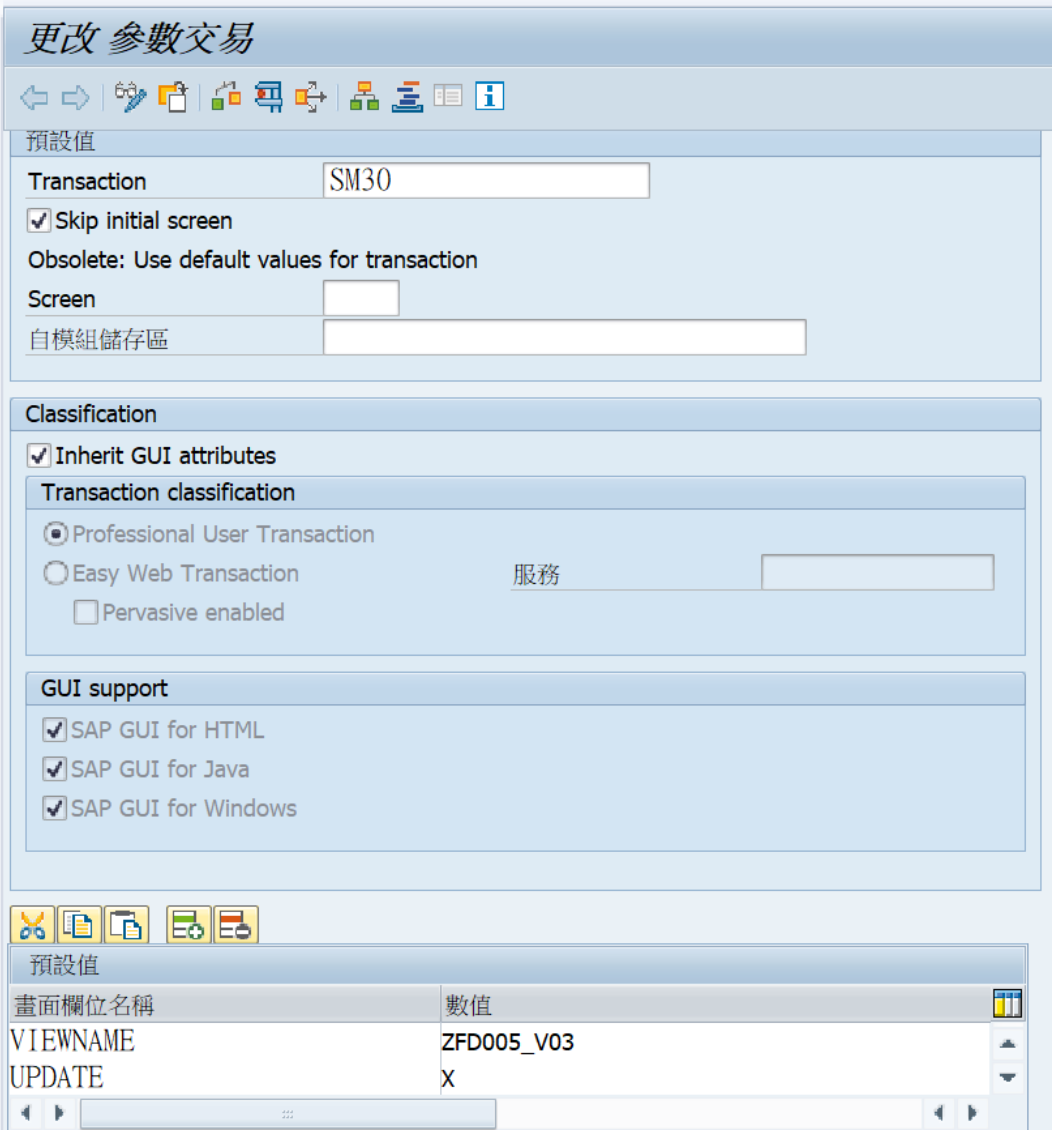


➢ 如果要自定義 T-Code，一樣也可以在此設定(也可以用 SM93)。



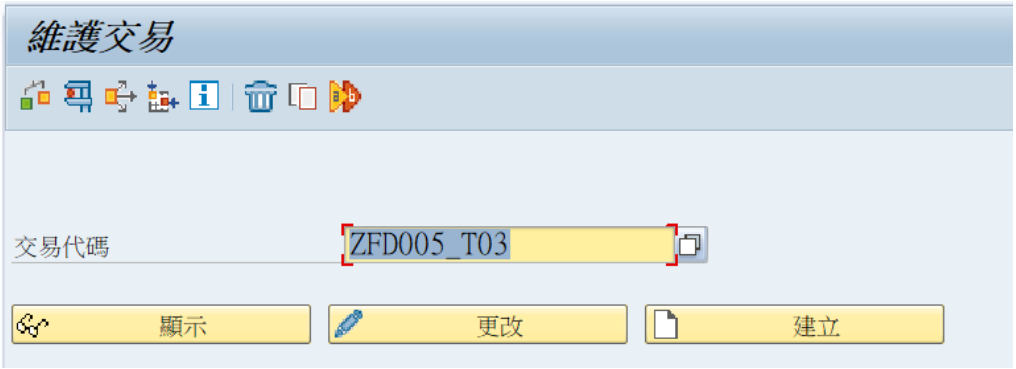
➢ 差別在要選擇 Transaction with parameter 這個選項

➢ 輸入下列內容，重點是下方的預設值內容



➢ 然後就可以輸入 T-Code 執行。

➢ 用 SE93 建立 T-Code，和從 SE80 建立 T-Code 是一樣的。



第 13 章 SAP 的自定義報表工具

在 ERP 導入的過程中，最難以解決的是使用者天馬行空、層出不窮的報表需求，無論系統預設提供多少報表，永遠都不可能滿足所有客戶的需求，所以大部分的 ERP 系統都會提供可以讓 Power User 能自定義報表的工具，當然，這些報表工具總有功能極限，也還是必須了解相關的 Table 結構，所以它可以降低 ABAP 的開發量，但一些報表還是需要客製程式來完成。

此類工具通常用來解決關聯幾個表、設定每個表取得的欄位、再建立關聯的條件，如果取得的欄位要經過一些計算過程，如 sum、count、average 等，雖然有辦法寫程式解決，但是工具本身設定的使用對象是 Power User，因為都要寫程式，我偏向直接寫 ABAP 程式即可，不要用本文介紹的工具解決。

如果是臨時性需要下 SQL 指令取得相關的資料，基本上可以透過 Eclipse 的 SQL Console 取得，不需要透過報表處理，這個部份請參考第 11 章的相關說明即可。

在 SAP 中較常用的報表工具可以分為 2 大類

1. 入門款：如 SQVI 及 Query 報表(SQ01➔SQ02)
   本文會針對這 2 個工具說明如後，其中 Query 有顧問上課講義，說明非常清楚，建議先看過一遍。
   \\10.0.8.14\sap 導入專案\專案公用資料夾\Phase 3 具體實施\3.X 具體實施階段影音檔\CO\
   20190214-Query 報表.wmv
2. 進階款有 GRRx 及 KE30 的進階報表繪製程式，可以參考下面顧問上課內容，本文不做說明。
   \\10.0.8.14\sap 導入專案\專案公用資料夾\Phase 3 具體實施\3.X 具體實施階段影音檔\CO\
   20190213-GRR2 報表繪製器.wmv
   20190213-KE30 報表繪製器.wmv

➢ SQVI 快速報表(Quick Viewer)

https://www.fenginfo.com/2523.html

- 說明

對資料表進行查詢通常使用 SE16，但只限於單張表，對於多表聯動的查詢，則需要通過創建 Query 的方式，方法有多種，而 SQVI 是一種簡潔快速的工具。

SQVI 全稱是 Quick Viewer，可以快速生成多表聯動的查詢。但使用有很多限制，主要有以下方面：創建的物件只歸屬於創建者，其它帳號無法發現及使用；只能在當前環境使用，無法進行傳輸。雖有以上缺點，但也有補救手段，SQVI 創建的 Query 可以生成程式，而用 SE93 可綁定此程式創建獨立事物碼。

SQVI 操作分為編輯和執行兩大部分，在編輯狀態可直接執行報表。
本文範例是採購文件的匯入 Log 表頭及明細檔的 Join，原 SQL 有 sum 及 count，實作時會省略，只會 Join 欄位顯示而已。

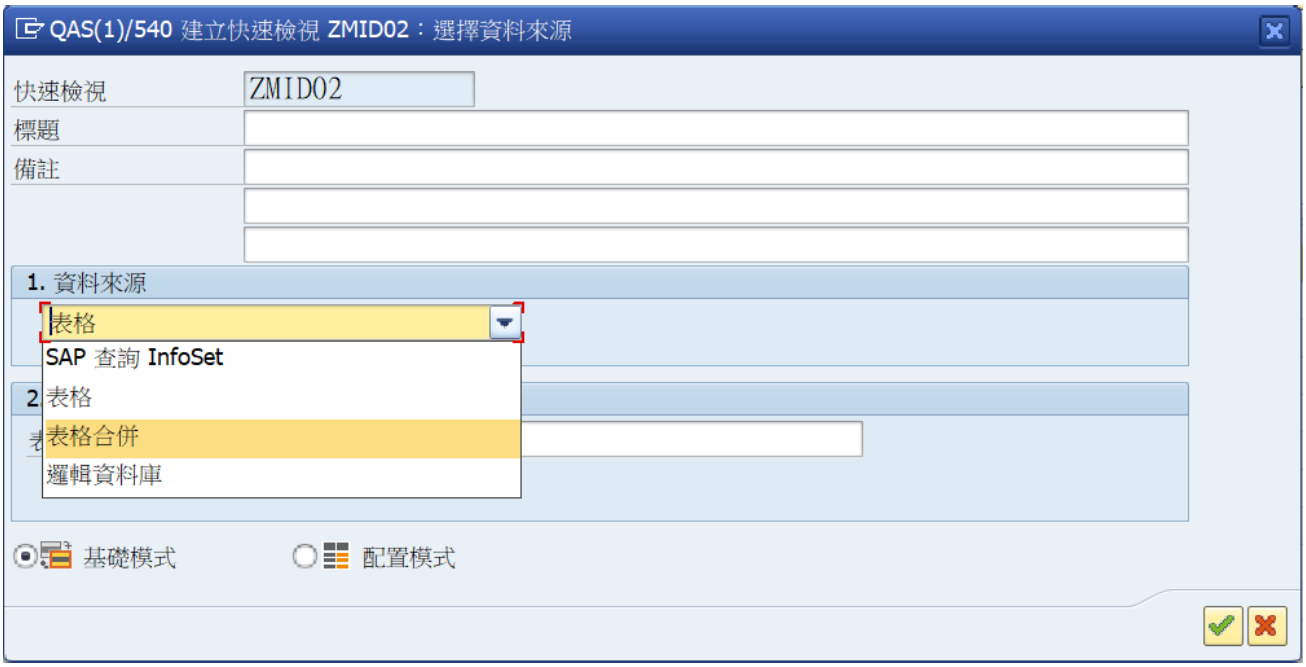
參考 SQL 如下

```
Select A~BEDAT AS BLDAT,A~BSART,B~WERKS,sum( B~MENGE ) AS QTY,count( * ) AS RECNO
        From ZMM013_T01 AS A Left Join ZMM013_T02 AS B ON A~IHREZ = B~IHREZ
        group by A~BEDAT,A~BSART,B~WERKS
        HAVING A~BEDAT in @d_BLDAT      "查詢條件
        ORDER BY A~BEDAT,A~BSART,B~WERKS.
```

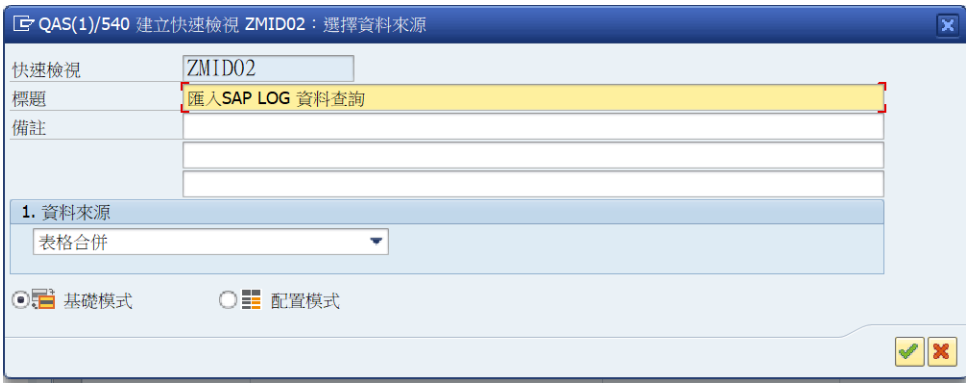
■ 編輯

輸入 T-Code=SQVI 並執行，進入後輸入需要創建的查詢物件名稱，如下圖所示，然後點擊 [建立] 按鈕。

系統彈出創建查詢物件的對話方塊，這裡有一個重要選項，也就是 [資料來源]，如下圖所示，其中 [表格] 是單張表，如同 SE16 查看一樣；而[表格合併] 則是允許兩張以上表進行聯集查詢，在此選擇 [表格合併]。

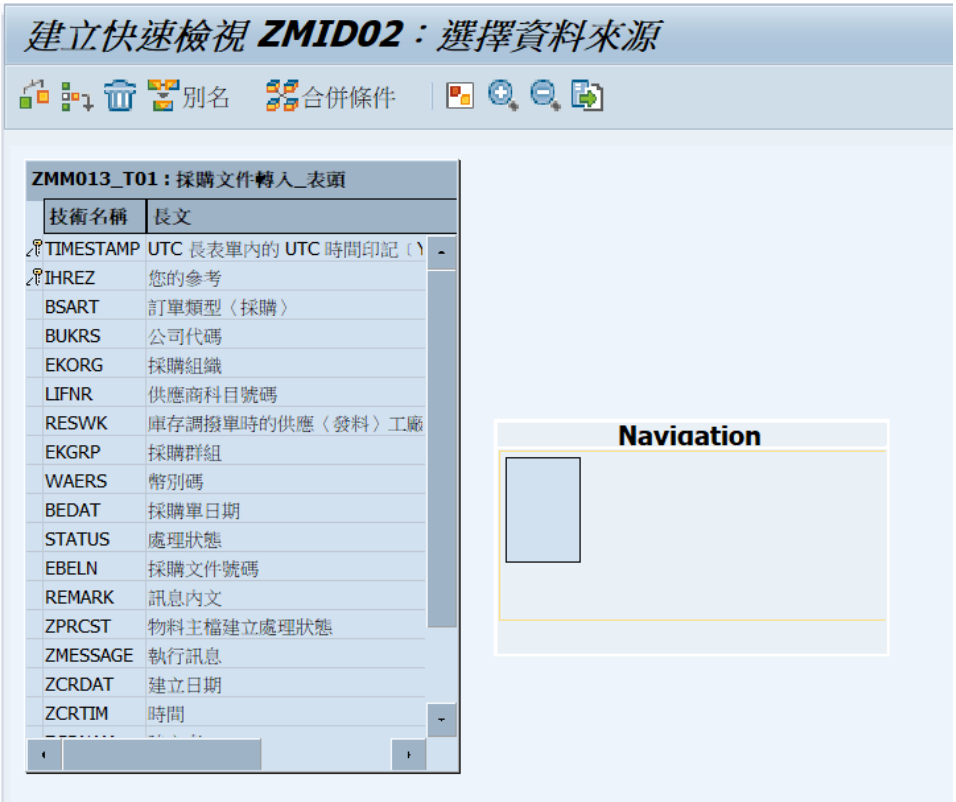


填上流覽物件的標題，以及編輯模式，如下圖所示，在此選擇“基本模式”，點擊確認鍵（ ）繼續。



進入資料來源選擇介面，如下圖所示，在此可以加入一個或多個資料表，並設定表之間的連續條件，資料表可以設定別名—當某個資料表需要同時出現兩次以上時，請先設定別名加以區別。圖中的 Navigation 灰色矩形是介面維護的導航視窗。工具列上可點擊放大鍵（ ）、縮小鍵（ ）用於視窗顯示效果的調整。

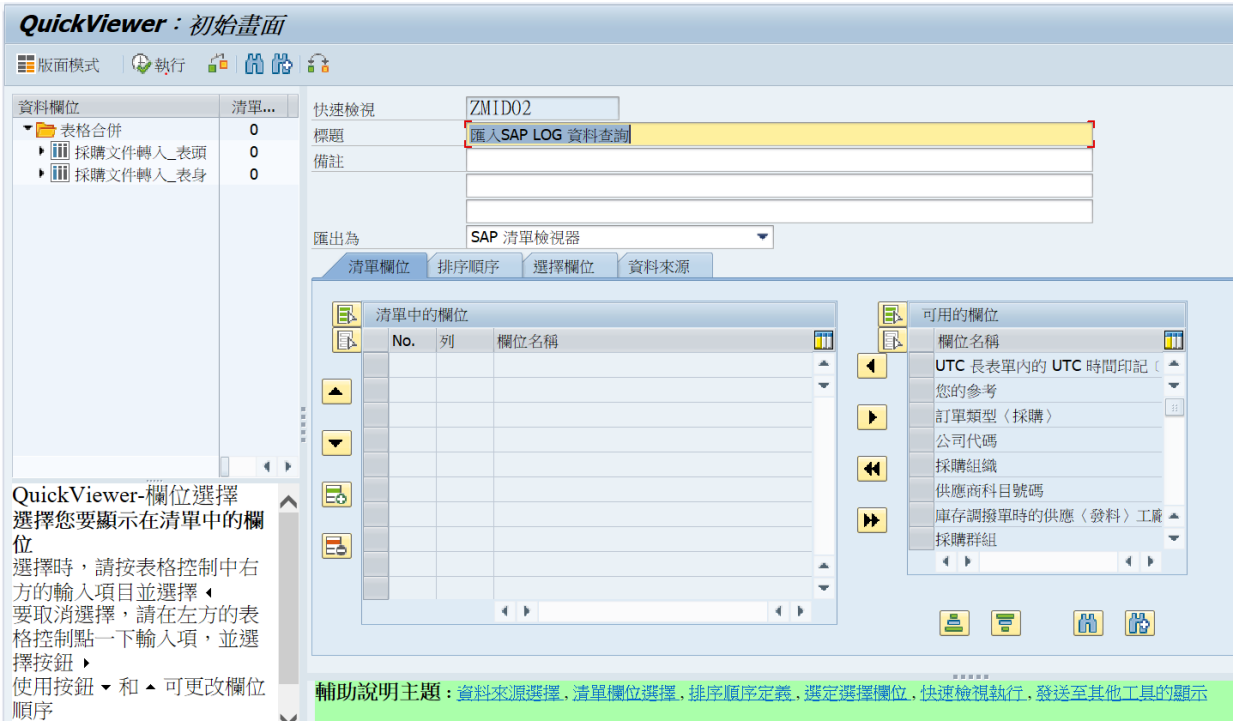
首先添加銷售訂單抬頭表 ZMM013_T01，點擊插入表按鈕（ ），彈出對話方塊，輸入表名，如下圖所示，點擊確認鍵（ ），表加入後如下圖所示，資料表是一個矩形，列出含有的欄位清單。



按上述步驟再次添加表 ZMM013_T02，如下圖所示。因為表頭表 ZMM013_T01、明細表 ZMM013_T02 的關鍵欄位的關係明確，系統自動添加連接條件，但對於沒有直接關係或系統添加錯誤的連接條件，則需要手工刪除或手工添加。

兩張資料表添加並確認連接關係後，點擊返回鍵，如下圖所示，左邊上部顯示資料表及欄位，右側下部有四個標

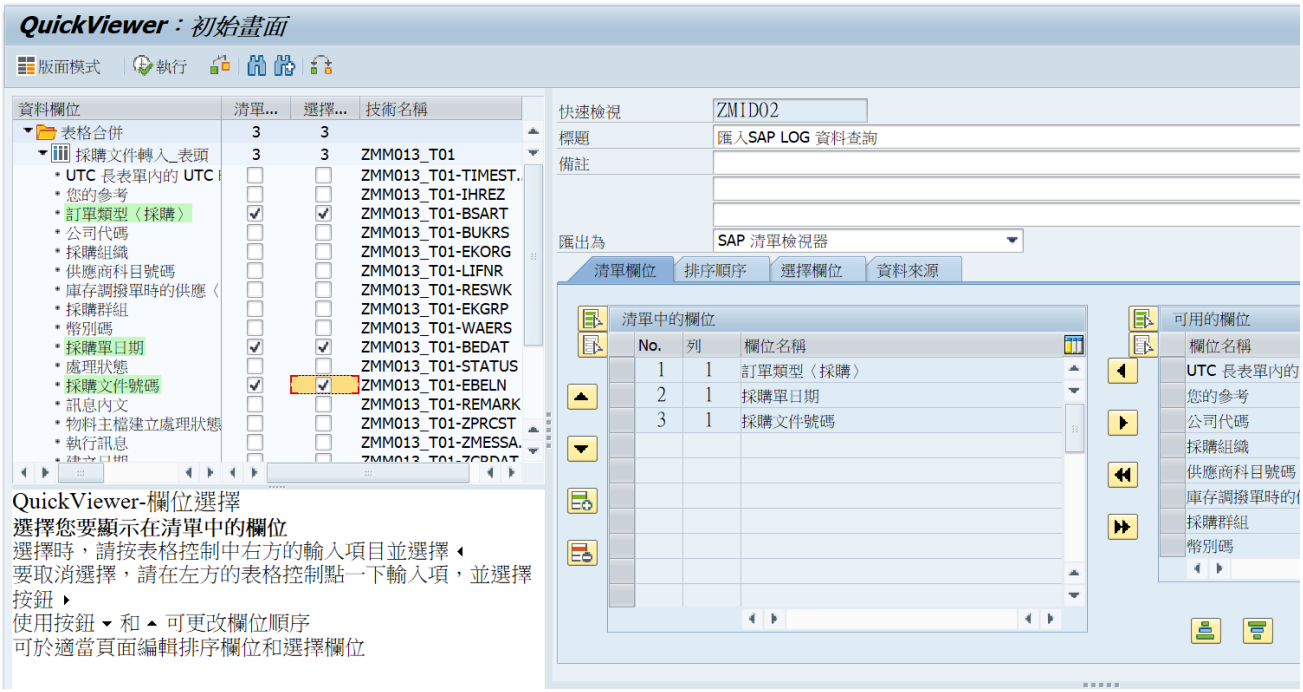
籤頁，分別設置顯示清單欄位、排序順序、選擇欄位，以及一個顯示資料來源。



首先轉至 [資料來源] 標籤頁，如下圖所示，在此點擊 [更改合併] 按鈕可返回前述基本模式編輯介面。

轉至 [清單欄位] 標籤頁，如下圖所示，在此設置顯示輸出的欄位，此下有兩列清單，左側是搜索欄位清單，右側是可用欄位清單，可用欄位中包含 ZMM013_T01、ZMM013_T02 兩張表的欄位。而在 [選擇欄位] 標籤頁下，是設置查詢準則欄位，介面與 [清單欄位] 標籤頁相同。

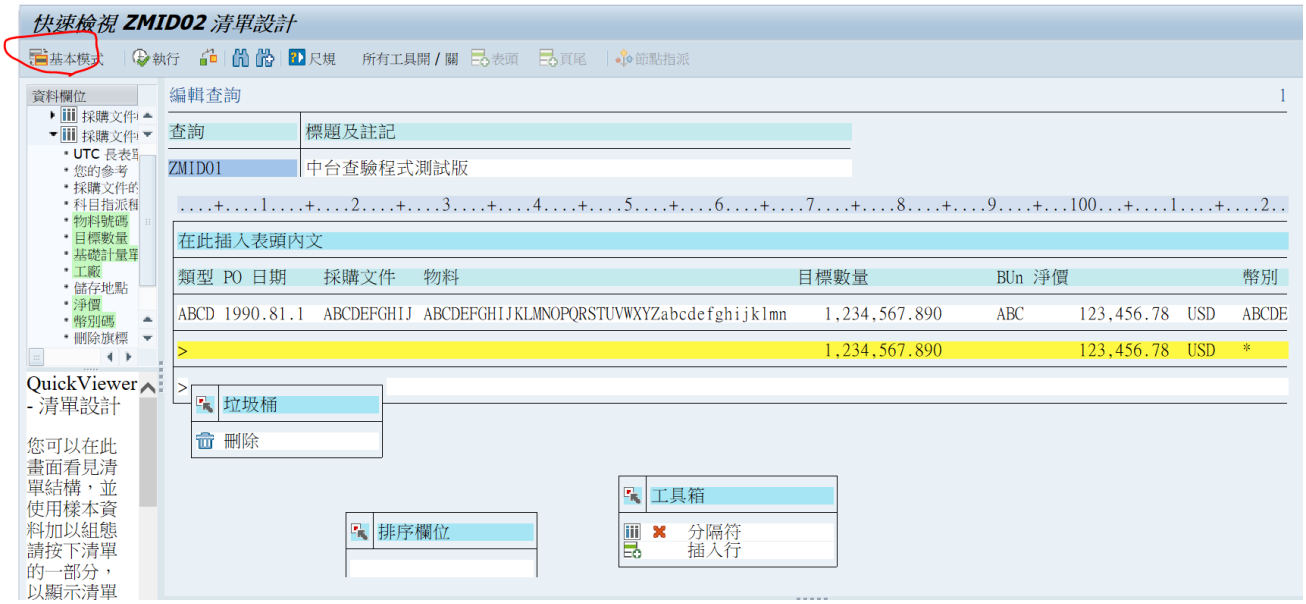
可在不同標籤頁中設置欄位，並可進行欄位添加、刪除及上下移動。也可以在左側樹形表中選擇，樹形表中的每個欄位有兩個選擇標誌欄位元，如下圖所示，[欄位清單] 欄打上選擇標誌後，則在 [清單欄位] 標籤頁下增加一個欄位，而 [選擇欄位]欄打上標誌，則在 [選擇欄位] 標籤頁中增加一個欄位。



逐一選擇各表要顯示的欄位，如下。

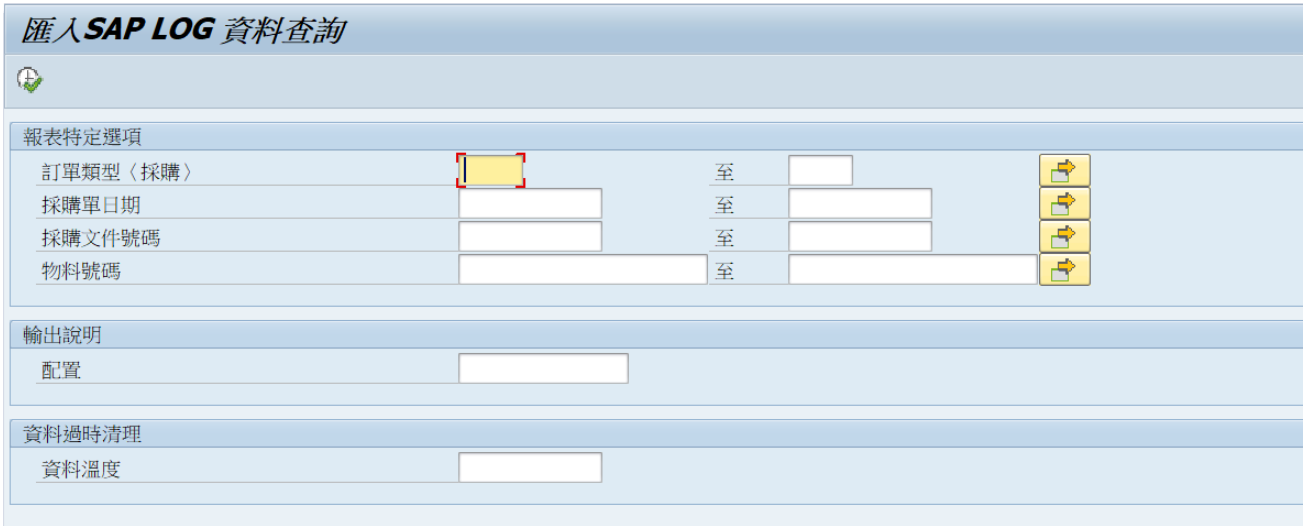
欄位設置完畢，還可以在 [排序順序] 標籤頁下設置用於排序的欄位。注意目前的匯出模式是 [SAP 清單檢視器]，在此還可以選擇 Word、Excel、本地檔等輸出方式。

以上設計用於 SAP 列表輸出、Excel 檔輸出是可以的，但輸出還可以採用 Word 格式，SAP 在此可以更加靈活的設計表格清單。點擊工具列的“格式模式”按鈕（ 格式模式 ），顯示如下圖所示，在此可增加表頭、頁腳，示例在表頭中編輯 [玖商集團 SAP LOG 查詢]，各欄位可以拖動，或選中後設置具體的位置以及顏色。



■ 執行

查詢設計好後就可以執行，可在編輯狀態直接運行，也可以在初始介面運行，執行畫面如下圖所示，輸入查詢物件名稱，點擊 [執行] 按鈕。



查詢結果

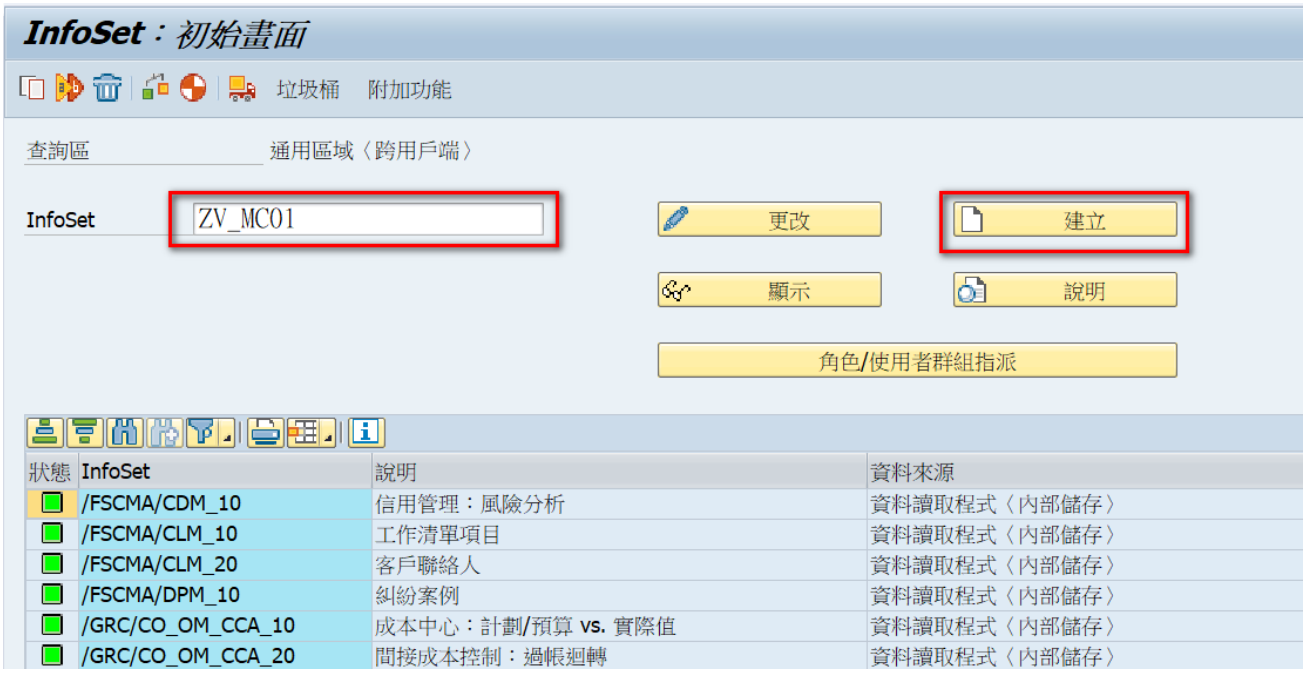
匯入SAP LOG 資料查詢

匯入SAP LOG 資料查詢

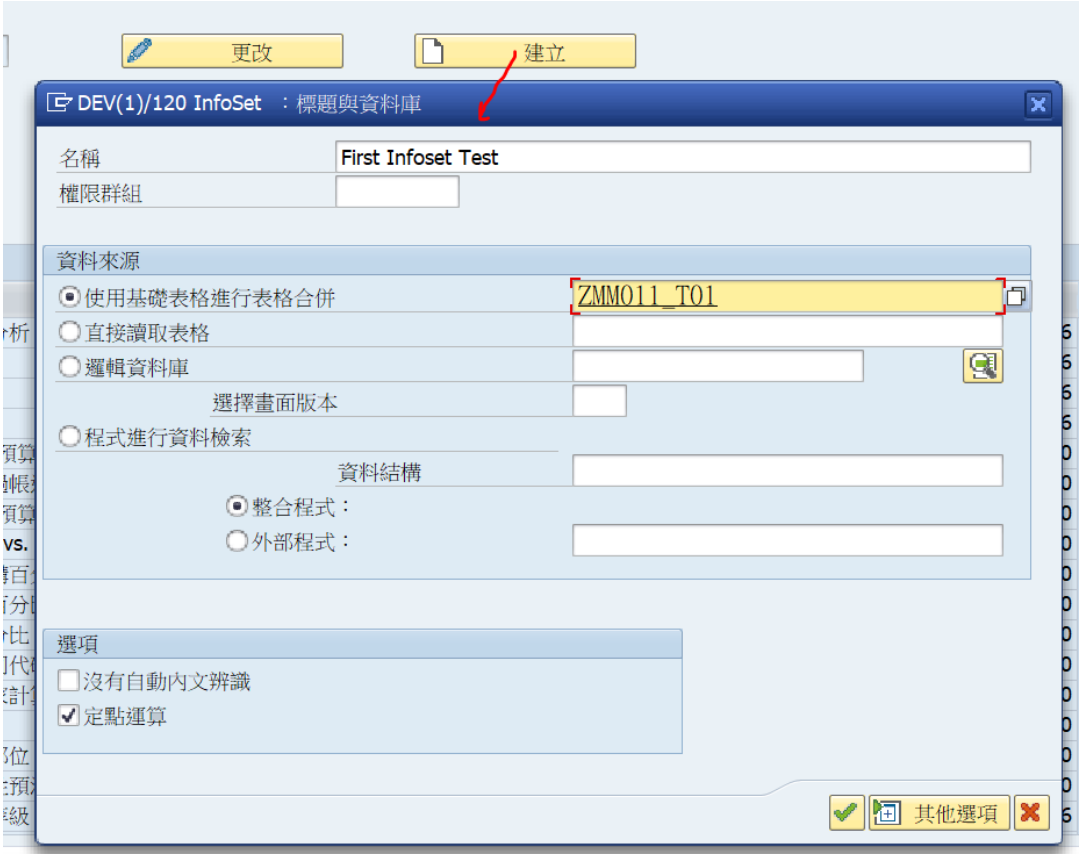
類型	PO 日期	採購文件	物料	Σ目標數量	BUn	BUn	Σ淨價	幣別	幣別	工廠
ZS03	2019.05.18		66858	2.000	瓶	瓶				6731
ZS03	2019.05.18	6300039727	65781	3.000	瓶	瓶				6732
ZS03	2019.05.18		65781	1.000	瓶	瓶				6733
ZS03	2019.05.18	6300039745	66858	2.000	瓶	瓶				6736
ZS03	2019.05.18	6300039746	65781	2.000	瓶	瓶				6738
ZS03	2019.05.18	6300039746	66858	2.000	瓶	瓶				6738
ZS03	2019.05.18		65781	1.000	瓶	瓶				6739
ZS03	2019.05.18		65781	1.000	瓶	瓶				6741
ZS03	2019.05.18		65781	1.000	瓶	瓶				6742
ZS03	2019.05.18	6300039701	66858	1.000	瓶	瓶				6745
ZS03	2019.05.18		65781	1.000	瓶	瓶				6746
ZS03	2019.05.18		66858	4.000	瓶	瓶				6746
ZS03	2019.05.18		66858	1.000	瓶	瓶				6750
ZS03	2019.05.18		65781	1.000	瓶	瓶				6752

➢ Query 的簡易說明

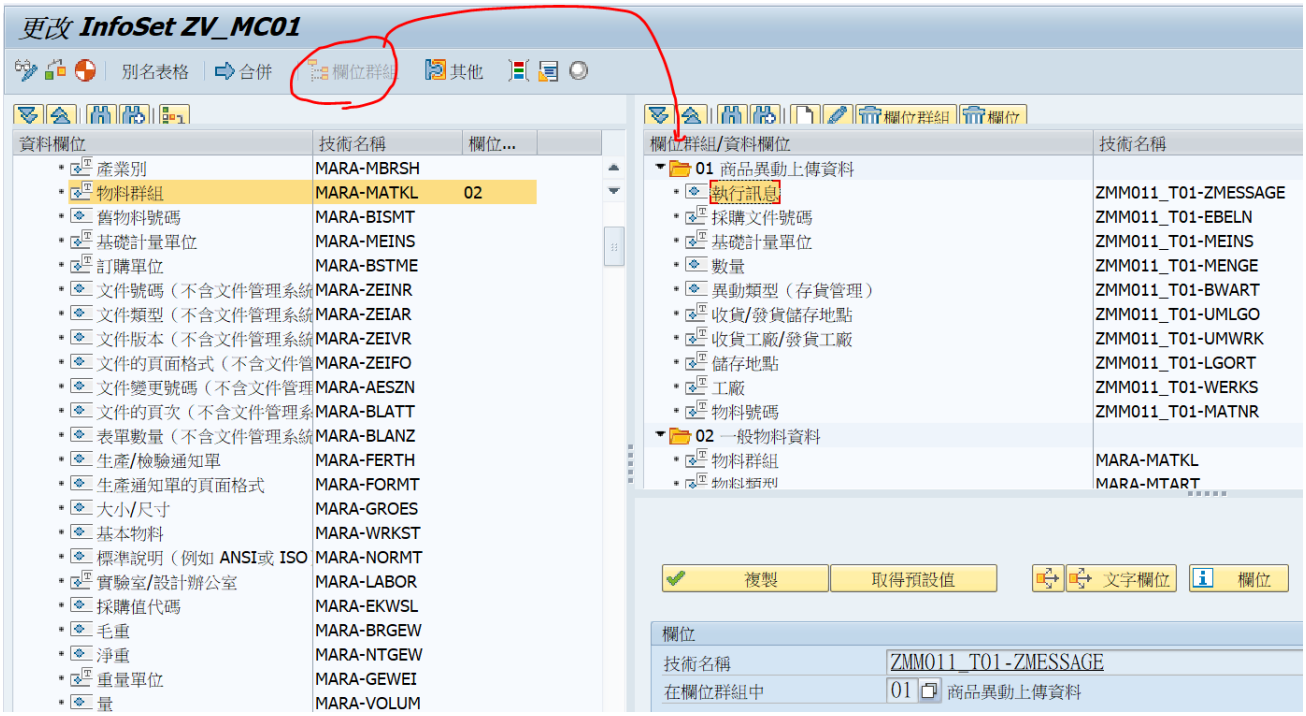
- 創建 InfoSet：SQ02



可以選擇多表 Join 或單一 Table，然後進入畫面可以新增 Table、建立關聯、選擇查詢條件欄位及顯示欄位，依畫面的指示操作即可。

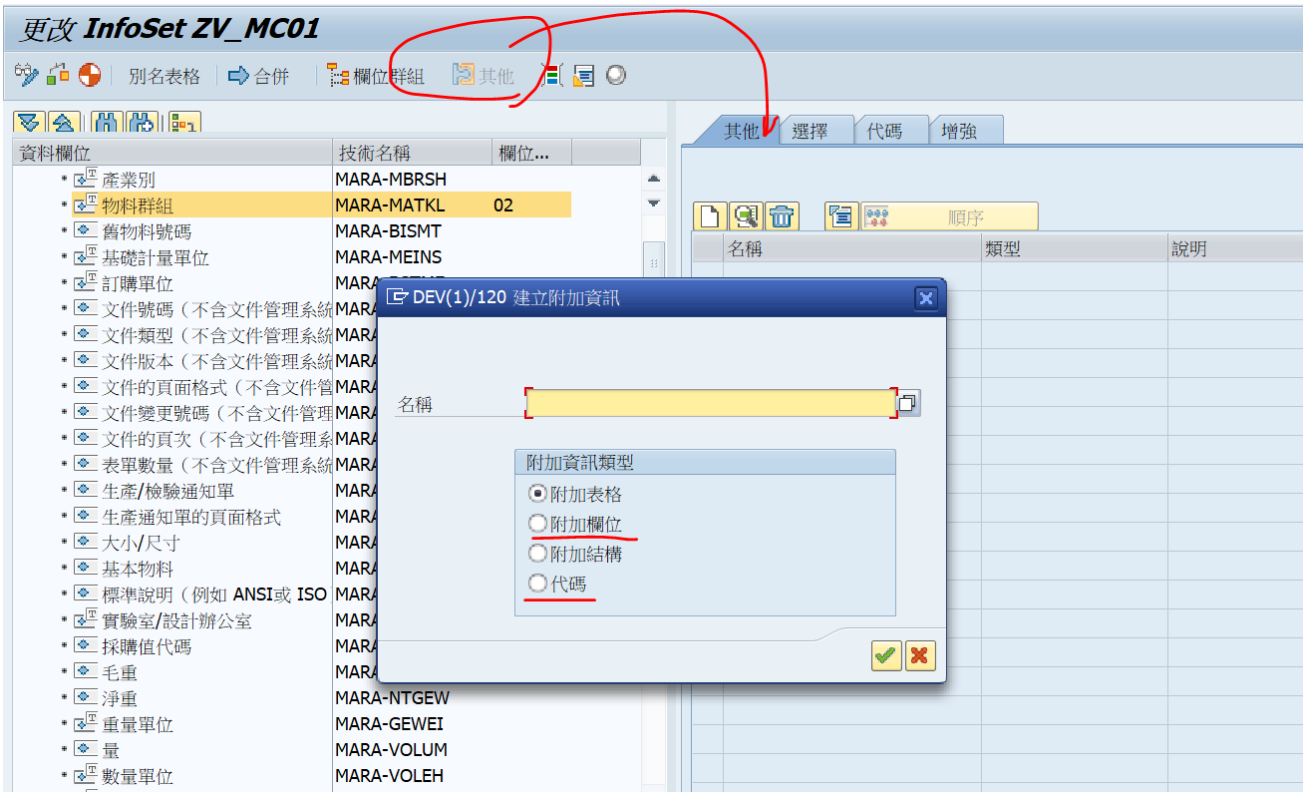


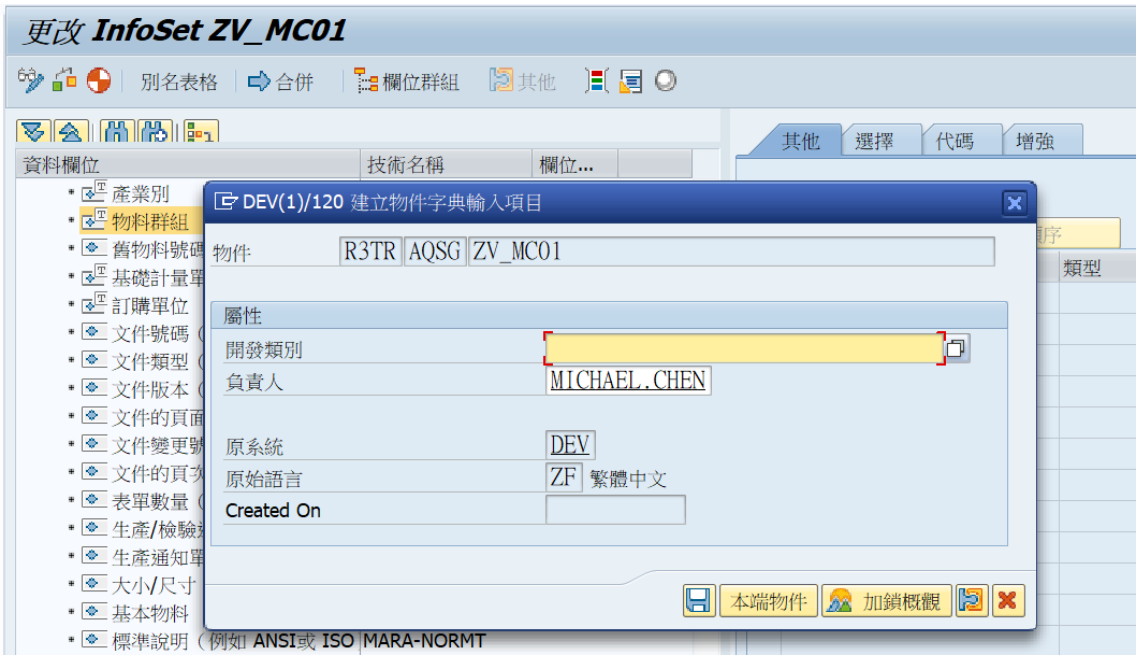
逐一選擇要使用的欄位，目前都是在 [欄位群組] 這個頁籤作業，比較沒有特殊的地方，注意取得主檔的內文描述要寫 Code 才能取得(國別代碼的關聯問題)。



