

南 華 大 學

資訊管理學系碩士論文

無線廣播有效查詢過程中的資料配置使用 Apriori 演算法

Data Allocation on Wireless Broadcast Channels for  
Efficient Query Processing Using Apriori Algorithms



研 究 生：張桐旗

指 導 教 授：吳光閔 博士

中 華 民 國 九 十 二 年 六 月

# 南 華 大 學

## 碩 士 學 位 論 文

資 訊 管 理 學 系

無線廣播頻道有效查詢過程中的資料配置使用 Apriori 演算法

Data Allocation on Wireless Broadcast Channels for Efficient  
Query Processing using Apriori Algorithms

研 究 生：張 桐 旗

經 考 試 合 格 特 此 證 明

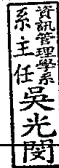
口 試 委 員：

邱 宏 彬

陳 志 榮

指 導 教 授：

吳 光 慶



所 長：

口 試 日 期：中 華 民 國 九 十 二 年 六 月 二 十 二 日

## 誌 謝

本論文撰寫與完成，非常感謝我的指導教授吳光閔博士用許多心力的指導批示，他親切誠懇、治學嚴謹的態度，引導我在論文研究上獲得明確的方向，使得本論文能順利如期完成，讓我永遠感念。

感謝邱宏彬教授、陳光榮教授，於百忙中詳加審閱我的論文，並在論文口試期間惠予良言卓見，並提供許多寶貴的修正意見。

更感謝前所長王昌斌教授，在我求學中給與相當多的教導。感謝同窗摯友信旭、佳昌、胤良、隆哥、全福、玉玲、昭邦、清涼、美燕、嘉文、健華、開德的相互支持與砥勵及系辦伊汝的協助。

感謝我的先父與母親，給與我精神上的鼓勵及民雄岳父及岳母在我求學階段給與生活起居上的照顧。在論文的撰寫過程中，感謝內人昭靜的體諒與幫忙做論文排版，聰穎的嘉晋及可愛的鳳書兩位寶貝犧牲了許多親子遊樂時間，讓我可以全心投入書本，使得論文的進度如期完成。謹以此文，獻上個人由衷的感激與敬意，並將這份榮耀與所有關心我的人一同分享。

張桐旗 謹謝

2003年6月於嘉義南華大學

# 無線廣播頻道有效查詢過程中的資料配置使用 Apriori 演算法

學生：張桐旗

指導教授：吳光閔 博士

南華大學資訊管理學系碩士班

## 摘 要

電腦科技進步，行動計算環境技術亦趨成熟，個人行動通訊設備的發展與普及，廣播資料將會被廣泛擴大的應用在網域。在本論文中，將資料庫發展由傳統資料庫、行動資料庫、廣播資料庫，探討資料庫演進以及資料探勘理論中關聯法則最主要的演算法之一 Apriori 演算法來探討無線廣播資料頻道的有效分配，使用者從廣播資料上執行複雜查詢時，可以呈現最佳查詢處理位置，將查詢的資料庫放在靠近每一個廣播頻道的位置，可以減少處理複雜的查詢存取時間與查詢處理的位置。實驗證明我們的方法與 Random 比，快 10%。

關鍵詞：無線廣播, 資料庫, 行動計算, 存取時間

# Data Allocation on Wireless Broadcast Channels for Efficient Query

## Processing using Apriori Algorithms

Student : Tung-Chi Chang

Advisor : Dr. Guang-Ming Wu

Department of Information Management  
The M..B..A..Program  
Nan-Hua University

### ABSTRACT

As computer technology is continue progress, the environment of mobile computing is more mature. The personal mobile communication equipment is development and popular, and the broadcast data will be widespread use on Internet. In this paper, we will probe into the evolution of database form its development: tradition database, mobile database, and broadcast database. And use Apriori algorithms, the one of the algorithms of data mining association rule to research the efficient of data allocation on wireless broadcast channels. When user make a complex query into the broadcast data, we can show the best position in the query processing. Put the query data near to each broadcast channel position, can reduce the access time of a complex query and query processing position. The result of experiment improve that our method is 10% faster than random.

**Keywords** : Wireless broadcast, Database, Mobile computing,  
Access time

# 目 錄

書名頁.....	i
碩博士論文授權書.....	ii
論文指導教授推薦函.....	iii
論文口試合格證明.....	iv
誌謝.....	v
中文摘要.....	vi
英文摘要.....	vii
目錄.....	viii
表目錄.....	xi
圖目錄.....	xii
第一章 緒論.....	1
第一節 研究背景與動機.....	1
第二節 研究目的與論文架構.....	2
第二章 資料庫發展.....	4
第一節 傳統資料庫的發展階段.....	4
壹、第一階段—無管理階段.....	5
貳、第二階段—檔案管理階段.....	5

參、第三階段—資料庫階段.....	6
第二節 分散式資料庫.....	7
壹、分散式資料庫系統.....	7
貳、分散式資料庫的資料分佈方式.....	8
參、分散式資料庫管理概.....	9
肆、分散式資料庫系統的架構.....	10
第三節 可以隨身攜帶的資料庫.....	11
壹、行動資料庫.....	11
貳、行動資料庫架構.....	11
參、行動資料庫系統的特色.....	12
肆、行動資料庫管理的挑戰.....	15
第四節 廣播資料庫.....	16
第三章 問題描述.....	18
第四章 資料探勘與 Apriori 演算法.....	20
第一節 資料探勘.....	20
第二節 關聯法則的相關技術.....	21
壹、關聯式規則.....	22
貳、Apriori 關聯法則.....	25
參、關聯法則的有關應用.....	26

肆、Apriori 演算法.....	27
第五章 實驗設計.....	32
第一節 實驗設計.....	32
第二節 排序方法.....	33
第三節 實    驗.....	35
第四節 實驗結果.....	44
第六章 結    論.....	51
參考文獻.....	52



## 表目錄

表 1	5 個 Client，資料欄位 10 與 Random 比較.....	44
表 2	10 個 Client，資料欄位 20 與 Random 比較.....	45
表 3	15 個 Client，資料欄位 30 與 Random 比較.....	46
表 4	20 個 Client，資料欄位 40 與 Random 比較.....	47
表 5	25 個 Client，資料欄位 50 與 Random 比較.....	48
表 6	資料長度與 Random 比較值.....	49

## 圖目錄

圖 1 無線廣播的基本架構圖.....	17
圖 2 知識發現之流程.....	21
圖 3 顧客被群聚成四個集群.....	25
圖 4 Apriori 演算法的過程.....	30
圖 5 我們的方法與 Random 比較效率曲線.....	50

# 第一章 緒論

## 第一節 研究背景及動機

無線廣播是我們平日所接觸到的媒體，與人們的生活緊密結合，是現代人不可或缺的中間媒體，GPRS為目前第三代無線通訊技術，在學術及產業界積極發展中的通訊技術。無線頻寬數據技術是未來的一個趨勢。第三代寬頻CDMA技術系統成熟之前，為改進目前第二代無線通訊技術，提供適當的數據服務。無線網路資源分配相關議題，在國立交通大學，電腦技術研發重點中心，於民國八十七年，整合型計劃中，有一項行動計算之研究，成功大學、中山大學、中央大學等也投入該研究領域中。

計算機工程於九十年代有一重大趨勢即是加強與通信工程之結合，其中以分散式計算 (Distributed Computing) 與行動通信 (Mobile Communication) 之結合最具挑戰性與趣味性。這結合即是所謂的行動計算 (Mobile Computing) [1,17]。行動計算為使人們在任何時間、地點，使用數位資源的一種技術，其發展乃基於無線通信技術之進步，其成功則有賴於通信及電腦軟硬體技術之整合。行動計算應用於資訊產業方面，分散式計算技術引進，使得行動通訊能提供更高層次的服務。行動計算系統包含無線系統、主幹網路、系統軟體以及應用軟體等四種領域，依此一階層分類，規劃行動計算的研究方向有：

### 一、無線網路資源管理：

頻道分配、handoff 管理以及資源規劃與排程。

二、行動電腦移動性管理：

位置追蹤與定位、漫遊管理、security mobile IP。

三、主幹網路設計：

智慧型網路、Wireless LAN, Wireless ATM。

四、分散式演算法：

同步問題、資料庫與檔案管理以及群體通訊協定。

無線廣播所面臨的問題分成調整時間與接收時間，調整時間就是 Clients 端在收聽頻道訊號所花的時間。接收時間是自 Clients 端提出一個廣播節目直到資料被收到[5]。本文選取無線網路資源管理領域中『頻道分配』為方向，將無線廣播頻道分配領域，探討其中因廣播頻寬不足引起通訊壅塞、容易斷訊、造成通訊品質不佳，為有效查詢資料配置，本研究擬以資料探勘理論中的關聯法則(Association Rule)[7]最主要的演算法-- Apriori 演算法作無線廣播頻道有效查詢過程中的資料配置。

