第二章 問題定義及背景知識

2-1 問題定義

令 $I = \{i_1, i_2, \ldots, i_m\}$表示資料流應用中所有的資料項（item）種類所成的集合，一筆交易 T 是指由 I 集合中一個或一個以上的資料項所組成的集合，亦即是 I 的一個非空子集合。本論文中所考慮的資料流型態如下：在每一個時間點 k（$k \geq 1$）皆會有一筆新的交易 T_k 輸入。至時間點 t 為止，資料流所形成的資料庫 $DS_t = \{T_1, T_2, \ldots, T_t\}$，包含了 t 筆交易，$|DS_t|$表示目前資料流庫的大小，亦指所包含的交易數目。

定一個視窗大小 w 值，且目前時間點為 t。令$t \geq w$，則**目前交易視窗** CTL_t（Current Transaction list）是由交易$\{T_{t-w+1}, T_{t-w+2}, \ldots, T_t\}$所構成的交易集合。$CTL_t$ 中的第一筆交易，以W_t^{first} 表示，即交易 T_{t-w+1}。當目前時間改為 t+1，CTL_{t+1} 相當於對 CTL_t 新增新交易 T_{t+1}，再移除原來的CTL_t^{first} ，CTL_{t+1} 中仍維持有 w 筆交易。舉例說明，當視窗大小 w 設為 5，從時間點 5 到 7 對應的 CTL 如圖 2.1 所示。

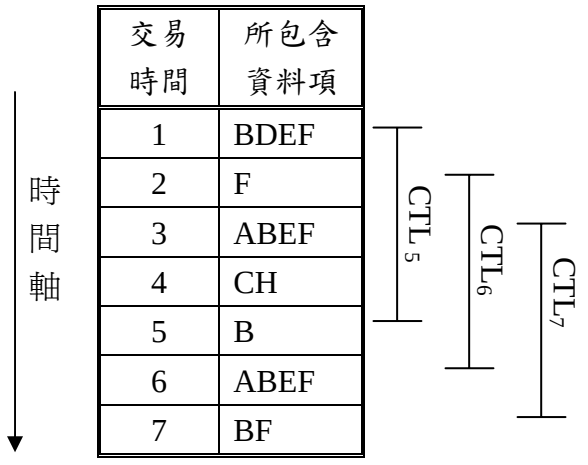

圖 2.1 目前視窗 CTL 移動情形

一個資料項集（itemset）是由 I 集合中一個或一個以上的資料項所組成的集合，為 I 的一個非空子集合。一個資料項集 e 內所包含的資料項個數稱為 e 的長度，以$|e|$來表示；而長度為 l 的資料項集，稱為一個 l-資料項集（l-itemset）。當一個資料項集 e 為交易 T 的子集合，則稱為 T **包含** e。在目前交易視窗 CTL_t 內包含資料項集 e 的交易筆數稱為 e 在 DS_t 中的**最近支持度計數值**，以 $RC_t^{DS}(e)$ 表示；**最近支持度**由 $RC_t^{DS}(e)$ 除以視窗大小 W 計算得之，以 $R\sup_t^{DS}(e)$ 表示。由使用者給定一個介於 0 到 1 之間的最小支持度門檻值（minimum support）S_{min} 及一個介於 0 到 S_{min} 之間的最大支持度誤差門檻值 ε。對一個資料項集 e，若 $R\sup_t^{DS}(e)$ 大於等於 S_{min} 資料項集，則稱 e 為 DS_t 中一個**最近常見資料項集**（Recent Frequent itemset）；若 $R\sup_t^{DS}(e)$ 小於 S_{min} 但大於等於 ε，則稱 e 為 DS_t 中一個**最近準常見資料項集**（Recent Sub-frequent itemset）；若 $R\sup_t^{DS}(e)$ 小於 ε，則稱 e 為 DS_t 中一個**非常見資料項集**（Recent Infrequent itemset）。

本論文的研究問題，便是要提出一個不需保留目前交易視窗，且在每個時間點能快速探勘出近似最近常見資料項集的方法。

2-2 FP-tree 結構及其動態調整方法

在本論文中，我們採用 FP-tree 結構來儲存目前交易視窗中出現資料項集的摘要資訊，並從中探勘出最近常見資料項集。因此在本節中以一個範例來說明 FP-tree 結構，及其動態調整方法。