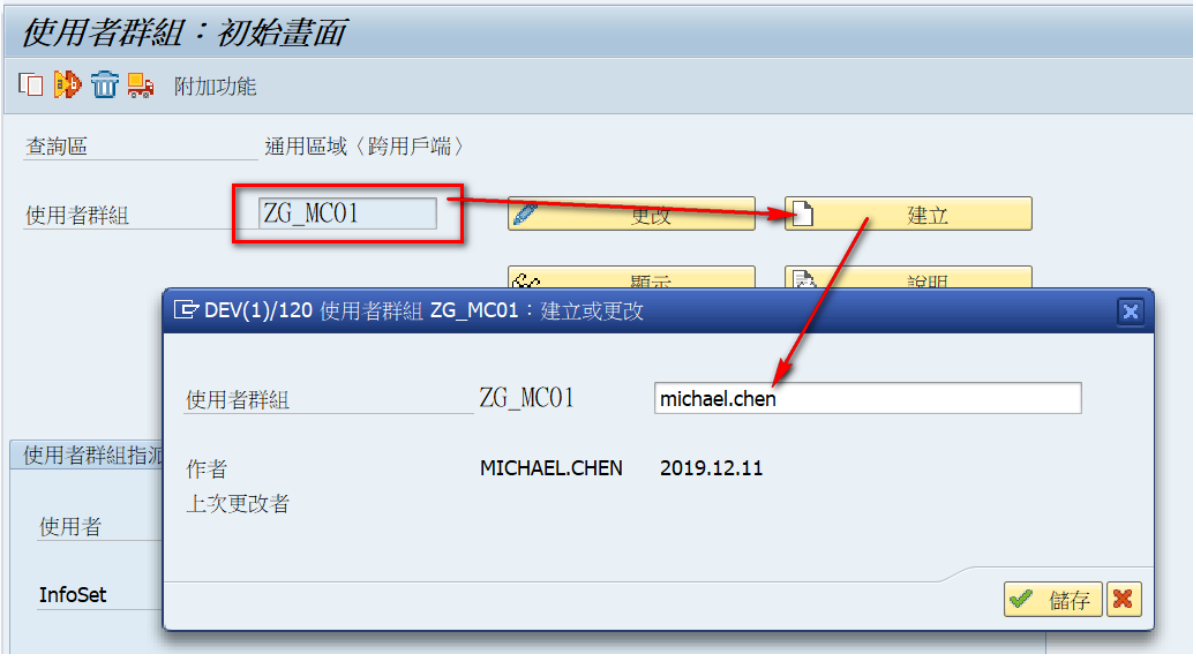
進階的設定都是在 [其他] 頁籤，這裡有 2 個地方要注意，1 是可以擴充欄位，另外則是可以擴充 abap 程式，解決無法由 Join 取得的問題，不過說實在的，如果都要寫程式才能解決，與其寫在此處，不如就直接新增 ABAP 程式，用這樣的方式寫 Code，很容易有效能問題。

如果都沒問題，存檔即可。因為 InfoSet 的變動與程式有關，所以會進入 Change Request 流程，此處不在贅述。

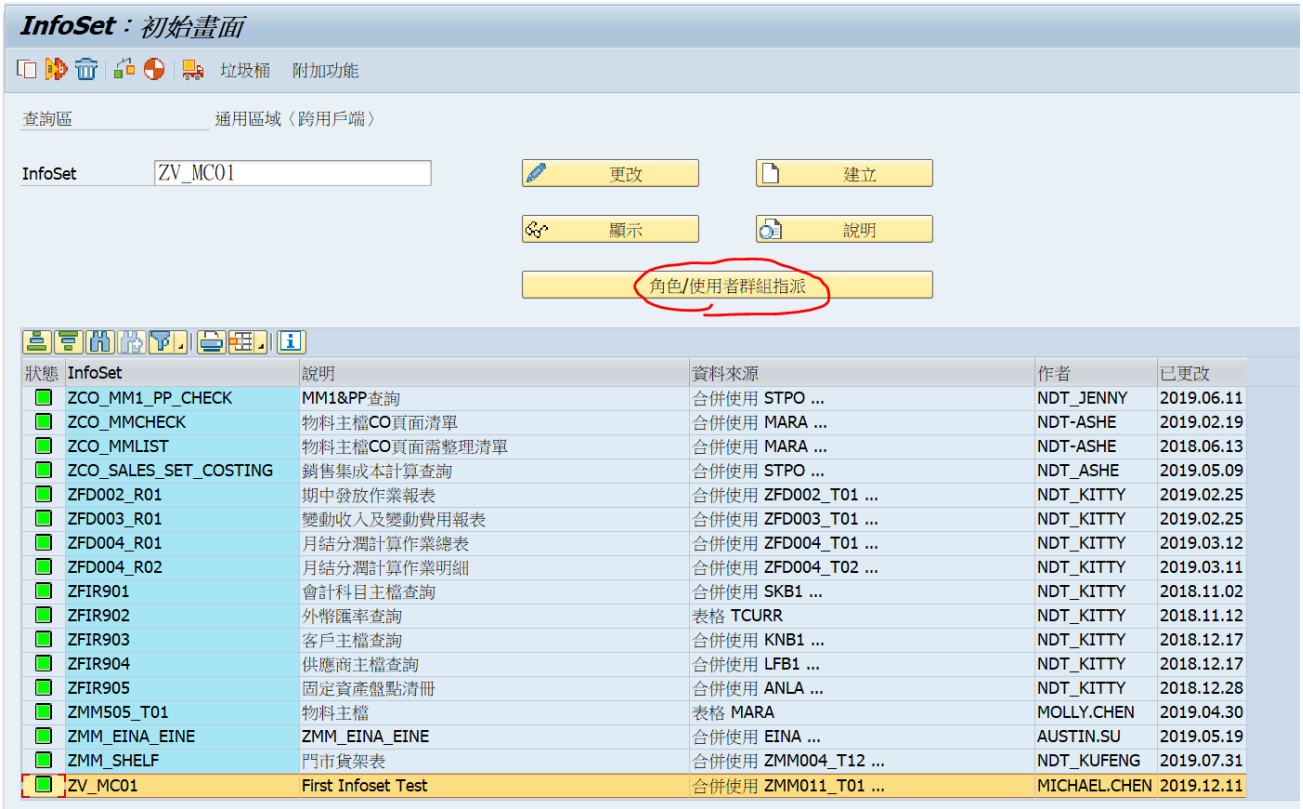




- 創建 UserGroup：SQ03

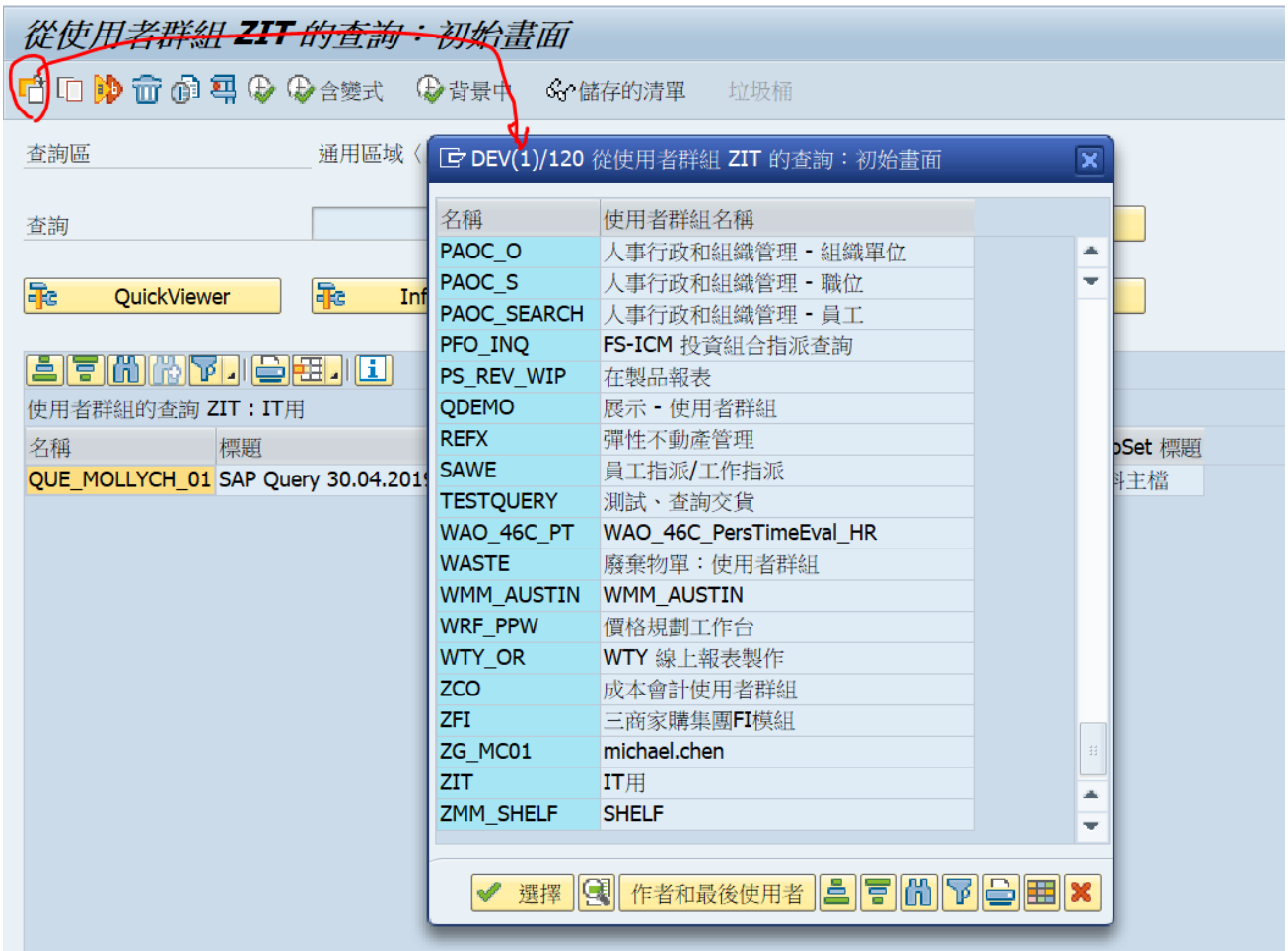


指派 InfoSet 的所屬 User Group(SQ02 的指派 button)，只有指定的群組人員，才能使用相關的 Query 及 SQVI。

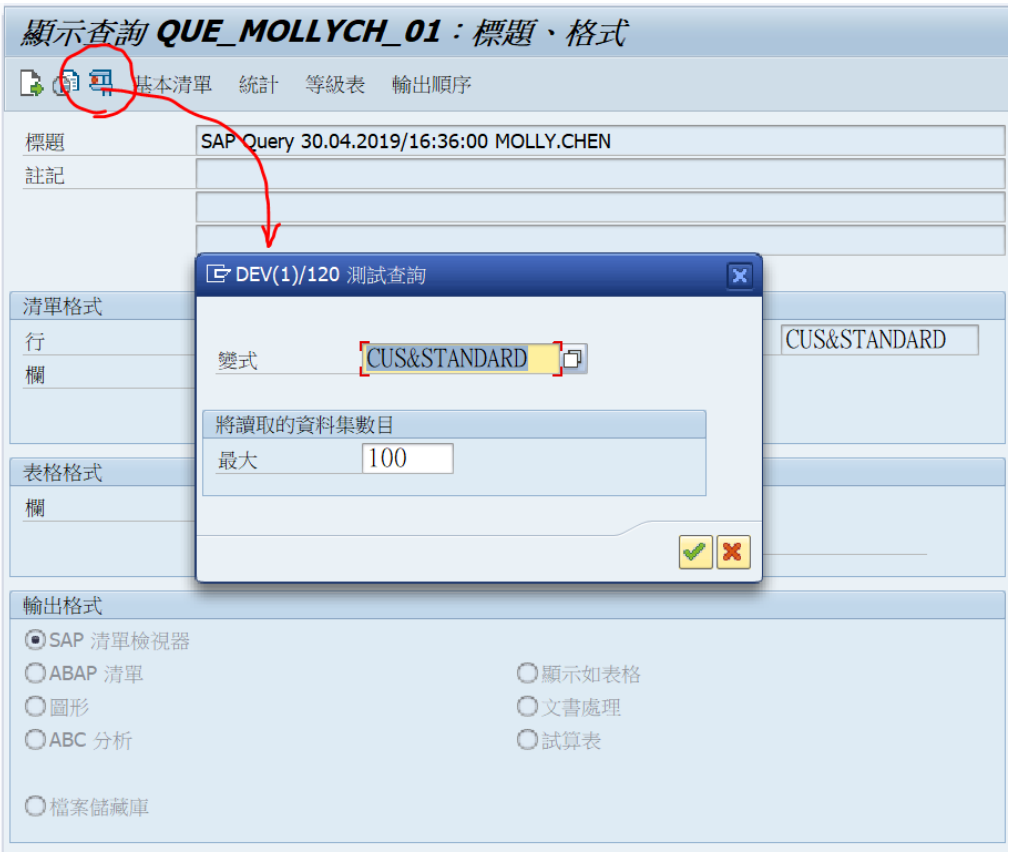


狀態	InfoSet	說明	資料來源	作者	已更改
	ZCO_MM1_PP_CHECK	MM1&PP查詢	合併使用 STPO ...	NDT_JENNY	2019.06.11
	ZCO_MMCHECK	物料主檔CO頁面清單	合併使用 MARA ...	NDT-ASHE	2019.02.19
	ZCO_MMLIST	物料主檔CO頁面需整理清單	合併使用 MARA ...	NDT-ASHE	2018.06.13
	ZCO_SALES_SET_COSTING	銷售集成本計算查詢	合併使用 STPO ...	NDT_ASHE	2019.05.09
	ZFD002_R01	期中發放作業報表	合併使用 ZFD002_T01 ...	NDT_KITTY	2019.02.25
	ZFD003_R01	變動收入及變動費用報表	合併使用 ZFD003_T01 ...	NDT_KITTY	2019.02.25
	ZFD004_R01	月結分潤計算作業總表	合併使用 ZFD004_T01 ...	NDT_KITTY	2019.03.12
	ZFD004_R02	月結分潤計算作業明細	合併使用 ZFD004_T02 ...	NDT_KITTY	2019.03.11
	ZFIR901	會計科目主檔查詢	合併使用 SKB1 ...	NDT_KITTY	2018.11.02
	ZFIR902	外幣匯率查詢	表格 TCURR	NDT_KITTY	2018.11.12
	ZFIR903	客戶主檔查詢	合併使用 KNB1 ...	NDT_KITTY	2018.12.17
	ZFIR904	供應商主檔查詢	合併使用 LFB1 ...	NDT_KITTY	2018.12.17
	ZFIR905	固定資產盤點清冊	合併使用 ANLA ...	NDT_KITTY	2018.12.28
	ZMM505_T01	物料主檔	表格 MARA	MOLLY.CHEN	2019.04.30
	ZMM_EINA_EINE	ZMM_EINA_EINE	合併使用 EINA ...	AUSTIN.SU	2019.05.19
	ZMM_SHELF	門市貨架表	合併使用 ZMM004_T12 ...	NDT_KUFENG	2019.07.31
	ZV_MC01	First Infoset Test	合併使用 ZMM011_T01 ...	MICHAEL.CHEN	2019.12.11

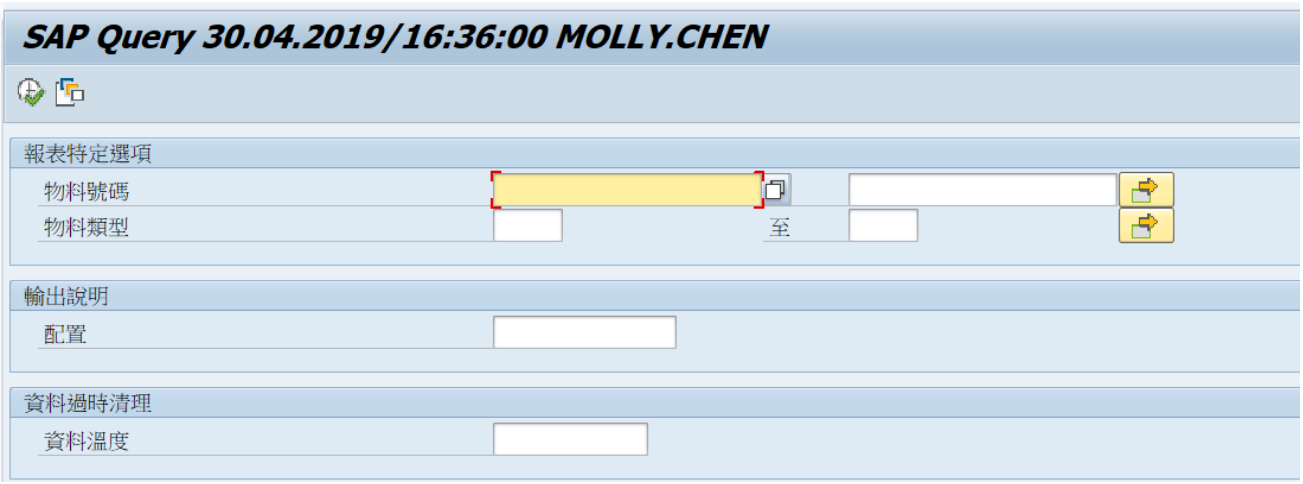
- 創建 Query：SQ01，先選擇使用者群組，會帶出該群組的 InfoSet。



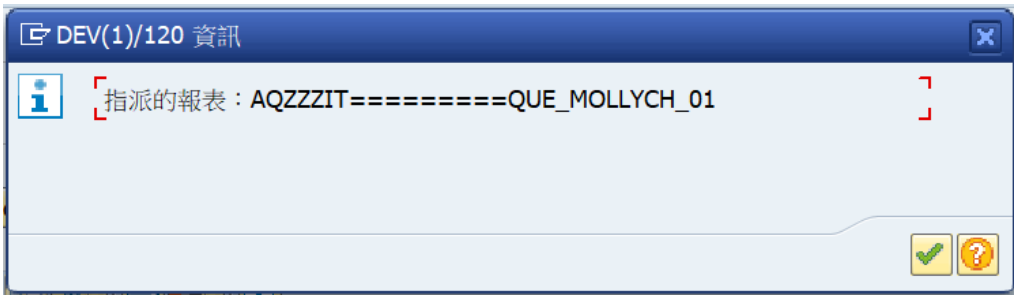
SQ01 的相關 button，可以自行測試，基本上是 All-In-One 的介面，把相關的功能都整合在一個介面。



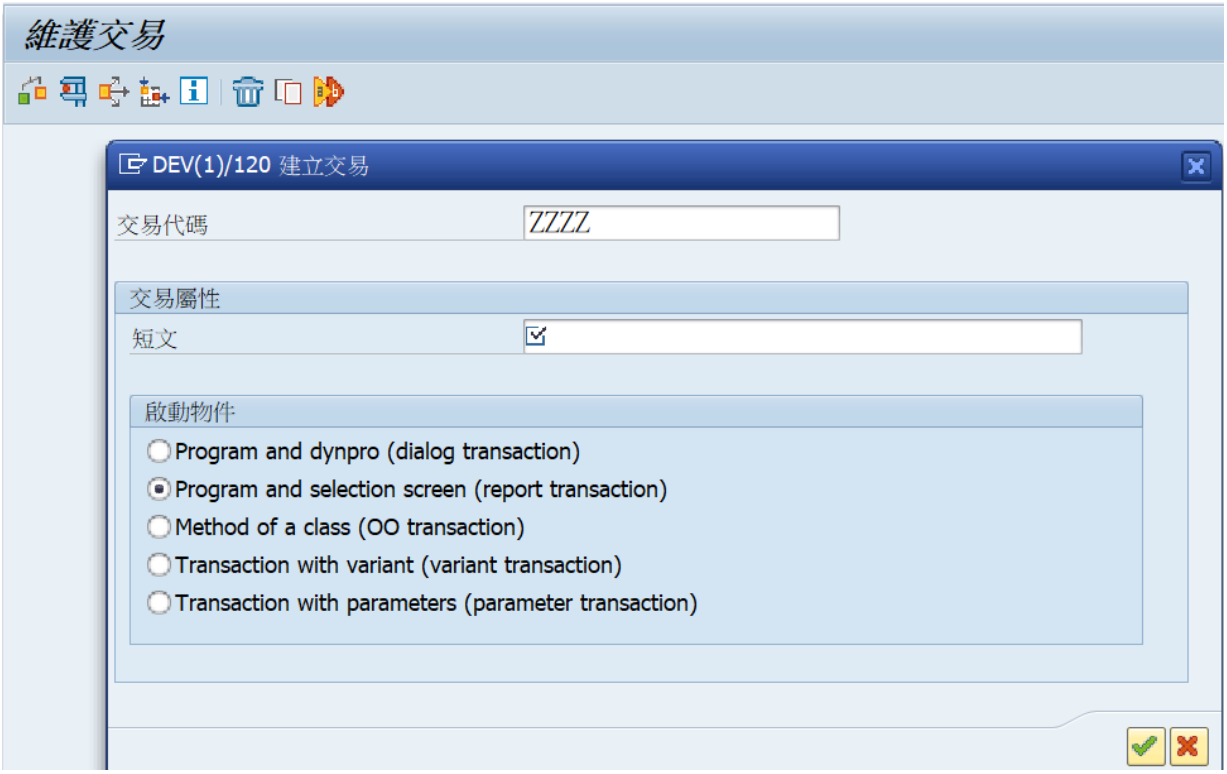
執行結果如下，效果幾乎和 ABAP 的程式一模一樣。



- 將 Query 設定 T-Code，首先，先取得報表名稱，並複製起來

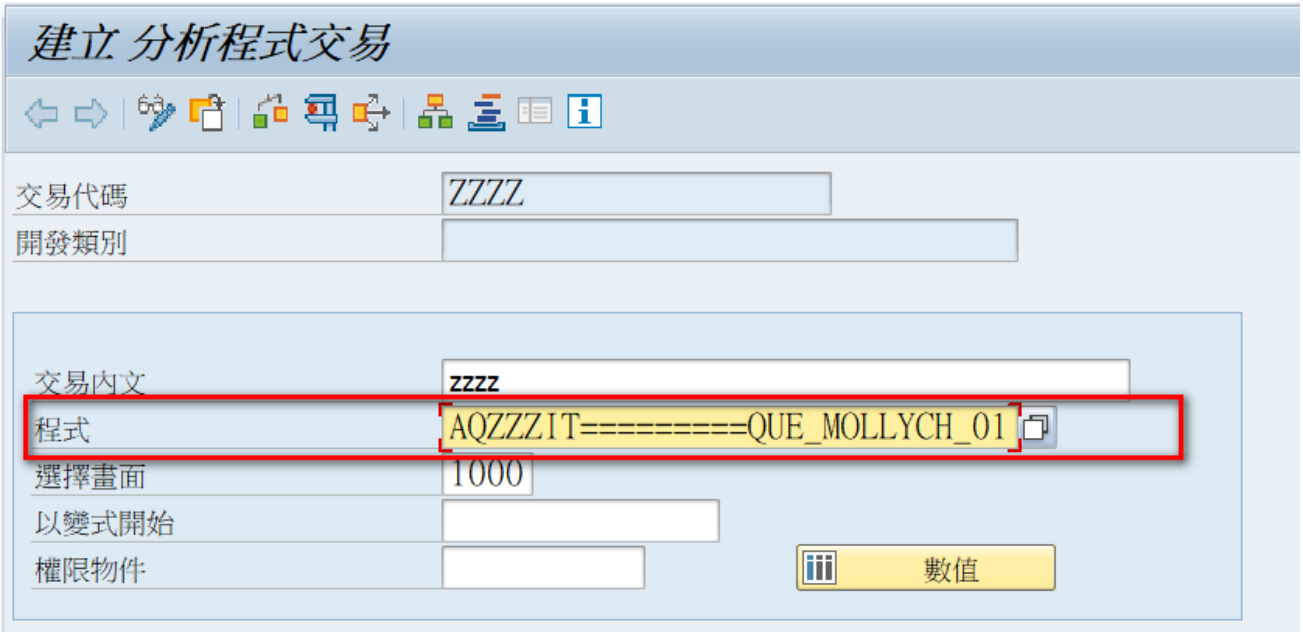


- 利用 SE93 設定 T-Code



將前步驟取得的報表名稱貼入即可。

PS：本作業是參考網路文件，並未實際設定測試，先假設此法可行。如果有問題再找時間研究。



第 14 章 LSMW 批次處理使用方法

https://www.fenginfo.com/1821.html

Session 1：總述

在 SAP 系統中，批次處理操作有多種方法。如果是對一個事物碼(TCODE)進行批次處理操作，常用的是 LSMW。LSMW 全稱是 Legacy System Migration Workbench。它能夠對靜態資料（如各類主資料）、動態資料（如初始化庫存）、業務資料（如銷售訂單）等進行成批操作，是上線資料準備和運維期間大規模資料操作的主力武器。

LSMW 的原理是設定批次處理的範本，再將準備好的格式化數據傳到 SAP 系統進行預轉換，如果合適就進行實際轉換。

批次處理範本可以有多種類型，包括系統已設定好的標準批輸入物件（Standard Batch Input Object）、BAPI（Business Object Method）、IDC（Intermediate Document）、批輸入記錄（Batch Input Recorder）等；本文以批輸入記錄（Batch Input Recorder）為例進行介紹，此方法也可稱為錄影法。

LSMW 錄影法批導入的原理是對需要批次處理操作的流程進行操作錄影，然後設定範本，再將準備好的數據傳入到 SAP 系統進行預轉換，如果合適就進行實際的轉換。

LSMW 導入資料操作分為以下幾大步驟：

- 準備需要導入的資料
- 轉換範本定義
  - ✓ 創建或打開轉換範本
  - ✓ SMW 中對事物代碼進行螢幕錄影，並定義變數
  - ✓ 定義來源資料的表結構
  - ✓ 將來源資料的表結構欄位與螢幕錄影的變數進行對應
- 讀取資料並預轉換
  - ✓ 將來源資料進行以預讀
  - ✓ 進行預轉換
- 實際轉換
  - ✓ 定義轉換進程
  - ✓ 實際轉換
  - ✓ 查看轉換結果

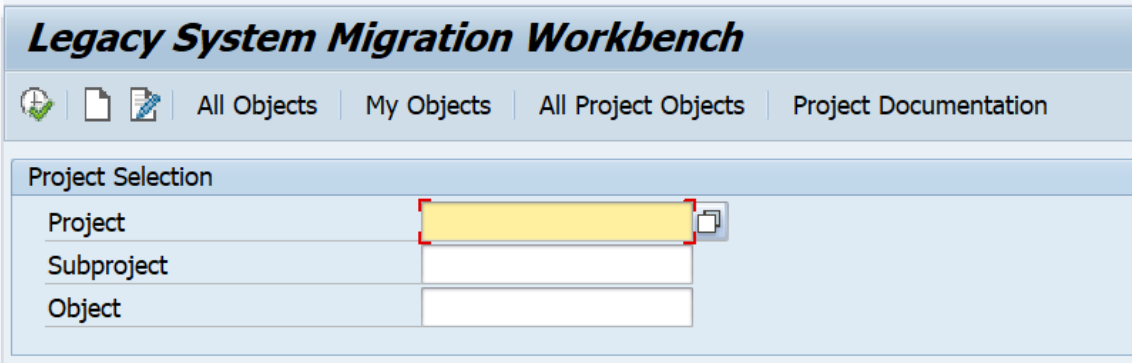
以下的實作是按經典的 14 個步驟進行講解。

錄影法只能對前台執行的事物碼 (T-CODE) 進行錄影，如果是後台配置，需轉化成前台可執行 T-CODE、SM30 維護視圖或是 SM34 維護視圖叢集方式進行操作。

Session 2：操作介面說明

- 管理介面

批導入的事物碼(TCODE)就是 LSMW，在主視窗介面輸入就可以進入 LSMW 的管理介面

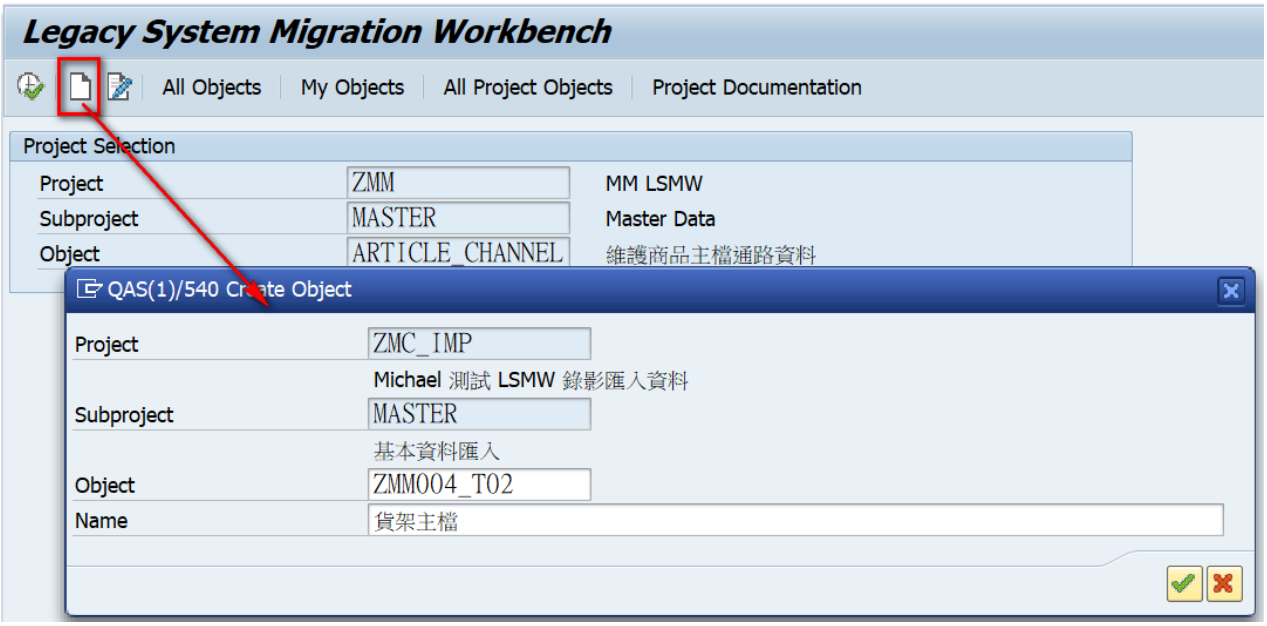


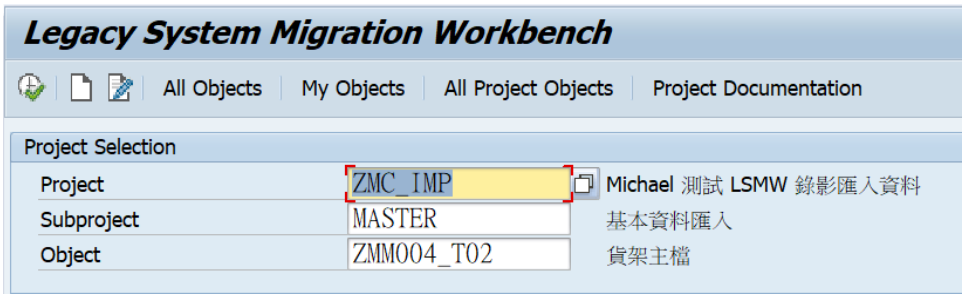
這個介面主要解決以下幾方面問題：

(1) 批導入物件的管理，包括新增、修改、刪除、查找、匯出、導入等操作。

(2) 進入到其它工作介面，主要有錄影操作介面、分步操作介面。

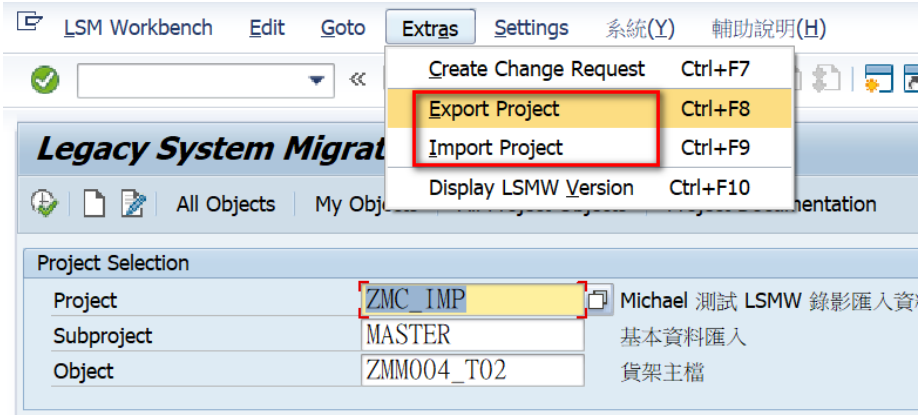
我們如果要進行一個批次處理操作，則需要輸入或通過選擇確定 Project、Subproject、Object，如下圖所示，然後再進行具體的操作。





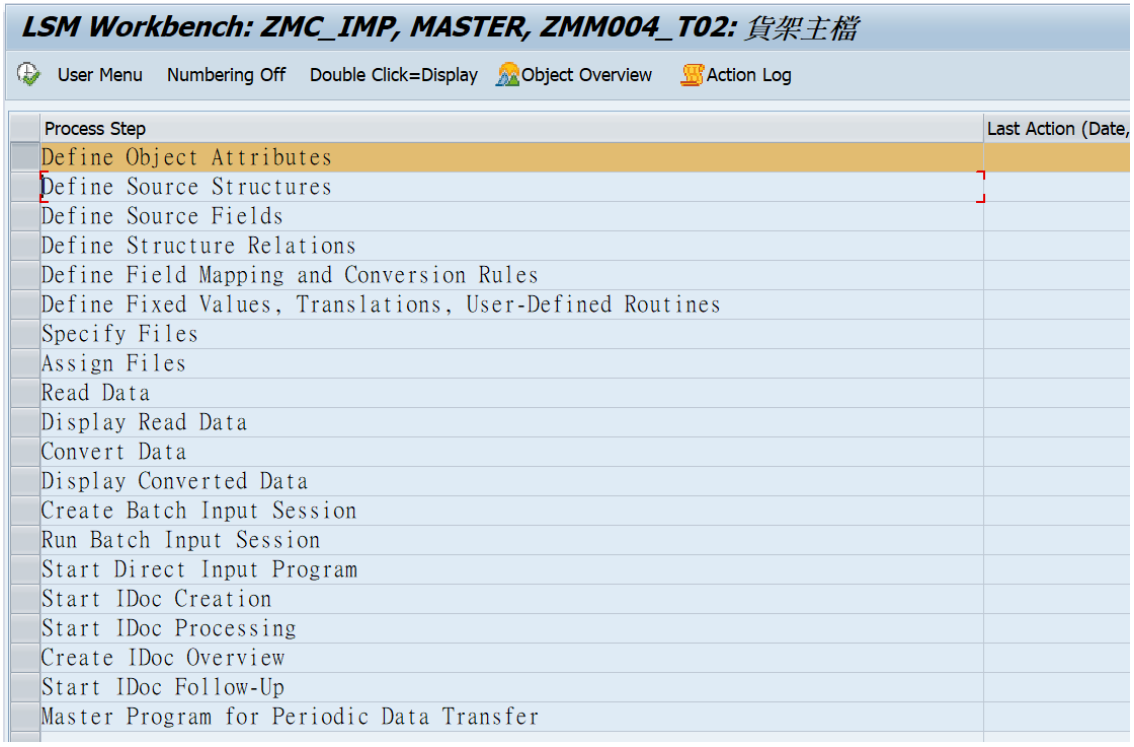
LSMW 管理介面—選擇 Object

當批匯出範本做好後，可以將其匯出備份成本地檔，也可將本地的範本檔導入進 ERP 系統。通過匯出、導入，可在不同 ERP 伺服器之間互換範本資源。匯出 (Export Project)、導入 (Import Projects) 的菜單如下圖所示。



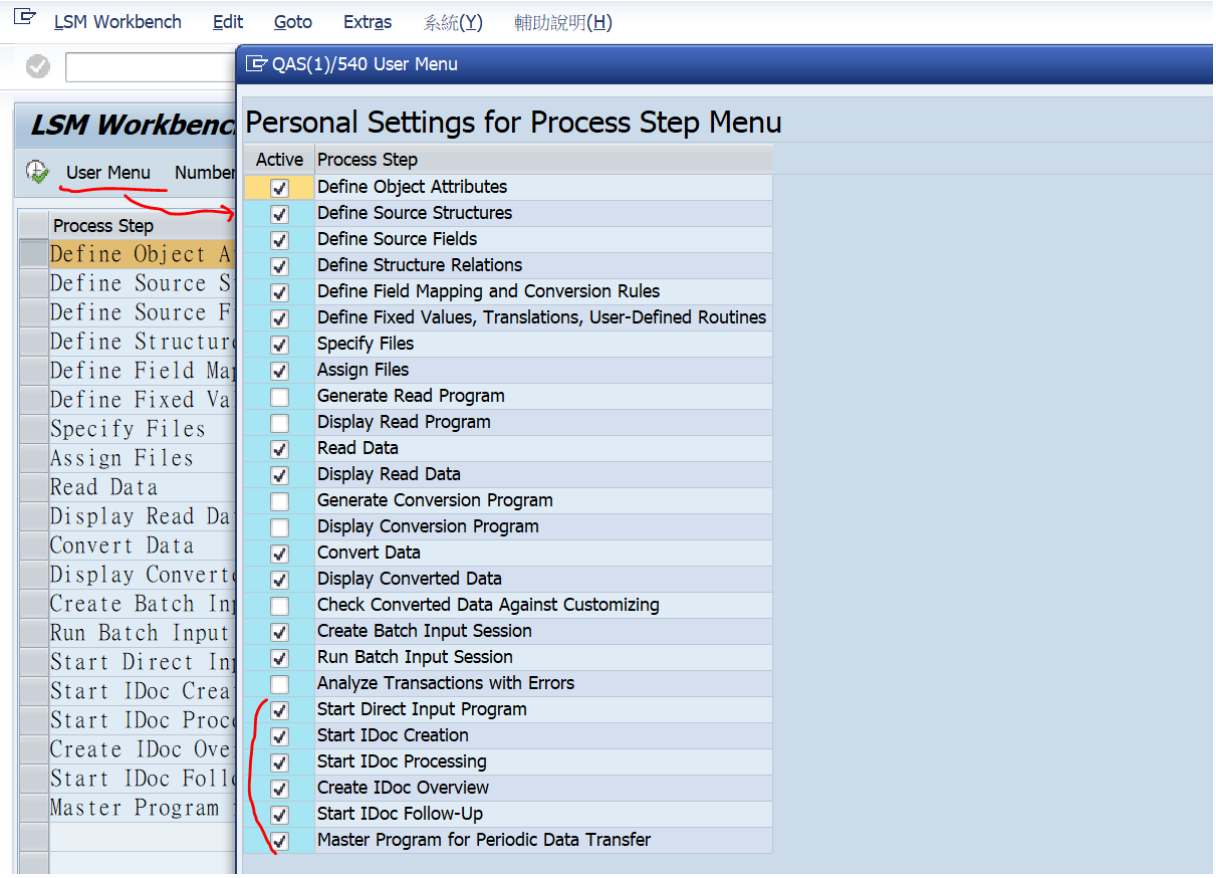
- 分步操作介面

在管理介面選擇 Project、Subproject、Object 後，按執行按鈕(⏲)進入分步操作介面，如下圖。

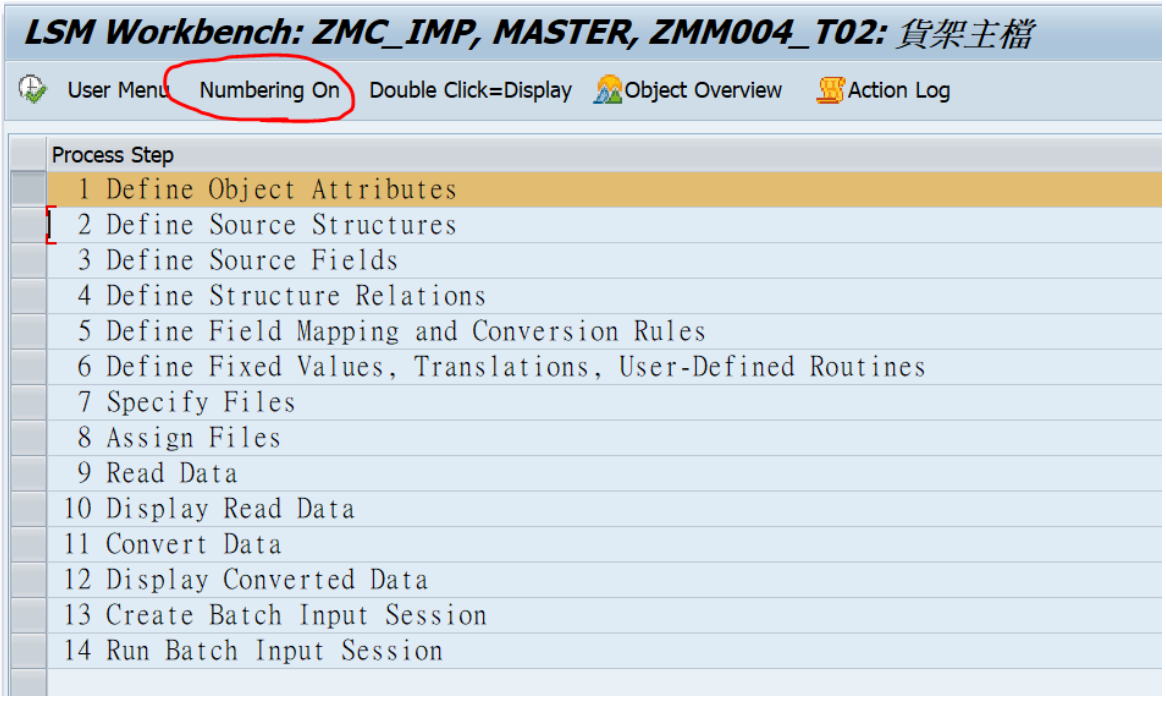


在上圖中顯示了分步操作的功能表，用滑鼠按兩下各功能表行可進入不同介面。這裡共有 20 個步驟，在實際操

作中我們不需要這麼多，只需要 14 個就可以了，點擊“User Menu”按鈕（ User Menu ）進入選擇使用者功能表的對話方塊，如下圖所示。



最後 6 個功能表項目取消勾選標誌，請勾選取消，確定後如下圖。



- **實際操作步驟**

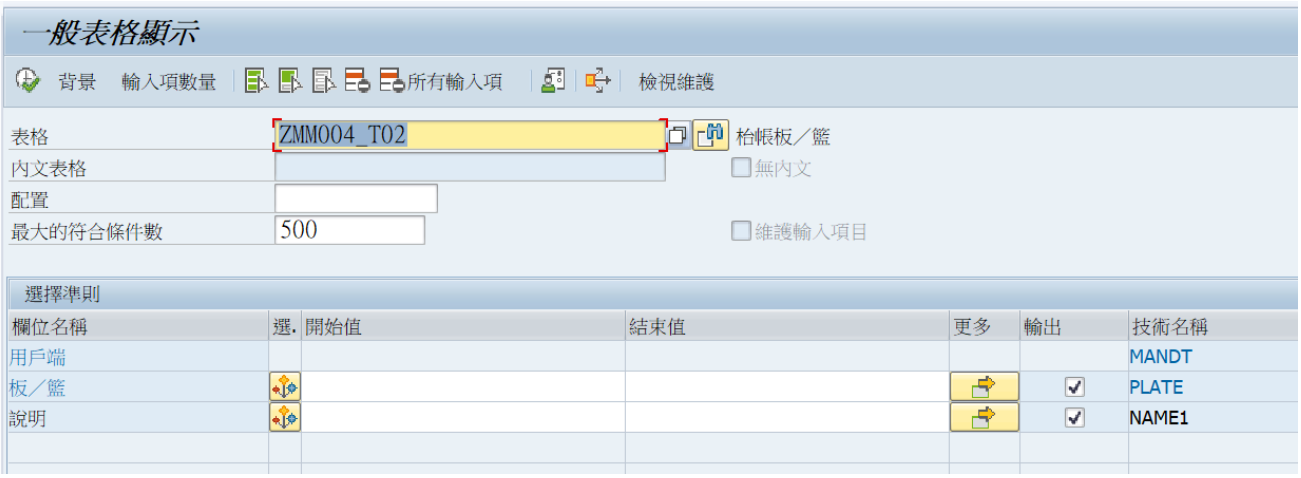
做一個完整的批次處理操作，需要很多步驟。這些步驟分為幾大部分：

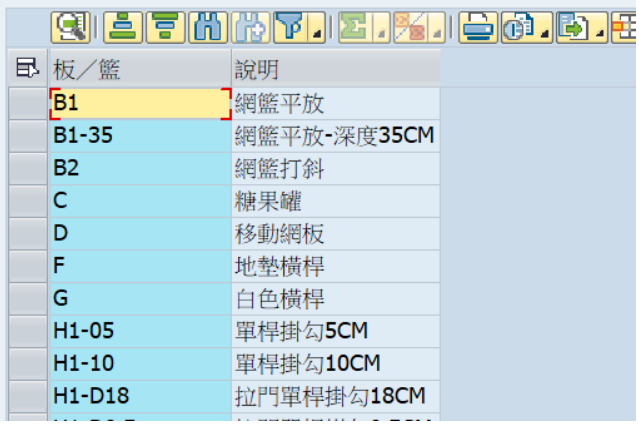
- ✓ 資料準備(步驟 1)；
- ✓ 創建批次處理物件(步驟 2)；
- ✓ 範本定義（步驟 3 至步驟 9），此部分又可分為三個小步驟；
  螢幕錄影(步驟 3)；
  源表定義(步驟 4 至步驟 6)；
  轉換欄位對應（步驟 7 至步驟 9）；
- ✓ 預轉換（步驟 10 至步驟 15），此部分又可分為兩個小步驟：
  導入源表資料（步驟 10 至步驟 13）；
  預轉換（步驟 14 至步驟 15）；
  實際轉換(步驟 16 至步驟 18)。