## 第二節 研究目標及論文架構

本研究，預期達成以下目標：

一、探討無線廣播頻道資料配置。

二、探討廣播資料庫的廣播封包有序化。

三、在眾多 Client 端分別於不同時間、地點、擷取資料庫資料，各 Client 端所需求的資料順序也不一樣，計算出 Client 端最佳解。

四、運用 Apriori 演算法計算出廣播資料庫最佳解。

五、本論文架構第一章說明研究背景、研究動機、研究目標。第二章是敘述資料庫的發展趨勢，傳統資料庫、行動資料庫及廣播資料庫。第三章問題描述。第四章資料探勘觀念，關聯法則等

相關文獻之探討及提出 Apriori 演算法導入無線廣播模式。第五章實驗設計與結果。第六章結論。

## 第二章 資料庫發展

### 第一節 傳統資料庫的發展階段

電腦科技的進步，使得資料庫的應用成為一門成熟的技術，也是一門新興的技術。資料庫的應用面，也在發展精進中，透過不斷增添新資料庫內容，出現各具特色的資料庫系統，這些特色也成為資料庫發展的趨勢。由網路傳輸連接，使得 Client 端和 Server 端相連，近年來，由於個人通訊設備發展快速，行動計算技術成熟，透過無線傳輸，另一新型的資料庫—行動資料庫，正快速成為人們的新需求，其相關技術，深具研究、探討之價值。現今通訊環境，有線傳輸頻寬快速，而無線網路頻寬不是很普遍的情形，影響到無線通訊品質。在無線網路環境內，頻寬有限的情況下，如何避免通訊中斷、網路壅塞等情況日形重要，因此廣播資料庫資料廣播模式技術，逐漸成為新的研究主題。

資料庫是一群經過一連串有系統整合，有規劃的組織，其相關資料的檔案，就如同一個電子抽屜櫃，資料以不重覆的方法，儲存許多有用的資訊，透過計算、查詢、檢索、排序、組織等方法，有效率的管理，轉成有用的資訊。資料庫的建立，不是僅將一堆資料，輸入電腦中就能完成建檔，而是要將資料有組織、有規劃的存放到資料庫中。重要的是能建立一套完整的管理規則。資料庫管理系統，除提供資料處理外，更重要是提供權限設定、密碼管理、維護整個資料庫達到正確與安全的使用。因此，好的資料庫管理系統（DBMS）應有以下條件：

- 一、保持資料完整性。

- 二、保持資料安全性。
- 三、使用者容易操作。
- 四、資料可匯入、匯出及相容性。
- 五、完整的復原能力。
- 六、可擴充及轉換。
- 七、程式設計資料結構完整。

隨著電腦科技的迅速發展，人們應用電腦進行資料處理，逐漸廣泛，滿足發展的資料處理需求，及現代化管理的需要，電腦對資料管理經歷三個階段。

#### 壹、五十年代中期以前是第一階段—無管理階段

本階段背景為：電腦主要應用在計算，其他工作還沒有開展。硬體背景是，儲存媒體只有磁帶、卡片、紙帶等。軟體背景是，沒有作業系統 (Operating System)、沒有管理資料軟體。此時期資料管理的特點是：

- 一、資料不保存。
- 二、沒有檔案概念。
- 三、一組資料對應一個程式，資料與程式沒有獨立性，如資料儲存方式稍有改變，即必須修改程式。
- 四、沒有軟體系統，資料邏輯結構和實體結構都是程式設計師直接管理，且要設計儲存方法及輸出、輸入方式，必須熟悉各種假設的特性、結構、容量、啟動步驟。

#### 貳、五十年代後期至六十年代中為第二階段—檔案管理階段

本階段背景：電腦不僅應用於計算方面，且大量被應用在管理領域。硬體方面，儲存媒體有磁碟、磁帶等直接存取設備。軟體方面，已有專門管理資料的軟體，稱為檔案系統。檔案系統包含在作業系統之中，還

具有線上即時處理之功能。這時期檔案管理資料的特點是：

- 一、大量使用在資料處理等方面，資料需長期保留在儲存媒體作反覆處理，即對檔案文件進行查詢、修改、插入、刪除等操作。
- 二、有專門的軟體檔案系統進行資料管理，程式與資料之間有存取方法、資料查詢、修改等常駐程式放在 RAM 之中。程式設計師不須考慮儲存形式。
- 三、資料在儲存結構改變，不一定反映在主程式，資料與程式均有一定的獨立性。
- 四、管理資料軟體中增加安全、保密檢查機構，採取必要措施，確保資料安全、可靠。
- 五、檔案已經多樣化，有直接存取設備，檔案的種類除循序檔案外，還有索引檔案、鏈式檔案、直接存取檔案、倒排檔案等等。

### 參、六十年代後期為第三階段—資料庫階段

本階段背景：資料庫是大量資訊的儲存體，特別是指可以利用電腦處理的形式。這時期檔案管理資料的特點是：

- 一、電腦的規模更加龐大，除資訊管理之外，人工智慧系統、專家系統的出現使資料的數量大增，資料結構更加複雜。因此，資料的共享性也更強，更加符合程式設計者的需求。
- 二、容量大的磁碟、光碟使硬體存取空間更好。
- 三、軟體技術有很大進步，朝商品化開發成長。
- 四、即時線上處理進一步的應用發展，開始出現分散式處理。



## 第二節 分散式資料庫

一般提到資料庫系統，都是集中式資料庫，整個資料庫是存放在一台電腦或伺服器上。資料採取集中管理方式，要求主機或伺服器有較大的儲存容量，容易實作和管理。隨著資料庫應用，不斷發展，資料庫應用規模不斷擴大，集中式資料庫有諸多缺陷和不便。例如銀行儲蓄系統，大部分地區都已經實現；甚至全國的通存、通匯業務，這系統完全採用集中式資料庫，顯然是不可取的，主機或伺服器會不堪負荷，網路或通訊線路也會不堪負荷，最後可能會由於連線用戶過多、網路頻寬不足甚至造成整個系統癱瘓。

### 壹、分散式資料庫系統

資料實際是分佈，儲存在不同的電腦上，這些分佈各地的資料在邏輯上又構成一個整體資料庫。分散式資料庫：分散式資料庫是一個分佈於電腦網路上不同地點、而邏輯上又屬於同一系統的資料集合。網路上每個地點的資料庫都有自我管理的能力，能夠完成獨立作業；同時每個地點的資料庫又屬於整個系統，透過網路也可以完成整體作業。

假設分散式資料庫是一個銀行的通存通匯儲蓄系統，每個大圓圈表示一個分行或儲蓄所的電腦系統，它自成一個可以個別運作系統，透過通訊網路它們又構成一個整體的完整的系統。如果是在自己的開戶分行或儲蓄所進行存、取款，則只需於該開戶分行系統就可以完成；如果是在異地進行存、取款，則必須要透過整體系統才可以完成。

分散式資料庫的資料儘管是實際上分佈在不同的地點，邏輯上構成一個整體資料庫。使用者事實上使用的是一個整體資料庫，他們不需要知道哪些資料存放在什麼地方，在分散式資料庫中把這種特性稱為位置透明性（location transparency）。應用程式只需要提出應用要求，至於資料

庫管理系統在哪能取到所需要的資料，則完全是由分散式資料庫系統決定和完成的。

分散式資料庫系統可以由一群“對等”的區域資料庫構成，也可以再有一個中央控制資料庫，負責這些分佈資料庫系統中的各區域資料庫的資源調度和協調。

#### 一、分散式資料庫具有以下特點：

- (一)、資料實際的分佈。資料庫中的資料不是集中存放在一個地點的一台電腦上，而是分佈在不同地點的電腦上。
- (二)、資料的邏輯整體性。資料庫中的資料雖然實際上是分佈在不同地點的電腦上，但這些資料不是不相關的，它們在邏輯上屬於同一個整體資料庫。
- (三)、資料的分佈透明性。資料的分佈透明性也稱作分佈獨立性，也就是說，在分散式資料庫中，除了有資料的邏輯獨立性和實體獨立性之外，還有資料的分佈獨立性。資料的分佈獨立性（分佈透明性、位置透明性）是指用戶只需關心整體資料庫中有哪些資料，而不必關心資料存放在什麼地方及存放細節，分佈的儲存實現是由系統自動完成的。