2-2.1 FP-tree 結構

FP-tree（Frequent-Pattern tree）結構將資料庫中常見資料項所形成的資料項集緊密的壓縮儲存，並記錄這些資料項集在資料庫中的累計支持度計數值。除了根節點外，樹中每一個節點表示一個資料項集，並儲存對應的支持度計數值。

表 2.1 範例資料庫 *DB*

交易編號	所包含資料項	所包含常見 1-資料項集（依計數值遞減排序）
100	ABEFJ	BFEA
200	BDEH	BED
300	BCFI	BFC
400	ABDEFG	BFEAD
500	ACEF	FEAC
600	BCDF	BFCD

以表 2.1 所示之交易資料庫為例，若最小支持度門檻值設為 0.5，所建構之 FP-tree 如圖 2.2 所示，建構步驟如下：

步驟 1) 掃描一次資料庫，找出常見資料項及其支持度計數值。

掃描資料庫後，所有的資料項以支持度計數值由大到小遞減排序為{B:5,

F: 5, E: 4, A: 3, C: 3, D: 3, G: 1, H: 1, I: 1, J: 1}，支持度計數值大於等於 3（即 $6 \times 0.5 = 3$）的常見資料項及其支持度計數值有{B: 5, F: 5, E: 4, A: 3, C: 3, D: 3}。

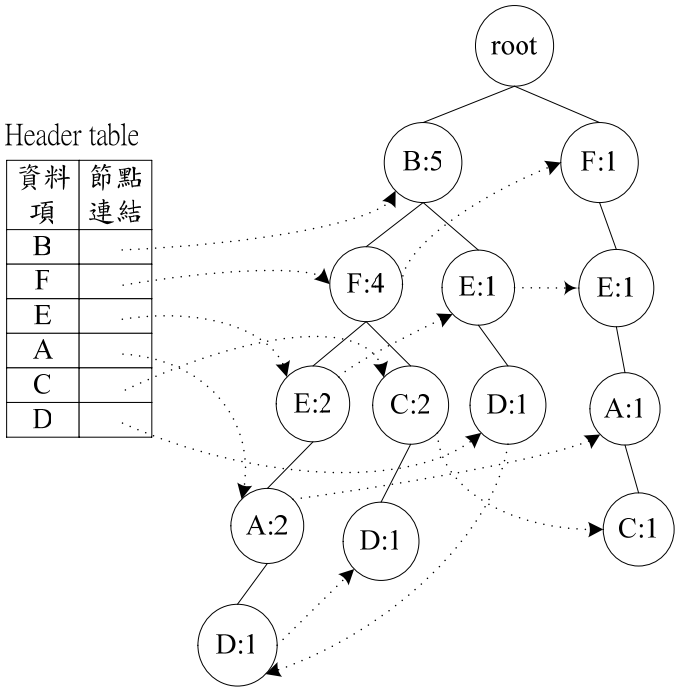

圖 2.2 範例資料庫 *DB* 之 FP-tree 結構

步驟 2) 建立 FP-tree。

首先，建立 FP-tree 的根節點，標示為空節點"null"。再次掃描資料庫，將各交易內容去除非常見資料項後，依常見資料項支持度計數值遞減順序排列，重新排列過的交易內容如表 2.1 最右欄所示。接下來將各交易中的資料項集加入 FP-tree 中。加入方式如下，從根節點開始，搜尋是否存在子節點包含與交易中第一個資料項相同的資料項。若已存在則將該對應子節點的計數值加 1，若不存則在根節點下新增一個對應此資料項的子節點，並將其計數值設為 1。再由目前所在的節點遞迴往下處理

交易中的其他資料項，直到交易中全部的資料項都處理完為止。以表 2.1 資料庫為例，最後建好的 FP-tree 如圖 2.2 中的節點及實線分支所示。

步驟 3) 建立 Header table。

為了方便探訪（traversal）FP-tree，FP-tree 中還會建立一個資料項表稱為 Header table，將具有相同資料項的節點串聯起來。Header table 中的資料是依常見資料項之支持度計數值遞減順序儲存。Header table 中有兩個欄位，第一個欄位為資料項欄，儲存常見資料項；第二個欄位為節點連結欄（node-link），儲存節點指標。如資料項欄的資料項為 X，透過節點指標所指向之串列，會將 FP-tree 中所有節點資料項為 X 的節點橫向串聯起來，如圖 2.2 中的虛線所示。