在以上操作中，除前三個步驟外，步驟 4 至步驟 18 均在分步操作介面下運行。

步驟 1：數據準備

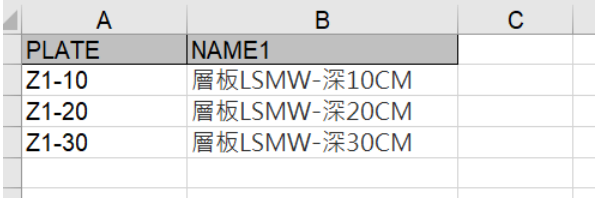
- ✓ 資料準備一般使用 Excel 整理資料，並轉換成標準文字檔(txt file)。以創建貨架主檔(ZMM004_T02）為例。





✓ 我們需要輸入以下欄位的資料：台帳貨架(PLATE) / 說明(NAME1)

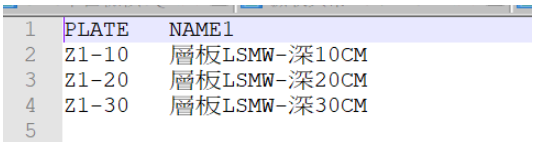
台帳貨架(PLATE)	說明(NAME1)
Z1-10	層板 LSMW-深 10CM
Z1-20	層板 LSMW-深 20CM
Z1-30	層板 LSMW-深 30CM



	A	B	C
	PLATE	NAME1	
	Z1-10	層板LSMW-深10CM	
	Z1-20	層板LSMW-深20CM	
	Z1-30	層板LSMW-深30CM	

✓ 將整理好資料的 Excel 表另存為 txt 檔，用定位字元(TAB 鍵)分隔。

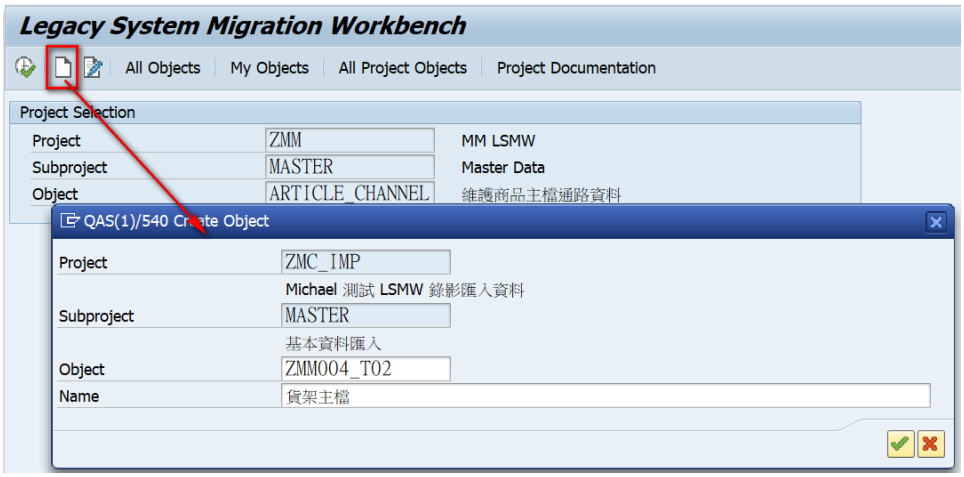
✓ 文字檔參考內容如下



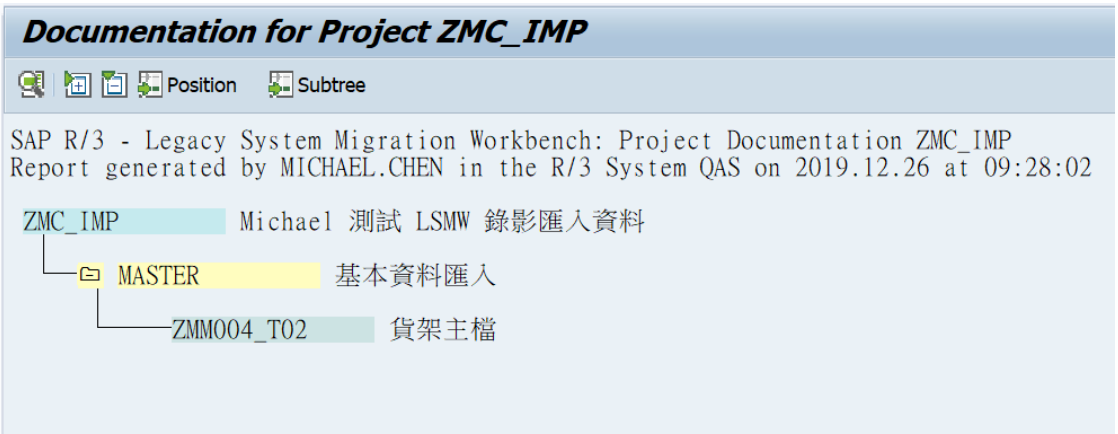
```
PLATE    NAME1
Z1-10    層板LSMW-深10CM
Z1-20    層板LSMW-深20CM
Z1-30    層板LSMW-深30CM
```

步驟 2：創建批次處理對象

一個批次處理物件是個樹形結構，包括三層：Project、Subproject、Object，我們分稱為專案、子專案、物件。本步驟就是創建一個批次處理物件(Object)。如果此物件(Object)已存在，可以不用創建，用輸入或選擇的方法找到指定的物件。請進入系統工作介面，如下圖所示。



點擊“Project Documentation”按鈕（Project Documentation）顯示物件命名資訊，如下圖所示。

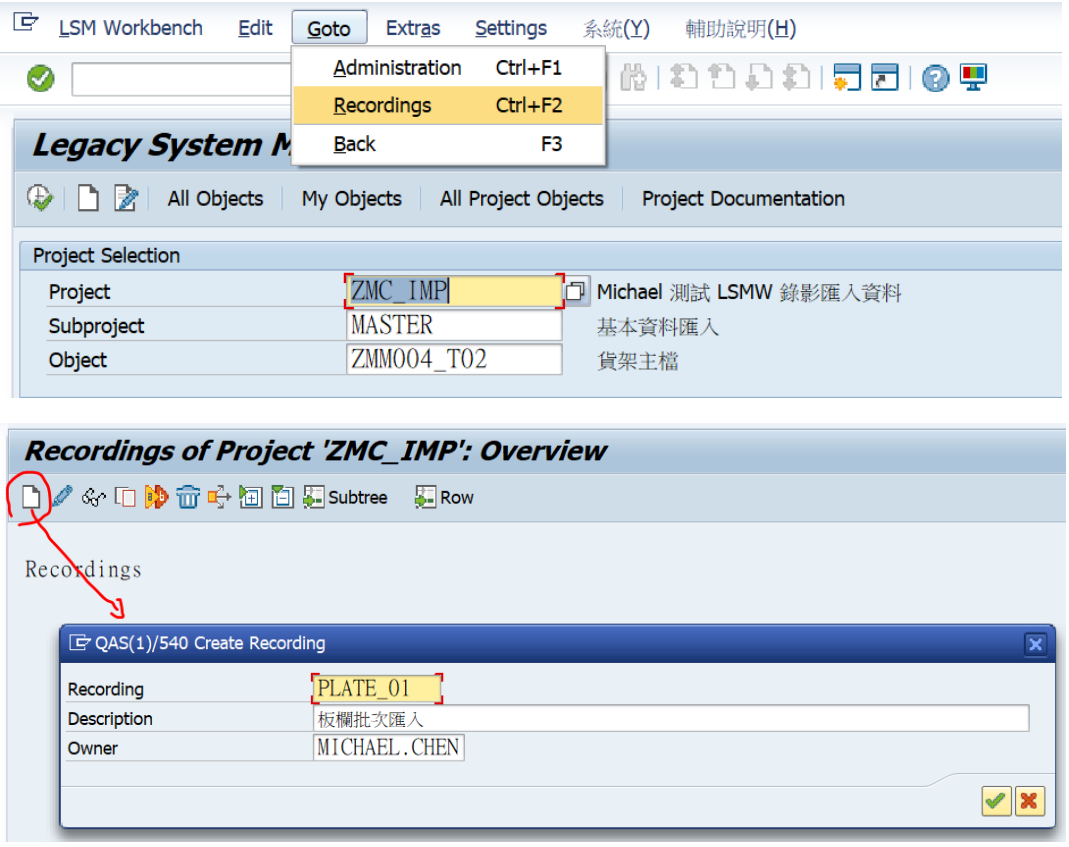


步驟 3：螢幕錄影並定義變數

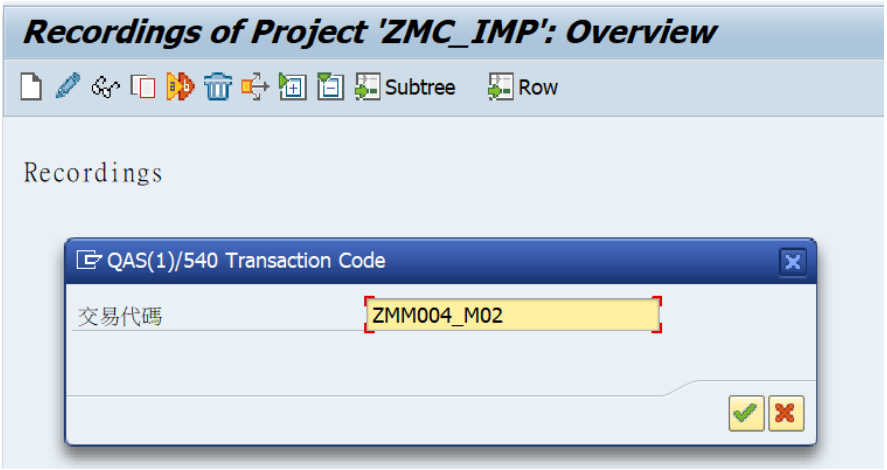
此步進入螢幕錄影階段，這是技術性、經驗性很強的步驟，在錄屏過程中做到儘量少的螢幕轉換；輸入項不要彈出查詢視窗；對於滾動列表應該使用 PageUP、PageDown 鍵翻頁，而不能用滑鼠點擊滾動行。

此步驟除了錄屏外，還需要對錄屏裡的欄位進行整理，並將某些欄位由常量變換為變數。

在管理介面點擊功能表 [Goto → Recordings] 進入螢幕錄影介面，如下圖所示。



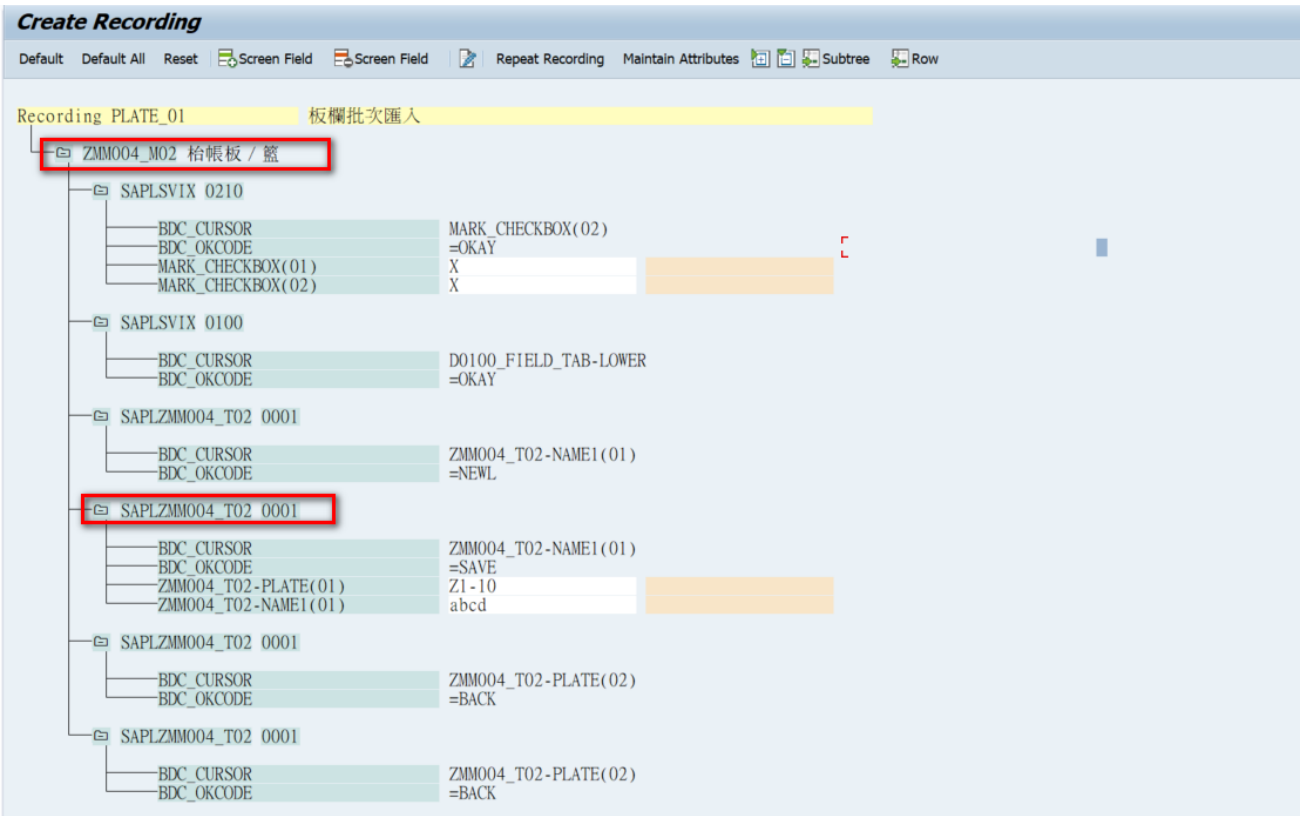
點擊確認按鈕（✔）進入下一屏，輸入需要錄影的事物代碼，本例事物代碼為 ZMM004_M02。



錄影的資料是準備資料中的第 1 個（當然可以是任意一個），依上述原則逐項輸入個欄位資料。

存檔後並於錄影後返回 LSMW 的操作介面，如下圖所示。

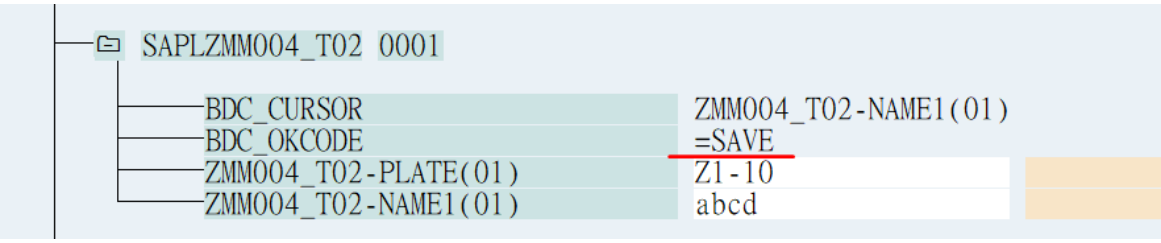
開始整理錄影的欄位，並將某些欄位由常量變更為變數，這個過程有許多細節要注意，但是重點就是操作一次指定 T-Code，新增一筆資料，然後配合整理好的 Excel 批次轉入資料。



上圖介面是一個樹形結構，它表示剛才錄影的資料，錄影後的資訊分為三層：

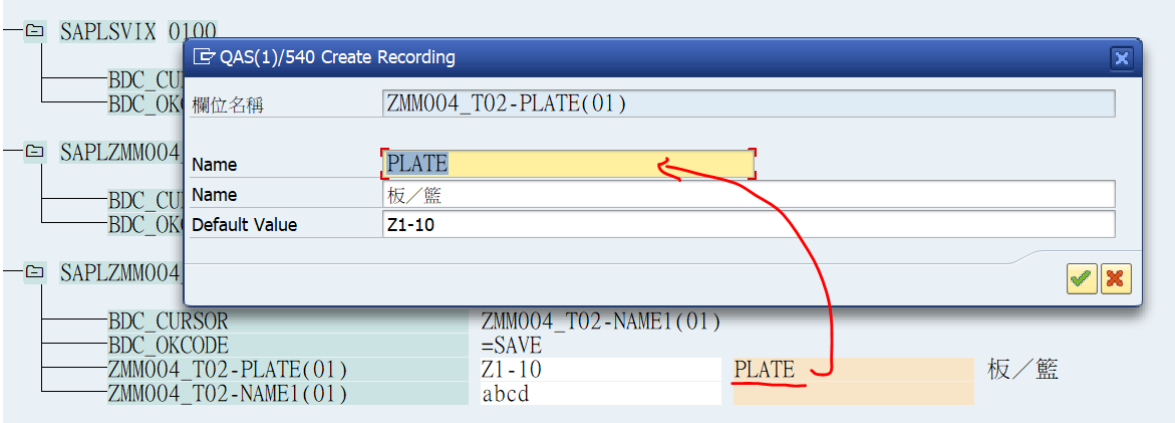
- ✓ 事物代碼(本例為 ZMM004_M02 台帳板/籃)
- ✓ 錄影螢幕(如 SAPLZMM004_T02 0001)
- ✓ 欄位資訊(如 ZMM004_T02-PLATE

我們將 Tree 收縮後詳細查看，就會看到錄影中的每一個螢幕在這裡都對應了一段資料，本範例因為只有 2 個欄位，基本上對照著看應該就可以理解紀錄的內容，如果是多頁籤的 T-Code 程式，就會一頁籤一段 Tree 內容。

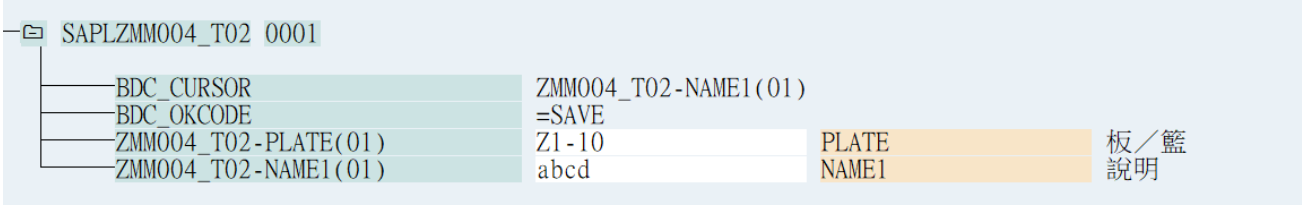


上圖 BDC_OKCODE [=SAVE] 代表按了確認按鈕。

接下來，要設定變數，基本上就是在匯入欄位和外部文字檔第一行間，建立關聯關係，請在要設定的欄位上 DbClick 或是按上方的 [Default] 按鈕，就會進入下面的設定畫面。



依序將槓欄位都用相同的方式設定好關聯。

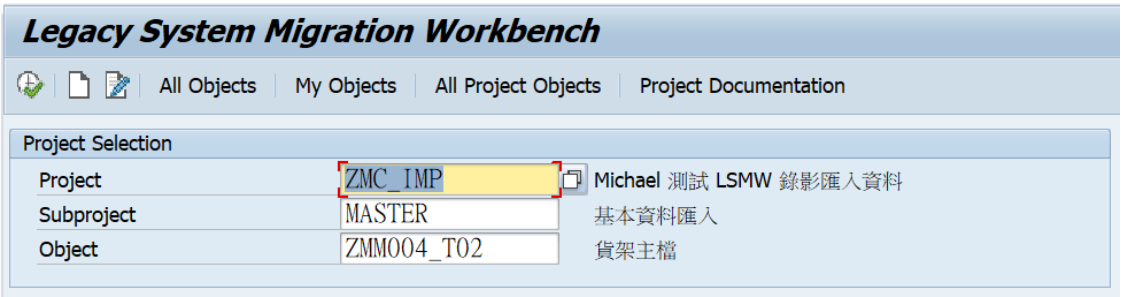


如果某個欄位的值是固定的，不必由外部文字檔傳入，例如計量單位是預設值 PC，就不需要定義為變數。

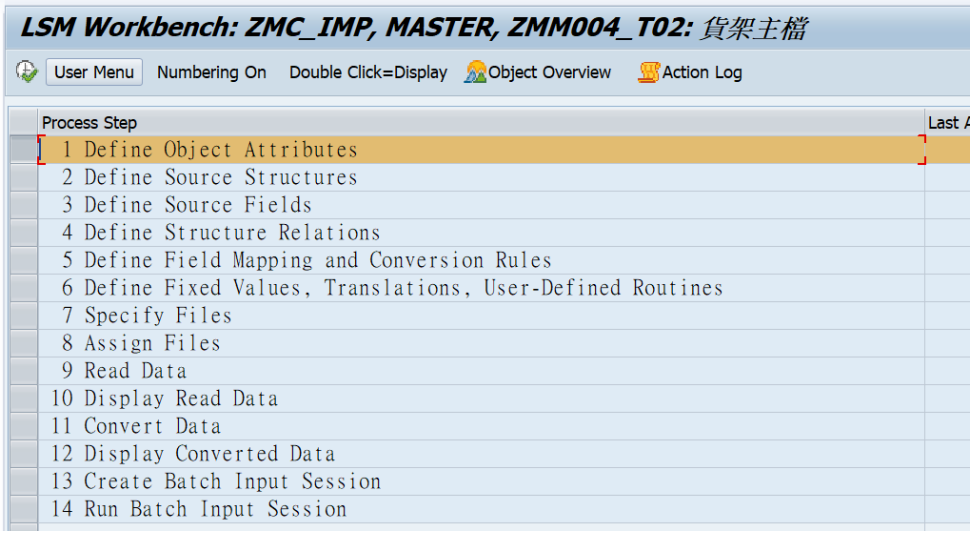
以上變數設定完成後按保存按鈕（ ）退出，返回到管理介面。

步驟 4：定義對像屬性

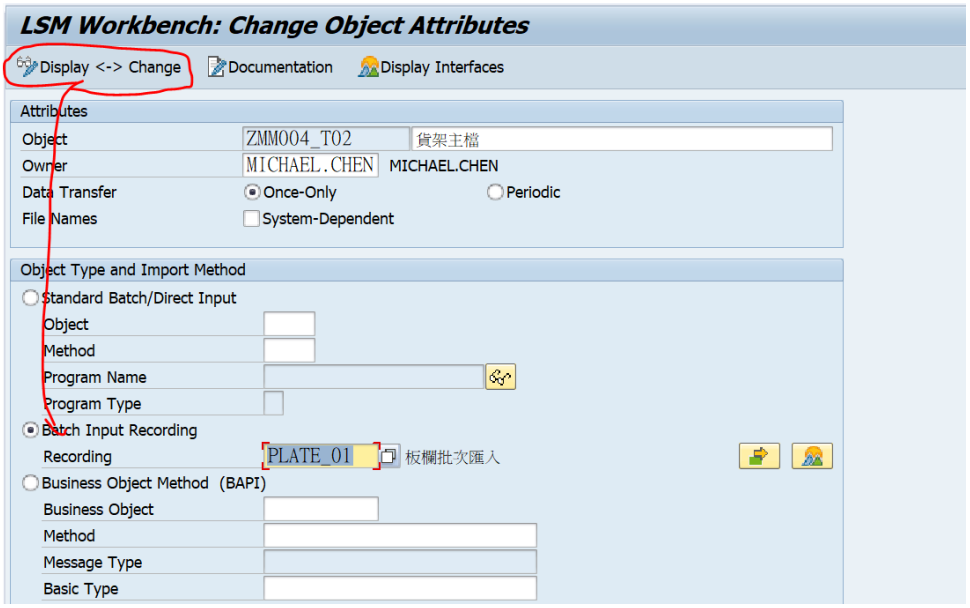
在以下步驟中，我們將從管理介面進入到分步操作介面。



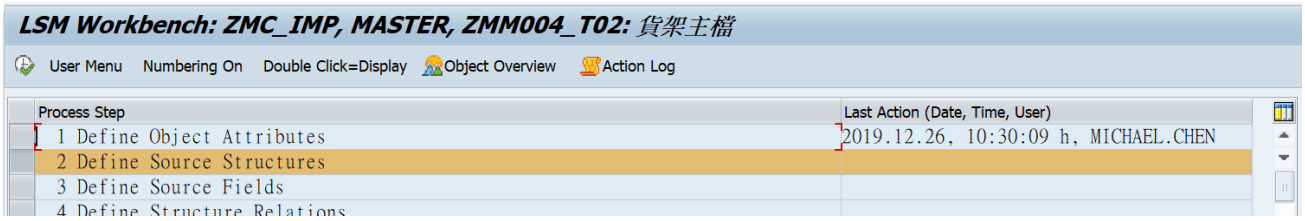
點擊執行按鈕（ ），進入分步操作介面，如下圖。



步驟一是定義物件的屬性，主要是將物件指明錄影的名稱。用滑鼠按兩下 Define Object Attributes 進入到定義物件屬性介面。



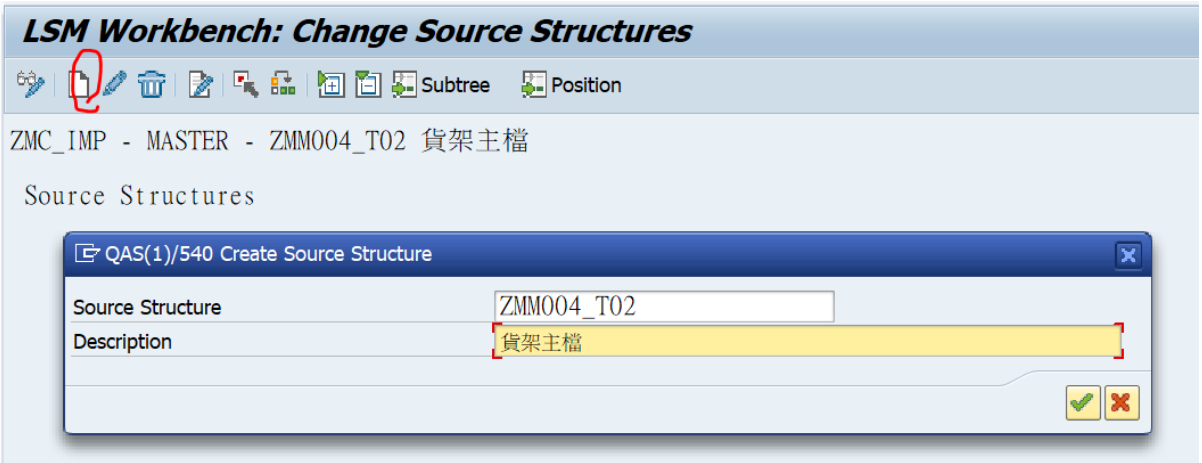
這時分步操作的介面的右部顯示了一行，表示最後操作的日期、時間和操作者，如下圖所示。

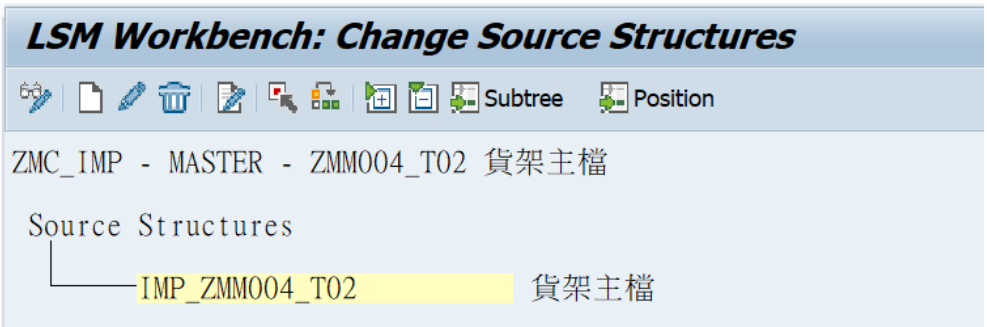


步驟 5：定義源表結構名稱

此步驟定義源表的結構名稱，在分步操作介面用滑鼠按兩下 Maintain Source Structures 進入操作介面。

如果進入的介面是顯示狀態，請點擊修改按鈕（ ）進入編輯狀態。再點擊新建按鈕（ ），輸入源表定義名稱及描述，如下圖所示(內容應該為 IMP_ZMM004_T02)。



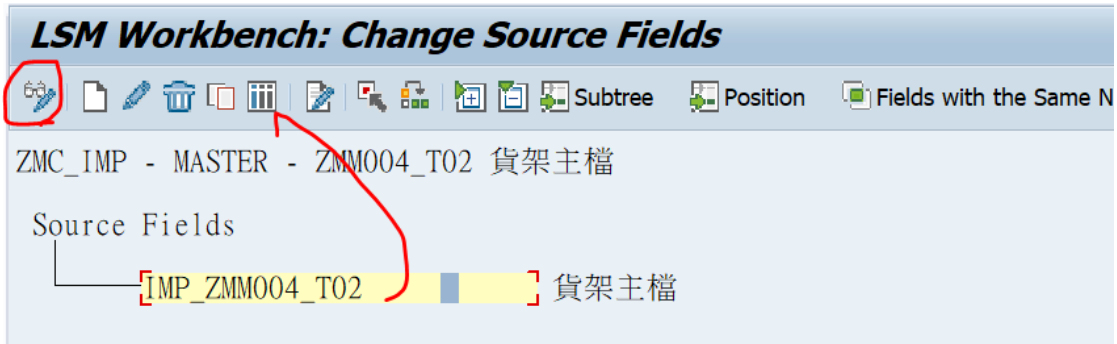


存檔後退出分步操作介面。

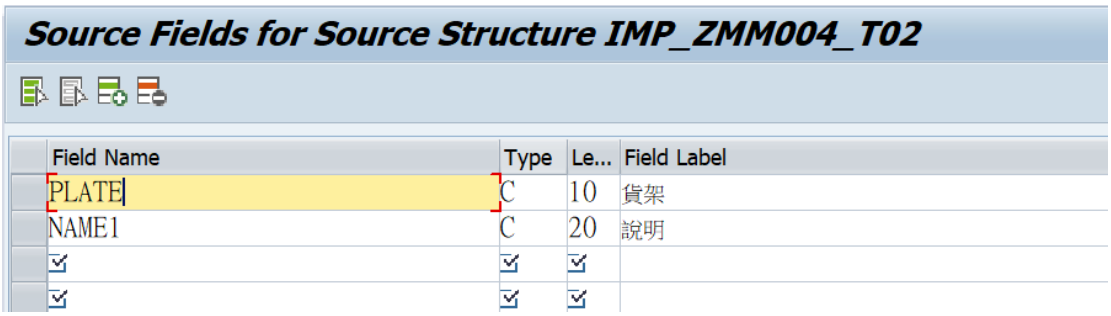
步驟 6：定義源表字段結構

此步驟是在 LSMW 物件中定義源表的資料結構，也就是定義第 1 個步驟資料準備時的欄位資訊。

在分步操作介面，用滑鼠按兩下 Maintain Source Fields，進入定義源表字段結構介面。如下圖所示。如果進入的是顯示狀態，按修改按鈕（ ）進入修改狀態。



選中來源資料結構名稱（IMP_ZMM004_T02）點擊維護按鈕（ ）進入欄位編輯介面。



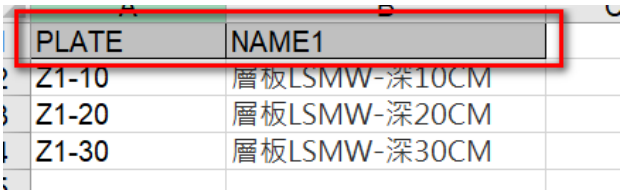
結構輸入介面，共有 4 列，需要分別填寫：

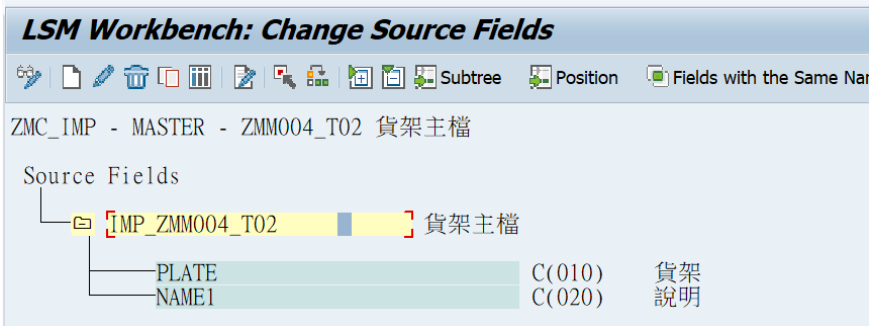
1．欄位名(Field Name)：輸入源表中的欄位名，詳見 excel 表中的表頭。

2．類型(Type)：資料類型，C 為字元型。

3．長度(Length)：欄位長度，可儘量大一些。

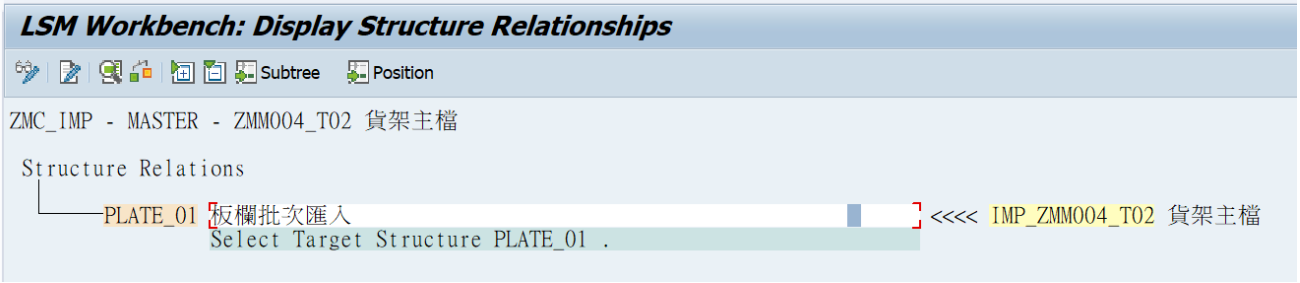
4．描述(Field Description)：欄位描述，可選項。





步驟 7：源表結構與錄影關聯

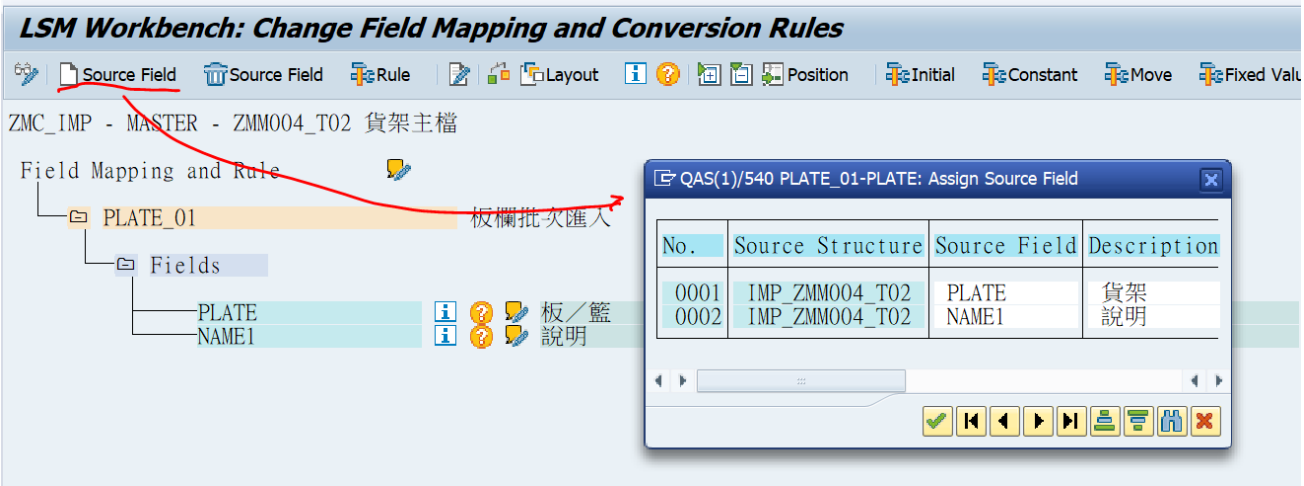
本步驟定義源表結構與錄影之間的關係。在 [Define Structure Relations] DbClick 進入操作介面，如圖。



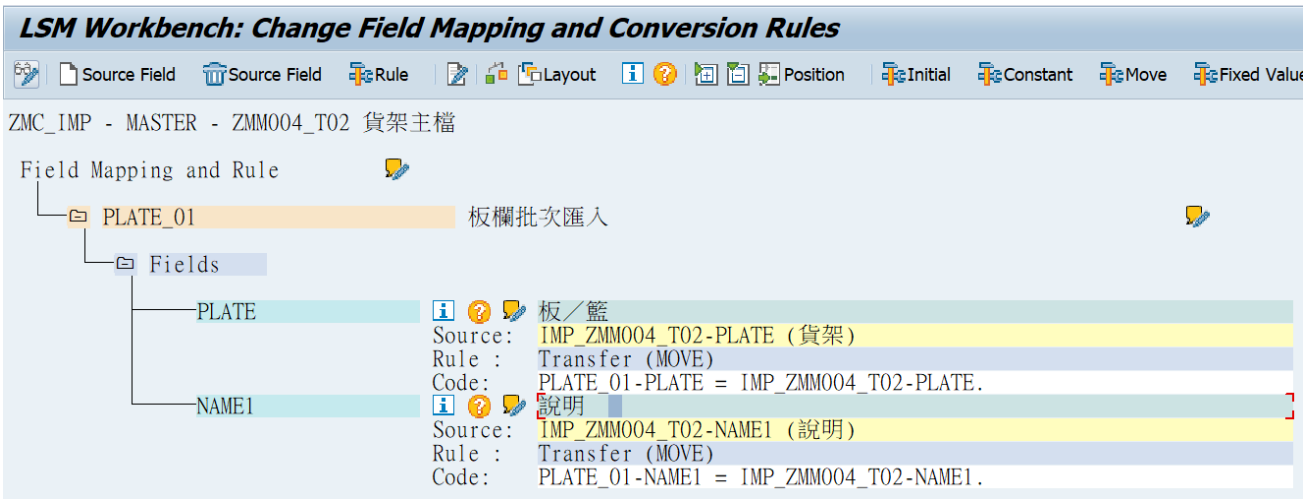
由於只有一個錄影與一個源表結構，系統自動對應，左邊是螢幕錄影設定的名稱（步驟 3），右邊是源表設計時的名稱（步驟 5、步驟 6）。如有多個需選擇對應。按返回按鈕（ ）返回分步操作介面。

步驟 8：源表字段與錄影欄位關聯

本步驟需要將源表的欄位結構與錄影中定義的變數相關聯，稱為欄位映射（Filed Mapping），並可以設定轉換規則（Conversion Rules）。在分步操作介面用滑鼠按兩下 Define Field Mapping and Conversion Rules，進入源表及錄影欄位關聯操作介面，如下圖。如果是顯示狀態，請按修改按鈕（ ）變成修改狀態。



依次類推，分別將各欄位分別逐一對應，全部完成後介面如下圖



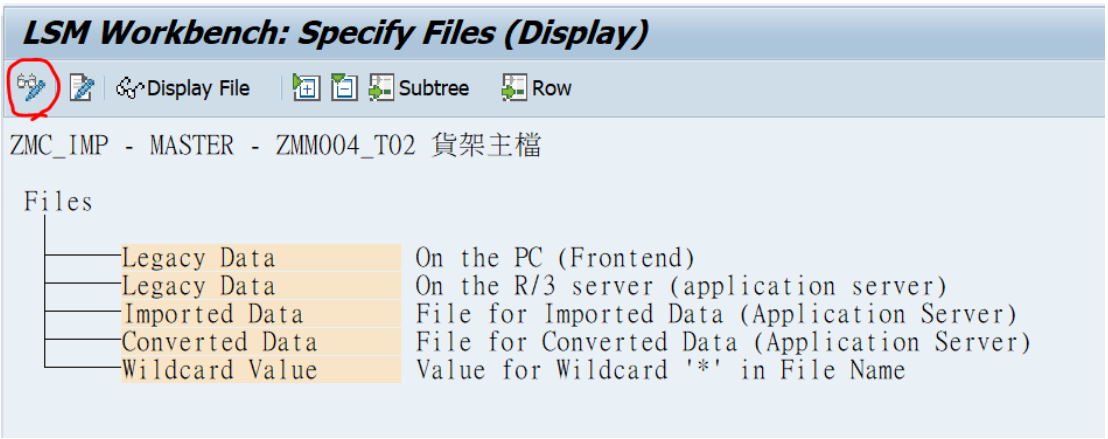
步驟 9：固定值，轉換條件，用戶定義

在此步驟中可以設定錄影中欄位的值來源，除對應源表字段外，還可以設定為固定值、轉換條件、或是更為複雜的用戶定義（用 ABAP 程式設計）。一般使用可跳過此節。

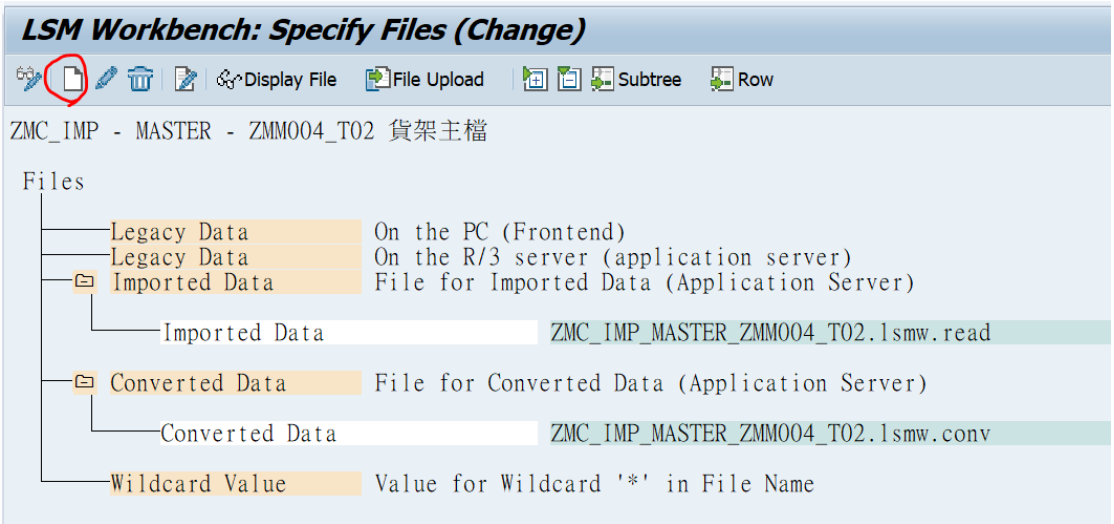
如需進入請在分步操作介面用滑鼠按兩下 Maintain Fixed Values, Translations, User-Defined Routines。