#### 貳、分散式資料庫的資料分佈方式

分散式資料庫的資料是實體上分佈在各不同的地點。在此資料庫中資料的分佈方式是分散資料管理的一種方式或策略，目前存在著四種分佈方式：

##### (一)、集中式：

將所有資料安排在一台伺服器上，是集中資料庫的管理方式。由於資料集中安排在同一台伺服器上，因此管理和控

制比較容易；但是由於所有的操作都將發生在一台電腦上，所以效率可能會較低，並且當儲存資料庫的電腦發生故障時，將會導致整個系統崩潰，因此其可靠性相對的也較差。

#### (二)、分割式：

將全部資料分割成若干部分，分別存放在若干地點的電腦上，或者說是將一整個資料庫分成若干子集，每個子集放在一台資料庫的伺服器上。以分割式的策略，每台資料庫伺服器都可以成為一個自主的區域系統，當需要整體查詢時，可能需要兩台、甚至多台資料庫伺服器，所以所需的存取時間要比集中式資料庫長；但是分割式策略可以充分發揮操作的潛力，並且系統的可靠性也有所提高（當某一個區域資料庫發生故障後整個系統仍能運行）。

#### (三)、複製式：

一個資料庫複製多個副本，在每個資料庫伺服器上都有一個完整的資料庫副本。分佈策略的可靠性最高，回應時間也較快，所有的操作都可以在本地資料庫完成。複製式分佈策略是目前最常用的分佈策略，SQL Server 支援複製式分散資料管理。

#### (四)、混合式：

即分割式與複製式的混合。混合式雖然可以兼顧分割式和複製式兩種方法，並且獲得了兩者的優點，從表面上看靈活性也較大，但這種方法管理起來比較複雜。

### 參、分散資料管理概述

主從式（Client/Server）架構本身就可以認為是分散式的架

構，在這類架構中，客戶與伺服器既可以置於同一個地點，也可以在不同的地點，透過區域網或遠端網路連接在一起。在實際應用中經常需要將多個 Client/Server 網路集成在一個整體網路系統下，形成一個完整的應用。這在諸多資料庫伺服器上就出現分散資料管理的需求，每個資料庫伺服器可視為一個獨立的系統，可以儲存和管理自己的資料；同時，每個資料庫伺服器又都有可能存取其他資料庫伺服器上的資料。

#### 肆、分散資料庫系統的架構

分散式資料庫系統（DDBS：Distributed Database System）

分散式資料庫系統是由數個資料庫系統所構成。每一個資料庫系統並不一定要由同一個 DBMS 管理。即使採用相同的 DBMS，同一個物件仍然有可能會使用不同的值。此外，也可以使用不同的管理體系。因此，分散式資料庫系統必須達成以下的獨立性：

##### （一）、異類性（heterogeneity）：

異類性的獨立性是指不讓使用者看到資料庫系統的相異處。

##### （二）、分散性（distribution）：

分散性的獨立性是指不讓使用者看到系統是由數個資料庫系統所構成的。

##### （三）、自律性（autonomy）：

自律性則是指資料庫系統可以自行管理或決定資料庫的設計與運算的執行順序。不讓使用者看到每一個資料庫系統的自律性或叫做自立性的獨立性。

### 第三節 可以隨身攜帶的資料庫

#### 壹、行動資料庫

行動資料庫系統的主要特性在於不受實體網路限制到處都可以接收與傳遞訊息。讓下載到行動設備的資料能被使用者管理及重複使用，才有行動資料庫的需求，其可讓資料的管理及資料的分享更加容易，適用於手機的行動資料庫。為了讓行動資料庫能夠在各種不同的行動設備上進行資料的交換及重複使用，需另外運用其他程式語言加以設計以符合行動資料庫的需求。再轉換可呈現在不同的平台而達到跨平台的特性。此外需採取處理器與處理規則分開的原則來設計行動資料庫，使用者可藉由處理規則對儲存在行動資料庫內的資料進行簡單的查詢、修改、新增及刪除的動作，而且行動資料庫具有不需要變更處理器就可以處理不同資料庫的能力。由於使用者的行動性多樣，一種新型的工具，行動計算機-「PDA」提供很好的行動方式，其質輕巧，容易攜帶的特性，空間顯示完善功能的使用者介面，遂發展一小到可用於行動式設備的關連式資料庫—(DB2 Everyplace) [2]，讓使用工作者不連線也能接上資料庫。

#### 貳、行動資料庫架構

無線區域網路提供在固定或緩慢移動的終端機間,小的涵蓋區域(幾十公尺直徑的大小)、高流通量(Mb/s)的通信。在1990年代發展的無線區域網路使用專有的傳輸協定,所生產的產品將按照二個標準:IEEE 802.11和HIPERLAN。此兩者均預期操作在2.5GHz,5GHz附近和較高頻率,無線區域網路由衛星和其他系統的分野是允許其終端機直接相互通訊,而不需經過一個包含基地台和交換設備的基礎架構。設計無線區域網路的主要問題,於無線區域是源至於系統架構的分佈特性。不

需要基地台的協調，終端機為取得同一無線電頻道的接取而競爭，協定是設計於促進接取的公平（所有終端機間相等的分享通道）和操作的可靠度，甚至當某些終端機超出其他終端機可查覺的範圍（隱藏終端機問題）。無線區域迴路（Wireless local loops）的技術挑戰比服務行動終端機的通信系統較不嚴格，廣域無線數據系統從 1980 年早期已經用以服務特殊的查證、廣播服務如道路交通諮詢、交互的服務如無線電子郵件。此系統提供雙向、低速、封包交換數據通信，在美國和其他很多歐洲國家用公眾數據網路的 Mobitex。Ardis 和 Mobitex 操作在特殊化的行動無線電頻段於 800MHz 至 1,000MHz，通信位元速率由 8kb/s 至 19.2kb/s。行動資訊庫上的資料：許多行業使用行動資料系統。這種系統有許多不同名稱：普及運算(pervasive computing)、無線運算、行動式設備或是掌上型應用。

#### 參、行動資料系統的特色

就是可攜帶性，使用者將產品帶到任何工作地點，將資料輸入小型、偶爾連線的行動式設備裡；其連線的設備不需要成為網路的一環，插入牆上插座或附接到其他物件就可以運作。該設備可同步更新中央資料庫的資料。推動此新運算最常用應用程式，就是 PDA 軟體，它可以儲存約會行事曆、連絡資料與筆記、檢索、組織與管理行動設備資料等，此系統還能同步更新傳統伺服器上關連式資料庫的資料。（IBM 首先在 1999 年 8 月提供網站下載）。IBM 已在 2000 年 7 月發行 Palm OS 平台的 DB2 Everyplace，2000 年 9 月已可支援其他執行 EPOC、Neutrino、Windows CE 與嵌入式 Linux 的行動式設備。

行動型資料庫系統的一大特質，就是體型輕巧。佔用空間是用 fingerprint(手指般)來形容的，因為它所佔體積小到無法用 footprint 形容。

DB2 Everyplace 屬關連式資料庫系統，佔用的空間極小，為 100 至 150K，專為小型的行動式設備而設計。一般的觀念，是將少量的重要資料儲存在行動式設備上，稍後再與其他較完整的長期性資料儲存庫同步更新。DB2 Everyplace 在行動式設備上提供本機資料儲存庫。另一個功能則是將行動式設備上的關連式資料，與其他 DB2 資料來源進行雙向式同步更新，例如在 Unix 系統、Windows NT 或 OS/390 平台上執行的 DB2 Universal Database(UDB)。

#### 一、DB2 Everyplace 的三大基本元件

三大基本的 DB2 Everyplace 元件：行動資料庫引擎(Handheld Database Engine)、同步更新伺服器 (Synchronization Server)、個人應用程式建構器(Personal Application Builder, PAB)。