2-2.2 FP-tree 結構調整方法

在資料庫變動（可新增及刪除）的環境中，本實驗室曾提出一個有效動態調整 FP-tree 結構的 AFPIM 演算法[2]，以提供資料庫更新時能漸進探勘出常見資料項集。只需掃描異動的資料部分，來更新調整 FP-tree 結構，而不需整個重建 FP-tree 結構。

當資料庫內容有變動時，常見或準常見資料項及其支持度計數值大小順序可能有改變，已違反原本 FP-tree 的建構定義，需依這些資料項的支持度計數值大小，重新調整 FP-tree 結構中各路徑節點的順序。

調整 FP-tree 結構的方法為，採用泡沫排序法（bubble sort）依更新後的支持度計數值大小遞減排序資料項，在泡沫排序過程當兩個資料項需要做交換時，則表示要調整 FP-tree 結構中對應節點的順序位置，即交換其在原 FP-tree 結構內互為父子關係的兩節點之位置。

表 2.2 異動資料庫 DB^+ 及 DB^-

交易編號	所包含資料項	所包含常見 1-資料項集 （依更新後之計數值遞減排序）
700	ABDEF	FABED
800	AF	FA
900	ACD	ACD

DB^+

交易編號	所包含資料項	所包含常見 1-資料項集 （依更新後之計數值遞減排序）
200	BDEH	BED

DB^-

以表 2.1 資料庫為原始資料庫，而表 2.2 為異動資料庫為例，若最小支持度門檻值 S_{min} 設為 0.5 且最大支持度誤差門檻值 ε 設為 0.25，資料庫異動前的 FP-tree 結構如圖 2.3 所示，接下來調整 FP-tree 結構的過程如下：

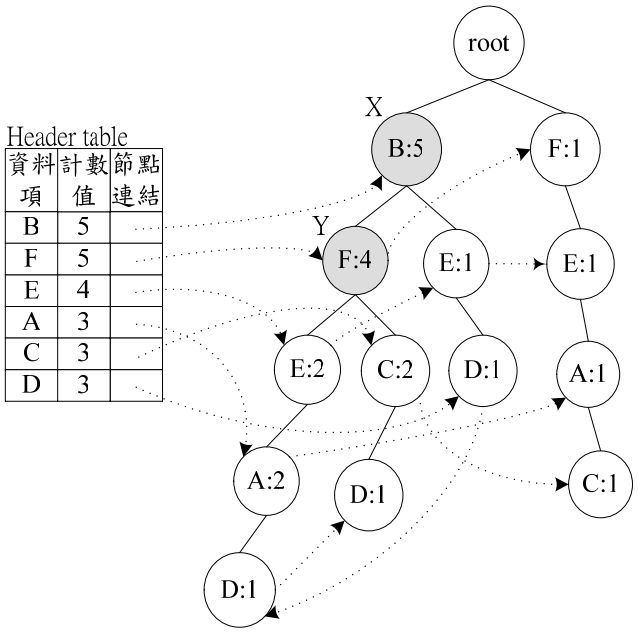

圖 2.3 資料庫異動前之 FP-tree 結構

步驟 1) 以氣泡排序法，決定需交換順序位置之節點。

在 Header table 新增一個計數值欄位，記錄常見及準常見資料項的支持度計數值，以方便當資料項之支持度計數值改變後依此值做遞減排序。原範例資料庫 DB 中支持度計數值大於等於 2（即 $\lceil 6\times 0.25 \rceil = 2$ ）的常見或準常見資料項有{B: 5, F: 5, E: 4, A: 3, C: 3, D: 3}，掃描一次異動資料庫後計數值更新為{ B: 5, F: 7, E: 4, A: 6, C: 4, D: 4}。以氣泡排序法，將資料項以新的支持度計數值由大到小做遞減排序，總共需要有三次資料項對(B, F)、(E, A)及(B, A)三次的交換調整順序，調整其對應節點在

FP-tree 結構中路徑上的順序。

步驟 2) 調整 FP-tree 結構內兩節點 X 與 Y 在路徑上的順序。

第一次調整，交換資料項 B 跟 F 之對應節點在 FP-tree 結構之路徑上的順序。延著資料項 B 的橫向節點連結檢查，以 X 表示正在檢查的節點。如果 X 有一個子節點 Y 的資料項為 F，則需調整節點 X 跟節點 Y 的順序位置，如圖 2.3 中標示為灰色的節點。調整的方法為，如果節點 X 中的支持度計數值大於節點 Y 中的支持度計數值，則執行以下步驟 2.1 到步驟 2.3；否則，則只需執行步驟 2.2 到步驟 2.3。