步驟 10：指定源表檔

以上步驟已完成資料準備和範本定義，以下將進入資料導入階段。本步驟指定源表的檔，也就是在步驟 1 中生成的文字檔。在分步操作介面用滑鼠按兩下 Specify Files 進入操作介面，如圖 3_10_1。

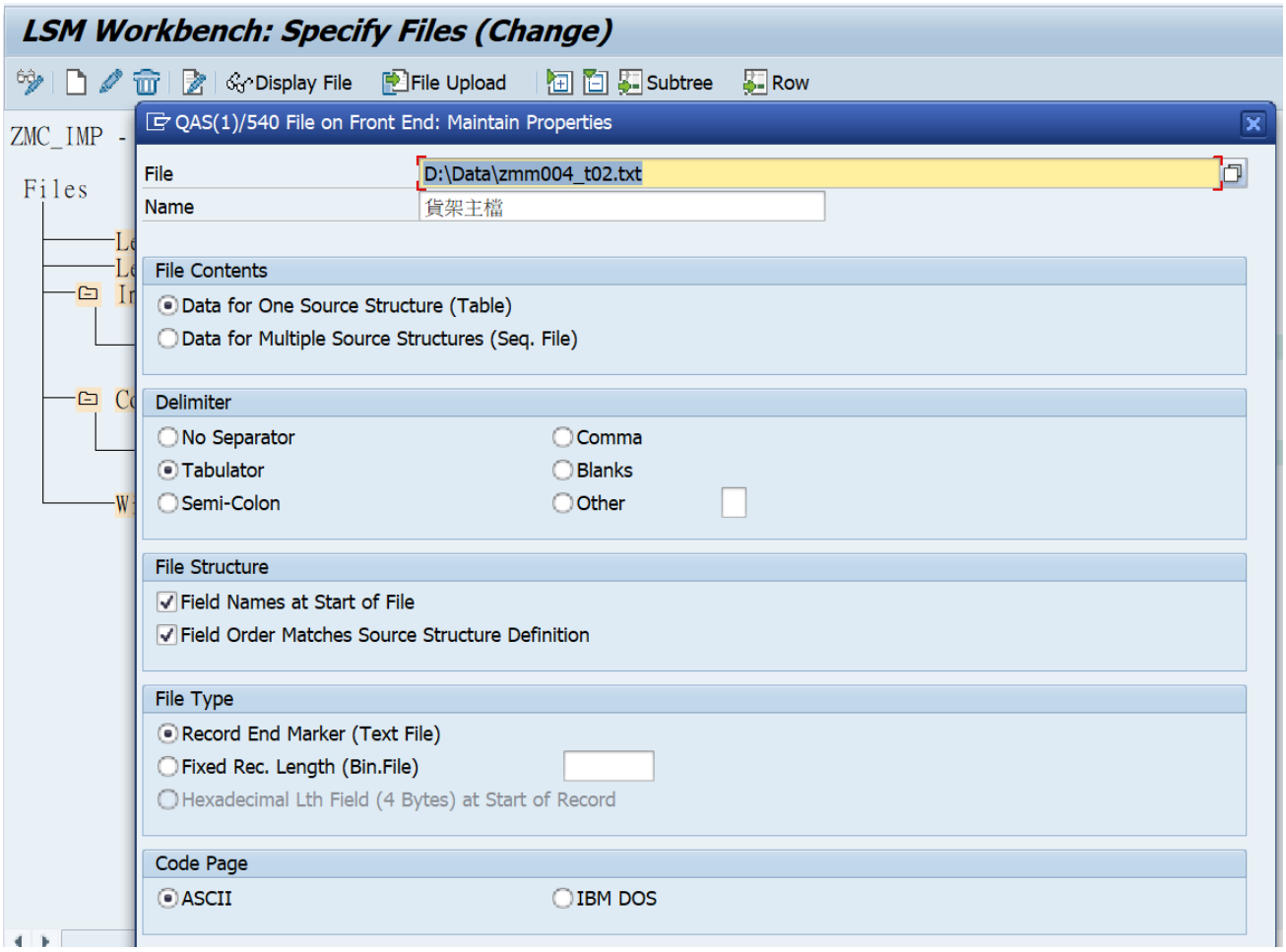


在本步驟中要指定三個值，其中一個需要手工指定，兩個自動生成。手工指定的 Legacy Data，自動生成的是 Imported Data、Converted Data。點擊修改按鈕（ ）進入修改狀態，如下圖所示。

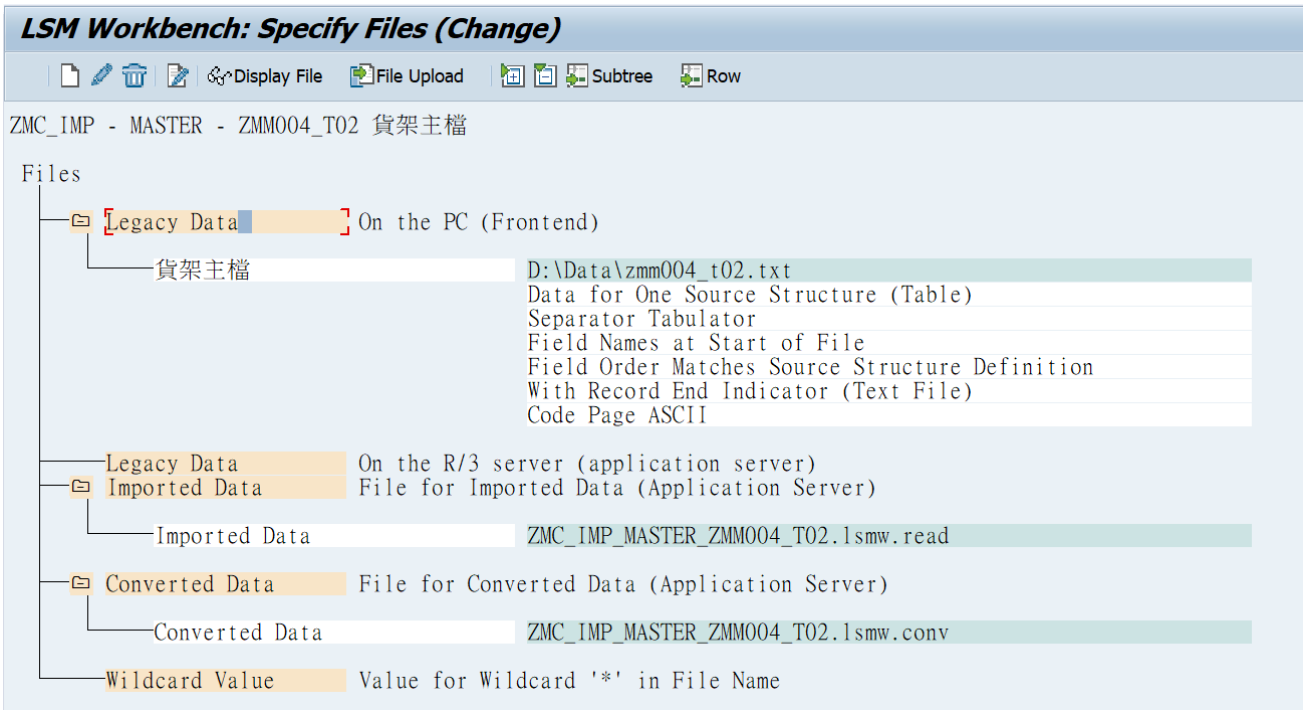


上圖中顯示了自動創建兩個系統檔。其中 Imported Data 設定了導入的資料檔案名；Converted Data 設定了轉換的資料檔案名。這兩個資料檔案都在本機上。

用滑鼠先指定 Legacy Data 行，再用滑鼠點擊創建按鈕（ ），螢幕彈出設定源表檔的對話方塊，需要輸入源表檔案名，及源表檔的屬性設置，全部輸入後如下圖。

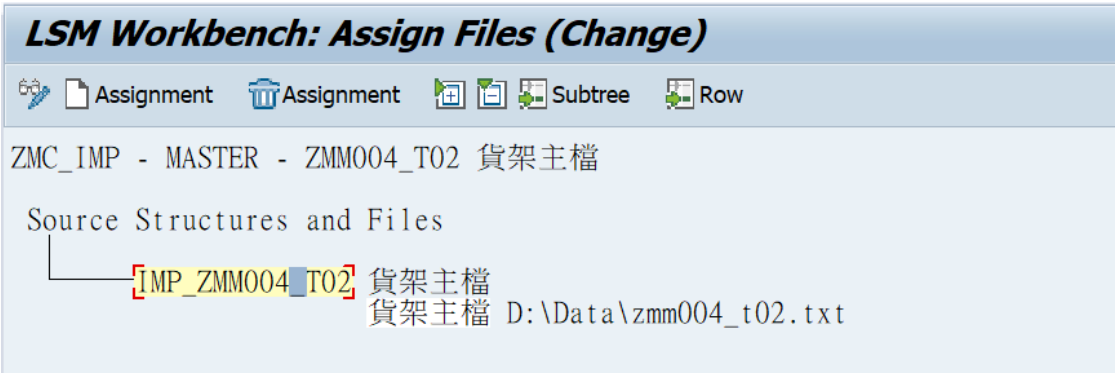


Separators 指定檔的分隔符號，我們的檔的分隔符號是 Tab 鍵，所以選中 Tabulator，可參見步驟 1；Field Name At Start Of File 項指定第一行是否有欄位名，我們的 txt 檔的第一行是欄位名，所以需要選中。Field Order Matches Source Structure Definition 項指定欄位順序是否與源表資料相同，我們進行選中處理。



步驟 11：指定檔

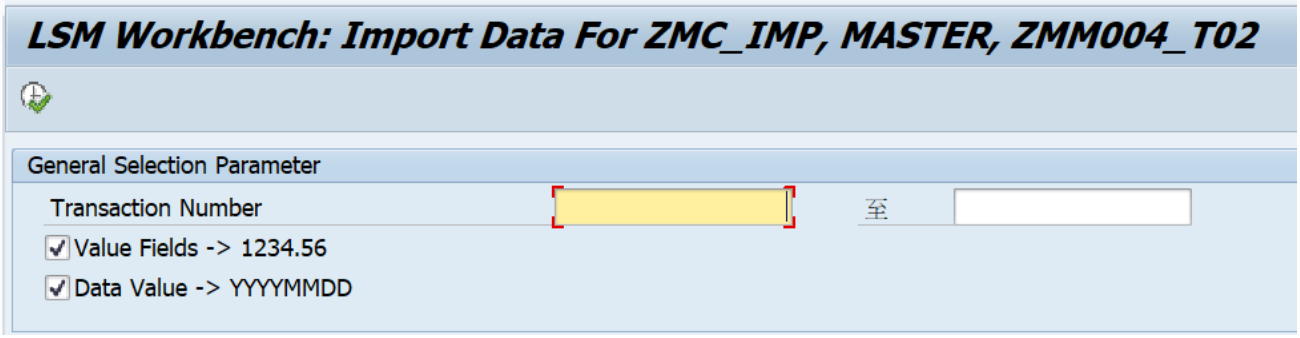
本步驟是指定源表資料結構和對應的資料檔案（.txt）。在分步操作介面用滑鼠按兩下 Assign Files，進入操作介面。如果是顯示狀態，請按修改按鈕。



由於批導入物件只定義了一個源表資料結構，並在上一步驟定義了一個資料檔案，所以系統自動進行了對應處理，此步驟可以不用操作。

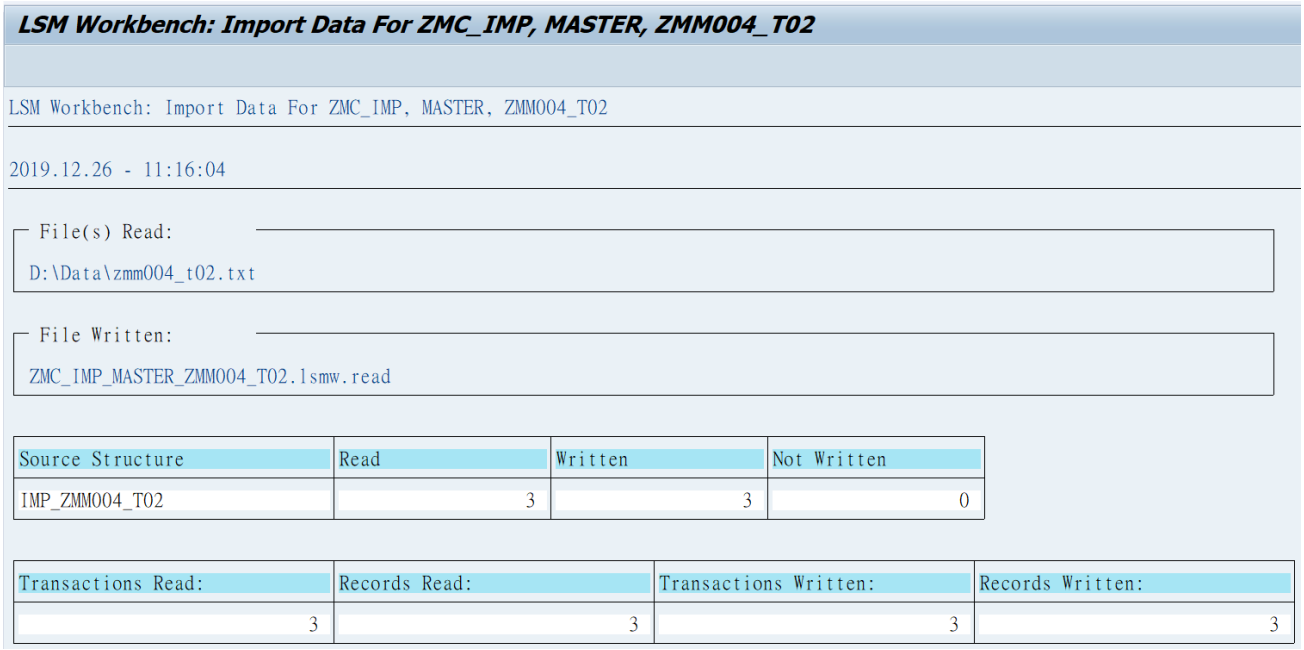
步驟 12：導入數據

此步驟是將源表資料讀取進本機的系統檔，也就是步驟 10 指定源表檔中的 Imported Data 指定的檔。在分步操作介面用滑鼠按兩下 Read Data，進入操作介面，如下圖所示。



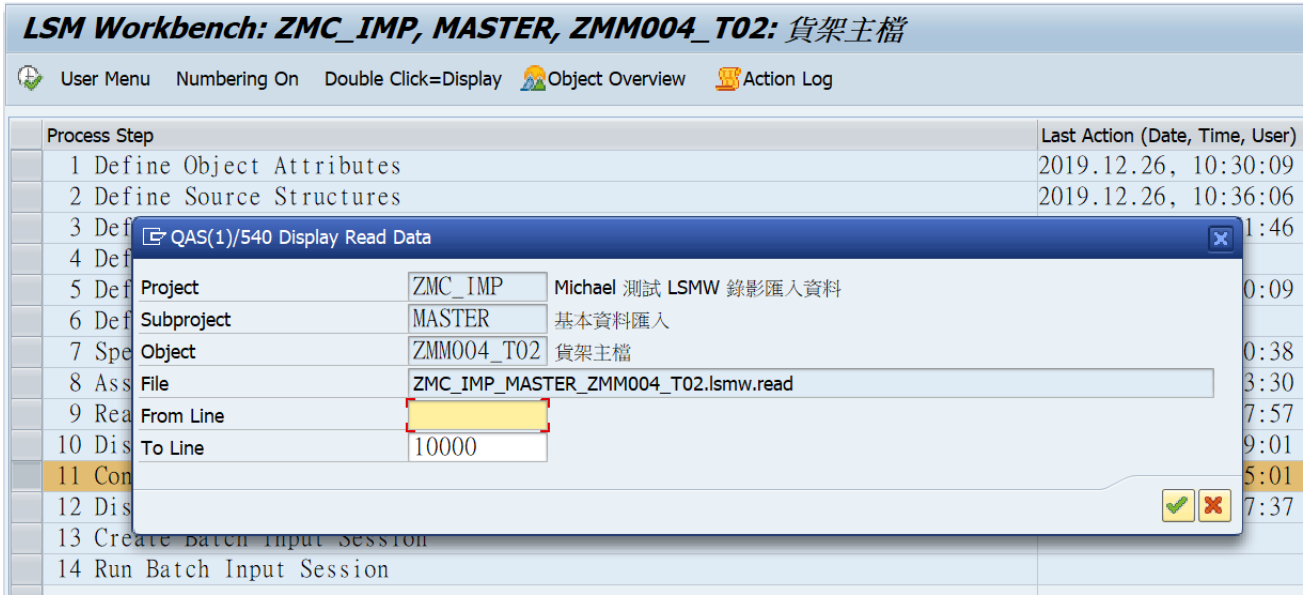
第一行填寫要讀取的起止行數，如不填則全讀取，按執行按鈕（ ）執行。系統彈出一個安全對話方塊，詢問是否授權訪問資料檔案，點擊“允許”按鈕繼續。

執行完畢介面顯示如下圖，表示正確導入了 3 行資料。再按返回按鈕（ ）返回到分步操作介面。

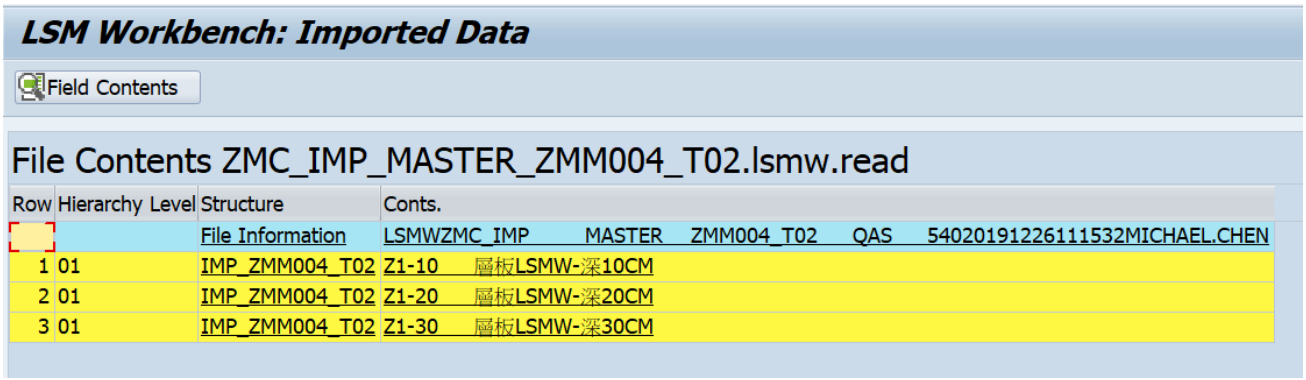


步驟 13：顯示導入的資料

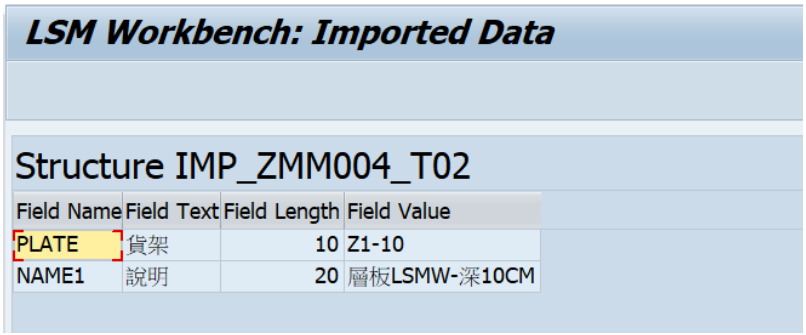
本步驟就是顯示上一步驟導入的資料。在分步操作介面用滑鼠按兩下 Display Read Data，彈出對話方塊，如下圖所示。



From Line 項和 To Line 要求填寫顯示的開始行數和結束行數，如不填寫則顯示全部。按確認按鈕（ ）進入顯示資料介面，如下圖所示。



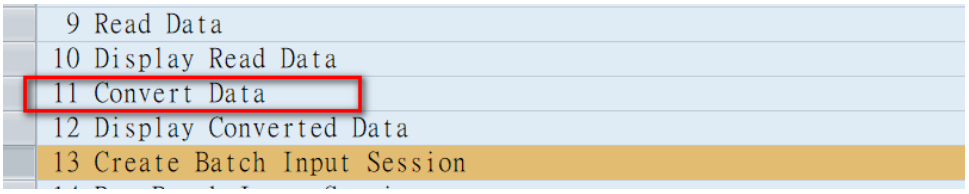
上一步驟導入資料，共有 3 行。用滑鼠按兩下任意一行，比如第 1 行，顯示詳細資訊，如下圖所示。

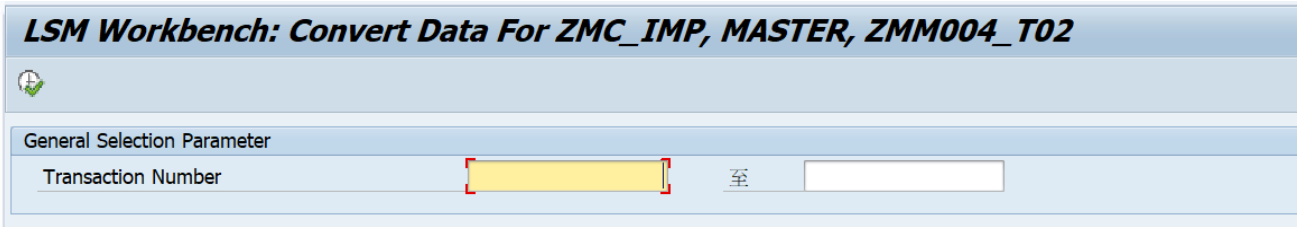


步驟 14：轉換資料

本步驟是將讀進系統檔的資料進行轉換，存放在步驟 10 指定源表檔 Converted Data 指定的轉換檔中。本步驟和下一步驟顯示可以查看轉換是否正確，如不正確可返回到以前步驟進行操作。本步驟操作的資料不會在 SAP 系統中真正執行。

在分步操作介面用滑鼠按兩下 Convert Data，進入操作介面。

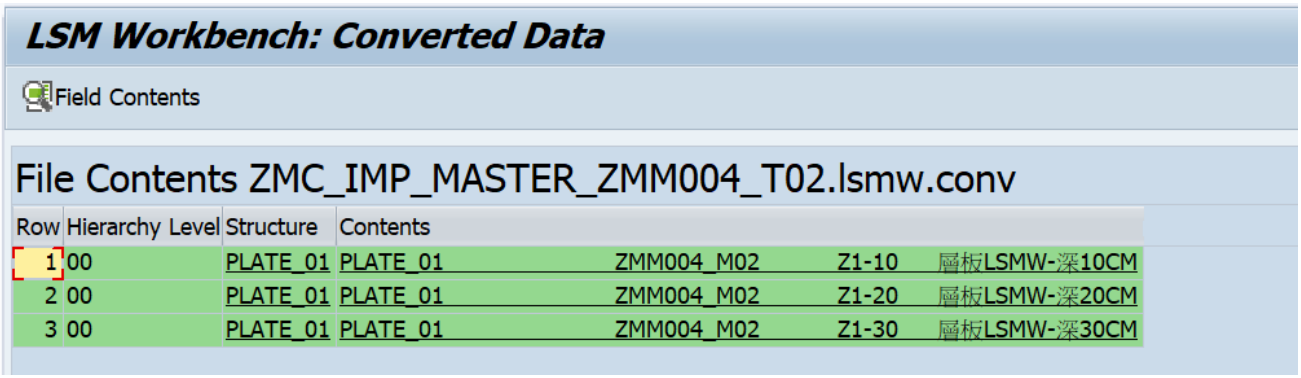


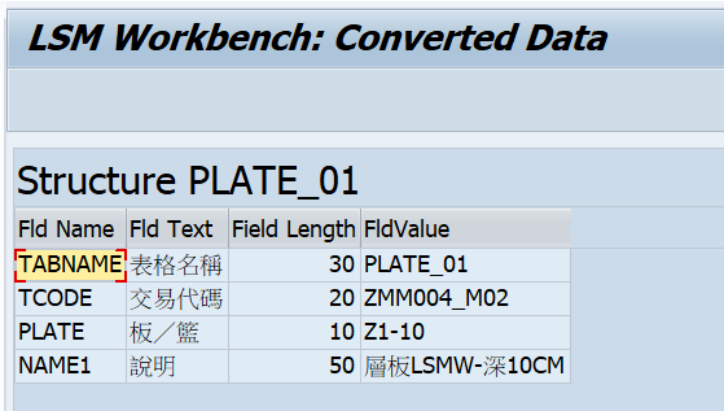


輸入轉換的開始和結束行數，如不填寫則全部轉換。按執行按鈕（ ）。執行完畢螢幕顯示轉換結果，如下圖。

步驟 15：顯示轉換資料

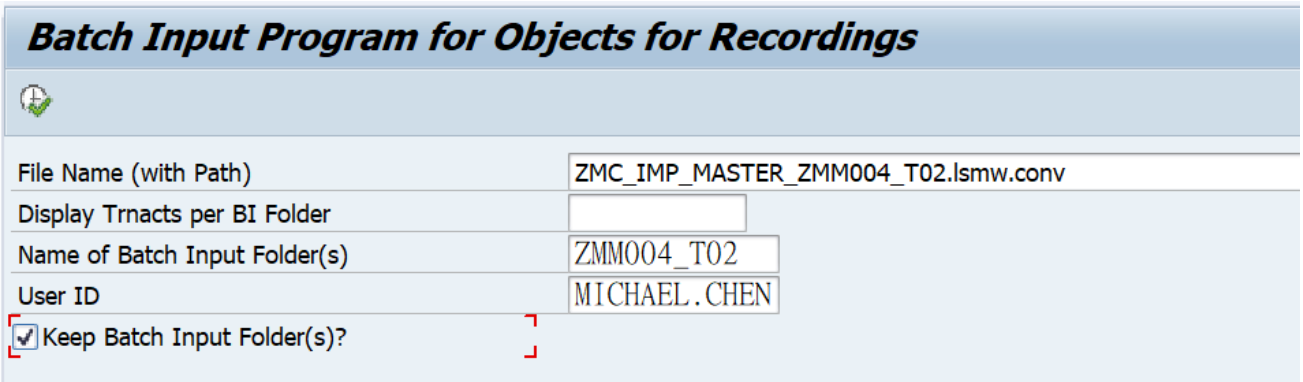
本步驟就是顯示上一步的以預轉換結果。在分步操作介面用滑鼠按兩下 Display Converted Data，彈出對話方塊，如下圖所示。



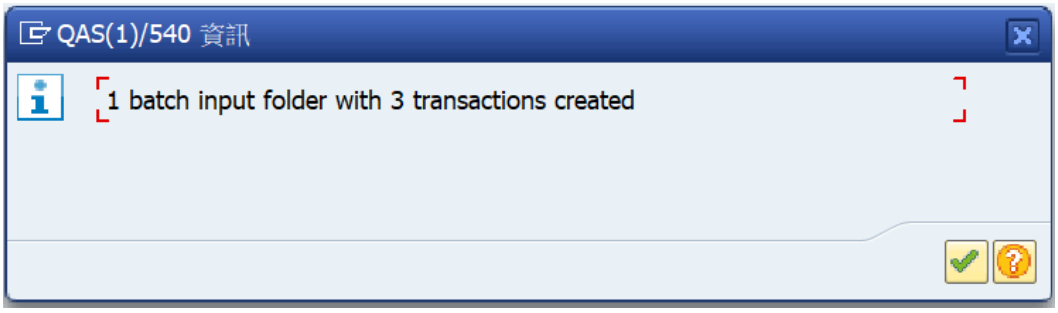


步驟 16：創建轉換任務

此步驟開始實際轉換。本步驟是創建一個轉換任務但不實際轉換，並將轉換的資料存放到 SAP 伺服器端。在分步操作介面用滑鼠按兩下 Create Batch Input Session，進入操作介面，如下圖所示。

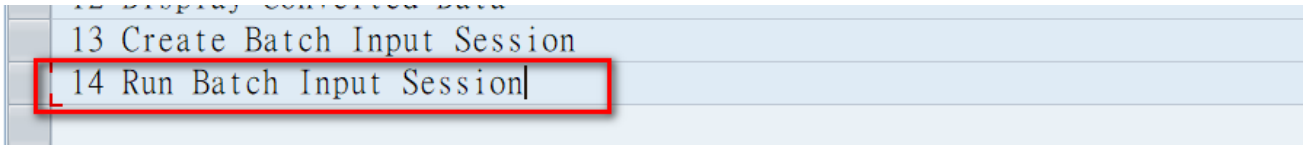


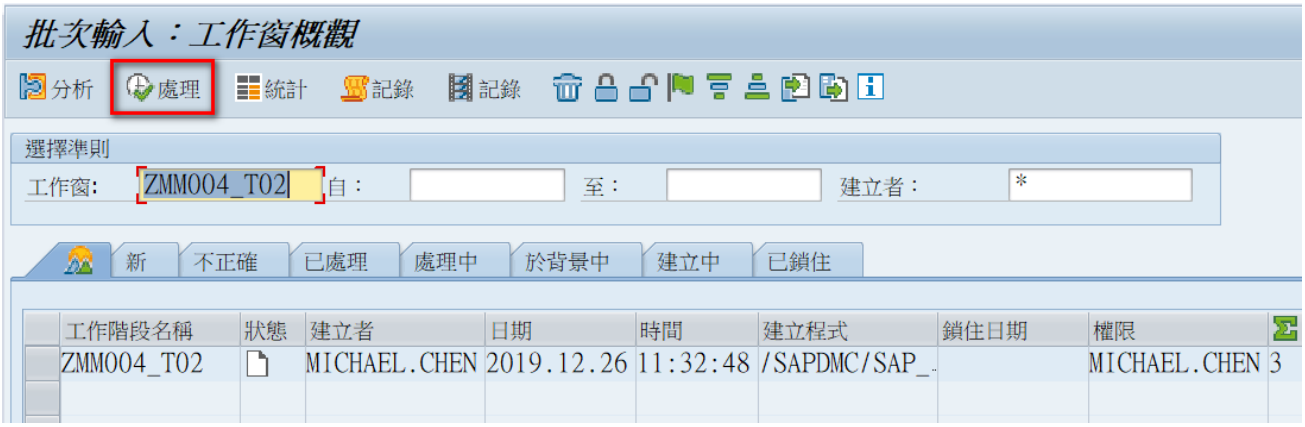
在 Keep Batch Input Folder(s)項打上勾。按執行按鈕（ ），運行後顯示對話方塊，再按確認按鈕（ ）返回到分步操作介面。



步驟 17：執行轉換任務

本步驟進行實際的轉換。除在 LSMW 集成環境操作外，此步驟也可在前臺運行 SM35 進入。在分步操作介面用滑鼠按兩下 Run Batch Input Session，進入操作介面，如下圖所示。





顯示已創建的轉換任務，尚未執行。用滑鼠選中此任務，並按[處理]按鈕，彈出執行選擇對話方塊，如下圖。

對話方塊中，運行模式 Processing Mode 有三個可選項

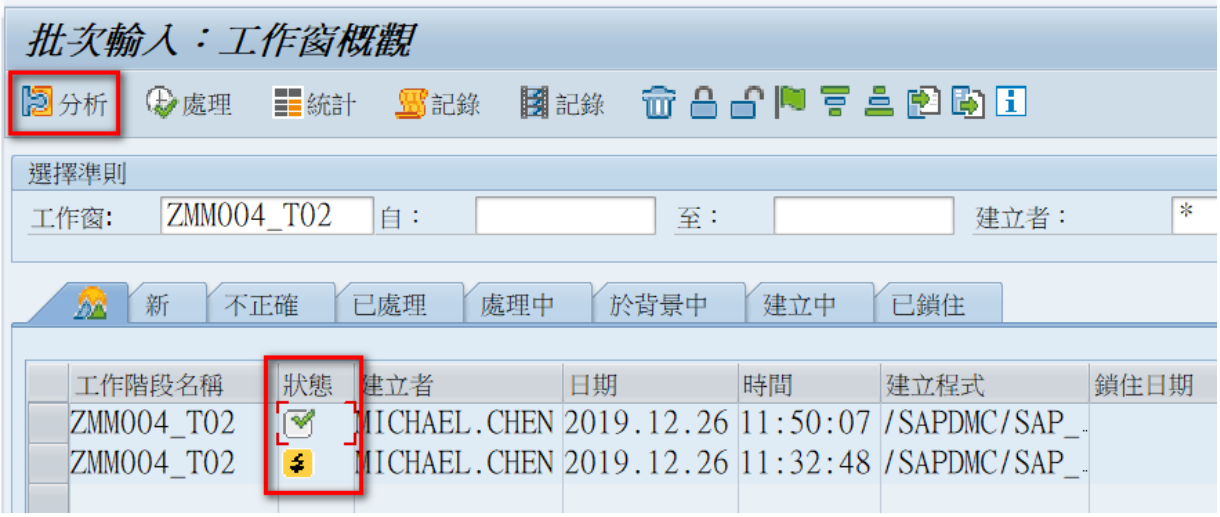
- ✓ 處理 / 前台：每個事物代碼運行在前台，可一步一步運行，可在運行時修改，可看其效果，並可以修改，也可中途退出，但速度慢，一般僅用於測試。
- ✓ 僅顯示錯誤：後台運行，錯誤時顯示到前台。
- ✓ 背景：後台運行，錯誤時也不報出，在全部運行完後可通過查看轉換結果看到錯誤。

本次操作選擇 [背景]，再選中專家方式。按 [處理] 按鈕執行。

由於此次轉換任務是在 SAP 伺服器後台運行，在運行期間 SAP GUI 可退出。

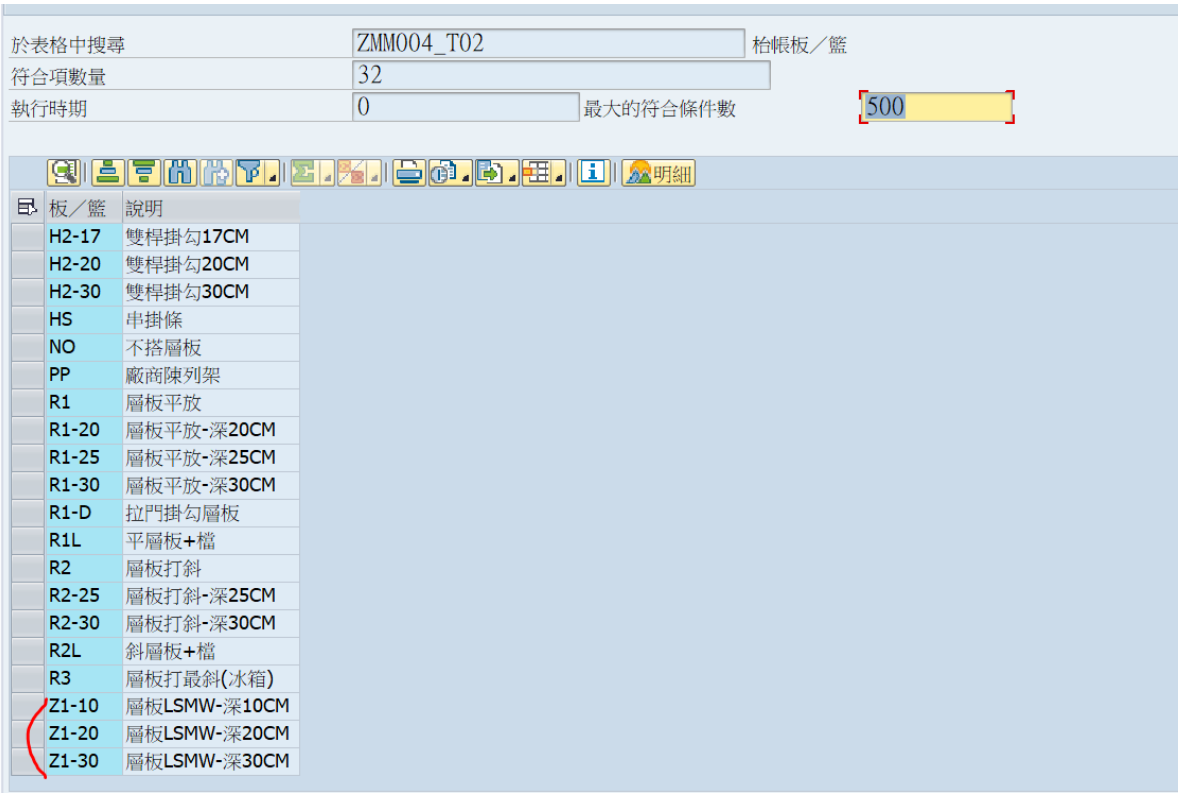
步驟 18：查看執行結果

等任務執行完畢我們可以查看批次處理的結果。也可以在執行過程中查看，當然資料是不完整的，但可以看到已執行部分的情況。和上一步一樣，在分步操作介面用滑鼠按兩下 Run Batch Input Session 再次進入轉換介面。如下圖所示，任務條的狀態會顯示成功或失敗。



用滑鼠選中此任務，按兩下條目或選中條目後點擊 [分析] 按鈕顯示轉換結果。

用 SE16N 確認資料有正確匯入



下圖是模擬錯誤發生的情況，最簡單的方式，是另開一個視窗，開啟同一功能(ZMM004_M02) 並進入修改模式，這樣再執行匯入時就會報錯，資料被鎖定(Locked)。

切換到 [畫面] 的頁籤，可以查看輸入錯誤的模擬畫面，有助於排查問題。

也可以查看整個匯入過程的 Log 紀錄。

配合這 2 個頁籤的查詢，基本上可以排除大部分因操作問題引起的錯誤，其他就是熟悉度的問題了。

- 整個流程自己跑過一遍後，雖然看起來步驟非常繁瑣，但是實際上都是一些必要的設定。根據以往的經驗，這個流程的確已考慮了絕大多數的批次匯入資料的需求，對於開帳時的主檔資料匯入的需求，應該是沒有問題的。當然，再更進一步的需求是匯入的欄位細項檢查條件、依 A 欄位的值給定 B 欄位等進階需求，這個日後再深入研究。

  至於單據類的匯入，會牽涉到多表資料、過帳、表頭表身同值等需求，我不確定是否能用本功能執行，此處先按下不表，先求能熟悉運用本功能匯入資料即可。
- 提醒：
  - ✓ 步驟 1-12 都是設定，一旦設定正確，就可以反覆執行，除非設定內容有變更。
  - ✓ 如果要執行批次匯入，每一次都必須先執行 Step 13 Create Batch Input Session 先創建一個 Session，依前面的說明設定相關參數。
  - ✓ 然後進入到 Step 14 Run Batch Input Session，先選擇剛新創建的 Session，再點擊 [處理] 按鈕，才會真正開始執行。
  - ✓ 過一段時間後，再重新進入系統(或執行 T-Code=SM35) 查詢結果。

Process Step	Last Action (Date, Time, User)
1 Define Object Attributes	2019.12.26, 10:30:09 h, MI
2 Define Source Structures	2019.12.26, 10:36:06 h, MI
3 Define Source Fields	2019.12.26, 10:51:46 h, MI
4 Define Structure Relations	
5 Define Field Mapping and Conversion Rules	2019.12.26, 11:00:09 h, MI
6 Define Fixed Values, Translations, User-Defined Routines	
7 Specify Files	2019.12.26, 11:10:38 h, MI
8 Assign Files	2019.12.26, 11:13:30 h, MI
9 Read Data	2019.12.26, 11:17:57 h, MI
10 Display Read Data	2019.12.26, 11:30:38 h, MI
11 Convert Data	2019.12.26, 11:25:01 h, MI
12 Display Converted Data	2019.12.26, 11:27:37 h, MI
13 Create Batch Input Session	2019.12.26, 12:17:49 h, MI
14 Run Batch Input Session	2019.12.26, 12:25:03 h, MI

第 15 章 BDC 批次處理使用方法

https://www.fenginfo.com/45.html

Session 1：說明

SAP 系統有多種批次處理工具，例如 LSMW、SCAT、BDC 等，各有側重點，本文專門介紹 BDC 的應用操作。BDC 是 Batch Data Conversion 的簡稱，對應的事物碼是 SHDB，是一款輕量化的批次處理工具。

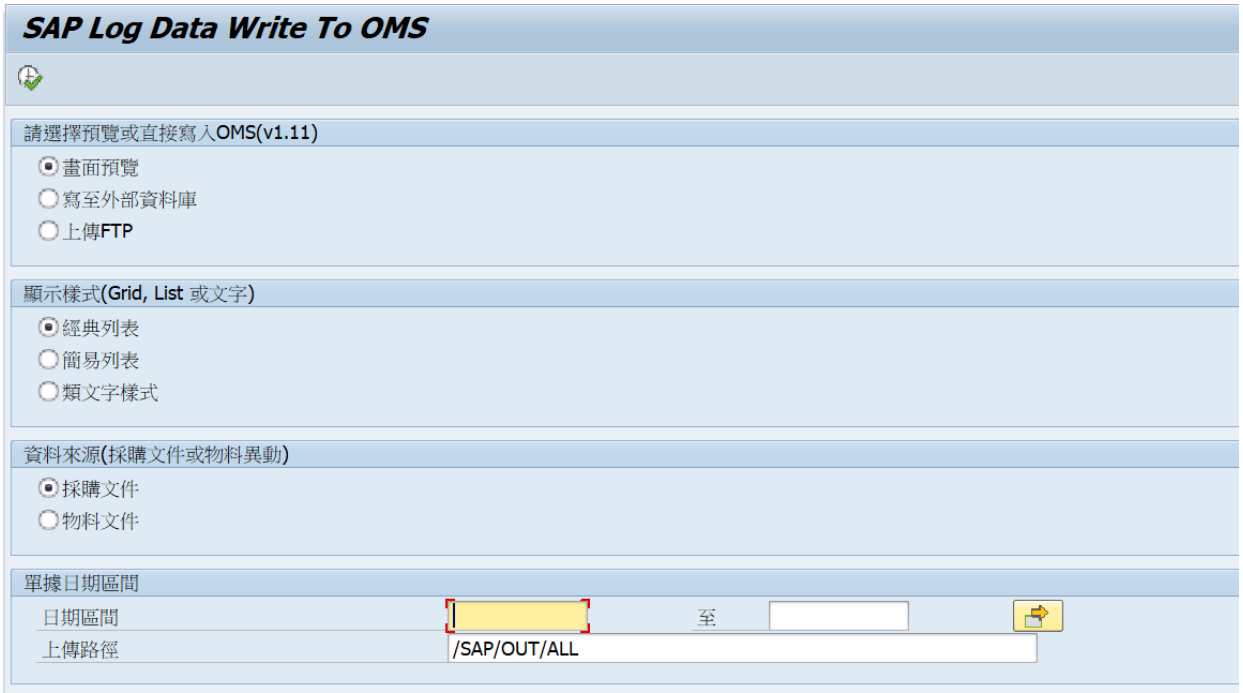
運用 SHDB，一般通過對業務操作（指定事物碼）進行錄影生成腳本，如果重複次數少，可以手工修改腳本中欄位的值再次執行；對於大量業務操作，則採用匯出腳本範本，加工成批量執行腳本後，再導入執行。