##### (一)、行動資料庫引擎(Handheld Database Engine)：

第一元件為「行動式資料庫引擎」，是真正的關連式資料庫引擎，能持續儲存紀錄組，也可以修改與檢索各筆紀錄。它還提供完整性機制，即使在使用行動式設備時突然遭遇斷電或不慎掉落，資料也不會遺失或損毀。其容量經過調整，記憶體只需 100 至 150K。且可讓數個應用程式共享，因此每個新的應用程式都不需要另外使用新的 DB2 Everyplace 資料庫引擎。可運用 SQL 修改並存取資料。SQL 是應用程式存取關連式資料的 API。行動式設備並不需要完整的 SQL 執行功能，因為大多數的資料存取作業只是單純的資料輸入與檢索要求而已。其支援所有基本的 SQL DML 敘述(INSERT、UPDATE、DELETE、SELECT)，及許多其他一般的作

業功能，包括 DBCS 支援，合併(joins)及游標(cursors)等。甚至提供捲動式游標。不過，標準的關連式功能 UNION 並未得到其支援。另頗具魅力的新特色是索引編製支援。一些影響 DB2 效能的最重要因素，就是適當的索引設計、建立及管理。IBM 已讓 DB2 Everyplace 管理容量高達 120MB 的資料庫。若干限制：譬如，無法支援子查詢，也不能製作檢視畫面。同時，一些物件關連式功能尚未提供，如觸動器(trigger)、預儲程序(store procedure)、大型物件(LOBS)及使用者定義函數。DB2 Everyplace 設計的目標，就是要存取小型資料庫的資料，因此目前並沒有急切的理由需要這些高階的功能。此外，不需要鎖定功能。

## (二)、Synchronization Server(同步更新伺服器)

第二元件是 Synchronization Server(同步更新伺服器)，或稱 Sync Server，是 IBM DB2 Everyplace V.7.1 的新功能。Sync Server 取代了 IBM Mobile Connect-它原來是用以同步更新行動式設備資料與中央伺服器的必要附加產品。Sync Server 為主從架構的程式，管理行動式資料庫及 DB2 原始資料庫之間的資料同步化過程。DB2 原始資料庫可以是任何一種 DB2 UDB 伺服器平台。Sync Server 必須在行動式設備上安裝用戶端，並在被更新資料所在的平台上安裝伺服器元件。不論被更新資料所在的主機伺服器為何，Sync Server 引擎都需要一個支援 Windows NT 伺服器的中間層(midtier)DB2 UDB。中間



伺服器的 Unix 支援，已在 2000 年上市。Sync Server 支援雙向的資料同步更新作業，可從行動式資料庫更新到 DB2 UDB 資料庫，或從 DB2 UDB 資料庫更新到行動式資料庫。為使資料同步，Sync Server 會啟動一個同步化協訊(session)，該協訊為雙向的流程，其間：行動使用者將手機上原始資料的變動送出。行動使用者收到企業伺服器上已變動的原始資料，而且是上次資料同步化之後的最新變動。

### (三)、個人應用程式建構器(Personal Application Builder)

第三元件個人應用程式建構器(Personal Application Builder)，PAB 為整合性工具組，可用以開發在行動式設備上執行的 DB2 Everyplace 應用程式；它可以為存取 DB2 的小型行動式設備建置應用程式。利用 PAB，開發人員在更強大的開發平台上(如 Windows PC)，撰寫應用程式，在行動式設備裡部署。

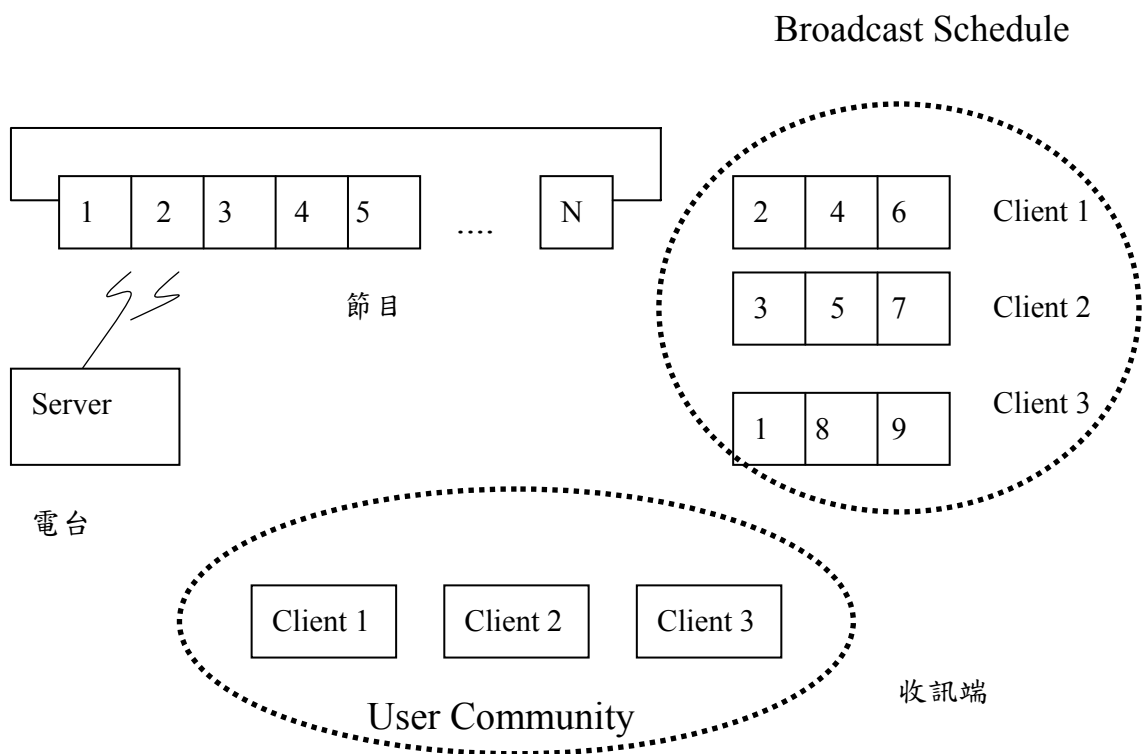
### 肆、資料庫管理的挑戰

到目前為止，在行動型設備上，雖然不像企業資料庫那樣，需做大幅的微調與管理工作，但它仍是一個關連式資料庫。資料庫的開發，應採用正確的邏輯與實質的設計技巧，包括資料模型建構與正規化(normalization)。最大的成效是，你可以對數百台甚至數千台行動式設備，規劃並管理資料的同步更新作業。如果未能同步更新資料，可能產生的問題包括：中央資料庫的資料過期，行動式設備的資料過期，行動式設備中的大型檔案最後必須更新時，會減緩資料同步更新的速度，行動式設備的應用程式將降低效能。

#### 第四節 廣播資料庫

廣播 (broadcast) 是指在網路通訊上，一個工作站對網路上 (不論是區域網路或廣域網路) 其他工作站發出訊息，如無線傳輸、衛星通訊傳輸，都是屬於這類型技術。通訊科技發達，硬體設備價位合理，使用者在現在環境快速移動。對於資料的廣播，傳播都和建立無線通訊的聯繫方式有關。一為存取時間 (access time) [3,4]，二為無線通訊聯繫時間 (Tuning time)，存取時間是 user 端送出一查詢指令詢問廣播通道封包到收到封包的這段時間流逝時段。無線通訊聯繫時間是被客戶端收聽廣播通道所花的時間數。現在已經有方法於減少建立無線聯繫時間，如索引方式、雜湊方式等。

廣播架構裡，由發射基地台 Server 端發送一連串廣播封包，自 Client 端發送頻道中所有的節目到空中，user 收訊端得到想要的節目封包。在這個架構裡得到如圖 1。



Client 1 :  $\{d_2, d_4, d_6\}$  ;

Client 2 :  $\{d_3, d_5, d_7\}$  ;

Client 3 :  $\{d_1, d_8, d_9\}$  ;

圖 1：無線廣播的基本架構圖

資料來源：周宏磬，民 91.6

### 第三章 問題描述

本篇文章在廣播資料庫裡要解決的是無線廣播頻道分配領域，因廣播頻寬不足引起通訊壅塞、容易斷訊、造成通訊品質不佳，為有效查詢資料配置的一個問題。先將相關資料庫 Server 端依封包長度切割為小區塊，把每一個相關的廣播封包循序放置在廣播頻道上播放，再將廣播頻道上不同長度封包順序送出，Client 端對所送出的封包依需求存取該封包，最後找出存取最符合的查詢順序，以前的查詢方式在選擇上先要減少暫時不必要的封包，因此為減少調頻時間挑選封包時，優先選擇處理廣播頻道上其他封包的價值。目標就是要找出查詢過程的物件資料理想的廣播順序，問題順序可以用 NP-complete 的方式證明[19]。在 Server 端或 Client 端中封包長度沒有控制先後順序。其對應的關係如下：

relation  $A = (r_1, r_2)$ , relation  $B = (x_1, x_2)$ , relation  $C = (y_1, y_2)$