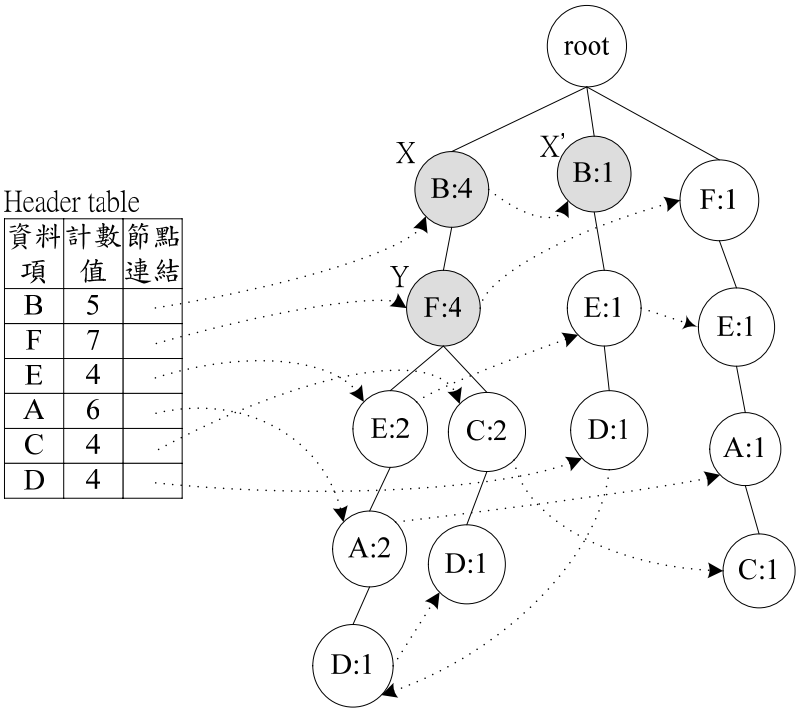

圖 2.4 新增節點 X'

步驟 2.1) 新增節點。

在節點 X 的父節點 P 下，新增一個子節點 X'。節點 X'的資料項設為節

點 X 的資料項，計數值為節點 X 的支持度計數值扣掉節點 Y 的支持度計數值。並把 X 下除了 Y 的其他子節點移到 X’下，並將 X’加入的節點橫向串連內。最後，把 X 的計數值設為與 Y 相同。