Session 2：操作

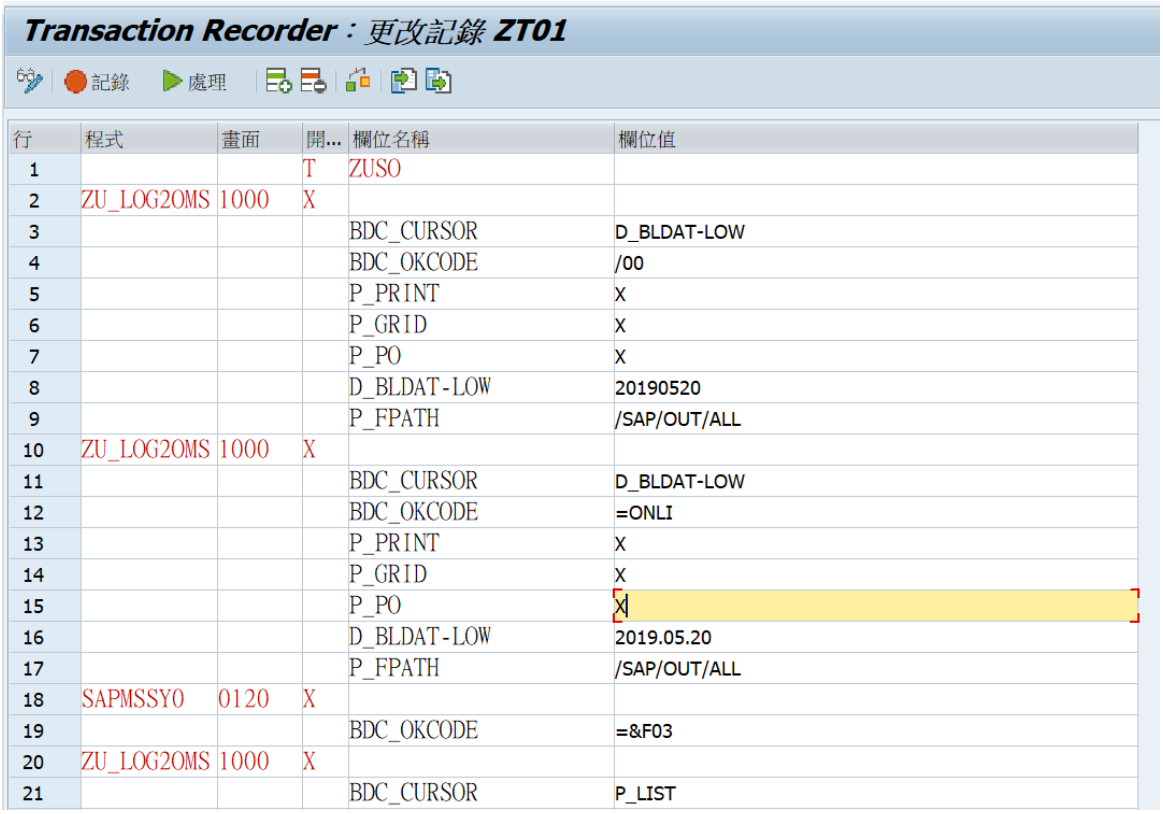
本文講解 SHDB 操作，使用 T-Code=ZUSO 為範例，簡單的說明錄製即撥放的完整操作的過程。

- 在 SHDB 環境中創建一個記錄，指定 T-Code 為 ZUSO

- 按 [開始記錄]，會直接進入 ZUSO 程式，就像平常一樣操作，SAP 會將過程記錄下來，所以盡量避免不必要的鍵盤的滑鼠的操作。



- 操作完後，可以逐步退回到 SHDB 的主畫面，可以看到 SAP 的紀錄內容，此時尚未存檔，如果變動不大，內容也可以直接修改。

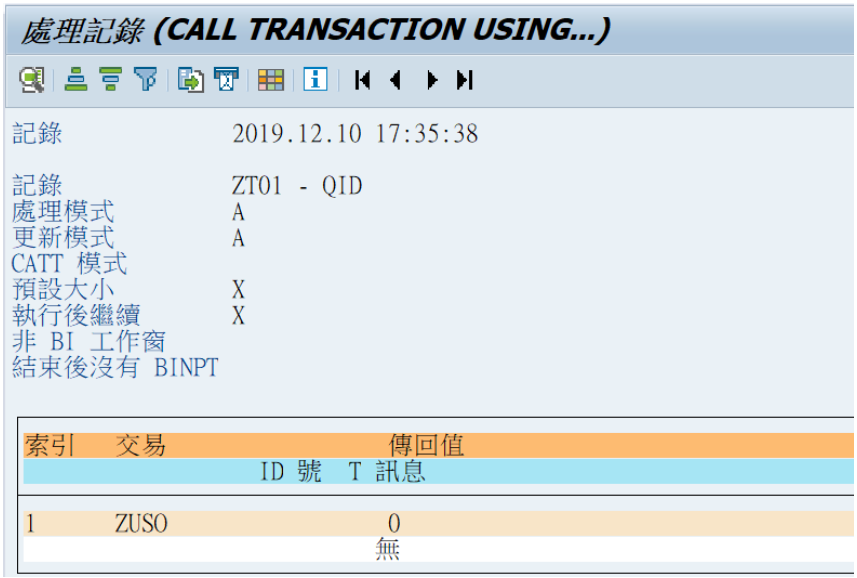


行	程式	畫面	開...	欄位名稱	欄位值
1			T	ZUSO	
2	ZU_LOG2OMS	1000	X		
3				BDC_CURSOR	D_BLDAT-LOW
4				BDC_OKCODE	/00
5				P_PRINT	X
6				P_GRID	X
7				P_PO	X
8				D_BLDAT-LOW	20190520
9				P_FPATH	/SAP/OUT/ALL
10	ZU_LOG2OMS	1000	X		
11				BDC_CURSOR	D_BLDAT-LOW
12				BDC_OKCODE	=ONLI
13				P_PRINT	X
14				P_GRID	X
15				P_PO	X
16				D_BLDAT-LOW	2019.05.20
17				P_FPATH	/SAP/OUT/ALL
18	SAPMSSY0	0120	X		
19				BDC_OKCODE	=&F03
20	ZU_LOG2OMS	1000	X		
21				BDC_CURSOR	P_LIST

- 對於大量的資料，一般採用匯出範本腳本，生成批量腳本，再重新導入加以完成。匯出請點擊匯出鍵( )，將錄屏的腳本生成文字檔存到本地電腦上。然後經由手工整理好完整的要重複執行的內容後，再將該文檔重新匯入，即可達到重複執行的目的。一般而言，這個作業是用來匯入多筆資料，先人工建立一筆資料，再透過外部整理的作業，然後達到批次匯入多筆資料的效果。

- 執行 [處理] button，即會開始一建立的腳本逐步執行。除了期初開帳會使用本作業，另外測試案例(Test Case) 也是使用本功能的大戶，尤其是專案程式在開發完後，要不斷的重複測試，此時如果有編寫好的 Test Case，會省掉測試人員相當多的時間。

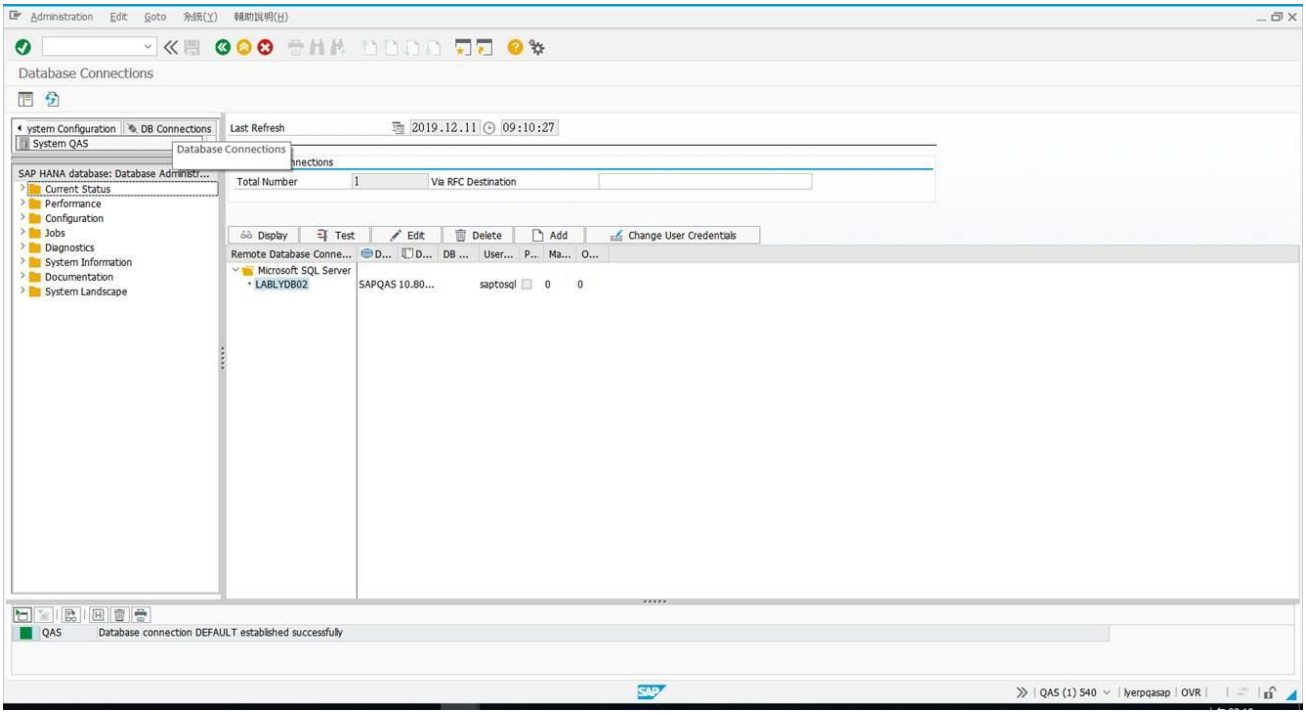
- 執行後會產生一張分析報表



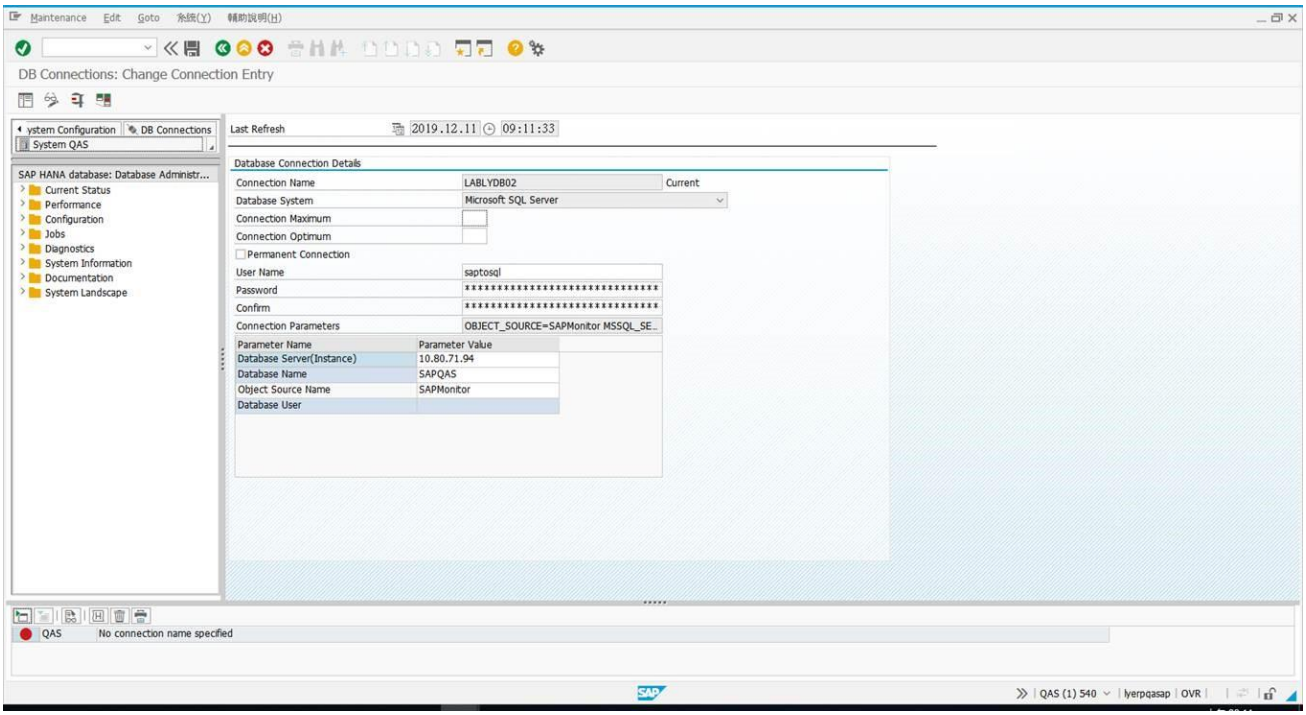
➢ 實際上，還有另外一種方式可以交換資料，當然這種方式並不是 SAP 的建議作法，而且可能會有 Licence 的問題，但本文只論作法，不論其他，所以仍稍加以說明。

1. 首先，請 Basis 建立一個類似 JDBC(ODBC) 的對外指定資料庫的設定連線 T-Code=DB02，並設定一個連線代號，如 wms_uat_db_conn。

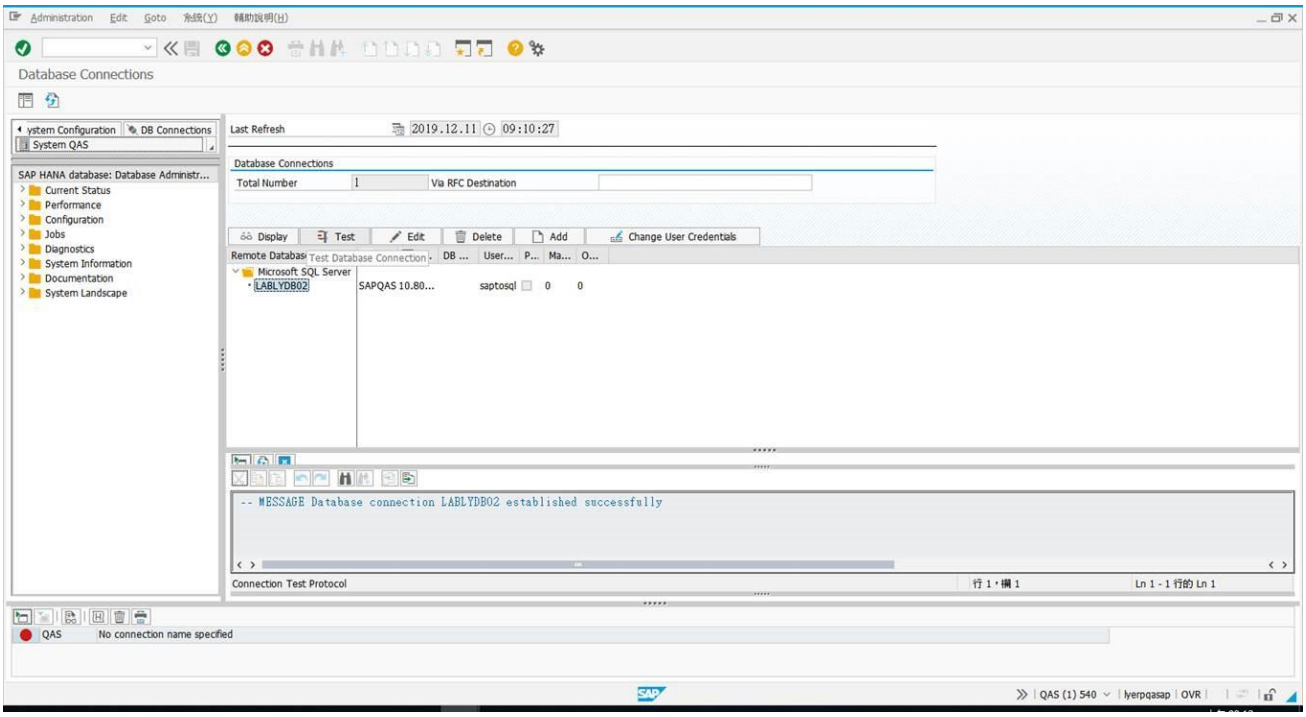
✓ **選擇 DB Connections**



✓ **Add Connectio,依連線需求填入欄位**



✓ **建立完成後可執行 Test,測試連線**



2. 在 ABAP 程式中，先建立連線，然後使用 EXEC 呼叫 Native SQL 的方式，直接讀寫到外部資料庫，完整範例可參考 ZU_LOG2OMS。

```
*&---------------------------------------------------------------------
*&      Form  create_connection
*&---------------------------------------------------------------------
FORM create_connection RAISING cx_sy_native_sql_error.
 EXEC SQL.
   CONNECT TO :dblink
 ENDEXEC.
ENDFORM.                    "create_connection
*&---------------------------------------------------------------------
*&      Form  close_connection
*&---------------------------------------------------------------------
FORM close_connection RAISING cx_sy_native_sql_error.
 EXEC SQL.
   DISCONNECT :dblink
 ENDEXEC.
ENDFORM.                    "close_connection

******* Variant ******************************************************
CONSTANTS dblink TYPE dbcon-con_name VALUE 'LABLYDB02'."中介 DB
```

```
**** 主程式(部分程式碼)
START-OF-SELECTION.
  PERFORM create_connection.
  PERFORM process_data.
  COMMIT WORK AND WAIT.
  PERFORM close_connection.

*&---------------------------------------------------------------------
*&       Form  PROCESS_DATA
*&---------------------------------------------------------------------
FORM process_data .
  "取得資料
  PERFORM get_data.

*** 以下是呼叫的程式片段
LOOP AT itab ASSIGNING <lfs_data>.
  TRY.
    CASE 'X'.
     WHEN p_po.
       EXEC SQL.
         INSERT INTO PO_LOG_B_SAP ( BILLDATE,EKPO_WERKS,EKKO_BSART,QTY,BodyRecNo ) VALUES
         ( :<LFS_DATA>-BLDAT, :<LFS_DATA>-BWART, :<LFS_DATA>-WERKS, :<LFS_DATA>-QTY, :<LFS_DATA>-RECNO )
       ENDEXEC.
     WHEN p_inv.
       EXEC SQL.
         INSERT INTO INV_LOG_B_SAP ( BILLDATE,MSEG_BWART,MSEG_WERKS,QTY,BodyRecNo ) VALUES
         ( :<LFS_DATA>-BLDAT, :<LFS_DATA>-BWART, :<LFS_DATA>-WERKS, :<LFS_DATA>-QTY, :<LFS_DATA>-RECNO )
       ENDEXEC.
    ENDCASE.
    lv_ok = lv_ok + 1.

   CATCH cx_sy_native_sql_error INTO lcx_sy_native_sql_error.
       lv_err = lv_err + 1.
       CONTINUE.
   ENDTRY.
 ENDLOOP.
```

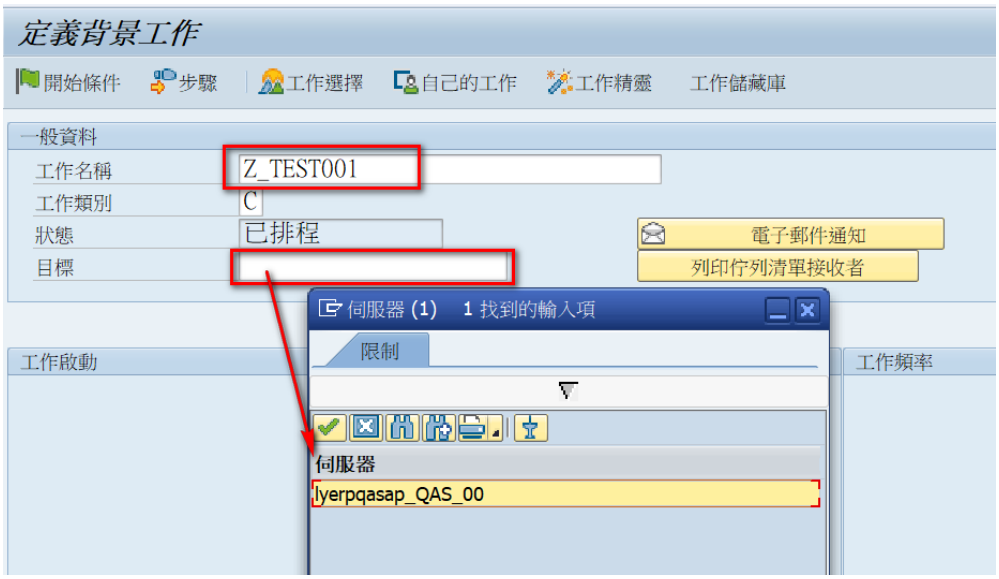
第 16 章 SAP 工作排程

在 ERP 系統中，有些需求是希望能定期(週期性)的執行某個作業，這個作業有可能是匯出或匯入資料，也可能是檢查某些異常數據，然後自動更新某些內容，再發 mail 通知負責人員。當有這種需求時，在 SAP 就會利用 SM36 來設定，這和訊光的 Scheduling 程式是相同的用途。SM36 本身只是個啟動器，並沒有任何實際的商業邏輯，它可以和任何 ABAP 的程式配合(甚至是外部程式，但是我沒實際用過)，所以要作什麼事情，就完全由它所呼叫的程式來完成。至於呼叫程式要傳入的參數，就由該程式的變式設定來完成，一般最常用的通常是日期，稍後會加以說明。

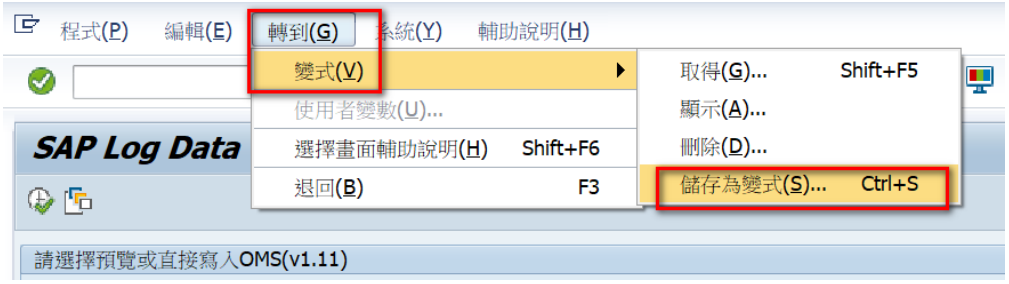
可參考 \\10.0.8.14\sap 導入專案\專案公用資料夾\Phase 5 上線與支援\5.1 上線問題處理\介面資料轉入檢查\商品異動作業處理流程.wmv

SAP Background Job T-Code=SM36

➢ 給定 Job Name 及 Server 資訊

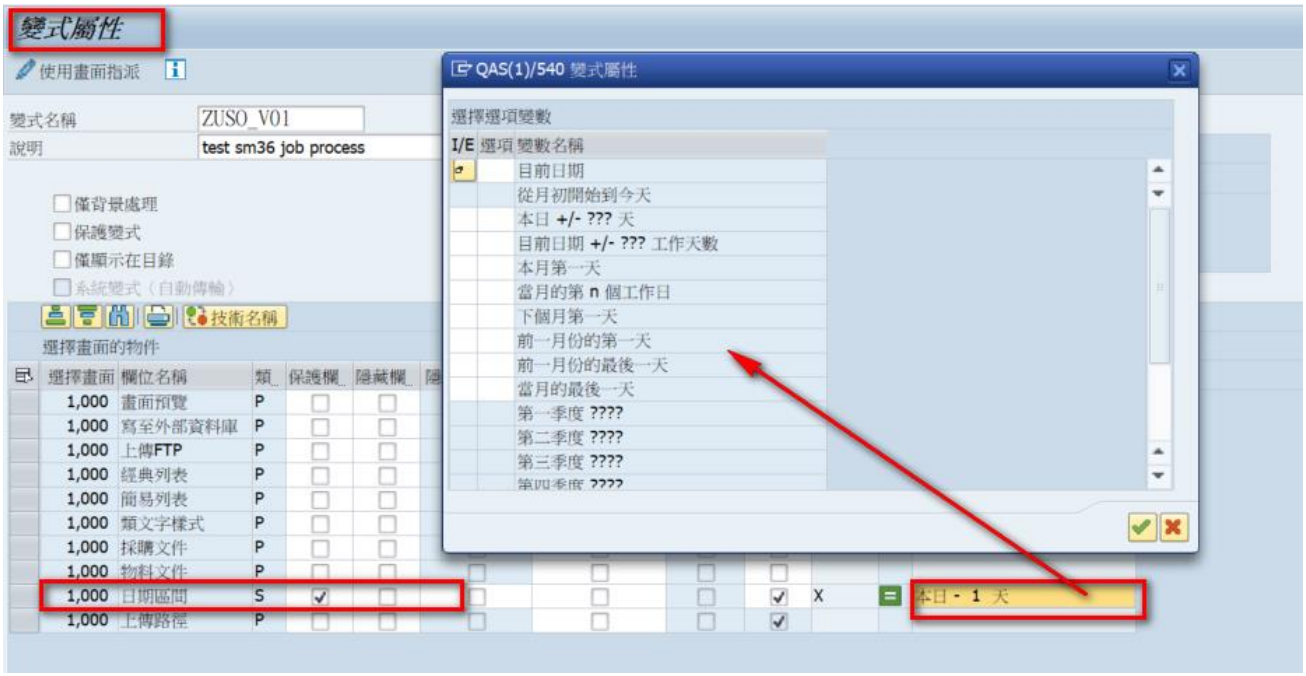


➢ **設定的重點在變式。以 ZUSO 程式為例，操作方式如下**

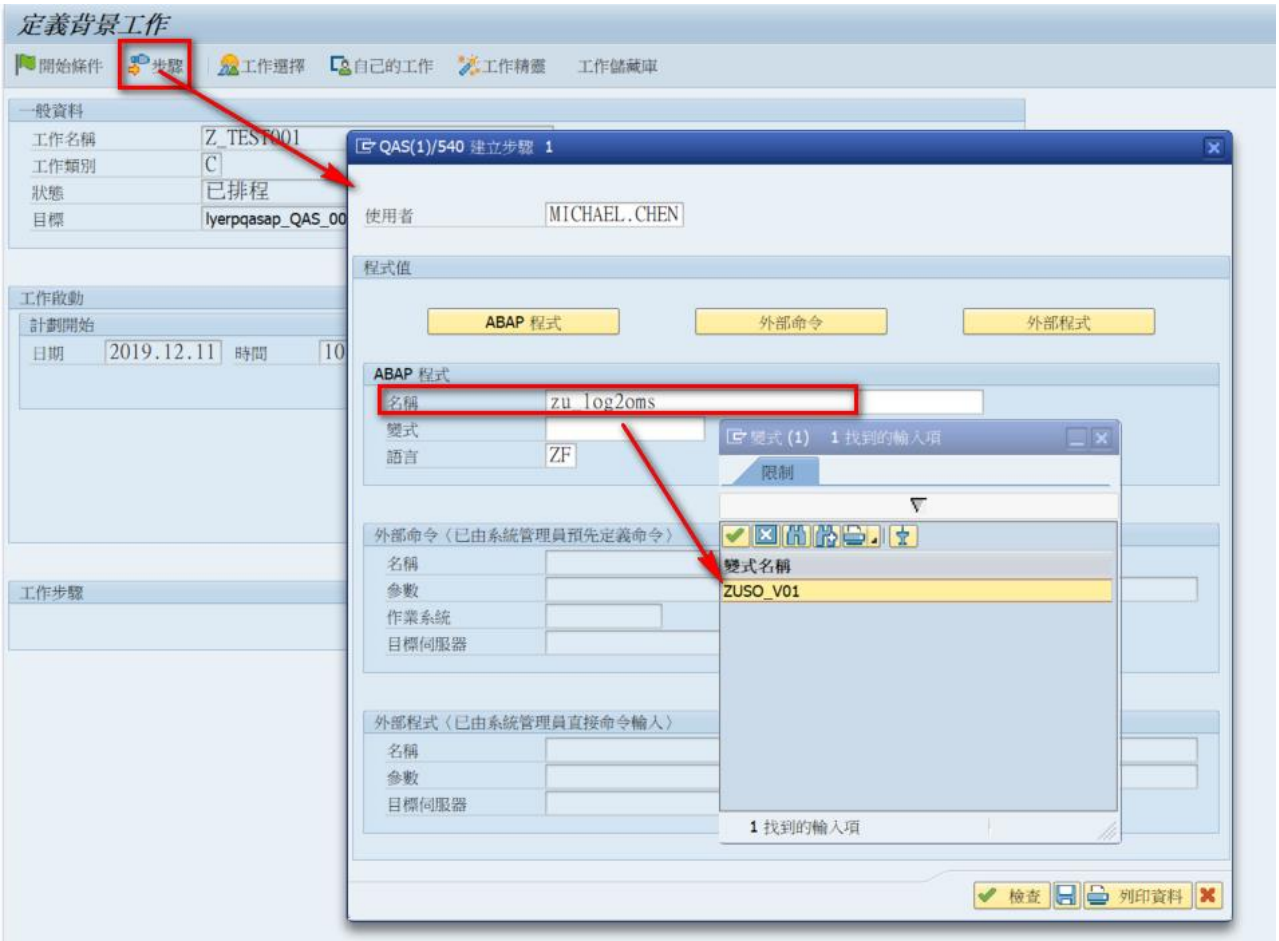


一但有設定變式，在 [執行] button 旁，就會多一個查詢變式的 button。

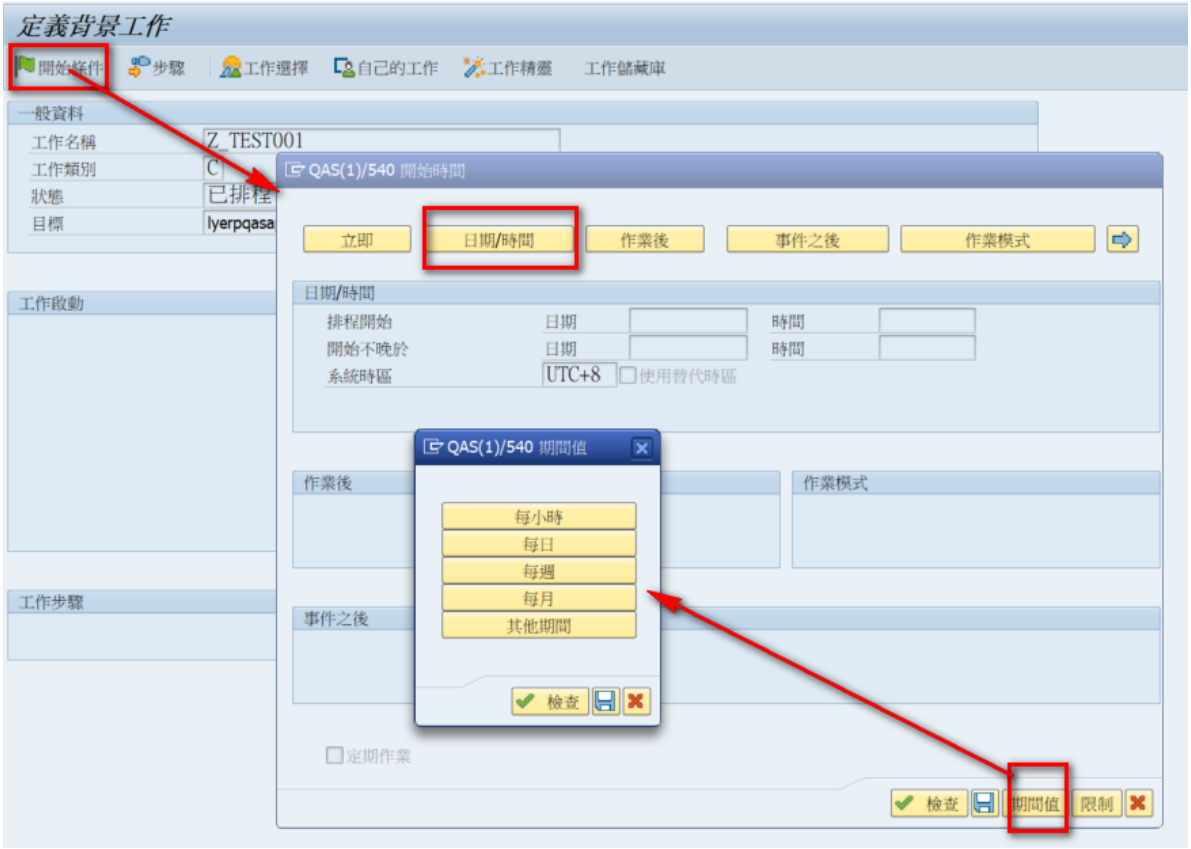
如果定期執行的內容和日期有關，請參考下面的畫面設定，即可於執行時，自動帶入指定的日期



➢ 設定要連線的 ABAP 程式相關資訊



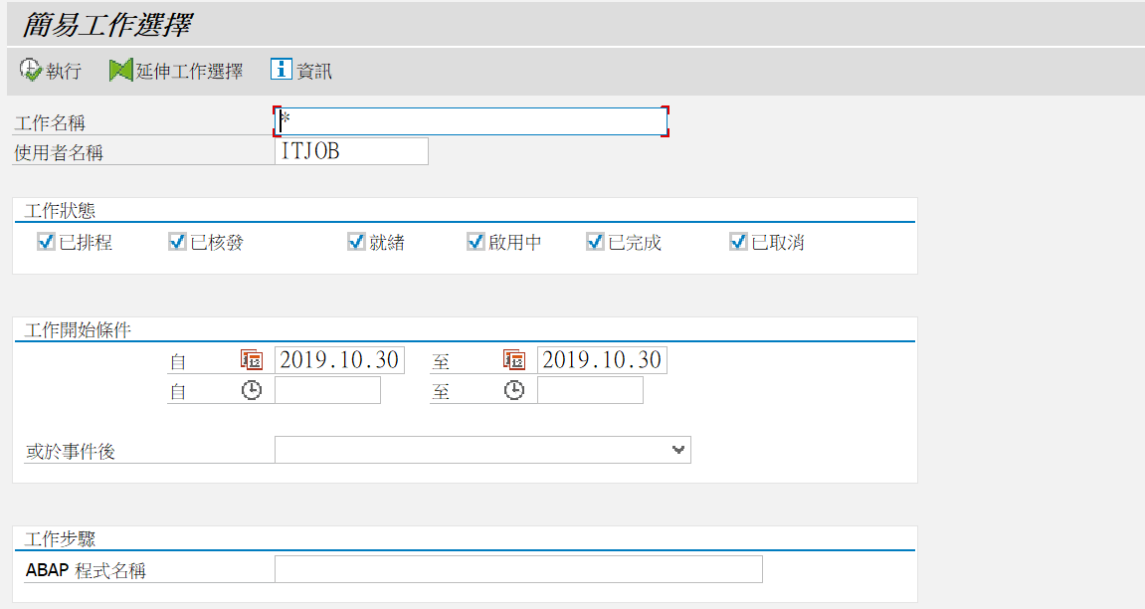
➢ 設定定期執行的條件



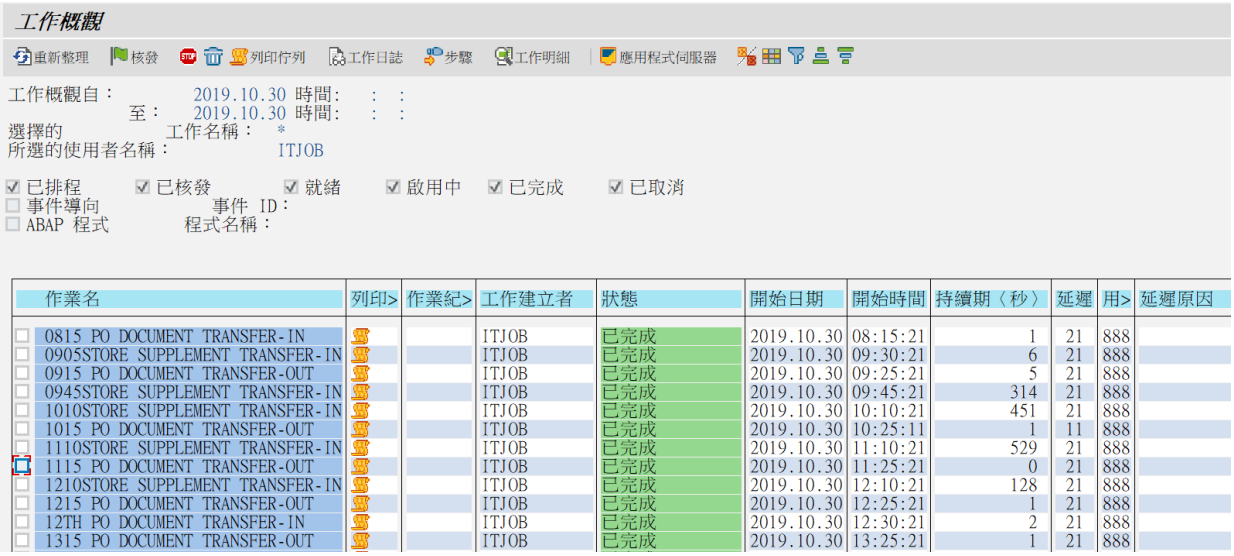
➢ 存檔後會顯示下面畫面

數	程式名稱 / 命令	程式類型	佇列清單	參數	使用者	語言
1	ZU_LOG2OMS	ABAP		ZUSO_V01	MICHAEL.CHEN	ZF

➢ T-Code=SM37 可以查詢目前有哪些排程，也可以按 [工作選擇] 進入



執行查詢後，會列出查詢區間的 Job 狀況



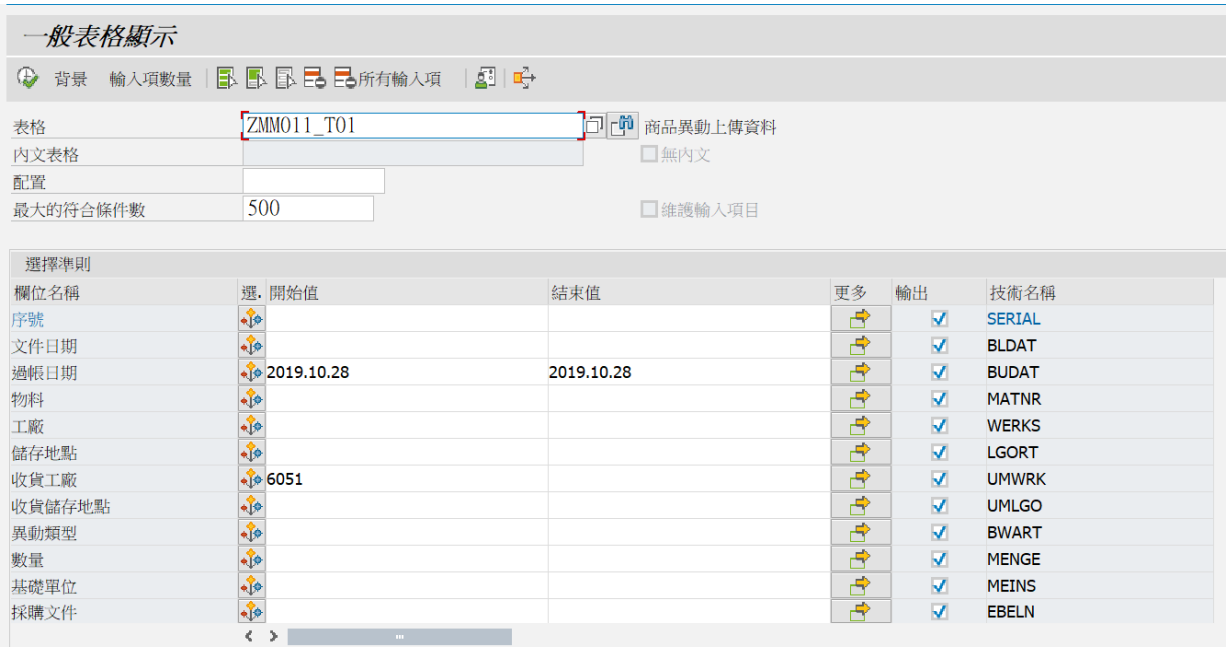
作業名	列印>	作業紀>	工作建立者	狀態	開始日期	開始時間	持續期（秒）	延遲	用>	延遲原因
0815 PO DOCUMENT TRANSFER-IN			ITJOB	已完成	2019.10.30	08:15:21	1	21	888	
0905STORE SUPPLEMENT TRANSFER-IN			ITJOB	已完成	2019.10.30	09:30:21	6	21	888	
0915 PO DOCUMENT TRANSFER-OUT			ITJOB	已完成	2019.10.30	09:25:21	5	21	888	
0945STORE SUPPLEMENT TRANSFER-IN			ITJOB	已完成	2019.10.30	09:45:21	314	21	888	
1010STORE SUPPLEMENT TRANSFER-IN			ITJOB	已完成	2019.10.30	10:10:21	451	21	888	
1015 PO DOCUMENT TRANSFER-OUT			ITJOB	已完成	2019.10.30	10:25:11	1	11	888	
1110STORE SUPPLEMENT TRANSFER-IN			ITJOB	已完成	2019.10.30	11:10:21	529	21	888	
1115 PO DOCUMENT TRANSFER-OUT			ITJOB	已完成	2019.10.30	11:25:21	0	21	888	
1210STORE SUPPLEMENT TRANSFER-IN			ITJOB	已完成	2019.10.30	12:10:21	128	21	888	
1215 PO DOCUMENT TRANSFER-OUT			ITJOB	已完成	2019.10.30	12:25:21	1	21	888	
12TH PO DOCUMENT TRANSFER-IN			ITJOB	已完成	2019.10.30	12:30:21	2	21	888	
1315 PO DOCUMENT TRANSFER-OUT			ITJOB	已完成	2019.10.30	13:25:21	1	21	888	

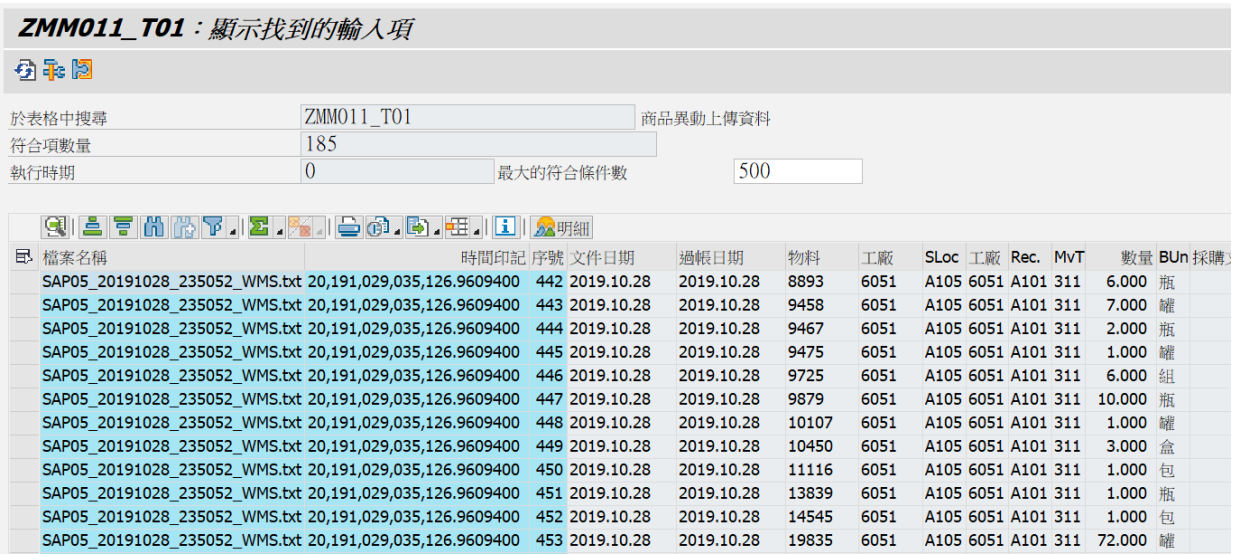
➢ 上面的設定完成後，就會在設定的日期/時間自動執行，其他的工作，就是 ABAP 程式的負責範圍，以三商家購目前的情況看來，目前有設定的 Job 參考如下：