廣播順序稱為廣播程式， $r_1 x_1 x_2 r_2 y_1 y_2$  或  $r_1 x_1 r_2 y_2 y_1 x_2$  都有可能是廣播順序。假設某一  $C_i$  下列查詢語句：

```
Select  $r_2, x_2$ 
```

```
From A ,B
```

```
Where  $r_1 = 10$  and  $r_1 = x_1$ 
```

在這個查詢中， $SA = \{ r_1 \}$ ,  $JA = \{ x_1 \}$  (因為  $r_1 \in SA$ ，這是從  $JA$  中區分出來的)， $PA = \{ r_2, x_2 \}$ 。Client 首先轉到廣播頻道去下載  $r_1$  篩選  $r_1$  的長度，然後 Client 下載  $x_1$  和加入  $r_1$  的呈現。之後，我們得到一對封包長度的數目，這個關係著  $A$  和  $B$  的封包長度就混在一起且滿足在 where-clause 中的查詢條件。從這些封包長度數目中，有關  $r_2$  和  $x_2$  就下載回應查詢。

$$D = \{r_1, r_2, r_3, \dots, r_n\}$$

$$K = \{C_1, C_2, C_3, \dots, C_k\}$$

$$C_i = \{C_{ri1}, C_{ri2}, C_{ri3}, \dots, C_{rim}\}, C_i \subset D$$

本論文所定義的長度：其指為一個廣播順序  $S_1$  所有的和，某一個 Client 端  $C_i$  長度為  $L(S_1, C_i)$ ， $\sum_{i=1}^k L(S_1, C_i)$ 。我們將無線廣播查詢頻道的資料配置問題定義如下：對於有 5、10、15、20、25 個 Client 端的資料庫查詢型態，到 Server 端廣播順序播放與隨機擷取的 Random 值作比較查詢樣式，選取廣播資料最佳順序。

## 第四章 資料探勘

### 第一節 資料探勘

本論文探討無線廣播頻道查詢運用資料探勘理論中的關聯法則之 Apriori 的演算法，為避免無線網路頻寬壅塞、通訊品質不良，提供較好的廣播途徑。

#### 壹、資料探勘 (Data Mining)

資訊科技快速的運算能力，為從大量資料中擷取出商品關聯、可供應用、新的資訊以提供組織決策上的支援。資料探勘 (Data Mining) 是近年來盛行的資料探勘技術，所盛行的新方法，將一般資料庫中看似無用的資料轉換成有用資訊，並找出隱藏之關聯性。資料挖掘亦有人稱之為知識發現 (Knowledge Discovery in Database, KDD)，由於目前關聯式資料庫管理系統 (Relational Database Management System, RDBMS) 所提供的是一個從組織功能程序的觀點出發，重要資訊可能儲存在多個功能性的資料庫中而不自覺，而且操作性不佳，過於技術化，無法直接提供真正高階的決策人員使用。另使用者自行開發系統的風氣盛行，且分散式資料庫的架構正符合大多數需求，使得資訊系統分散獨立，造成資訊資源無法共享。加上整合的需求迫切，所以為了結合主從式運算的開放環境，不受限於集中式的限制，且提供資訊資源運用更高階的整合觀點，將以所需的資訊主題為主，完全以作業資料 (Operational Data) 為核心的決策支援工具，設計者以多維式模組或加上統計學、人工智慧等資訊科技的理念，針對歷史資料來模擬真實情況，以建立出高階決策主

管人員所需的資料模型，對未知預測可能情況。這一連串的過程就稱之為知識發現。

根據Fayyad et al 等人的研究，針對知識發現提出了一個參考的流程。其進行步驟說明如下：

- (一)、資料淨化(Data Cleaning)
- (二)、資料整合(Data integration)
- (三)、資料選擇(Data Selection)
- (四)、資料轉換(Data transformation)
- (五)、資料探勘(Data mining)
- (六)、型樣評估(Pattern evaluation)
- (七)、知識呈現(Knowledge representation)

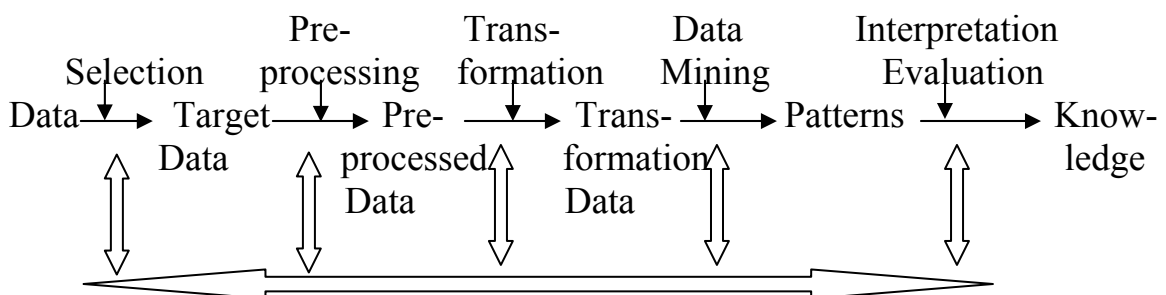


圖 2：知識發現之流程

由以上步驟可看出，資料挖掘涉及大量準備工作與規劃過程，許多專家皆認為整個資料挖掘的進行有80% 的時間精力是花費在資料前置作業階段，其中包含資料的淨化與格式轉換甚或表格的連結。可知資料挖掘只是資訊挖掘過程中的一個步驟而已，在進行此步驟前還有許多的工作要先完成。

## 第二節 關聯法則的相關技術

資料挖掘步驟中，所使用的相關技術，包含關聯式法則[5] (Association Rule)、時間序列 (Time Sequence)、分類式法則 (Classification Rule)、組群式法則 (Clustering Rule)、序列型樣 (Sequential Pattern) 等，每一種技術都有許多不同的方法來實作它。以下針對不同的技術做說明：

### 壹、資料挖掘的相關技術

#### 一、關聯式規則 (association rule)

關聯規則的探討，在資料挖掘中，最早被提出的是找出資料關聯規則，關聯分析技術為找出各項目 (Item) 之間的關聯性，從資料庫中尋找出現頻率較高的樣型 (Pattern)，然後再利用這些樣型分析出關聯規則，因果分析技術則是在關聯分析中加上時間的因素，尋找事件發生的先後關係，而異常偵測技術是要找出資料庫中不尋常的記錄或不正常的改變。

例如資料挖掘是要從企業銷售交易資料庫中，來發掘不同資料彼此間的關係，同時也可以一事件為中心，找出和其有關的資料模式，尤其是當有大量、詳細的交易資料時，更能將此功能發揮的淋漓盡致舉例來說，在分析過每天自POS(Point of Sales)交易系統傳來的資料後，零售業者可能發現，若消費者同時購買低脂牛奶和脫脂優格時，有80%的機會會同時購買土司麵包；又購買油漆的消費者中，有一半的人會同時購買刷子；當購買洋芋片時，有60%的人也會同時購買可樂，尤其是在促銷期間，更有高達80%的機率。在挖掘關聯規則中我們可以把它分成兩個次問題，一是從資料庫中找出所有滿足最小支持度 (Support) 的高頻項目集合



(large itemset)；二是從這些高頻項目集合中找出符合最小信賴度 (Confidence)的關聯規則。關聯規則主要是要找出資料項間的相關性，定義如下：

假設 $D$ 為所有交易記錄 $T$ 的集合， $T$ 是在 $I$ 中任意物品項目的子集合，一個集合 $X \subseteq I$ 稱為項目組 (itemset)，此項目組所包含的項目個數稱為該項目組的長度，若長度為 $k$ ，則稱此項目組為 $k$ 項目組 ( $k$ -itemset)。一筆交易若可以支持一個項目組 $X$ ，則此交易記錄必包含於此項目組的所有項目， $X \subseteq T$ ，記為 $\text{support}(X)$ 。