以圖 2.3 所示之例，新增步驟完成後之結果如圖 2.4 所示。

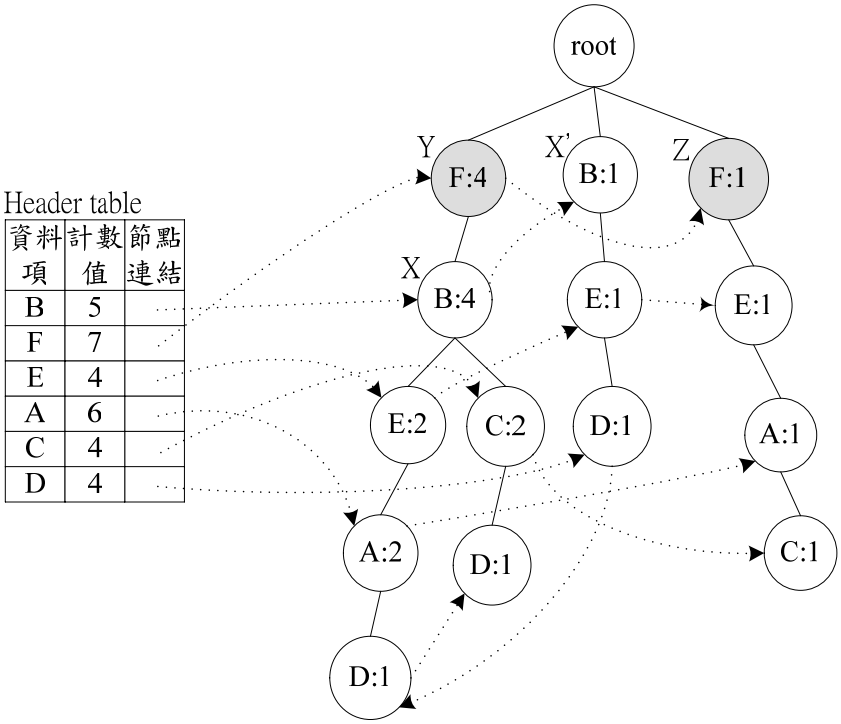

圖 2.5 交換節點 X 跟 Y

步驟 2.2) 交換節點。

交換節點 X 跟 Y 的父連結及子連結。

以圖 2.4 所示之例，交換步驟完成後之結果如圖 2.5 所示。

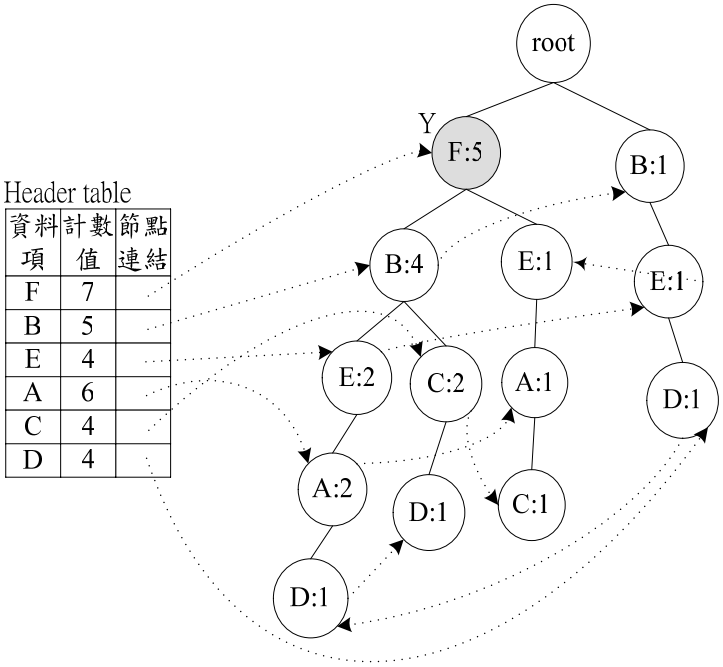

圖 2.6 合併節點 Y 跟 Z

步驟 2.3) 合併節點。

交換步驟後，在節點 P 下如果存在一個節點 Z 與 Y 具有相同的資料項，則合併 Y 跟 Z。將 Z 的計數值加到 Y，且把 Z 的所有子節點移到 Y 下。最後把 Z 從 FP-tree 結構內刪除。