作業名	列印>	作業紀>	工作建立者	狀態	開始日期	開始時間	持續期（秒）	延遲	用>
ZMM011_F01_1			ITJOB	已核發			0	0	888
ZMM011_F01_1			ITJOB	已核發			0	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	09:32:06	80	1	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	11:02:10	51	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	12:32:21	117	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	14:01:10	87	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	14:10:19	101	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	15:04:02	142	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	16:45:00	139	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	17:15:09	33	9	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	20:01:51	15	0	888
ZMM011_F01_1			ITJOB	已完成	2019/12/10	20:09:03	11	0	888
ZMM014_F01			AUSTIN.SU	已完成	2019/12/10	16:54:09	1	0	888
ZS101 20191210 AM RETRY			ITJOB	已完成	2019/12/10	10:40:08	45	8	888
ZS641 20191210 AM RETRY			ITJOB	已完成	2019/12/10	10:20:08	427	8	888
ZSTATUS			AUSTIN.SU	已完成	2019/12/10	18:10:46	85	0	888

➢ 查看匯入錯誤的 error log，利用 T-Code=SE16N 查相關 Table 的內容，不同的匯入功能會寫到不同的 Error Log Table，以下整理一個參考的列表(功能有 Log 檔)，基本上應該是在規格書上紀錄

Error Log Table	功能說明	關聯 T-Code
ZMM003_T06	OPL 備貨系統 FTP import log: MM003:轉入 門市補貨 欄位	ZMM003* 等
ZMM007_T02	物料主檔建立處理狀態	ZMM007
ZMM010_T01	FTP import log: MM010: 轉入 銷售變價 欄位	ZMM010
ZMM010_T02	FTP import log: MM010: 轉入 採購變價欄位	ZMM010
ZMM011_T01	商品異動上傳資料	ZMM011
ZMM013_T01	採購文件轉入_表頭	ZMM013
ZMM036_T02	商品採購資料轉入	ZMM036





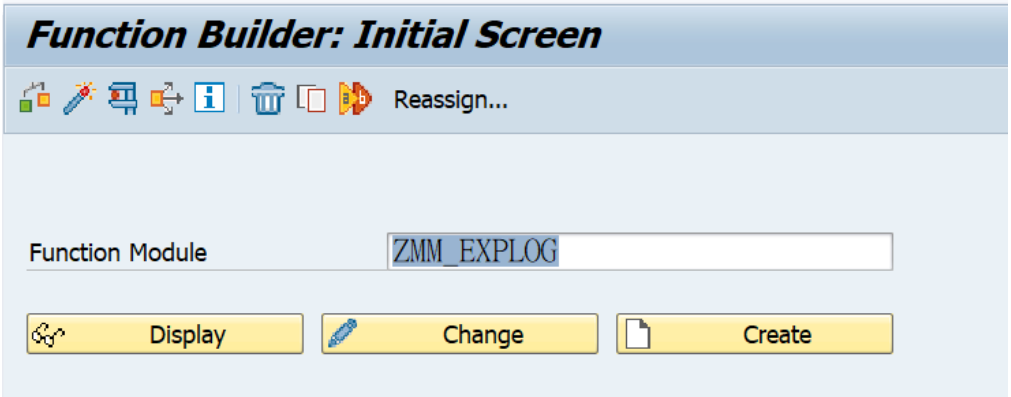
第 20 章 一步一步教你寫 SAP RFC 程式

https://blog.csdn.net/szlaptop/article/details/8093033
https://blog.csdn.net/szlaptop/article/details/8059480
https://blog.csdn.net/szlaptop/article/details/8678905
https://www.cnblogs.com/VerySky/articles/2137935.html

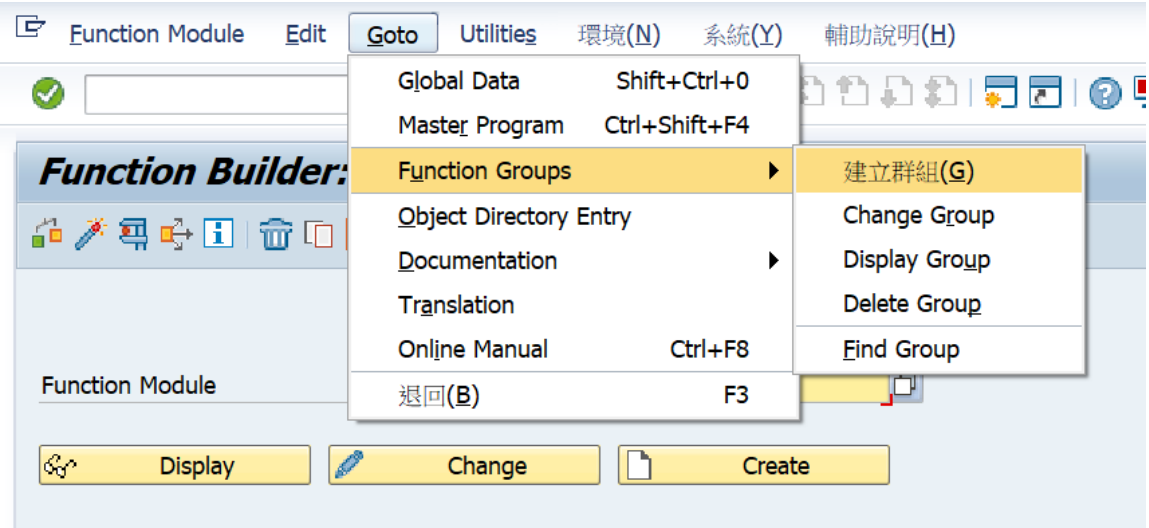
前面我們已經用多個實例簡單介紹過 ABAP 程式開發的基本概念。在撰寫 ABAP 的過程中，SAP 提供了非常豐富的共用程式庫供程式設計師呼叫使用。只要幾行程式就可以執行非常豐富的功能，這是 ABAP 非常重要的核心重點。除了系統提供的 RFC，SAP 也允許我們自行開發 RFC，如此我們就可以建立自己的共用程式庫，當我們有很多的客製程式要開發時，就應該先逐步建立自己的共用程式以建立一致的開發標準。

接下來我們就以一支簡單的程式，寫一支取得採購文件、物料文件匯入異常的資料內容。

- 執行 T-Code=SE37，進入 RFC 的開發主頁面



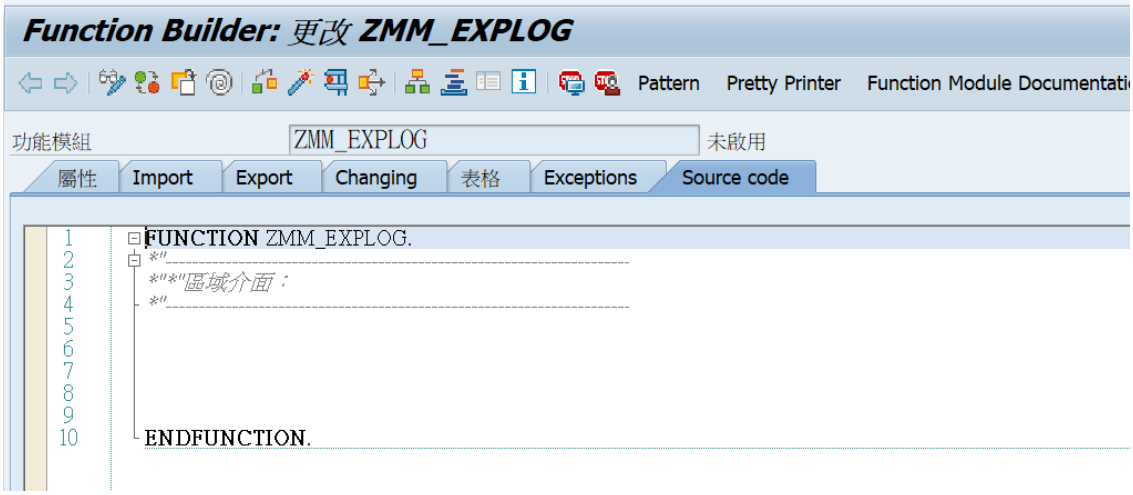
- 與 Report 不同的是，新建 RFC 程式前必須先定義一個 FuncitonGroup，一個 Group 下可包含多個 Function。創建 Function Group 要執行上方的 Menu，路徑為：Goto-Function groups-Create group



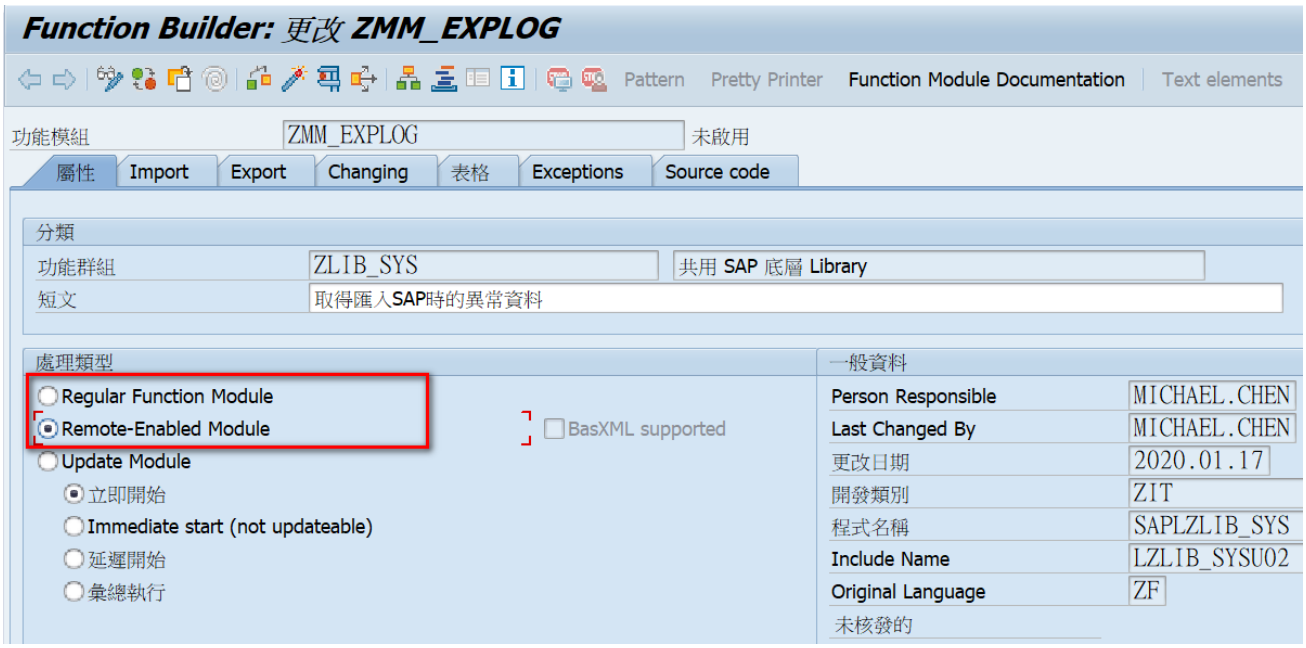
建立 ZLIB_CMM 共用群組，如下。

歸屬於某個開發類別，此處選 ZIT(或 ZMM 皆可)，然後會記錄 CR，此處不在贅述。

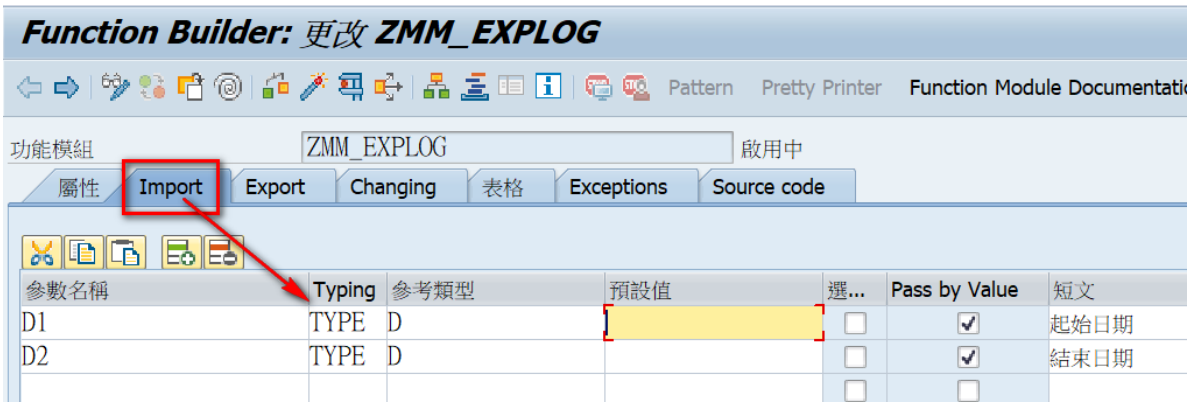
➢ 建立實際的 RFC 程式，存檔後就會進入程式開發介面，和之前的 report 差異不大，請參考下圖



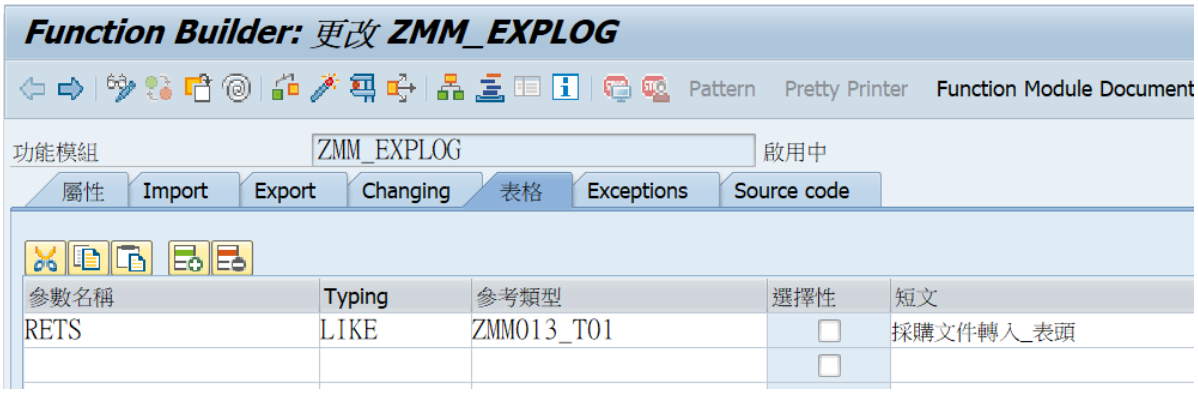
- 到此為止，該 Function 已被創建成功，系統進入初始介面。可以看到系統自動生成了預代碼，但是還不具備任何功能，需要對 Function 的輸入輸出參數及資料處理邏輯程式進行設置。
  - Function 編輯器共分為 7 個頁面，在 Attributes 屬性設置視窗，如下圖所示。允許該 Function 能否被 SAP 及非 SAP 程式調用。



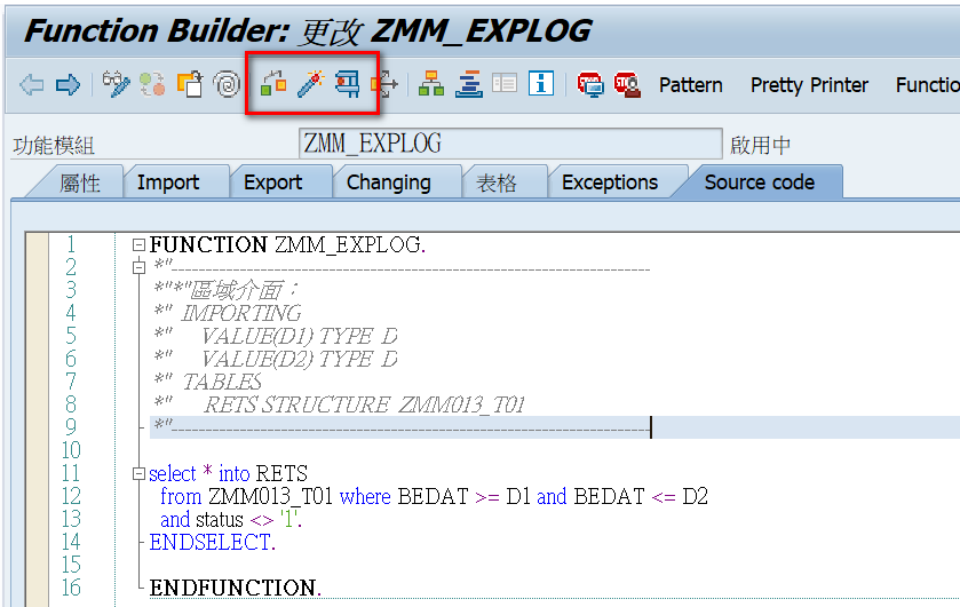
- Import 為資料登錄介面，其介面參數可以為單個變數或者為一個結構體。本例為 2 個變數，輸入 log 的查詢日期區間。



- Tables 可同時作為輸入或輸出的介面，其參數可為單個變數或 Struct 或內表。本例為 Table。



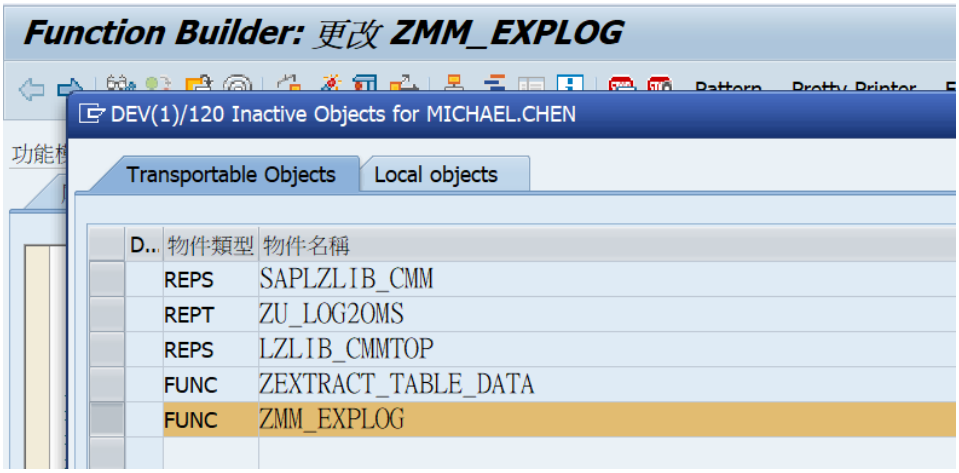
➢ 再來切到 Source code 寫相關的程式(如下)，介面中所定義的參數都是該 Function 代碼的私有變數，可以在 Function 中直接引用，由此實現與外部程式資料交換。



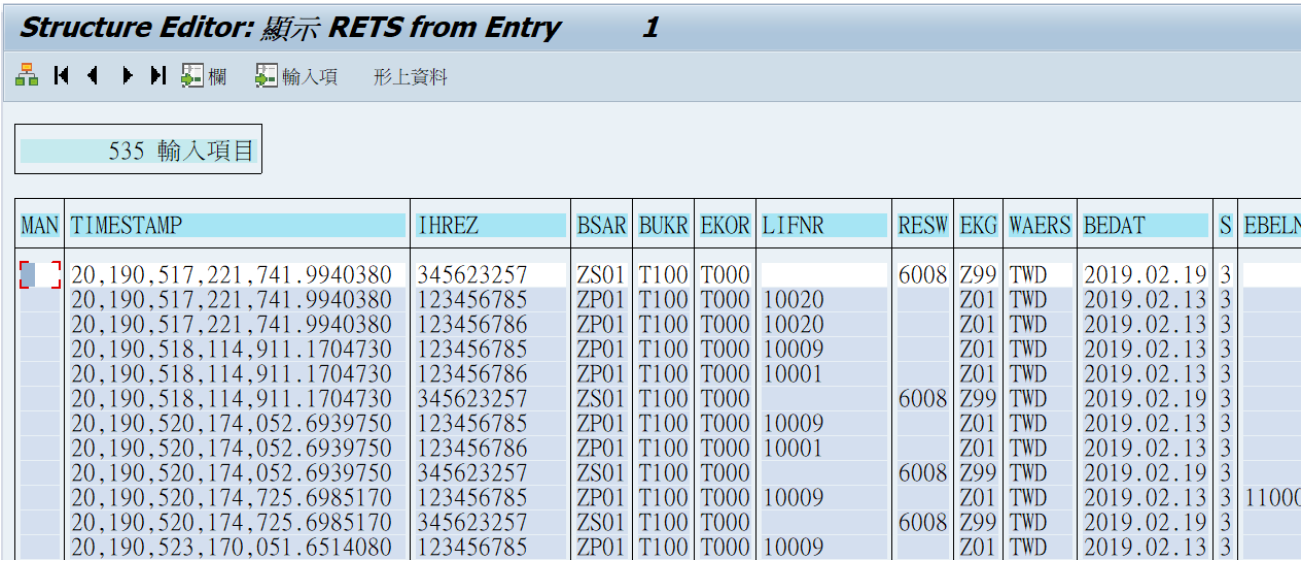

```
FUNCTION ZMM_EXPLOG.
*"----------------------------------------------------------------------
*"*"區域介面：
*"  IMPORTING
*"      VALUE(D1) TYPE   D
*"      VALUE(D2) TYPE   D
*"  TABLES
*"       RETS STRUCTURE   ZMM013_T01
*"----------------------------------------------------------------------

DATA: BEGIN OF ITAB OCCURS 0,
      TIMESTAMP LIKE ZMM013_T01-TIMESTAMP,
      IHREZ LIKE ZMM013_T01-IHREZ,
      BSART LIKE ZMM013_T01-BSART,
      BUKRS LIKE ZMM013_T01-BUKRS,
      EKORG LIKE ZMM013_T01-EKORG,
      LIFNR LIKE ZMM013_T01-LIFNR,
      RESWK LIKE ZMM013_T01-RESWK,
      EKGRP LIKE ZMM013_T01-EKGRP,
      WAERS LIKE ZMM013_T01-WAERS,
      BEDAT LIKE ZMM013_T01-BEDAT,
      STATUS LIKE ZMM013_T01-STATUS,
      EBELN LIKE ZMM013_T01-EBELN,
```

```
      REMARK LIKE ZMM013_T01-REMARK,
      ZPRCST LIKE ZMM013_T01-ZPRCST,
      ZMESSAGE LIKE ZMM013_T01-ZMESSAGE,
      ZCRDAT LIKE ZMM013_T01-ZCRDAT,
      ZCRTIM LIKE ZMM013_T01-ZCRTIM,
      ZCRNAM LIKE ZMM013_T01-ZCRNAM,
      ZCDDAT LIKE ZMM013_T01-ZCDDAT,
      ZCDTIM LIKE ZMM013_T01-ZCDTIM,
      ZCDNAM LIKE ZMM013_T01-ZCDNAM,
      END OF ITAB.

select * INTO CORRESPONDING FIELDS OF TABLE ITAB
   from ZMM013_T01
   where BEDAT >= D1 and BEDAT <= D2
   and status <> '1'.

LOOP AT ITAB.
      MOVE-CORRESPONDING ITAB TO RETS.
      APPEND RETS.
   ENDLOOP.

ENDFUNCTION.
```

- Function 程式編寫完成後可直接對其進行測試，保存 Function 後並啟動，啟動時要順帶將涉及本 Function 的所有都選中。



➢ Function 的執行及資料查看



MAN	TIMESTAMP	IHREZ	BSAR	BUKR	EKOR	LIFNR	RESW	EKG	WAERS	BEDAT	S	EBELN
	20,190,517,221,741.9940380	345623257	ZS01	T100	T000		6008	Z99	TWD	2019.02.19	3	
	20,190,517,221,741.9940380	123456785	ZP01	T100	T000	10020		Z01	TWD	2019.02.13	3	
	20,190,517,221,741.9940380	123456786	ZP01	T100	T000	10020		Z01	TWD	2019.02.13	3	
	20,190,518,114,911.1704730	123456785	ZP01	T100	T000	10009		Z01	TWD	2019.02.13	3	
	20,190,518,114,911.1704730	123456786	ZP01	T100	T000	10001		Z01	TWD	2019.02.13	3	
	20,190,518,114,911.1704730	345623257	ZS01	T100	T000		6008	Z99	TWD	2019.02.19	3	
	20,190,520,174,052.6939750	123456785	ZP01	T100	T000	10009		Z01	TWD	2019.02.13	3	
	20,190,520,174,052.6939750	123456786	ZP01	T100	T000	10001		Z01	TWD	2019.02.13	3	
	20,190,520,174,052.6939750	345623257	ZS01	T100	T000		6008	Z99	TWD	2019.02.19	3	
	20,190,520,174,725.6985170	123456785	ZP01	T100	T000	10009		Z01	TWD	2019.02.13	3	11000
	20,190,520,174,725.6985170	345623257	ZS01	T100	T000		6008	Z99	TWD	2019.02.19	3	
	20,190,523,170,051.6514080	123456785	ZP01	T100	T000	10009		Z01	TWD	2019.02.13	3	

➢ 這樣就完成了一支標準的 RFC 程式，大部分的概念和我們寫 report 差異並不大。接下來，我們就要測試由外部程式來呼叫這支 RFC 程式。網上有很多的範例可供參考，大多是 Java 和 C#，接下來我就用我熟悉的 Delphi 7 來測試看看，請參考下列 blog 的相關說明(一併附上參考 blog)。

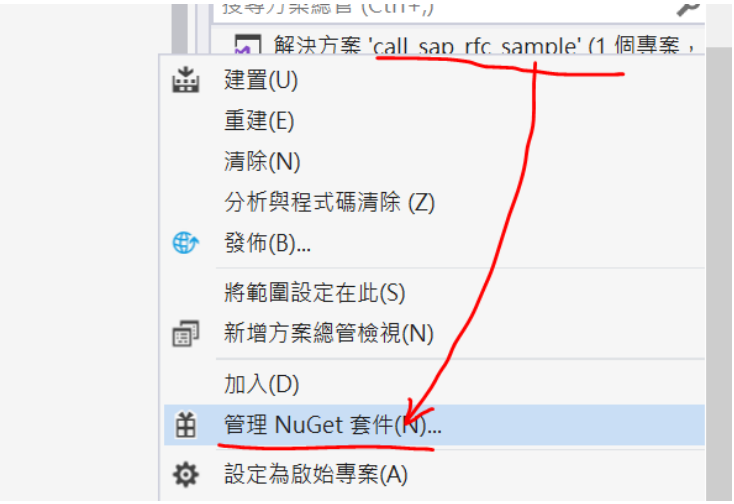
https://bbs.csdn.net/topics/310195606

https://robertlan.com/programming/sap-net-connector-3-0%E9%80%A3%E6%8E%A5rfc/

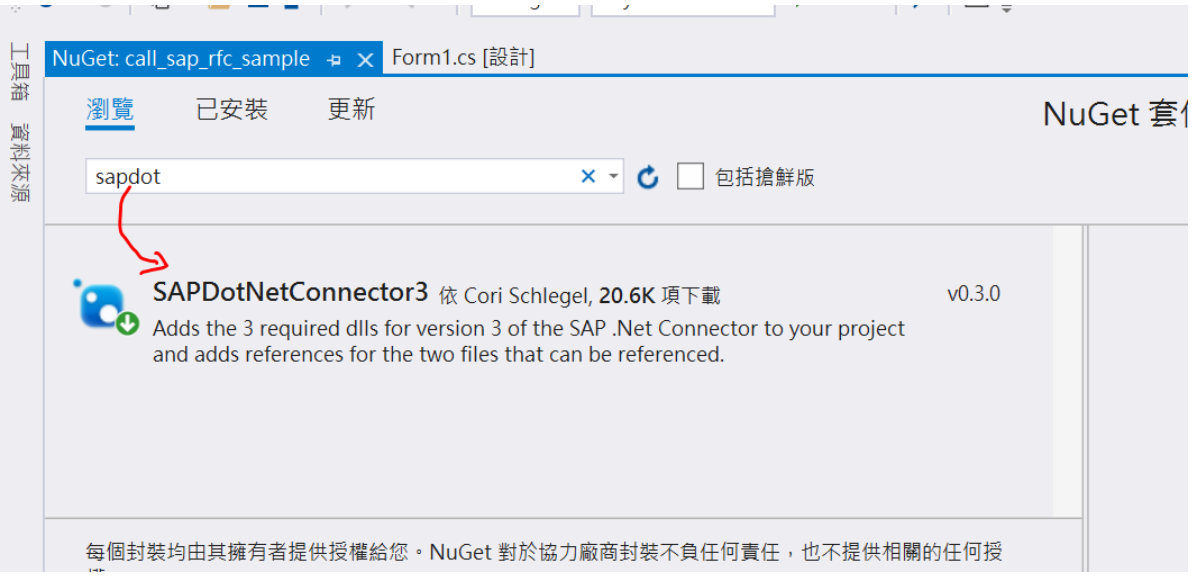
https://www.codeproject.com/Articles/824928/SAP-Net-Connector-NCo-Example

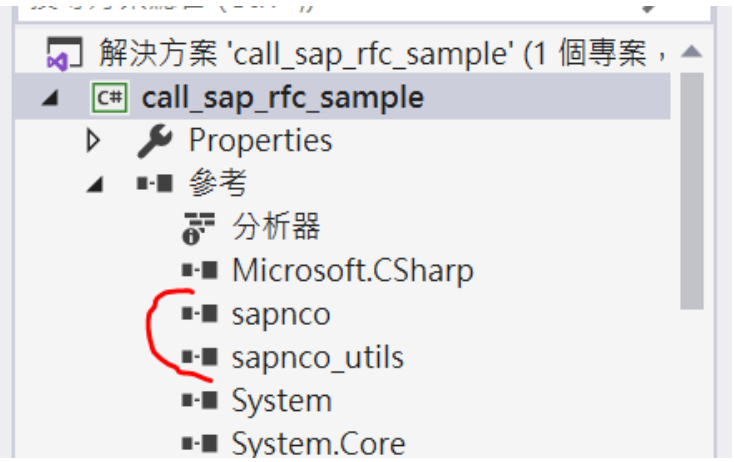
https://www.jianshu.com/p/11bf74e86a3e

- Visual Studio201x 下，安裝 SAPDotNetConnector3 -Version 0.3.0
  - 新建一個 C# WinForm 的 Solution，在該 Solution 上按右鍵，選擇 [管理 NuGet 套件]



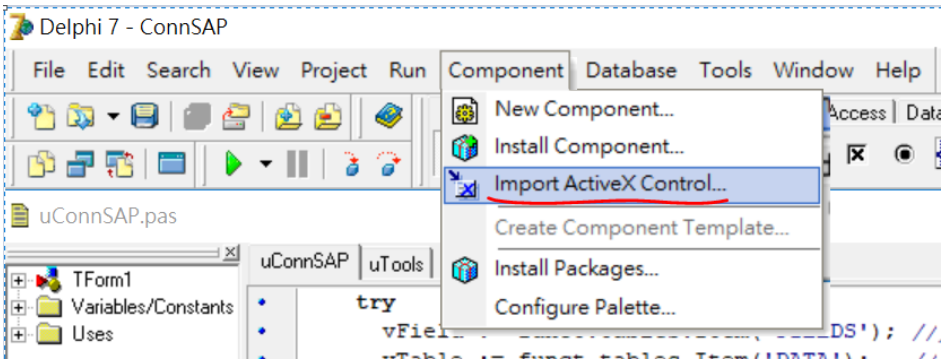
  - 搜尋 SAPDotNetConnector 並安裝，安裝後，會新增 2 個參考(sapnco、sapnco_utils)

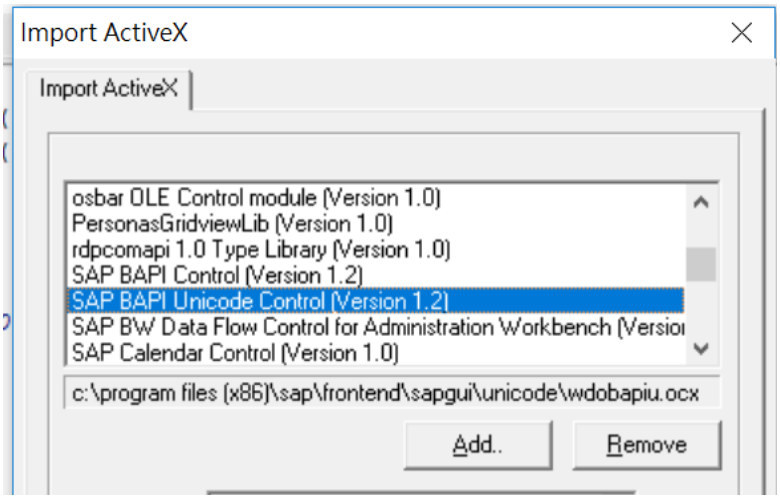


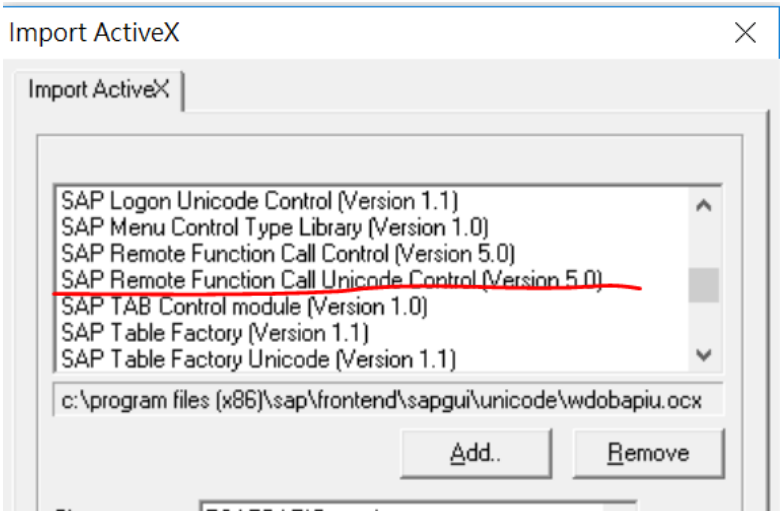


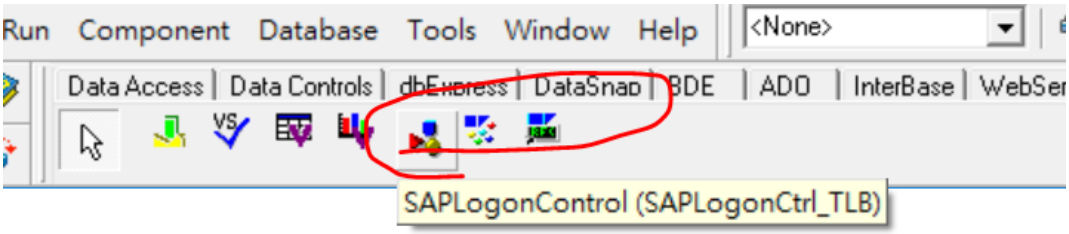
➢ Delphi 7 安装 ActiveX 元件

https://www.cnblogs.com/VerySky/articles/2137935.html

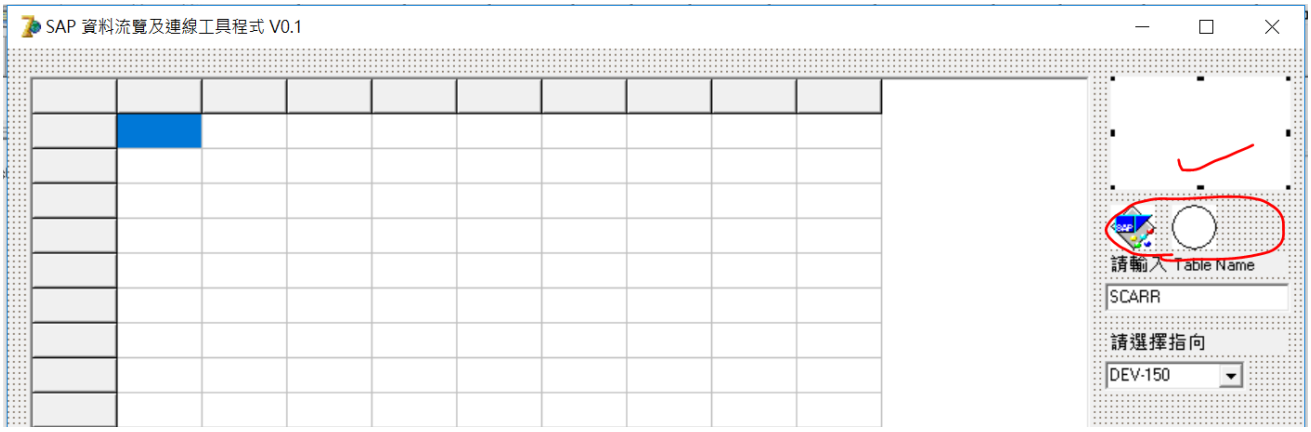








➢ 連接 SAP，先拖拉元件到 UX 上



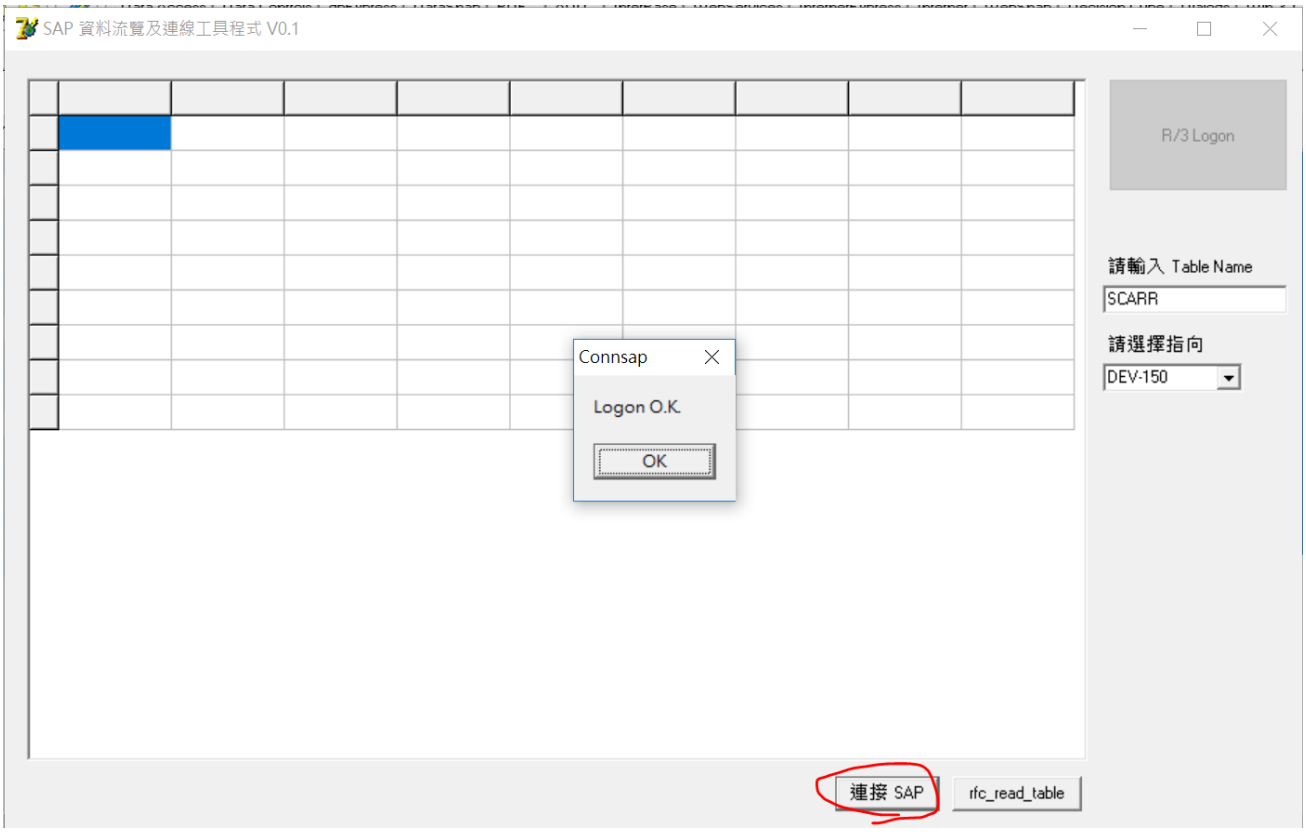

```
//參考 blog ==> https://www.cnblogs.com/VerySky/articles/2137935.html
//連線 SAP QAS-888
Connection                    := SAPLogoncontroll.newConnection;
Connection.User               := Ansiuppercase(sUser);
Connection.System             := sSys;
Connection.Client             := sID;
Connection.ApplicationServer  := sIP;

Connection.SystemNumber       := '00';
Connection.Password           := sPwd;
SAPLogonControll.Enabled      := false;

if Connection.LogOn(0,true) = True then
begin
  ShowMessage('Logon O.K.');
  button2.Enabled:= true;
  button3.Enabled:= true;
  SapBapiControll.Connection  := Connection;
  sapFunctions1.Connection    := Connection;
end
else begin
  ShowMessage('Error on logon :-(((');
end;
```