關聯法則的表示式可以定義為 $X \Rightarrow Y$ ， $X, Y \subseteq I$ 且 $X \cap Y = \emptyset$ 。關聯法則靠支持度 (support) 與信賴度 (confidence) 兩個參數來判斷此關聯法則是否有意義。支持度為資料庫中包含 $X \cup Y$ 的交易記錄所佔的百分比，記為 $\text{support}(X \cup Y)$ ；信賴度 (confidence) 則為定義此關聯法則可信的程度，也就是 $X$ 出現， $Y$ 就出現的機率，記為 $\text{support}(X \cup Y) / \text{support}(X)$ 。一個有效的關聯法則，其支持度與信賴度必須大於或等於使用者所定最小的限制，只有滿足此條件的關聯法則才是有意義的。

接著，我們必須確定 $X \cup Y$ 的支持度是否超過最小支持度 (minimum support)，若是，則為高頻項目集合 (large itemset)，再把 $X \cup Y$ 的支持度除以 $X$ 的支持度，所得的值為 $X \cup Y$ 的信賴度 (confidence)  $c$ ，即在交易資料庫 $D$ 中，有 $c\%$ 筆數的交易資料包含 $X$ 也包含 $Y$ ，我們的目的是在於找出所有能滿足使用者指定最小支持度和最小可信賴度門檻限制的關聯規則。

## 二、時間序列分析(Time Series Analysis)

時間序列是找出在連續的一段時間中，事件發生的相似性。

我們可以採用其相關的分析方法探討當時間改變時其他屬性值的變化，進一步達到預測的目的。目前資料挖掘在時間序列上的研究，大部份是以關聯法則為基礎來發展時間序列的規則，主要目的是針對時間性資料上的分析以找出有用的時間序列。

### 三、分類式法則(Classification Rule)

分類的目的在於利用訓練資料(Training Data)中的各種特徵或屬性來建構一個分類器(Classifier)，再使用與訓練資料特徵屬性相同，但與訓練資料內容不同的測試資料(Testing Data)來決定此分類器所做的分類判斷是否與測試資料中所給定的分類相同，若達到使用者可以接受的正確率時就可以利用此分類器來預測以後新資料集的分類。一般的分類演算法是採用決策樹，先將原始資料分成二部份：Training Data 和 Test Data。Training Data 是用來建立決策樹，將 Training Data 中的每筆資料依其屬性及其分類屬性建立決策樹。然後利用 Test Data 來驗證該樹的正確性。

### 四、組群化法則(Clustering)

針對要分析的資料，利用合適的組群化演算法，有效的將資料集分成一個個的組群，使得各個組群的特徵(Centroid of Cluster)能有效的突顯出來，而不同的組群其所顯示的特徵亦不相同。在資料挖掘的應用上，這些被找出來的特徵可作為決策之參考或資料分析時之前處理器。

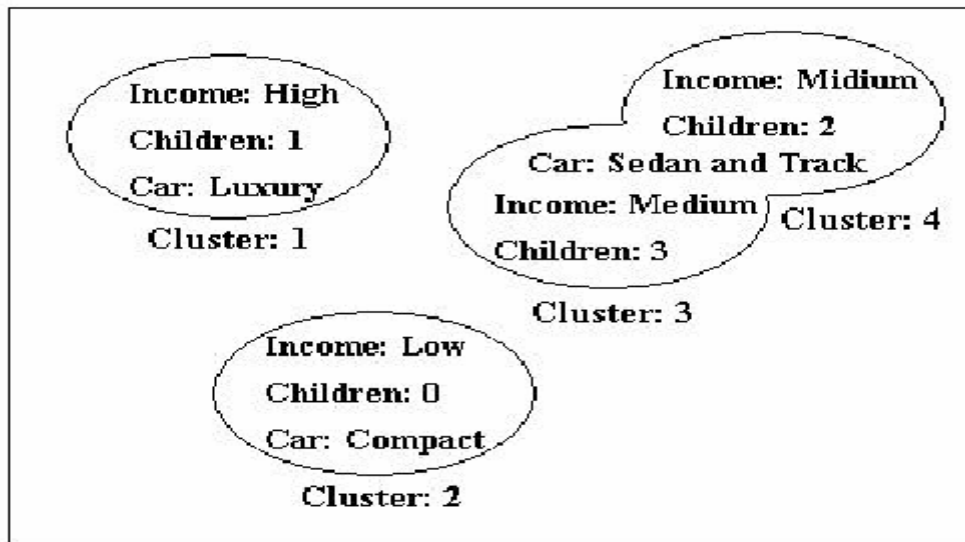


圖 3：顧客被群聚成四個集群(資料來源：Groth,2000)

## 五、序列型樣(Sequential Pattern)

序列型樣是要找出經常發生的關聯順序。其重點在於考慮了時間的因素，所處理的是建構在時間上之資料，利用此方法去分析在不同時間點上各事件之關聯性。序列型樣主要可分為順序性型樣以及週期性型樣，分別針對時間的順序以及時間的區段做分析。順序性型樣考慮時間的先後的問題，只需對時間的順序做資料關聯的分析，而週期性型樣分析時間區段內所發生的事件，是否在其他相同大小的時間區段內也會重覆發生，其著重於不同時間區段的變化。目的在於找出某些事件發生的先後關係，如甲事件發生後就會有乙事件也一起發生，或買了A商品後會接著買B商品，買了B商品後會接著買C商品。

### 貳、Apriori關聯法則

傳統的Apriori關聯規則挖掘演算法有兩個階段[8]，第一階段是找出所有超過最小support的項目集合(即高頻項目集合)，第二階段再從高頻項目集合找出關聯規則。其中第一個步驟決定了整個作業的效能，它佔了

作業的大部分時間，所以在探討關聯規則的挖掘時，均將焦點放在如何有效率的找出高頻項目集合。本篇論文是以Apriori演算法為基礎，在掃描資料庫時即記錄每個項目集的次數，使得第一次掃描資料庫後即可得到所有的項目集，並得到最初項目集的統計次數，並且可以得到長度為 1 且滿足最小支持度的項目集，再以此項目集為基礎，結合並掃描所有的項目集合，並加以統計長度為 2 的項目集的發生次數，得到滿足最小支持度的長度 2 的項目集合，以此類推，直到無法找到滿足最小支持度的長度  $N + 1$  的項目集合，即得到最大長度為  $N$  的項目集合。如此，可改善Apriori中每次都去資料庫中掃描的次數及時間，且每次掃描之後的項目集及次數都能夠保留給下次掃描來使用，可有效減少掃描的時間。

### 叁、關聯法則的有關應用

關聯法則的形式易懂，許多應用領域都會利用此技術來協助資料分析與決策制定，相關應用領域說明如下：

#### 一、商業上的應用

(一)、在進貨與存貨管理，可由分析貨品的銷售情形及其關聯性做適當調整。

(二)、商業促銷方面，可由分析顧客的消費習性以寄出特別為顧客設計的廣告郵件，達到促銷的目的。

#### 二、醫學上的應用

(一)、醫院管理上，可分析藥物使用情形是否異常，以利藥品管理。

(二)、醫師診斷管理上，可分析以往診療情形及診療技術，以達有較好的醫療效果。

### 三、學術上的應用

(一)、電子商務發展技術應用上，可幫助其開發商場及企業 e 化。

(二)、網路教學上，分析學生學習情形，以改善其教學方法及品質。

### 四、其他應用方面

(一)、防治犯罪上，可分析各刑案的關聯性，提高破案率改善社會治安。

(二)、都市規劃上，協助政府施政規劃，使人民享有優質生活。

### 肆、Apriori 演算法

本論文研究問題擬以一個在資料探勘 (Data mining) 領域受到許多廣泛運用的 Apriori 演算法分析大量的資料，以建立有效的規模及規則。在企業上透過 data mining 來瞭解他們的客戶，改進他們的行銷運作。運用 data mining 由大量繁多的網路行為紀錄中，找尋可能隱藏的行為模式。

Data mining有許多不同的演算法，目前實作方式採用關聯法則。依物件間的關聯性 (支持度和信賴度)。即信賴度是一種量測關聯法則強弱的標準，而支持度則是表示統計上出現的頻率，事實上實作時會訂定信賴度與支持度的門檻，擷取出來之關聯法則的信賴度與支持度均必須大於此門檻。Agrawal et al.首先於1994年提出了Apriori演算法[7]以找出關聯法則。此演算法是由單一項目組開始，逐層(level-wise)去擴展其他的相關項目組，優點是可以減少非相關項目組的產生，節省CPU的時間。演算法中的表示符號說明：