以圖 2.5 所示之例，合併步驟完成後之結果如圖 2.6 所示。

FP-tree 結構內所有資料項 B 跟 F 的節點交換後，則交換 Header table 內資料項 B 跟 F 及其對應節點連結的儲存位置。

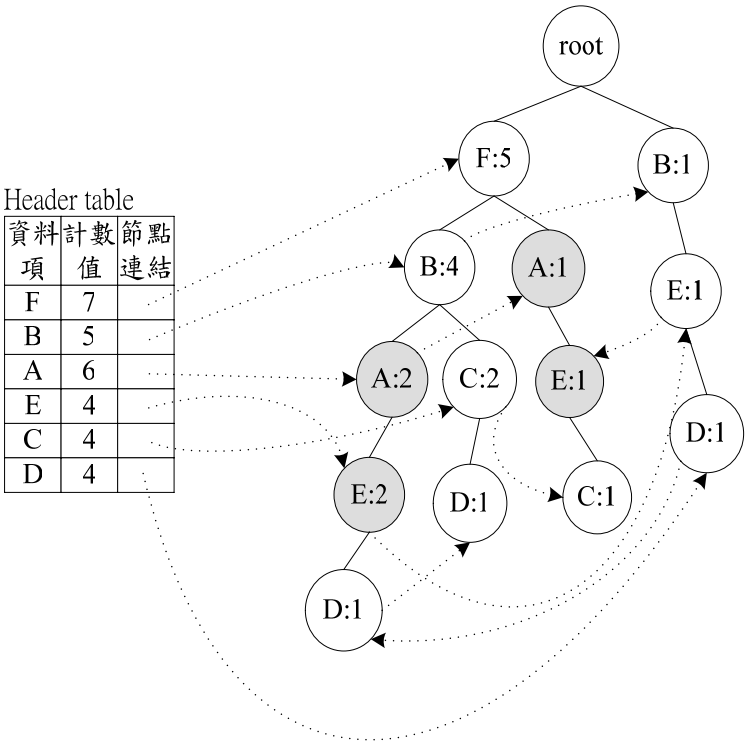

圖 2.7 第二回調整，交換資料項 E 跟 A

第二回調整，以同樣步驟交換資料項 E 跟 A 之對應節點在 FP-tree 結構路徑上的順序，交換結果如圖 2.7 所示。

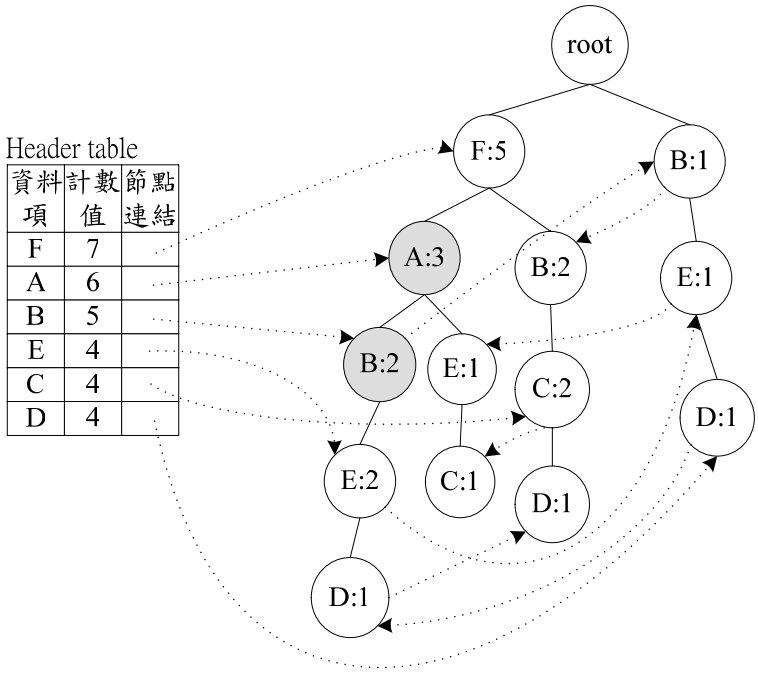

圖 2.8 第三回調整，交換資料項 B 跟 A

第三回調整，以同樣步驟交換資料項 B 跟 A 之對應節點在 FP-tree 結構路徑上的順序，交換結果如圖 2.8 所示。

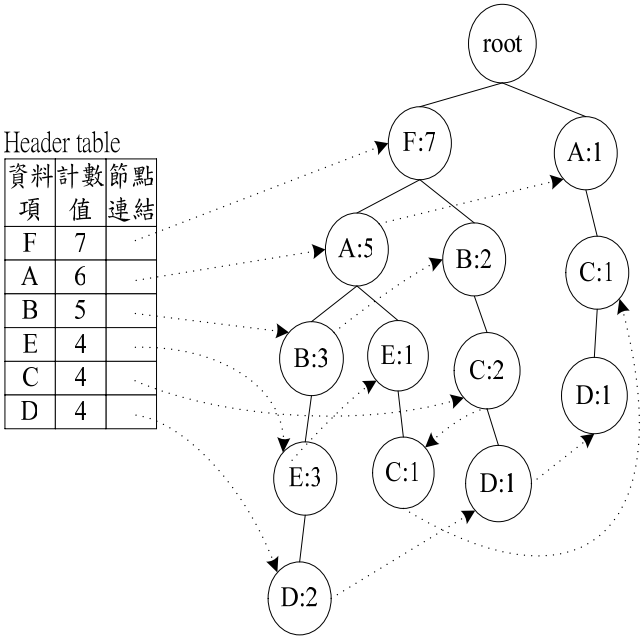

圖 2.9 資料庫異動後之 FP-tree 結構

步驟 3) 處理異動資料庫。

再次掃描異動資料庫，將各交易內容去除非常見資料項後，依更新後之支持度計數值遞減順序排列，重新排列過的交易內容如表 2.2 最右欄所示。將 DB^+ 及 DB^- 內所有交易分別加入 FP-tree 及從 FP-tree 中移除，處理完之結果如圖 2.9 所示。