➢ 呼叫 RFC_READ_TABLE 共用程式，這是 SAP 提供的一支通用 RFC，顧名及可思其義。

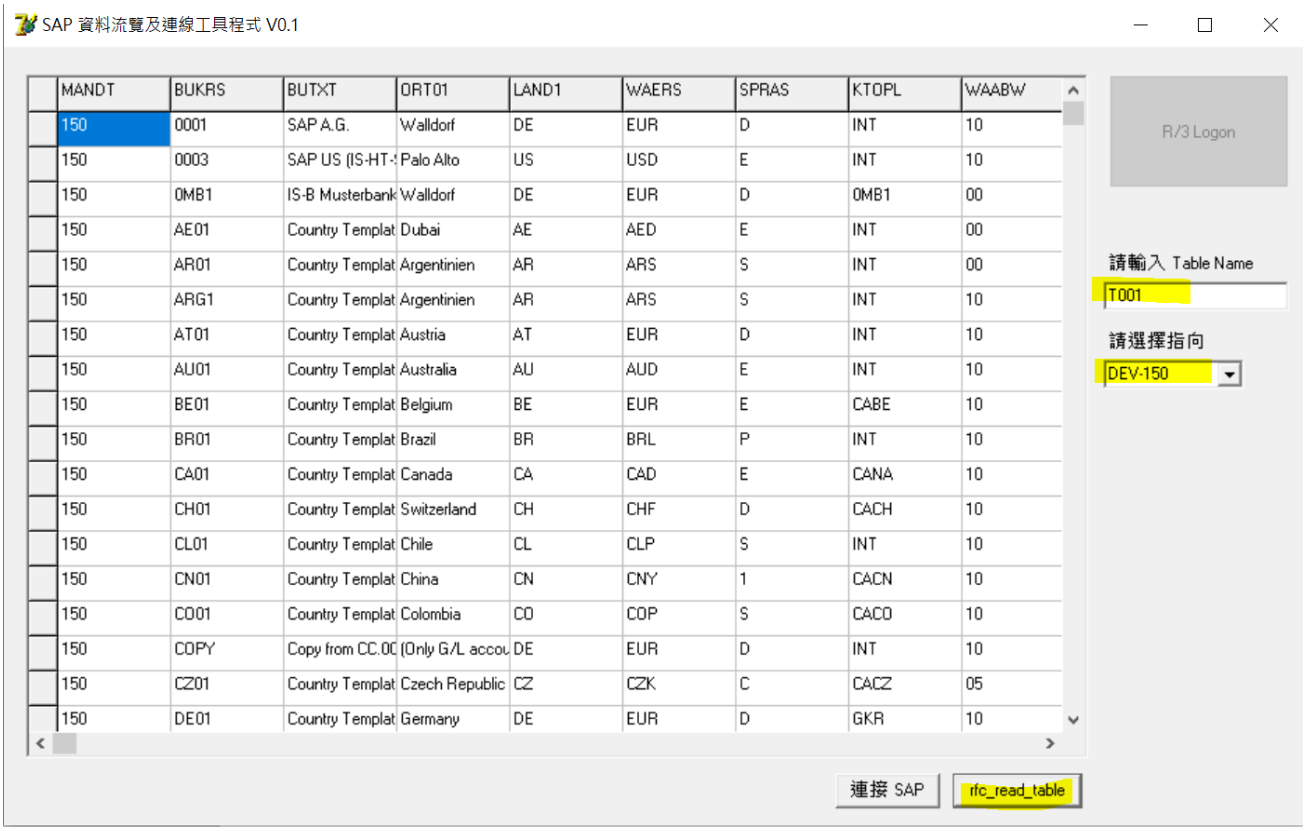
```
SAPFunctions1.RemoveAll;
funct :=SAPFunctions1.add('RFC_READ_TABLE');  // function 名稱
funct.exports('QUERY_TABLE').VALUE:= edTable.Text;  // 輸入的參數
funct.exports('DELIMITER').VALUE:= '|';  // 輸入的參數 2

if not funct.CALL then
  ShowMessage('呼叫 RFC error')
else begin
  try
    vField := funct.tables.Item('FIELDS'); //連結成功 返回的 Table
    vTable := funct.tables.Item('DATA');   //連結成功 返回的 Table
  except
    ShowMessage('error')
  end;

  bEnd := False;  //欄位, 最多 100 個欄位
  sFields := '';
  iCnt := 1;
  while not bEnd do begin
    // 可直接使用 TABLE  的值
    try
      sTmp := vField.Value(iCnt,'FIELDNAME');
      sFields := sFields + sTmp + '|';
      Inc(iCnt);
    except
      bEnd := True;
    end
  end;

  bEnd := False;  //最多取 100 筆資料為限
  iCnt := 1;
  while not bEnd do begin
    // 可直接使用 TABLE  的值
    try
      sTmp := vTable.Value(iCnt,'WA');
      Inc(iCnt);
    except
```

```
        bEnd := True;
      end
    end;

    vFields := StringToArray(sFields,'|');
    StringGrid1.ColCount := VarArrayHighBound(vFields,1)+1;
    for x := 0 to VarArrayHighBound(vFields,1)-1 do begin
      StringGrid1.Cells[x+1,0] := Trim(vFields[x]);
    end;

    for i := 1 to iCnt-1 do begin
      sTmp := vTable.Value(i,'WA');
      vData := StringToArray(sTmp,'|');

      for x := 0 to VarArrayHighBound(vData,1) do begin
        StringGrid1.Cells[x+1,i] := Trim(vData[x]);
      end;
    end;
    StringGrid1.RowCount := iCnt;
  end;
```

➢ 程式執行的外觀



連線後即可呼叫 RFC，取得指定 SAP Table 的資料。

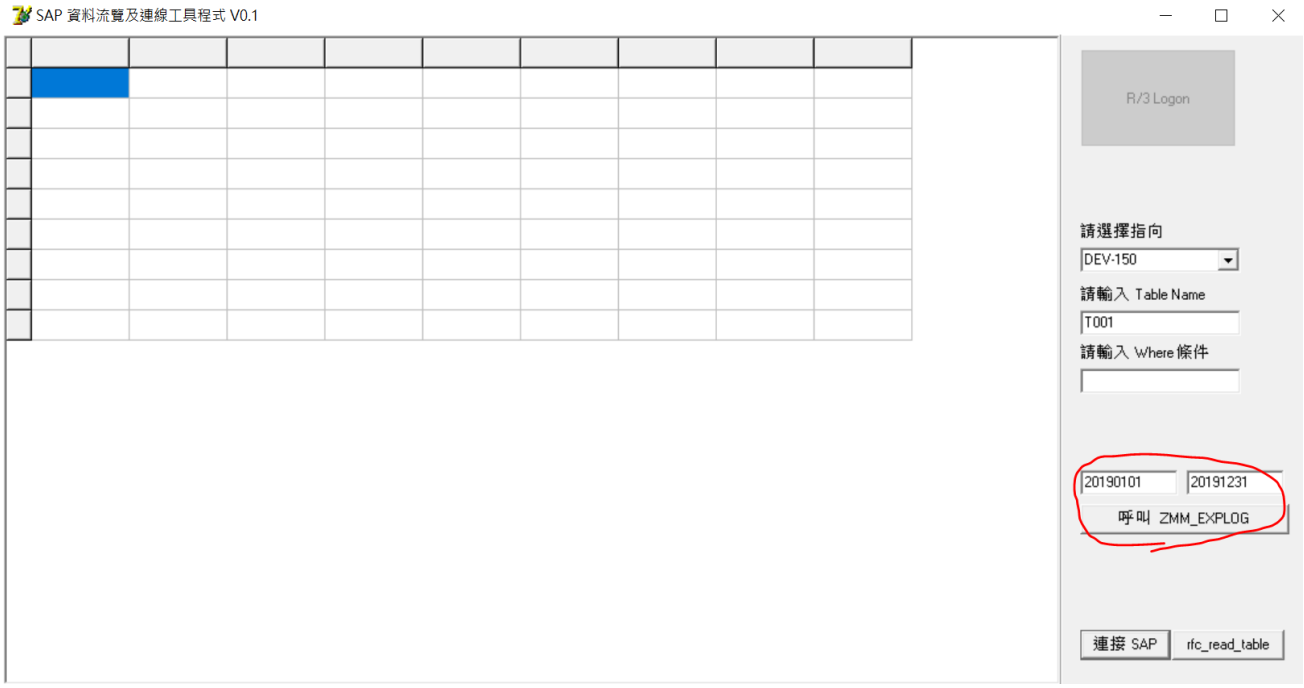


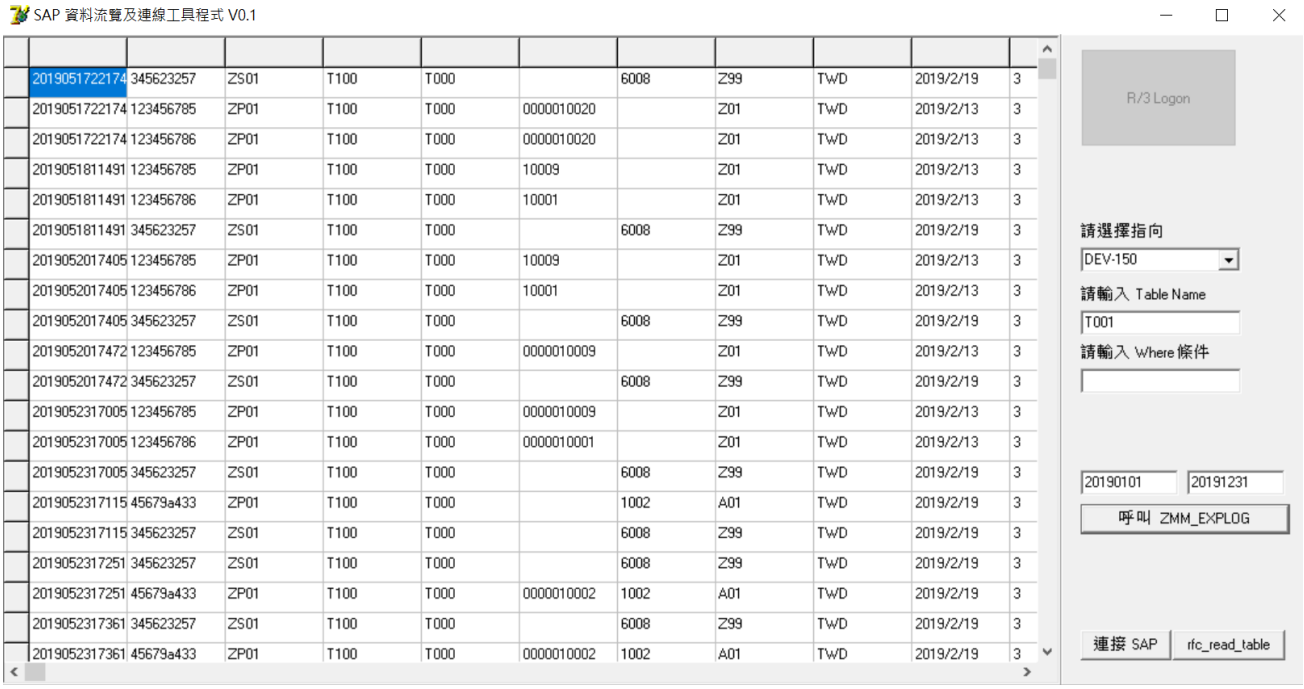
MANDT	BUKRS	BUTXT	ORT01	LAND1	WAERS	SPRAS	KTOPL	WAABW
150	0001	SAP A.G.	Walldorf	DE	EUR	D	INT	10
150	0003	SAP US (IS-HT-	Palo Alto	US	USD	E	INT	10
150	0MB1	IS-B Musterbank	Walldorf	DE	EUR	D	0MB1	00
150	AE01	Country Templat	Dubai	AE	AED	E	INT	00
150	AR01	Country Templat	Argentinien	AR	ARS	S	INT	00
150	ARG1	Country Templat	Argentinien	AR	ARS	S	INT	10
150	AT01	Country Templat	Austria	AT	EUR	D	INT	10
150	AU01	Country Templat	Australia	AU	AUD	E	INT	10
150	BE01	Country Templat	Belgium	BE	EUR	E	CABE	10
150	BR01	Country Templat	Brazil	BR	BRL	P	INT	10
150	CA01	Country Templat	Canada	CA	CAD	E	CANA	10
150	CH01	Country Templat	Switzerland	CH	CHF	D	CACH	10
150	CL01	Country Templat	Chile	CL	CLP	S	INT	10
150	CN01	Country Templat	China	CN	CNY	1	CACN	10
150	CO01	Country Templat	Colombia	CO	COP	S	CACO	10
150	COPY	Copy from CC.0(	(Only G/L accou	DE	EUR	D	INT	10
150	CZ01	Country Templat	Czech Republic	CZ	CZK	C	CACZ	05
150	DE01	Country Templat	Germany	DE	EUR	D	GKR	10

➢ 下面是呼叫自行開發的 RFC=ZMM_EXPLOG 及執行後的畫面(Sample 只顯示前 100 筆資料)

```
//如何呼叫?
SAPFunctions1.RemoveAll;
funct :=SAPFunctions1.add('ZMM_EXPLOG');   // function 名稱
funct.exports('D1').VALUE:= edD1.Text;   // 輸入的參數
funct.exports('D2').VALUE:= edD2.Text;   // 輸入的參數 2

if not funct.CALL then
  ShowMessage('呼叫 RFC error')
else begin
  try
    vTable := funct.tables.Item('RETS'); //連結成功 返回的 Table
  except
    ShowMessage('error')
  end;

  // 可直接使用 TABLE  的值
  for i := 1 to 100 do begin
    StringGrid1.Cells[1,i] :=vTable.Value(i,'TIMESTAMP');
    StringGrid1.Cells[2,i] :=vTable.Value(i,'IHREZ');
    StringGrid1.Cells[3,i] :=vTable.Value(i,'BSART');
    StringGrid1.Cells[4,i] :=vTable.Value(i,'BUKRS');
    StringGrid1.Cells[5,i] :=vTable.Value(i,'EKORG');
    StringGrid1.Cells[6,i] :=vTable.Value(i,'LIFNR');
    StringGrid1.Cells[7,i] :=vTable.Value(i,'RESWK');
    StringGrid1.Cells[8,i] :=vTable.Value(i,'EKGRP');
    StringGrid1.Cells[9,i] :=vTable.Value(i,'WAERS');
    StringGrid1.Cells[10,i] :=vTable.Value(i,'BEDAT');
    StringGrid1.Cells[11,i] :=vTable.Value(i,'STATUS');
    StringGrid1.Cells[12,i] :=vTable.Value(i,'EBELN');
    StringGrid1.Cells[13,i] :=vTable.Value(i,'REMARK');
    StringGrid1.Cells[14,i] :=vTable.Value(i,'ZPRCST');
    StringGrid1.Cells[15,i] :=vTable.Value(i,'ZMESSAGE');
    StringGrid1.Cells[16,i] :=vTable.Value(i,'ZCRDAT');
    StringGrid1.Cells[17,i] :=vTable.Value(i,'ZCRTIM');
    StringGrid1.Cells[18,i] :=vTable.Value(i,'ZCRNAM');
    StringGrid1.Cells[19,i] :=vTable.Value(i,'ZCDDAT');
    StringGrid1.Cells[20,i] :=vTable.Value(i,'ZCDTIM');
```

```
      StringGrid1.Cells[21,i] :=vTable.Value(i,'ZCDNAM');
    end;
    StringGrid1.RowCount := 100;
    StringGrid1.ColCount := 21;
  end;
```





2019051722174	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019051722174	123456785	ZP01	T100	T000	0000010020		Z01	TWD	2019/2/13	3
2019051722174	123456786	ZP01	T100	T000	0000010020		Z01	TWD	2019/2/13	3
2019051811491	123456785	ZP01	T100	T000	10009		Z01	TWD	2019/2/13	3
2019051811491	123456786	ZP01	T100	T000	10001		Z01	TWD	2019/2/13	3
2019051811491	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019052017405	123456785	ZP01	T100	T000	10009		Z01	TWD	2019/2/13	3
2019052017405	123456786	ZP01	T100	T000	10001		Z01	TWD	2019/2/13	3
2019052017405	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019052017472	123456785	ZP01	T100	T000	0000010009		Z01	TWD	2019/2/13	3
2019052017472	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019052317005	123456785	ZP01	T100	T000	0000010009		Z01	TWD	2019/2/13	3
2019052317005	123456786	ZP01	T100	T000	0000010001		Z01	TWD	2019/2/13	3
2019052317005	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019052317115	45679a433	ZP01	T100	T000		1002	A01	TWD	2019/2/19	3
2019052317115	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019052317251	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019052317251	45679a433	ZP01	T100	T000	0000010002	1002	A01	TWD	2019/2/19	3
2019052317361	345623257	ZS01	T100	T000		6008	Z99	TWD	2019/2/19	3
2019052317361	45679a433	ZP01	T100	T000	0000010002	1002	A01	TWD	2019/2/19	3

第 21 章 ABAP 進階要點

在第 10 章中，我們有用大量的簡單範例來快速說明 ABAP 程式的基本概念，如果您有仔細看過並認真練習，基本上寫一些基本報表應該是沒有什麼問題。當然，這遠遠不是 ABAP 的極限，任何語言的學習，都是由淺入深的，所以接下來，我會將學習過程中有遇到的問題，屬於較進階的部分，逐步以範例的方式說明，本章的內容由於是經驗總結，所以會比較雜亂，請隨興閱讀即可。

- 在 ABAP 報表程式中，如果希望按兩下某行或某欄位跳轉就跳轉到其他 tcode 的需求(類似 Drill Down 功能)，然後可能需要傳參數(例如單號等)，此功能該如何實現?
  參考 blog http://www.baidusap.com/abap/alv/3992

```
 "以下是主要的程式碼片段
CONSTANTS: gco_callback_user_command TYPE slis_formname    VALUE 'FRM_USER_COMMAND'.

CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY_LVC'
      EXPORTING
        i_callback_program          = sy-repid
        is_layout_lvc               = gwa_layout
        it_fieldcat_lvc             = git_fcat
        i_callback_user_command = gco_callback_user_command
      TABLES
        t_outtab                    = git_ekko[]
      EXCEPTIONS
        program_error               = 1
        OTHERS                      = 2.

FORM frm_user_command USING i_ucomm          TYPE sy-ucomm
                            i_wa_selfield TYPE slis_selfield.
   CASE i_ucomm.
      WHEN '&IC1'.   "Double click
*       if click on PO field, jump to me23n
        IF i_wa_selfield-fieldname = 'EBELN'.
          SET PARAMETER ID 'BES' FIELD i_wa_selfield-value.
          CALL TRANSACTION 'ME23N' AND SKIP FIRST SCREEN.
        ENDIF.
      WHEN OTHERS.
   ENDCASE.
```

基本上，程式碼並不難懂(這個範例寫得非常好)，雖然沒用過 CALL TRANSACTION 功能，但看過範例再 Google 大概就能理解。主要的卡殼點是在 SET PARAMETER ID 'BES' FIELD i_wa_selfield-value.
呼叫不同的 T-Code，相當然爾要傳入的參數(如單號)，一定不一樣，但是我怎麼曉得在呼叫不同的 T-Code 時要傳入 PARAMETER ID？可能是因為太簡單了，所以找了幾個 blog 都沒有說明，最後在下面這篇 blog 找到相關的說明： http://www.sapdoc.cn/sap/theme/86.html

```
SET PARAMETER ID 'LIB' FIELD p_rs_selfield-value.

CALL TRANSACTION 'SE37' AND SKIP FIRST SCREEN .

ENDFORM .                    " frm_alv_call_tran
```

其中，" **LIB** "是指：参数ID（PARAMETER ID）。

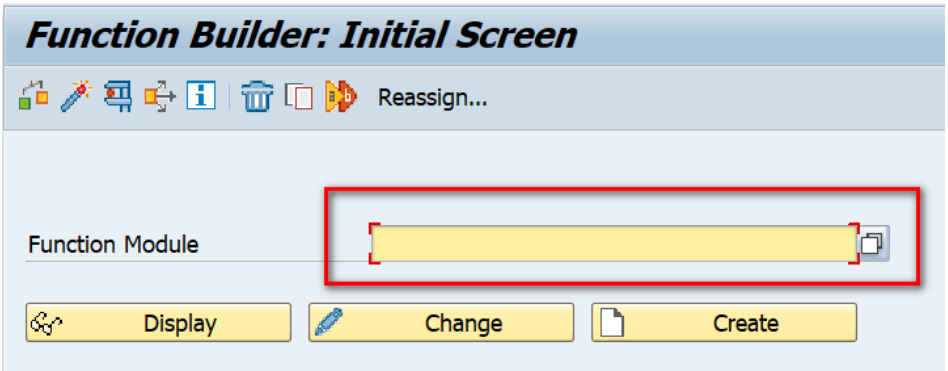
查找方法：

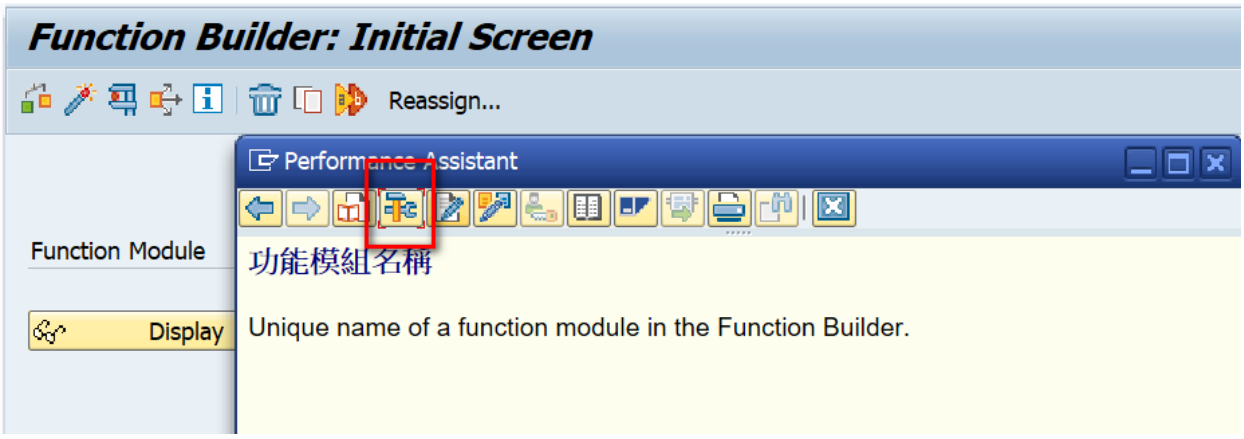
01). 用一个事务码进入屏幕界面，在输入框内按 F1；

02). 在弹出来的界面中点击" 技术信息" 按钮，在字段数据的最后一行，即为参数ID；

"SE37" 是指我们要调用的事务代码，**AND SKIP FIRST SCREEN** 表示跳过第一屏屏幕；

首先要先確定被呼叫的表單，要有查詢的介面。例如上例的 SE37，一進入就會有一個查詢畫面，只有一個欄位，在該欄位上按 F1，就和查表單的欄位對應的欄位名稱一樣。

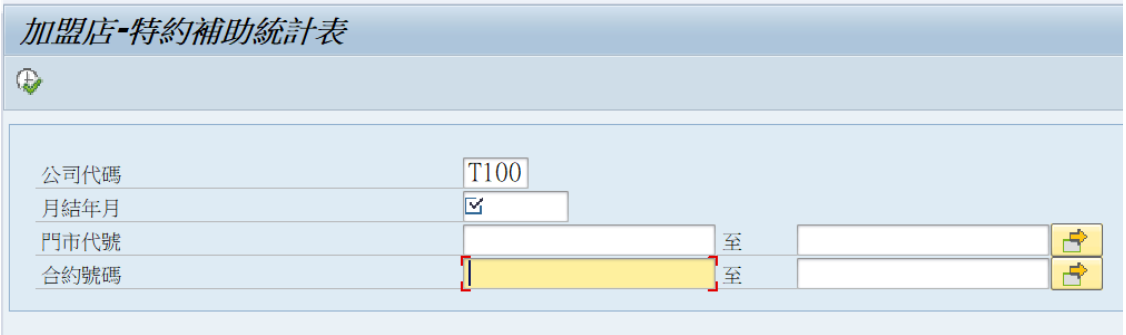




如果是 ME23N，可以由 [其他 XX 單] 顯示查詢畫面，其他的操作都是一樣的。

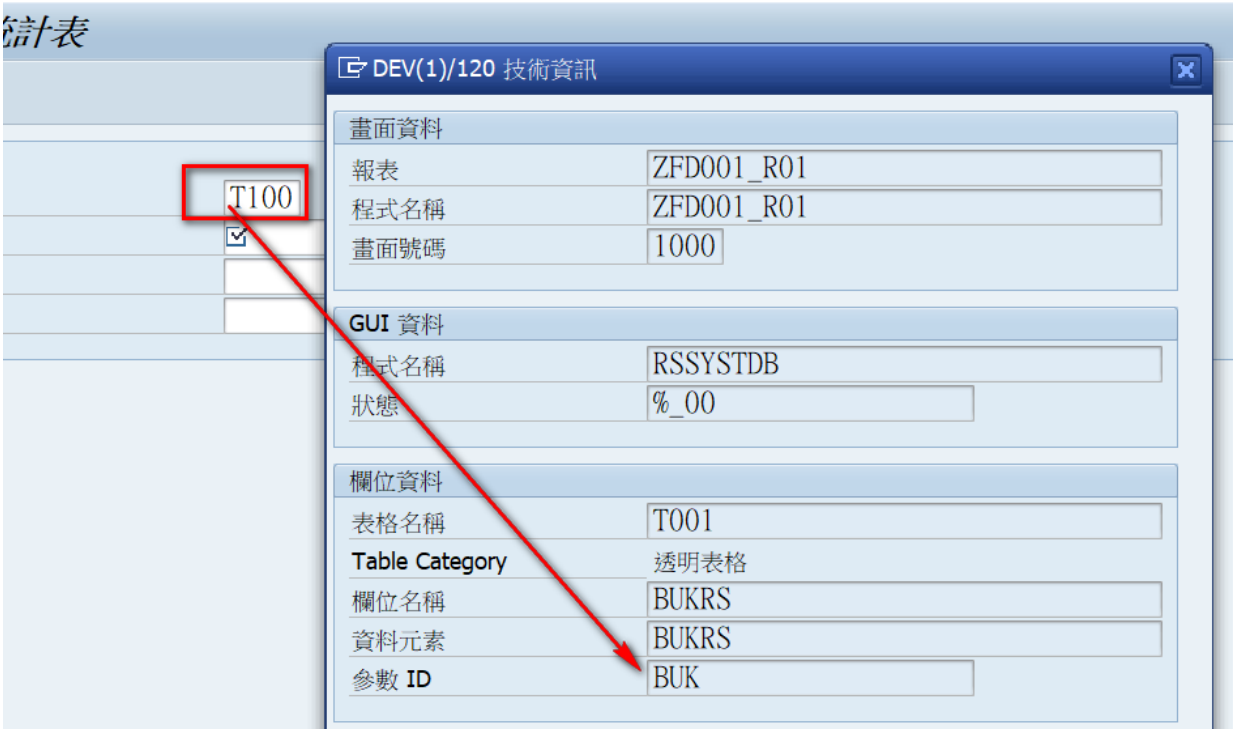
現在還有一個問題困擾我，如果我要呼叫的 T-Code 是我自己寫的 Report，也可以這樣查嗎?

試試 ZFD001_R02



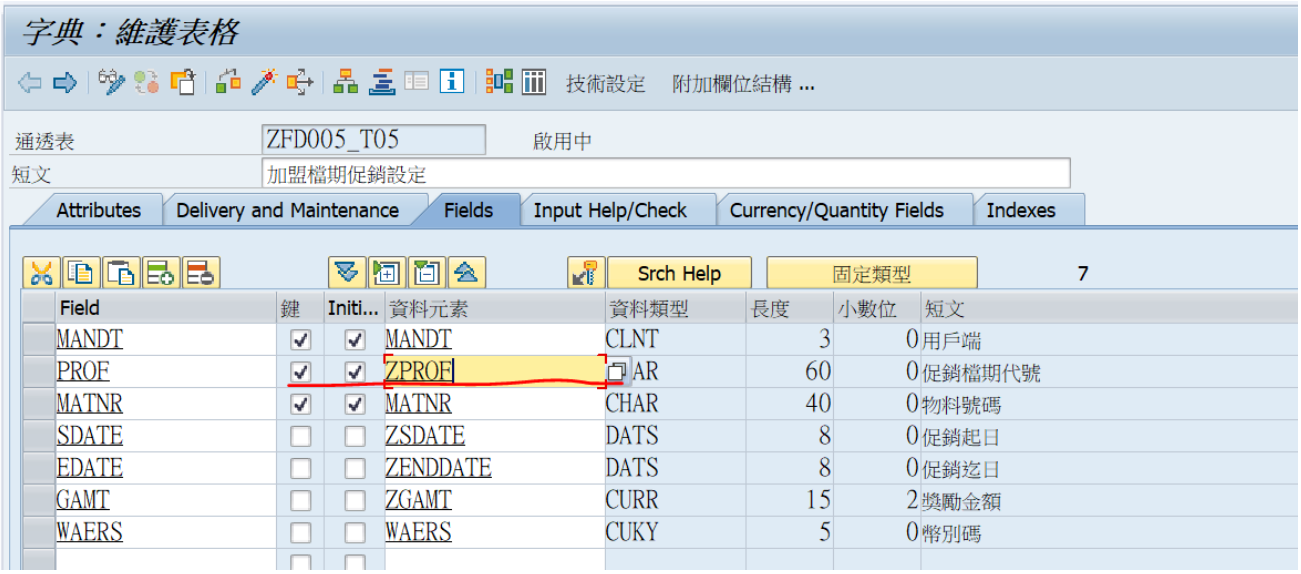
實作發現，前 2 個欄位可以用相同操作，查出相關的 PARA ID，但是後 2 個欄位不行，我就明白了，在程式中宣告為 PARAMETERS 的就可以，宣告為 SELECT-OPTIONS 就不行。

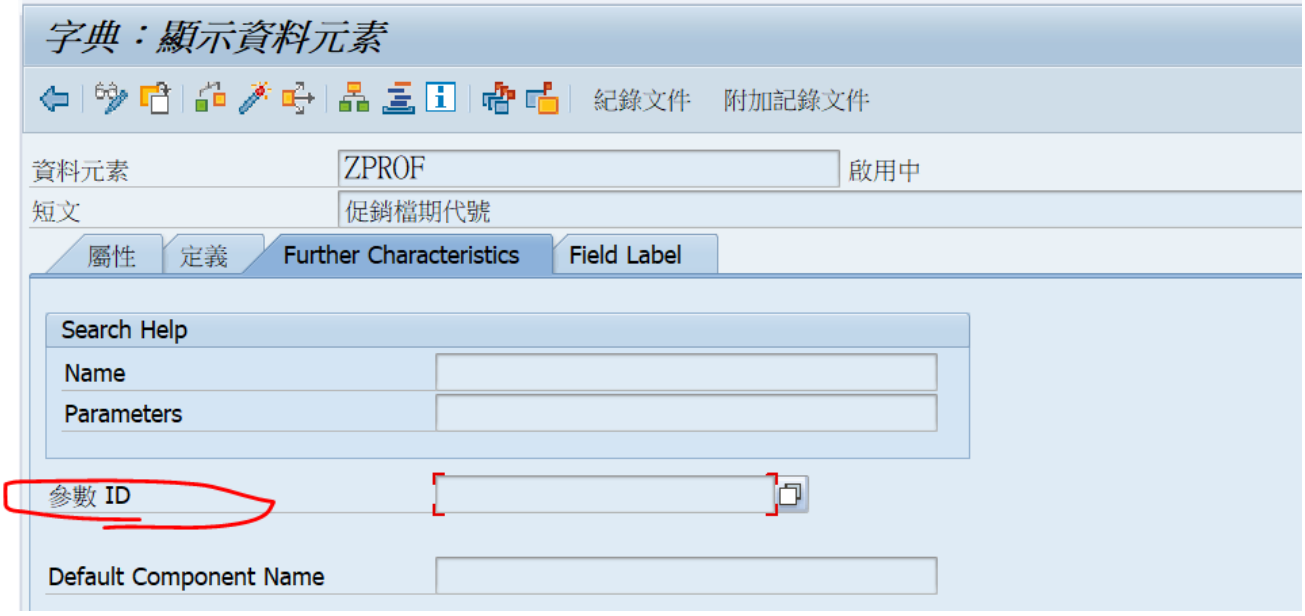
```
56  *&---------------------------------------------------------------------*
57  *& Selection-screen.
58  *&---------------------------------------------------------------------*
59  SELECTION-SCREEN BEGIN OF BLOCK bk WITH FRAME TITLE text-001.
60  PARAMETERS :
61    p_bukrs TYPE t001-bukrs OBLIGATORY DEFAULT 'T100',
62    p_zbuper TYPE zfd004_c01-buper OBLIGATORY.
63
64  SELECT-OPTIONS :
65    s_werks FOR itab-WERKS,
66    s_zcon FOR itab-zcontract.
67
68  SELECTION-SCREEN END OF BLOCK bk.
```



修正前述關於 Parameter ID 的錯誤觀念

關於 Parameter ID 事實上是定義在 SAP 的 SE11 中，一般新增的欄位是空白。

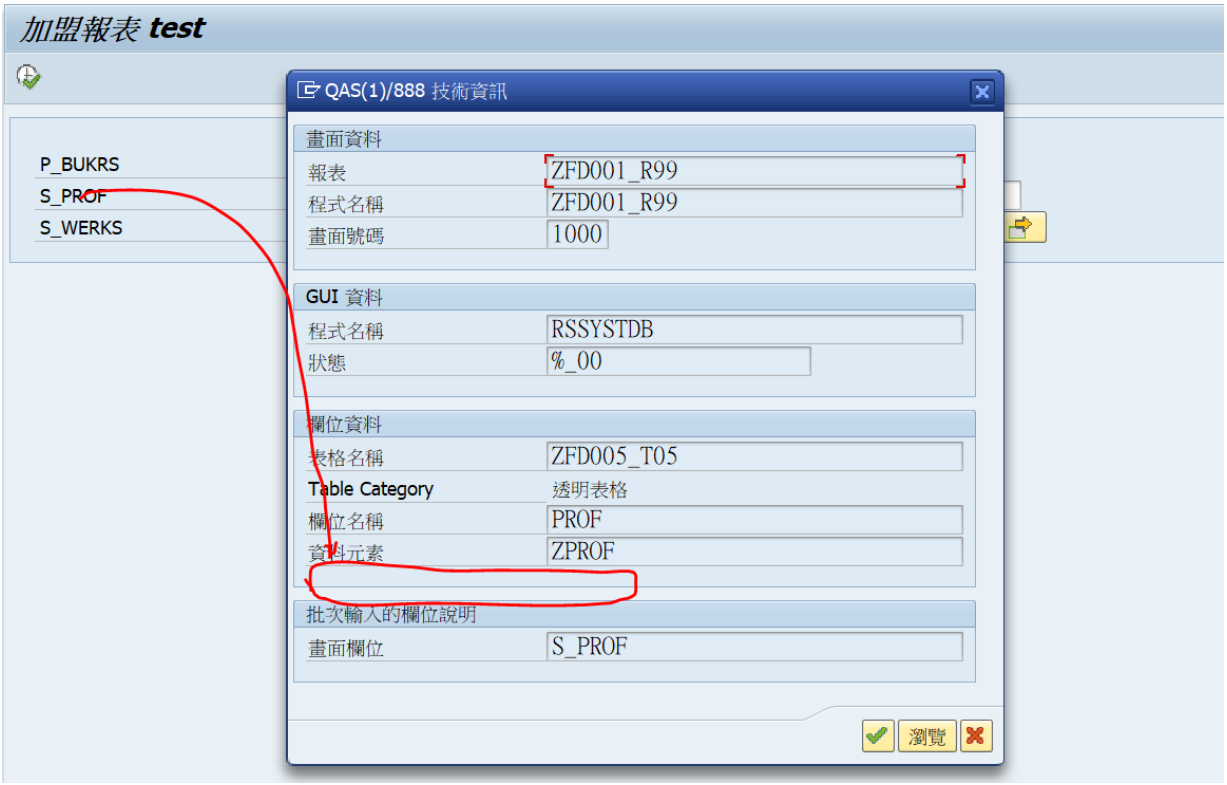




程式碼要注意 TYPE 後的宣告方式，一定要繼承自 DD，才能讀到 Parameter ID。

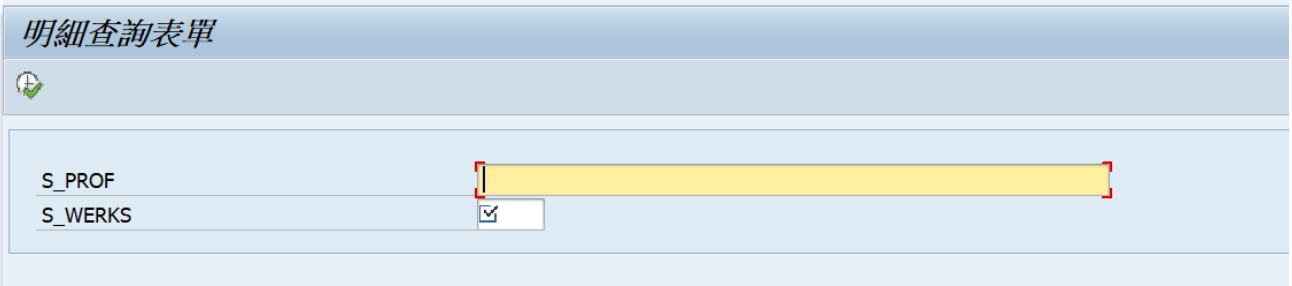
```
*&---------------------------------------------------------------------*
*& Selection-screen.
*&---------------------------------------------------------------------*
SELECTION-SCREEN BEGIN OF BLOCK bk WITH FRAME TITLE text-001.
PARAMETERS :
  p_bukrs TYPE t001-bukrs OBLIGATORY DEFAULT 'T100',
  s_prof TYPE ZFD005_T05-prof OBLIGATORY.
  "s_prof(10) TYPE c OBLIGATORY.
  "p_month(2) TYPE C DEFAULT '01'.

SELECT-OPTIONS :
  s_werks FOR zfd002_t01-werks.
  "s_zcon FOR itab-zcontract.
```

如果沒有設定，當用 F1 去查詢時，就沒有參數 ID 這個欄位，如下圖



在 SE11 設定後，再查詢就可以看到 Parameter ID

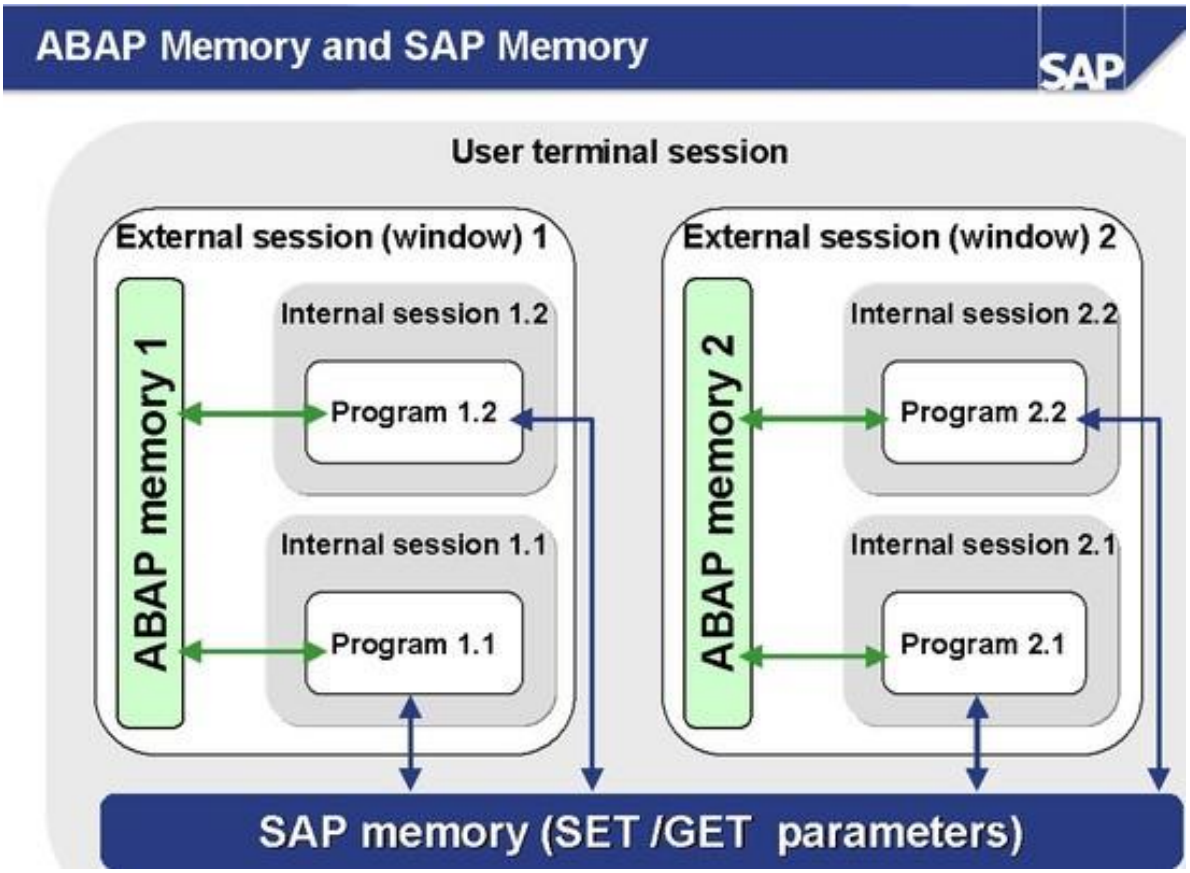
原本以為這樣就大功告成了，沒想到又碰到攔路虎。呼叫端的寫法是沒問題了，參數也正確傳了，但是被呼叫的報表，卻沒有自動帶入參數，而是顯示報表輸入條件的畫面，並沒有自動帶入剛才傳入的參數，如下圖。

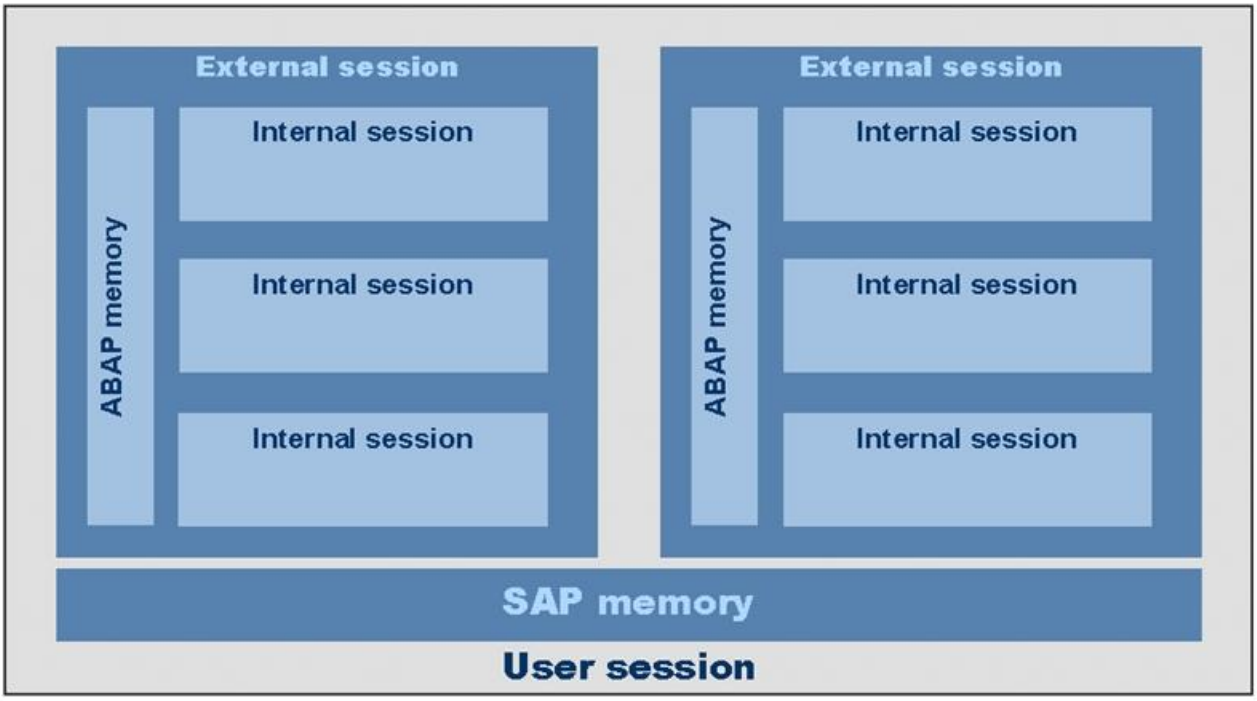


這就是不求甚解的下場，就是沒搞清楚 SAP 和 ABAP 的記憶體參數，請參考相關 blog 說明。

以下說明的圖片來源

https://sapcodes.com/2015/11/17/sap-abap-memory/





相關的 ABAP 記憶體運作原理說明

https://www.itread01.com/content/1549456383.html

https://blog.csdn.net/louisking111/article/details/51328548

在被呼叫端的報表，要加入下列程式，才真的大功告成。這樣才總算完成一支有 Drill-Down 的完整報表功能。

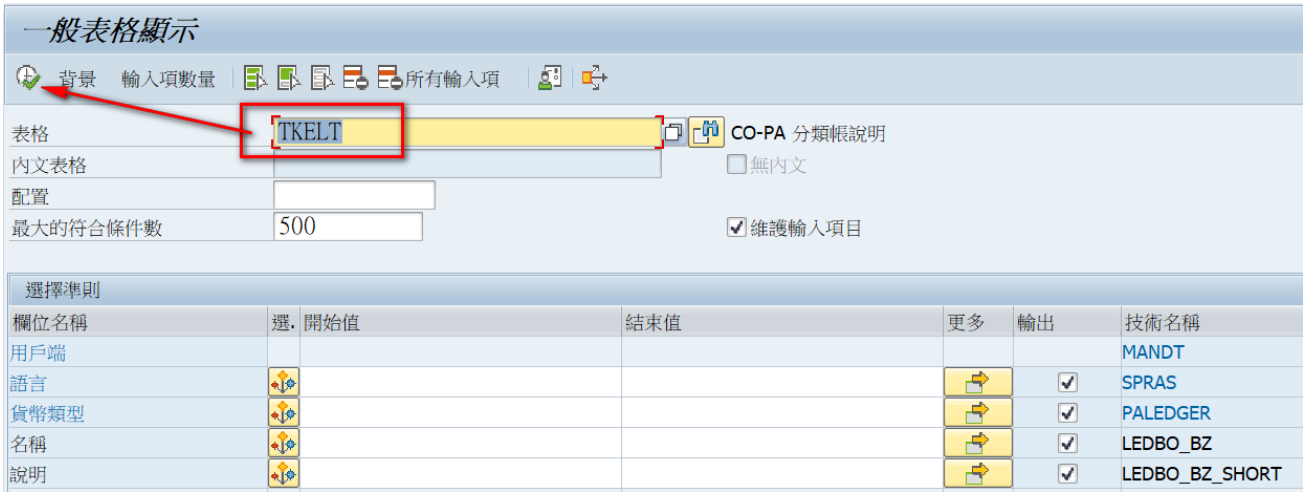
請注意，此時 SQL 語法的 Where xxx 只能用等於不能用 in ➔ `and k~WERKS = @s_werks`

```
PARAMETERS :
 s_prof TYPE ZFD005_T05-prof OBLIGATORY,
 s_werks TYPE zfd002_t01-werks OBLIGATORY.

*&---------------------------------------------------------------------*
*& Initialization.
*&---------------------------------------------------------------------*
INITIALIZATION.
 GET PARAMETER ID 'PRF' FIELD s_prof .
 GET PARAMETER ID 'WRK' FIELD s_werks.
```

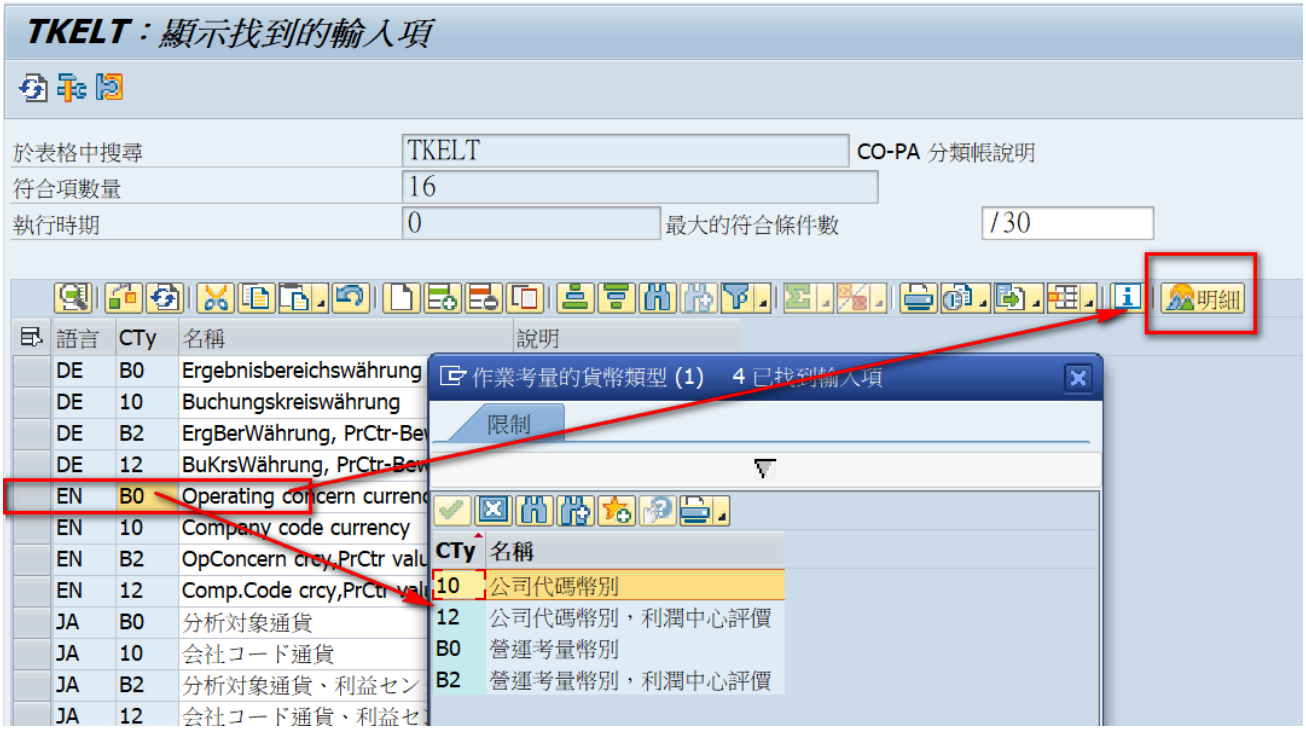
➢ 在寫 ABAP 的 SQL 語法時，很多時候在畫面上的下拉內容和資料庫欄位內容不符，如何取得真正的值?