1.  $L_k$ : 由  $k$ -wordset 所組成的集合，此集合的每個元素都要大於最小支持度之門檻，而  $L_l$  則是由所有大於支持度的集合所形成的集合。

2.  $C_k$ : 由  $k$ -wordset 所組合成的集合，此集合為  $L_{k+1}$  的候選者集合

3. Apriori-gen( $L_{k-1}$ ) 為一子方程式，裡面包含了兩個步驟：

(1) 結合步驟: 此步驟會將兩個  $L_{k-1}$  集合做結合的動作，以數學形式表示為  $L_{k-1} * L_{k-1}$ 。

(2) 修剪步驟: 此步驟會將  $C_k$  中的所有的  $k$ -wordset 做檢驗，如果其所產生的子集合於  $L_{k-1}$  中找不到，則將其刪除。

在交易資料庫。每一個 iteration，Apriori 演算法建立一個 large itemsets 的 candidate set，計算出每個 candidate itemset 的出現次數。而後依據一個預先決定的最小的支持度來決定 large itemsets。第一個 iteration，Apriori 演算法 scan 所有的交易來計算出每個 item 的出現次數。C1 為 candidate 1-itemset 的集合。假設最小交易支持度的限制是 2。L1 是由達到最小支持度限制的 candidate 組成的 large 1-itemset 集合，依照這個方法就可產生。接下來要找出 large 2-itemset 集合，有鑑於任何 large itemset 一定都有最小支持度，Apriori 演算法使用  $L1 * L1$  來產生 itemsets C2 的 candidate set。這裡的運算指的是連鎖運算。接下來，scan D 裡的四個交易，同時算出 C2 裡的 candidate itemset 的支持值。第二列中間的 table 就呈現出計算後的結果。因此依據 C2 裡每個 candidate 2-itemset 可決定出 large 2-itemset 的集合 L2。依此類推，C3 從 L2 裡產生。從 L2 開始，有相同 first item 的兩個 2-itemset，如 {BC} 和 {BE}，首先被定義。然後 Apriori 演算法去測試 {CE} 這個 2-itemset，是否構成一個 large 2-itemset。既然 {CE} 自己就是個 large itemset，又所有 {BCE} 的 subsets 都是 large，因此 {BCE} 即為 candidate 3-itemset。L2 中再也沒有其他 candidate 3-itemset。這時候 Apriori 演算法掃描所有的交易並在 L3 找出 large 3-itemset。因為 L3 中沒有 candidate 4-itemset 被構成，Apriori 演算法因此結束找尋 large itemset。

可寫成式子如下：

```

L1={large 1-itemsets} ;
For all (k=2 ; Lk-1≠ϕ ; k++) do begin
Ck=Apriori-gen ( Lk-1 ) ; //New candidates
For all transactions t ∈ D do begin
Ct=subset ( Ck , t ) ; //Candidates contained in t
For all candidates c∈Ct do
c.count++ ;
end
Lk={c∈Ck | c.count≥minsup}
End
Answer=UkLk

```

以下為Apriori演算法於資料探勘的一個例子：

Database	D
TID	Items
100	AC
200	BCE
300	ABCE
400	BE
500	ADCE

C1

Item sets	Sup.
{A}	3
{B}	3
{C}	4
{D}	1
{E}	4

L1

Item sets	Sup.
{A}	3
{B}	3
{C}	4
{E}	4

Scan D →

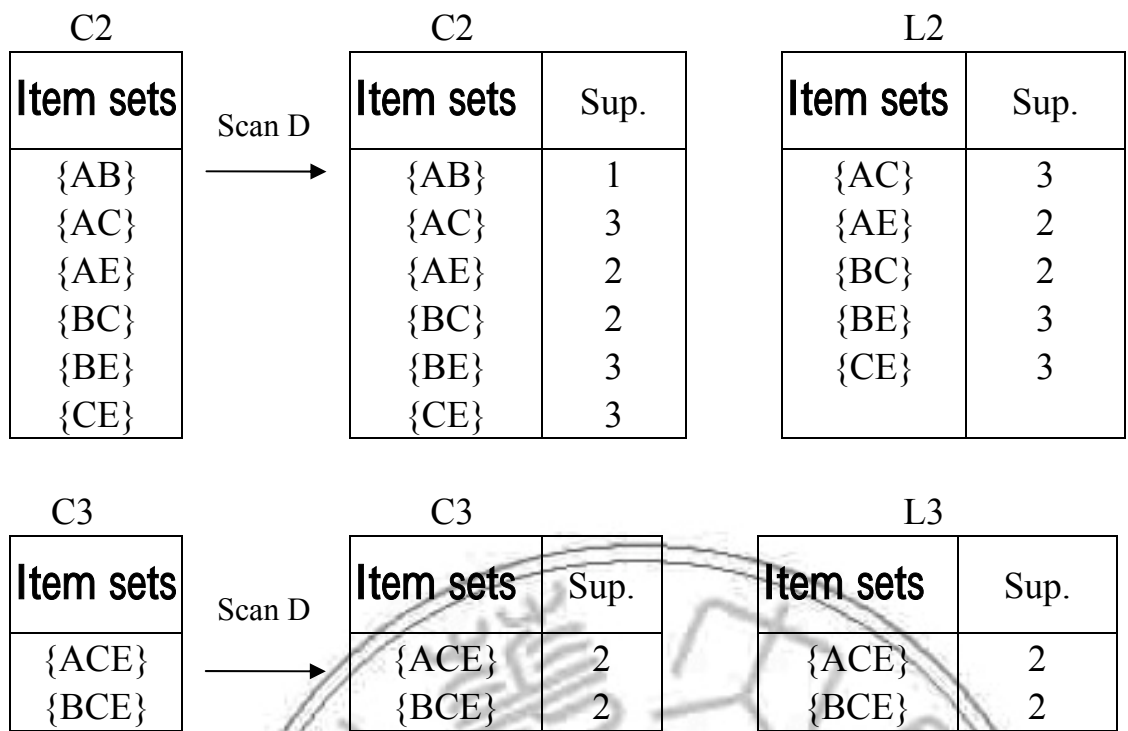


圖 4：Apriori 演算法過程

#### 伍、Apriori 演算法在無線廣播的應用

無線廣播資料順序經由 Apriori 的演算過程，可以得到 large itemsets, 我們將 Apriori 演算法代入無線廣播頻道配置問題中，來找到最佳廣播資料庫的廣播順序。如圖 5，Server 端之外連著許多 Client 端，輸入 m 筆資料分別是  $C_i = \{r_1, r_2, r_3, r_4, \dots, r_m\}$ ，個數是  $K = \{C_1, C_2, C_3, \dots, C_k\}$ ，對應到 data mining Apriori 在 TID 的資料表中，會得到所要的廣播順序。

$$D = \{r_1, r_2, r_3, \dots, r_n\},$$

$$K = \{C_1, C_2, C_3, \dots, C_k\},$$

$$C_i = \{C_{ri1}, C_{ri2}, C_{ri3}, \dots, C_{rim}\}, \quad C_i \subset D$$

依 Server 順序  $r_1 \sim r_{10}$  有 10 個廣播封包，最終目標要把 10 個封包排進廣播順序，將 Apriori 演算法算出最佳廣播組合 large-itemset 要集成一組，之後再看  $L_2$  條件應用。若另有其他的 item 未排入廣播順序時，再依



據各 Client 端所需的廣播順序需求，加以依序排列在適當位置。所得集合組合即為我們在本研究中所求得實驗的最佳解。

## 第五章 實驗設計

### 第一節 實驗設計

本研究實驗設計運用 Apriori 演算法，將各 client 端的廣播長度計算出來，依演算法運算結果中 large-itemsets 產生得到的廣播順序，為各 client 端所需求的廣播資料查詢。

$D = \{r_1, r_2, r_3, \dots, r_n\}$ ，(Server 端長度)

$K = \{C_1, C_2, C_3, \dots, C_K\}$ ，(Client 端個數)

$C_i = \{C_{ri1}, C_{ri2}, C_{ri3}, \dots, C_{rim}\}$ ，(Client 端長度)

設 Server 長度為 10，Client 端個數 k 有 5，10，15，20，25。D 的個數=10~50。個數是 10 的時候， $D = \{r_1, r_2, r_3, \dots, r_{10}\}$ ，用 Random 方法挑出 D 的 Itemsets，