例如在查 ke30 報表時，貨幣類型 Combo 的內容和實際資料不同，先 F1 找出該欄位所屬 Table

再利用 SE16N 查詢該 Table 內容



顯示如下，再按 [明細] 可以查對照代號



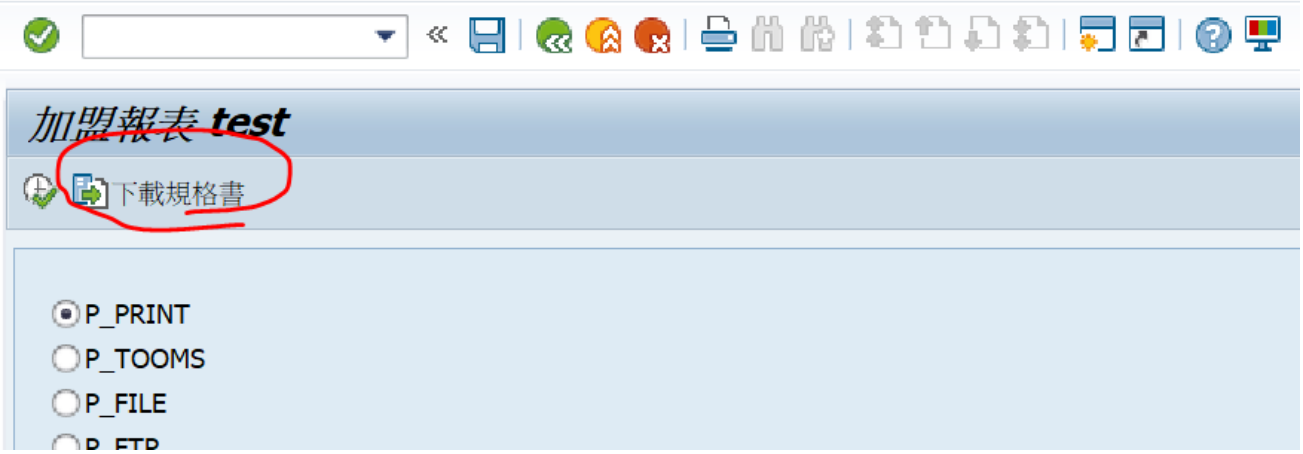
但是資料庫欄位 PALEDGER 的內容卻是

欄位名稱	值	未轉換值	技術欄位名稱
用戶端	888	888	MANDT
語言代碼	EN	E	SPRAS
貨幣類型	B0	01	PALEDGER
名稱	Operating concern currency	Operating concern currency	LEDBO_BZ
說明	Op. concern currency	Op. concern currency	LEDBO_BZ_SHORT
輸入項數量	5	5	LINE_INDEX

Combo 下拉顯示內容	資料庫實際欄位內容
B0	01
10	02
B2	03
12	04

我記得 NTT Michael 有在講義中說過，但我實在沒空在重聽一次，所以現在只要碰到這種問題，就先用這個方式去取得資料庫正確的對照值，我不清楚為何會有這種轉換(感覺很像是紀錄 Index 值)，想不懂目的何在?但是要寫 ABAP 直接撈資料庫，就要知道正確的欄位內容值，等日後了解原由後，再來補充說明。

➢ 如何在 SAP 報表新增一個按鈕，並執行某些程式？如下圖



參考 blog：https://blog.csdn.net/xuwj2006/article/details/49927013

簡單的說，直接完全照抄程式碼即可，我自作聰明少貼 TABLES: sscrfields. 當然就報錯了。

```
13    *&---------------------------------------------------------------------*
14    *& Version: V1.12
15    *&---------------------------------------------------------------------*
16    REPORT ZU_LOG2OMS.
17
18    TYPE-POOLS: slis.
19    TYPE-POOLS: icon.
20
21    TYPES: BEGIN OF sumReport,
```

```
32    DATA wa_data LIKE LINE OF itab.
33
34    TABLES: sscrfields.
35    DATA: functxt TYPE smp_dyntxt.
36
37    DATA: wa_fieldcat TYPE slis_fieldcat_alv ,   " 相?于工作?
```

```
92      P_FPATH TYPE EPSDIRNAM OBLIGATORY DEFAULT '/SAP/OUT/ALL'.
93    SELECTION-SCREEN END   OF BLOCK blk3.
94
95    SELECTION-SCREEN FUNCTION KEY 1.  "Button: 下載規格書
96
97    *---------------------------------------------------------------------*
98    *  Initialization
99    *---------------------------------------------------------------------*
100   INITIALIZATION.
101
102     functxt-icon_id   = icon_export.
103     functxt-quickinfo = '下載需求規格書'.
104     functxt-icon_text = '下載規格書'.
105     sscrfields-functxt_01 = functxt.
106
107     CLEAR:
108       T_DOWN,  T_DOWN[],
109       T_BINARY, T_BINARY[].
```

```
*&---------------------------------------------------------------------*
*& Report ZCK001_R02
*&---------------------------------------------------------------------*
```

```
REPORT ZCK001_R02.

TYPE-POOLS: slis.
TYPE-POOLS: icon.

TABLES: sscrfields.
DATA: functxt TYPE smp_dyntxt.

SELECTION-SCREEN FUNCTION KEY 1.  "Button: 下載規格書

TYPES: BEGIN OF sumReport,
  ZDATE TYPE D,

*&---------------------------------------------------------------------*
*& Initialization.
*&---------------------------------------------------------------------*
INITIALIZATION.
    functxt-icon_id   = icon_export.
    functxt-quickinfo = '促進價結算'.
    functxt-icon_text = '促進價結算'.
    sscrfields-functxt_01 = functxt.

*&---------------------------------------------------------------------*
*& Initialization.
*&---------------------------------------------------------------------*
AT SELECTION-SCREEN.
  CASE sscrfields-ucomm.
    WHEN 'FC01'. "..........下載規格書
      "PERFORM .....
  ENDCASE.

START-OF-SELECTION.
  "設定 alv 參數
  PERFORM frm_set_fieldcat USING:
```

另外，如果要設定 icon，程式是

```
functxt-icon_id    = icon_export.
```

請參考下面的 blog 設定即可

https://www.erpworkbench.com/abap/icons_list.htm

```
ICON_2 ICON_PAGE_LEFT                '@0G@'."  Page left
ICON_2 ICON_PREVIOUS_VALUE           '@0H@'."  Previous value;
ICON_2 ICON_NEXT_VALUE               '@0I@'."  Next value;
ICON_2 ICON_ANNOTATION               '@0J@'."  Note; remark
ICON_2 ICON_CREATE_NOTE              '@0K@'."  Create note
ICON_2 ICON_DISPLAY_NOTE             '@0L@'."  Display note
ICON_2 ICON_CALCULATION              '@0M@'."  Costing
ICON_2 ICON_GRAPHICS                 '@0N@'."  Graphics
ICON_2 ICON_CREATE_TEXT              '@0O@'."  Create text
ICON_2 ICON_DISPLAY_TEXT             '@0P@'."  Display text
ICON_2 ICON_CHANGE_TEXT              '@0Q@'."  Text
ICON_2 ICON_VARIANTS                 '@0R@'."  Variants
```

- SAP 的商品檔 SKU 系統會自動補 0 為 18 碼的動作，導致 Open SQL 在 JOIN 時，如果有自行新增的 Z-Table 中的商品欄位沒有補 0，就會無法 join，今天搞了半天才搞懂要如何處理。
  - ✓ 前情提要

    自行新增 ZFD005_T05 檔期促銷檔，內含 MATNR 欄位，並寫了一支 Excel 匯入程式(範例如下)

	A	B	C	D	E	F
1	1191101	9533	20190601	20190630	2.0	
2	1191101	62248	20190601	20190630	0.2	
3	1191101	9473	20190601	20190630	0.5	
4	1191101	46217	20190601	20190630	0.5	
5	1191101	315913	20190601	20190630	3.0	
6	1191101	61314	20190601	20190630	3.0	
7	1191101	66900	20190601	20190630	3.5	
8	1191101	1004753	20190601	20190630	2.5	
9	1191101	66243	20190601	20190630	4.5	
0	1191101	66244	20190601	20190630	3.0	
1	1191101	49771	20190601	20190630	2.0	
2	1191101	47393	20190601	20190630	1.0	
3	1191101	47394	20190601	20190630	1.0	

第一版的 ZFD006_U03 匯入程式沒有特別處理，導致下列 SQL 無法 JOIN

```
ORDER BY C01~WERKS,C01~ZCONTRACT.

"計算業績收入
select
  prctr,c~MATNR,sum( case when c~shkzg = 'H' then dmbtr * 100 else dmbtr * -100 end ) AS dmbtr,
     sum( case when c~shkzg = 'H' then menge else menge * -1 end ) AS QTY INTO CORRESPONDING FIE
from zfd004_c02 AS c
left join kna1 AS k on k~kunnr = c~prctr
where k~katr1 = '20'   "加盟
  and c~budat >= ( select DISTINCT SDATE from ZFD005_T05 where prof = @s_prof )
  and c~budat <= ( select DISTINCT EDATE from ZFD005_T05 where prof = @s_prof )
  and c~MATNR in ( select MATNR from ZFD005_T05 where prof = @s_prof )
  and c~hkont >= '0041110010' and c~hkont <= '0041110020'
group by prctr,c~MATNR.
```

  - ✓ 解決方式

    https://blog.csdn.net/xuwj2006/article/details/49927383

修改匯入程式，呼叫 RFC 讓 MATNR 補足，因為系統本身會這樣處理，如果客製程式搞特殊，就會造成不一致，後患無窮，所以還是要乖乖地照規矩來。

```
126  FORM frm_save.
127    * Uploading the data into the database table
128    ok_no = 0.
129    err_no = 0.
130    LOOP AT T_UPLOAD INTO WA_UPLOAD.
131      "判斷資料是否已存在,若已存在 skip
132      select count( * ) into @recno from ZFD005_T05 where PROF = @WA_UPLOAD-PROF and MATNR =
133
134      if recno = 0.
135        CALL FUNCTION 'CONVERSION_EXIT_MATN1_INPUT'   "前綴补零
136        EXPORTING
137          input  = WA_UPLOAD-MATNR
138        IMPORTING
139          output = WA_UPLOAD-MATNR.
140
141        ZFD005_T05-PROF = WA_UPLOAD-PROF.
142        ZFD005_T05-MATNR = WA_UPLOAD-MATNR.
143        ZFD005_T05-SDATE = WA_UPLOAD-SDATE.
```

- 當有大資料要計算、更新到 DataBase 時，如果分批(如每天更新一次 DB) MODIFY，實際上會等到程式全部執行完才會真得更新到資料庫，如何強迫真的寫入 DB?
  https://blog.csdn.net/wq122289014/article/details/78183381

```
……
IF FLAG = 'X'.
    ROLLBACK WORK.
ELSE.
    COMMIT WORK.
ENDIF.
```

在示例中每個 INSERT 語句後，程 序將檢查操 作是否成功 ，或檢查具 有相應的主 碼欄位（CARRID 和 CONNID）的行是否已 經在 SPFLI 中存在。稍 後，程式將 FLAG 設置為 “X” 。如果對於 每一個 INSERT 語句，SY-SUBRC 沒有設置為 0，那麼最 後一個 IF 語句中的 ROLLBACK WORK 語句將取消 對資料庫的所有更改。 否則，用 COMMIT WORK 語句確認這 些更改。

- 續上，Internal Table 使用 CLEAR 和 FREE 錯誤，會嚴重的影響效能

內表：如果使用有表頭行的內表，CLEAR 僅清除表格工作區域。要重置整個內表而不清除表格工作區域，使用 REFRESH 語句或 CLEAR 語句 CLEAR <itab>[]. ; REFRESH 加不加中括弧都是只清內表，另外 REFRESH 是專為清內表的，不能清基本類型變數，但 CLEAR 可以

以上都不會釋放掉內表所佔用的空間，如果想初始化內表的同時還要釋放所佔用的空間，請使用：FREE <itab>.

- Excel 匯入功能，當大筆數時效能很差，請改用文字檔匯入
  實測程式，資料越大(超過千筆) 效能會差到 10 倍以上，所以盡量使用文字檔匯入，此處不在舉例。
- 要特別小心 NUMC 欄位的補 0 問題，一不小心就會無法 Join 到資料安具
  一般要注意的是下面的欄位
  MANTR 商品編號

LIFTA 廠商編號

單據表身的 ITEM，我看到一些設定大多是 NUMC 3-5，如物料文件和採購文件就不同，當新開 Z-Table 時，如果用到表身項次欄位時，一定要特別注意 NUMC 的長度，否則 Join 會取不到資料，更麻煩的是 OpenSQL 的 udf 非常弱，很多語法都不支援，之前犯過幾次錯誤都是發現後只能重新變更 DE，非常麻煩。

https://blog.csdn.net/xuwj2006/article/details/49927383

```
CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'   "前缀补零
EXPORTING
  input  = i_lifnr
IMPORTING
  output = i_lifnr.

CALL FUNCTION 'CONVERSION_EXIT_ALPHA_OUTPUT'  "前缀去零
EXPORTING
  input  = i_lifnr
IMPORTING
   output = i_lifnr.

CALL FUNCTION 'CONVERSION_EXIT_MATN1_INPUT'   "料号前缀补零
EXPORTING
  input  = i_matnr
IMPORTING
  output = o_matnr.

CALL FUNCTION 'CONVERSION_EXIT_MATN1_OUTPUT'  "料号前缀去零
EXPORTING
   input  = i_matnr
IMPORTING
  output = o_matnr.
```

- 直接合併多個 Internal Table 的寫法

  因為一般要合併 Internal Table 的作法(網上查到的大部分範例) 都是跑 LOOP 迴圈，然後逐筆寫到另一個 Internal Table

```
DATA: BEGIN OF C OCCURS 0.
   INCLUDE STRUCTURE A.
   INCLUDE STRUCTURE B.
   end of C.

LOOP AT A.
```

```
  MOVE-CORRESPONDING A to C .
  LOOP AT B where <condition>.
    MOVE-CORRESPONDING B to C .
    APPEND C.
  ENDLOOP.
ENDLOOP.
```

有沒有辦法直接合併?

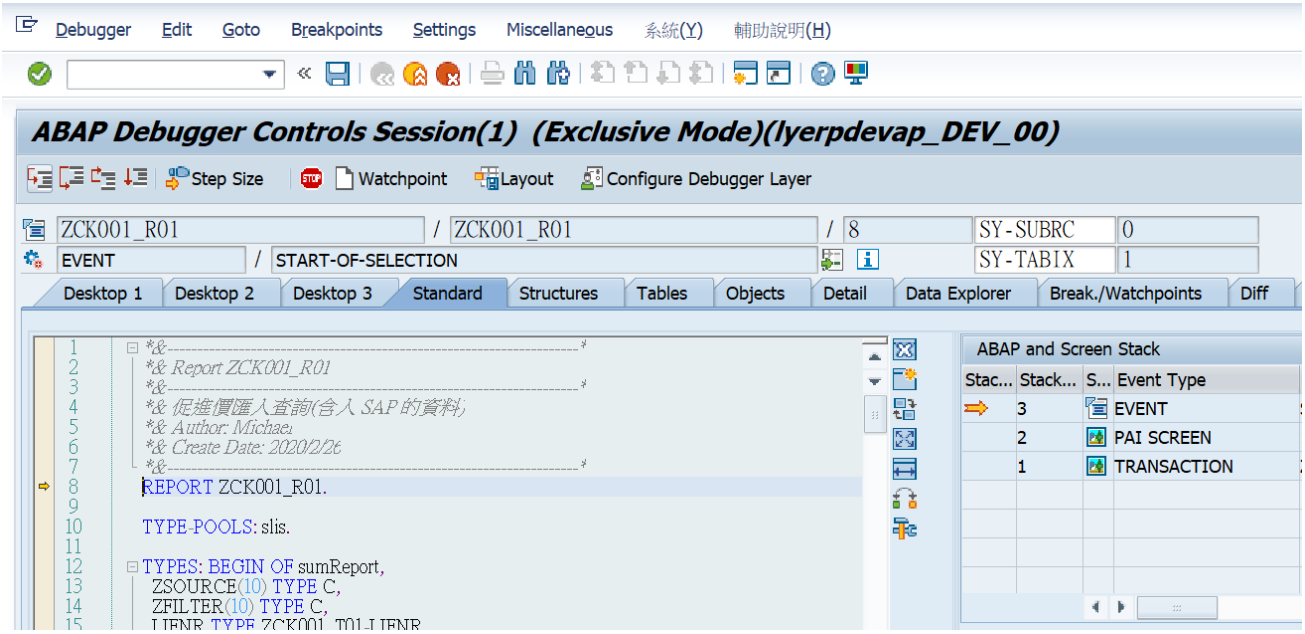
```
TYPES tpl_tab TYPE STANDARD TABLE OF sumReport WITH DEFAULT KEY.
DATA itab TYPE tpl_tab WITH HEADER LINE.
DATA wa_data LIKE LINE OF itab.
DATA:
  itab_rule1 TYPE tpl_tab WITH HEADER LINE,
  itab_rule2 TYPE tpl_tab WITH HEADER LINE,
  itab_rule3 TYPE tpl_tab WITH HEADER LINE,
  itab_rule4 TYPE tpl_tab WITH HEADER LINE.
DATA: gd_tabix type sy-tabix.

APPEND LINES OF itab_rule1 TO itab.
```

- READ TABLE 有很多使用的語法限制，可以改用 FILTER 過濾 Internal Table 的內容，基本上大部分的類 SQL 的 Where 都可以使用。

```
  DATA :
  itab_rule1 TYPE STANDARD TABLE OF t_tab with NON-UNIQUE SORTED KEY pk_price COM
PONENTS LIFNR MATNR ZSDATE ZEDATE,
  itab_rule2 LIKE itab_rule1,
  itab_rule0 LIKE itab_rule1,

  FREE itab_rule0.
  itab_rule0 = FILTER #( itab_rule1 USING KEY pk_price where LIFNR = lv_LIFNR AND
 MATNR = lv_MATNR AND  ZSDATE <= lv_date AND ZEDATE >= lv_date ).
  LOOP AT itab_rule0 INTO wa_return1.
    lv_P1 = wa_return1-INPRICE.
    lv_D1 = wa_return1-ZCDATE.
  ENDLOOP.
```

- 如何不進入 se38，就可以執行程式並直接進入 Debug Mode?
  - 先執行要 Debug 的 T-Code，如 ZCK001
  - 在 Command 下 /h

- 再執行就會進入 Debug Mode(程式的最開頭)



- 關於 Internal Table(內表) 及 Working Area(工作區) 的一些更明確的用法定義，此處以目前 Coding 有用過的語法說明觀念及寫法
  - 標準宣告 wa 和 itab
  - Table Type(Standard、Sorted、Hash)
  - LINE vs WITH HEADER
- 善用指標可提高執行效能 (itab=Internam Table wa=Working Area)

  剛開始寫 ABAP 會先由 INTO wa_xxx 入手，後來會改用 FIELD-SYMBOLS 以加強程式的執行效能。因為用 Cursor 首先就不用宣告 wa，然後少掉了在 itab 和 wa 之間要不斷的讀寫，當資料量大時，執行的效能會有顯著的差異。

用 Cursor 甚至連 DATA 都可以不必事先宣告(如下寫法)

READ TABLE itab INDEX 1

ASSIGNING FIELD-SYMBOLS <fs>

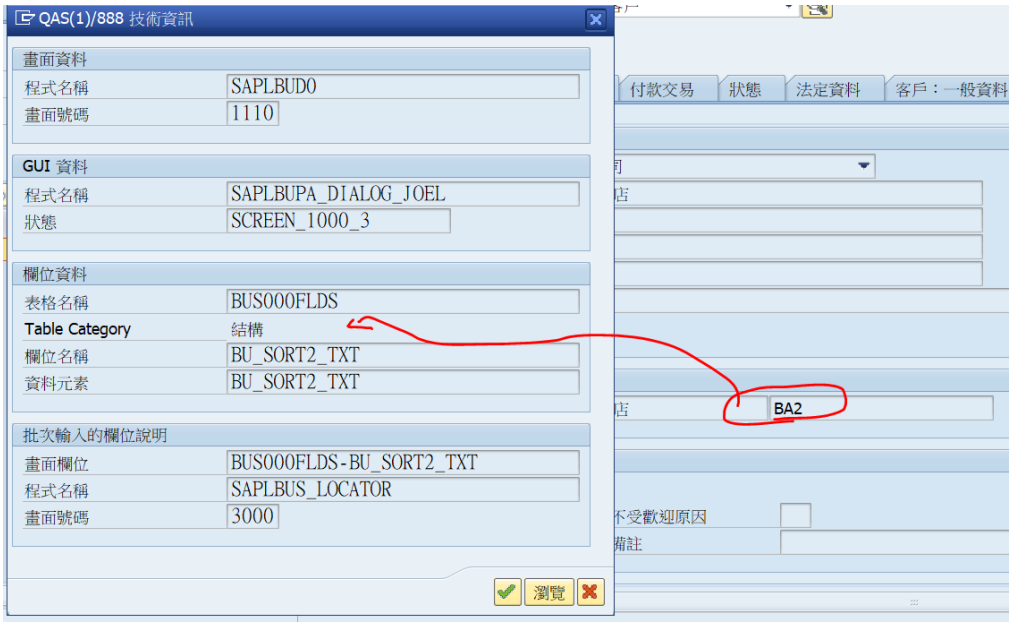
- 另類的 Function Module 的寫法(external subroutine)

Function Module 的寫法因為要求嚴謹，導致過於繁複，開始寫程式前一堆設定很讓人煩惱，今天 Spancer 有提到一種簡易的寫法，就像是以前呼叫 Lib 的寫法，只要呼叫時多指定程式代號即可，語法及範例如下。

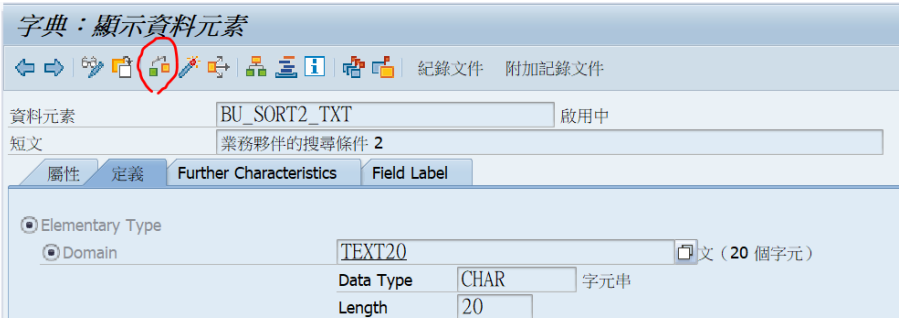
```
PERFORM <f_select_data> IN PROGRAM <program_b>.
```

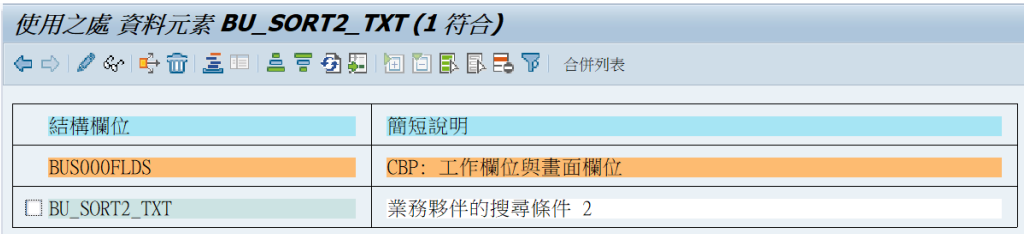
可以把例如殘月計算，取得加盟相關報表基本資訊等，都統一寫在一支 report 中，這樣可以大幅簡化程式的開發。目前很多程式都是複製修改，這對後續的維護不是件好事。

- 如果用 F1-技術資訊的欄位內容是 [結構]，該如何取得真正的 Table 及 Field 資訊?

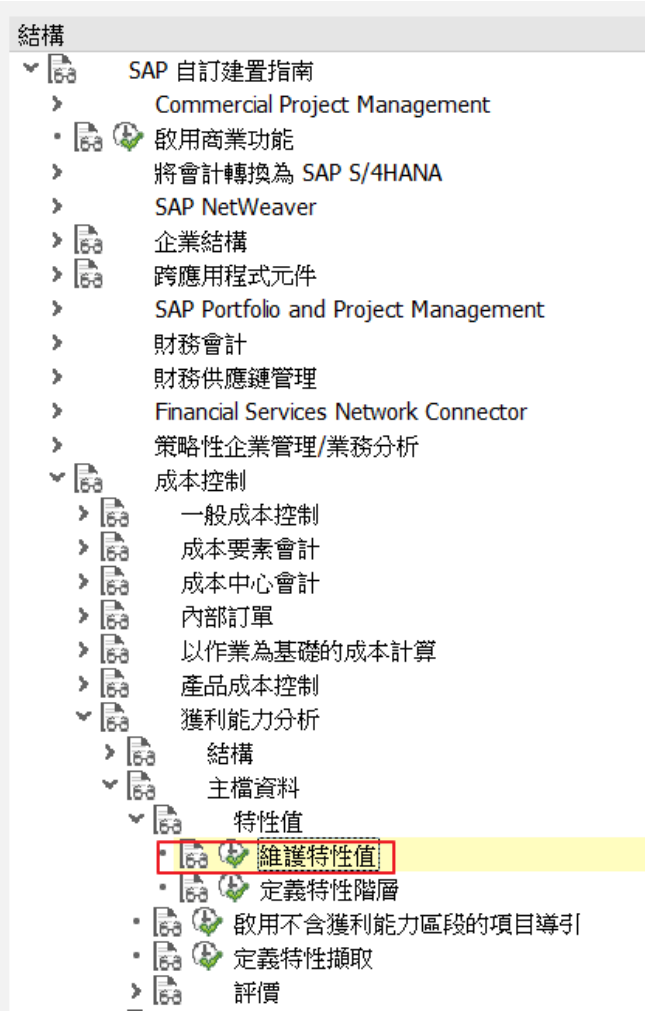


Spancer 建議： SE11 找出哪個 Structue 中的哪個欄位,再用 where-used 去找出所在程式去看哪個 Table 搬入的 Structure，但這樣我只查到結構定義，還是查不到對應的 Table-Fields。





Wade 則是叫我直接到 spro 找，因為他知道實際對照的設定處



至於該欄位的欄位名稱，我是根據內容(BA2) 比對查詢 ZMM501_T01 查詢實際資料，但是，這個欄位我是從 BP 的地址頁籤啟動，這可能是 enhance 吧? 我目前無解，後續再補充說明。

ZMM501_T01	門市相關日期及證照資料	HKONT	總帳科目		C	000020
ZMM501_T01	門市相關日期及證照資料	REGION	銷售處		C	000002
ZMM501_T01	門市相關日期及證照資料	ZDEPT	銷售區		C	000006
ZMM501_T01	門市相關日期及證照資料	SUPP_SITE	供應工廠		C	000008

- end

附錄一 SAP 常用 T-Code 列表

Module	T-Code	Description
MM	MM01	創建一般物料 Create Material – General
	MM02	修改一般物料 Change Material
	MM03	顯示一般物料 Display Material
	MM04	顯示修改 Display modify
	MK01	建立供應商 Create vendor (Purchasing)
	MK02	修改供應商 Change vendor (Purchasing)
	MK03	顯示供應商 Display vendor (Purchasing)
	Mk05	凍結供應商 Block vendor (Purchasing)
	MK06	在供應商上作標記以備刪除(採購) Mark vendor for deletion (purch.)
	ME11	建立採購資訊記錄 Create Purchasing Info Record
	ME12	改變採購資訊記錄 Change Purchasing Info Record
	ME13	顯示採購資訊記錄 Display Purchasing Info Record
	ME28	採購審批 Release Purchase Order
	ME29N	採購審批（單個） Release Purchase Order(single)
	ME9F	消息輸出: 採購訂單 Message Output: Purchase Orders
	ME51N	建立採購申請 Create Purchase Requisition
	ME57	分配和處理申請 Assign and Process Requisitions
	ME58	訂單: 分配的請求 Ordering: Assigned Requisitions
	MIGO	庫存運做 Stock(貨物移動)
		MIGO_GR 採購單收貨
		MIGO_GI 發料
		MIGO_TR 移轉過帳
		MIGO_GS 外包後續調整
	MB51	進出庫查詢 Stock view
	MB52	顯示工廠庫存可用量
	MMBE	庫存概觀
	MB5B	過帳日庫存(即歷史庫存)
	MBLB	外包商的庫存
	MB54	供應商的託售庫存
	MB58	客戶的託售庫存
	MB5T	顯示在途庫存
	ZMMT020	查詢物料倉庫庫存和儲位
	ME1L	每供應商的資訊記錄 Info Records Per Vendor
	ME2L	按供應商分類採購訂單 Purchase Orders by Vendor
	ME1M	每物料的資訊記錄 Info Records Per Material

	ME2M	按物料分類採購訂單 Purchase Orders by Material
	MB1A	發貨
	MB1B	轉倉
	MB1C	收貨
	ME81N	訂單值分析
	MCE1	採購組分析
	MCE3	供應商分析（採購）
	MCE5	物料組分析（採購）
	MB5B	庫存按過帳日期
	MB90	來自貨物移動的輸出
	MB1B	輸入轉帳
	ME23N	顯示採購訂單
	MBST	取消物料憑證
	ME2C	每個商品組的採購憑證
	ME2N	每個憑證號碼的採購憑證
	MI01	創建實地盤存憑證
	MI02	改變實物盤存憑證
	MI03	顯示實地盤點憑證
	MI04	輸入庫存計數
	MI05	改變庫存計數
	MI06	顯示庫存計數
	MI07	庫存差額總覽計數
	MI09	輸入無參考憑證計數
	MI20	庫存差額清單
	MI21	列印庫存盤點憑證
	MI22	顯示物料的實際盤點憑證
	MI23	顯示商品的庫存盤點資料
	MI24	庫存盤點清單
	MB54	顯示供應商寄售庫存
	MCBA	工廠分析
	MCBC	庫存地分析
	MCBE	物料分析
	MCBR	批次分析
	MC.1	工廠分析：庫存
	MC.2	工廠分析：收貨/發貨
	MB21	創建預留
	MB22	修改預留
	MB23	顯示預留

	MB25 MBVR	預留清單 管理預留
	MIRO MIR4 MR8M MRBR MIR5 MIR6 MRRL MRKO ZMMR020	發票驗證 顯示發票文件 取消發票文件 核發凍結發票 發票文件查詢 發票概觀 定期收貨結算 列印對帳單 發票驗證金額差異分析報表
	MI01 MI02 MI03 MI21 MI04 MI05 MI06 MI07 MI20	建立盤點文件 更改盤點文件 顯示盤點文件 列印盤點文件 輸入帶有文件的存貨計數 更改存貨盤點 顯示存貨盤點 處理差額清單 列印差額清單
	MI22 MI23 MI24	盤點清單　顯示物料的實際存貨文件 顯示物料的實際存貨資料 盤點清單
FI	FS00 FSP0 FSS0	編輯總帳科目集中地 編輯總帳科目在科目表中 編輯總帳科目在公司代碼中
	FI01 FI12	創建銀行 創建銀行主記錄
	FD01 FD02 FD03 FD05 FD06	創建客戶主資料 更改客戶主資料 顯示客戶主資料 凍結/解凍客戶主資料 刪除客戶主資料
	FB70 FB75	創建客戶發票 貸方憑證輸入收款
	F-36 F-28 F-32	創建客戶匯票收款----銀行承兌匯票 客戶收款清帳 單純的清帳業務，沒有銀行存款的介面

	F-48 F-40 F-53 F-53	創建供應商一般預付款 辦理匯票對供應商付款 供應商付款清帳 公司員工報銷付款
	FB60 FBRA FB65	創建供應商發票 重置已清項 創建公司員工借款
	AS01 AS05 AS06	固定資產主數據 固定資產主資料的鎖定解鎖 固定資產主資料的刪除
	F-92 F-90	各子公司調撥固定資產，或者直接對外銷售固定資產 資產有供應商&無供應商的購置
	ABAVN ABAON ABAA AW01N	資產的無收入報廢 資產的有收入無客戶報廢 對資產進行計畫外折舊 固定資產查詢
	AIAB	在建工程建立
	MMPV MMRV	物料期間的開關帳 物料期間是否允許記帳進行控制
	S_ALR_87003642	新財務期間帳打開
	FAGLF101	應收應付帳款月末 AR/AP 重組
	F.13 F.19 F.07	自動清帳無清算貨幣說明 GR/IR 清帳重組 客戶/供應商餘額的結轉
	FAGL_FC_VAL S_BCE_68000174 FAGLGVTR	外幣評估(新) 匯兌損益月結兌換比率 總帳科目餘額的結轉
	AJRW AJAB OAAQ	固定資產財政年度更改 固定資產年終結帳 撤銷固定資產年終結
SD	VD01 VD02 VD03	建立客戶 Create customer 更改客戶 Change customer 顯示客戶 Display customer
	VAP1 VAP2 VAP3	建立聯絡人(客戶) 更改聯絡人 顯示聯絡人
	VPE1 VPE2	銷售人員建立 銷售人員更改

	VPE3	銷售人員顯示
	VK11 VK12 VK13	創建條件 Create Condition 修改條件 Change Condition 顯示條件 Display Condition
	VV11 VV12 VV13	銷售文件建立 銷售文件更改 銷售文件顯示
	VV21 VV22 VV23	出貨建立 出貨更改 出貨顯示
	VA21 VA22 VA23	創建報價 Create Quotation 更改報價 Change Quotation 顯示報價 Display Quotation
	VA01 VA02 VA03 VA05	創建銷售訂單 Create Sales Order 修改銷售訂單 Change Sales Order 顯示銷售訂單 Display Sales Order 銷售訂單清單 List of Sales Orders
	VL01N VL02N VL03N VL06O	創建交貨 Create Delivery 更改交貨 Change Delivery 顯示交貨 Display Delivery 查詢銷售訂單 Query Sales Order
	VLSP VL22 VL09 VL10D VKM5	拆分銷售單 Split Sales Order 拆分發貨單 Split Delivery 取消發貨過帳 Cancel Goods Issue for Delivery Note STO 發貨單創建 Extended Mult. Processing Deliveries 對因信用控制凍結的發貨單解凍 Block Delivery
	VT01N VT02N VT03N	創建發運 Create shipment 更改發運 Change shipment 顯示發運 Display shipment
	VF01 VF02 VF03 VF04	創建出具發票憑證 Create Billing Document 更改出具發票憑證 Change Billing Document 顯示出具發票憑證 Display Billing Document 處理出具發票到期清單 Process Billing Due List
	VF11	取消出具發票憑證 Cancel Billing Document

附錄二 SAP 常用 Table 列表

Module	Table	Description	Table	Description
MM	MARA	常規物料數據	EBAN	採購申請
	MARC	物料的工廠數據	EBKN	採購申請帳戶設置
	MARD	物料的倉儲位置數據	EKBE	採購憑證歷史
	MARM	物料的計量單位	EKBEH	刪除的 PO 歷史記錄
	MARV	物料控制記錄	EKBZ	每個採購憑證的歷史:交貨費用
	MBEW	物料評估	EKBZH	每個採購憑證的歷史:交貨費用
	MAKT	物料說明	EKKI	採購條件索引
	MAST	BOM 連接物料	EKKN	採購憑證中的帳戶設置
	MVKE	物料銷售數據	EKKO	採購憑證擡頭
	MAPR	有關預測的物料索引	EKPA	採購中的合夥人規則
	MSTA	物料主記錄狀態	EKPB	採購憑證中的"已供物料"項
	LFA1	供應商主數據(一般地區)	EKPO	採購憑證項目
	LFAS	供應商主數據(增值稅登記號的一般部分)	EKPV	採購憑證項目的庫存轉儲裝運數據
	LFAT	供應商主數據(稅務分組)	EKRS	ERS 過程:將被開票的貨物(商品)移動
	LFB1	供應商主數據(公司代碼)	MKPF	擡頭:物料憑證
	LFB5	供應商主數據(催款數據)	MSEG	憑證段:物料
	LFBK	供應商主數據(銀行細目)	RBKP	憑證表頭:發票收據
	LFBW	供應商主記錄(預扣稅類型)X	RSEG	憑證項目:收款發票
	LFC1	供應商主數據(業務額)	EQUK	配額文件:標題
	LFC3	主供應商(特殊總分類帳交易圖表)	EQUP	配額文件:項目
	LFCPR	每一步驟的功能代碼參數文件	T027A	裝運須知,採購
	LFM1	供應商主記錄購買組織數據	T027B	文本:裝運須知
	LFM2	供應商主文件記錄:採購數據	T027C	與裝運須知相符的碼
	EINA	採購信息記錄 - 一般數據	T027D	與裝運須知相符的碼
	EINE	採購信息記錄 - 採購組織數據	IKPF	擡頭:實地盤存憑證
	EIPA	訂購價格歷史:信息記錄	ISEG	盤點庫存憑證項目
	EORD	採購貨源清單	MCHA	批次
	EKUB	有關物料的庫存運輸單的索引	MCHB	批量庫存
	ESKN	服務包中的帳戶分配	EKAB	批准憑證
	MBPR	生產倉位庫存	EKAN	供應商地址:採購憑證
	MLAN	物料的稅分類	EKEH	計劃協議批准憑證

	MLGN	每一倉庫號物料數據	EKEK	計劃協議批准的標題數據
	MLGT	每一存儲類型的物料數據	EKES	供應商確認
	MVER	物料消耗量	EKET	計劃協議計劃行
	MVKE	物料銷售數據	EKETH	計劃協議計劃:歷史表
	T001L	倉儲地點	LTBK	轉儲請求擡頭
	T001W	工廠/分支機構	LTBP	轉儲請求項目
	NRIV	號碼範圍間隔	LTAK	WM 轉儲單擡頭
	RKPF	憑證頭:預定	LTAP	轉儲單項目
	RESB	預定/相關需求	MARD	查主庫存
	T024	採購組	MARDH	主庫存歷史記錄
	T024E	採購組織	MSLB	特殊庫存 O
	T024W	工廠的有效採購組織	MKOL	批次庫存 K/M
	T024Z	採購組織	MSKA	批次庫存 E
	T023	物料組	MSKU	批次庫存 V/M
	T156	移動類型	MCHA	批次庫存
	T157H	移動類型的幫助文本	MCHB	批次庫存
	KONP	條件(項目)	T16FC	批准代碼

附錄三 參考 SAP 相關網站列表

	網址	說明
1	https://help.sap.com/viewer/index	SAP Online Help
2	https://www.fenginfo.com/	楓竹丹青
3	https://www.591sap.com/	SAP 顧問圈(陸)
4	https://www.w3cschool.cn/sap_abap/sap_abap_overview.html	W3C School
4	file:///E:/SAP 文檔/SAP-Learn.html	屠夫的作品彙總(單機執行)
5	http://blog.sina.com.cn/sapba	SAP 吧的博客
6	http://www.sapjx.com/	SAPJX 技術博客
7	https://blog.csdn.net/heng0757	SAP 道長
8	https://www.cnblogs.com/hhelibeb/	氢氦
9	https://blog.csdn.net/szlaptop	兔寶學習 ABAP 整理筆記
10	https://blog.csdn.net/sapliumeng/article/details/94605519	BAPI DEMO 文章汇总
11	https://blog.csdn.net/xuwj2006	Evan 技術專博