假設將 Server 端資料庫有 10 個 itemsets 為  $\{r_1, r_2, \dots, r_{10}\}$ ，有 k 個 Client 端和其連接。Apriori 演算法建立一個 large itemsets 的 candidate sets，計算出每一個 candidate itemset 的出現次數。依據，預先決定的最小支持度決定 large itemsets。第一個 iteration，Apriori 演算法 scan 所有的交易來計算出每一個 item 的出現次數。C1 為 candidate 1-itemset 的集合假設最小交易支持度的限制是 3。L1 是由達到最小支持度限制的 candidate 組成的 large 1-itemset 集合，依照這個方法就可產生。接下來我們要找出 large 2-itemset 集合，有鑑於任何 large itemset 一定都有最小支持度，Apriori 演算法使用  $L1 * L1$  來產生 itemsets C2 的 candidate set。這裡的運算指的是連鎖運算。接下來，scan D 裡的交易，同時算出 C2 裡的 candidate itemse

的支持值。第二列裡中間的 table 就呈現出計算後的結果。因此依據 C2 裡每個 candidate 2-itemset 可決定出 large 2-itemset 的集合 L2。依此類推，C3 從 L2 產生。

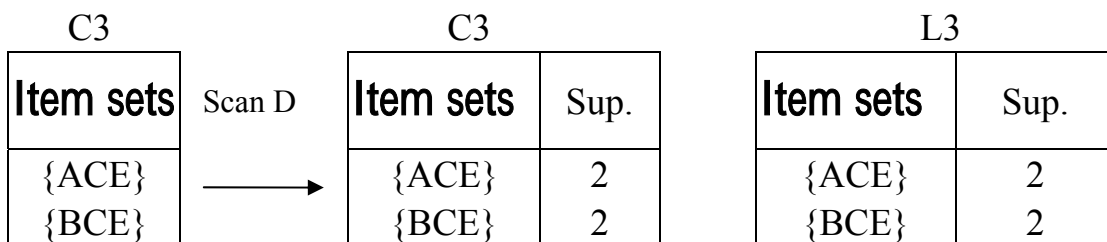
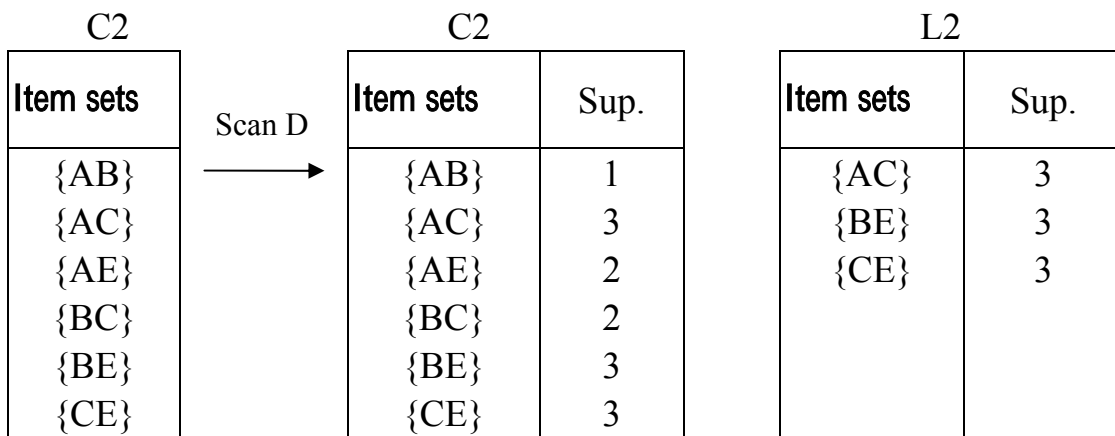
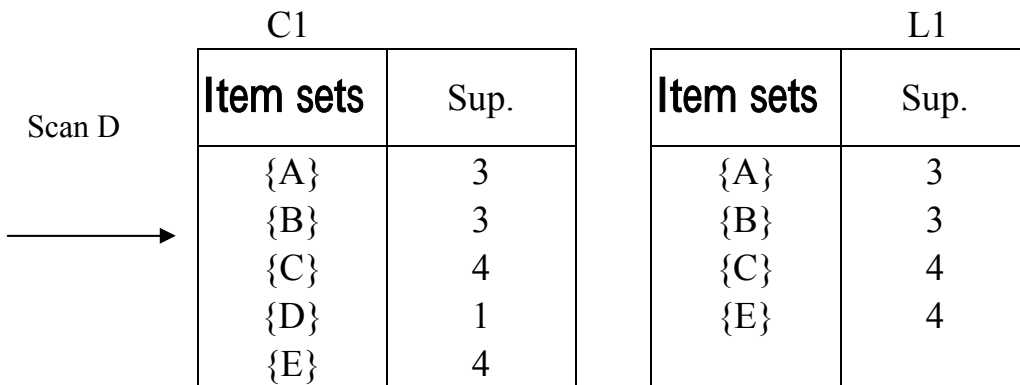
依據前面所提的 Server 廣播封包順序與隨機亂數產生的封包順序作實驗測試效率比較。

在實驗設計中，我們設計 Server 端資料庫中的資料欄位有 10~50 個，當 Clinet 端查詢資料欄位為 5 個時，Server 端資料欄位最多 10 個，Clinet 端查詢資料欄位為 10 個時，Server 端資料欄位最多 20 個，Clinet 端查詢資料欄位為 15 個時，Server 端資料欄位最多 30 個，Clinet 端查詢資料欄位為 20 個時，Server 端資料欄位最多 40 個，Clinet 端查詢資料欄位為 25 個時，Server 端資料欄位最多 50 個，資料欄位 10 個時以  $r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}$  代表，設每個 Clinet 端所查詢不一樣的資料欄位，也有可能查詢相同的資料欄位，分別以資料欄位 10 個及 20、30、40、50 個。Clinet 端作模擬產生我們要的資料欄位順序，經由隨機亂數產生 Random 資料庫廣播時的資料欄位順序，分成長度 10 和我們的方法做比較，來進行實驗。

## 第二節 排序方法

在 Apriori 無線廣播的排序方法中，把 large itemset 的這些資料集中，因為它是最多的 Client 要的資料欄位，所以在考慮 Itemset 長度由長到短的排序方式，還要同時考慮 Itemset 的支持度大小，後來加入的支持度以不破壞前面高頻率順序為主，原來支持度 60% 的順序為主，調整中間的順序以符合下一個 Items 進來的順序，如下列步驟：

Database D	
TID	Items
100	AC
200	BCE
300	ABCE
400	BE
500	ADCE



但{D}尚未排入，把  $L_3$  中 {ACE} 和 {BCE} Join 變成 {ABCE}，在  $L_2$  中 {BE} 是最大的 itemset，所以把 {ABCE} 調整成 {BECA}，最後剩下 D 尚未加入，一開始 D 就被刪除，所以它是少數 Clinet 需要的資料，因為 D 的支持度只有 1，Scan 資料庫發現{ACD}，那 {BECA} 是固定不可變動，所以把 D 加在後面變成 {BECAD} 這就是我們要的答案。

### 第三節 實 驗

(一)、先以 Client 5 為一組，分別是  $C_1, C_2, C_3, C_4, C_5$ ，最小交易支持度是 2，Client 端的 item 值由 Random 產生，透過 Arpioir 演算法得到 large itemsets 為 {r1,r3,r4}，將這 large itemsets 和 Server 端  $D=10$  對應，再和 Random 做比較，看其順序長度是否為 Client 所需求的長度，則該長度即為廣播順序。

Database D

TID	Items
$C_1$	$r_1, r_2, r_3, r_4$
$C_2$	$r_5, r_4, r_1, r_7$
$C_3$	$r_7, r_3, r_9, r_6$
$C_4$	$r_8, r_1, r_3, r_6,$
$C_5$	$r_4, r_5, r_9, r_8, r_{10}$

#### Client 5 端的實驗排序

Server 端以 10 個資料欄位為基礎，以 5 個 Clinet 產生 5 組不一樣的資料欄位，在 Clinet 端所需要的資料欄位也是經由隨機亂數產生，資料欄位最多 10 個，套入我們的方法和隨機亂數作資料庫廣播時的資料欄位比較。結果是我們的方法比隨機亂數加總起來的長度還要短，相差 2。

Client 5，我們的方法和隨機亂數產生的資料順序作比較

Client	我們的方法 ( $r_1, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_2, r_{10}$ )	隨機亂數 $R_1(r_1, r_7, r_3, r_{10}, r_5, r_6, r_4, r_8, r_9, r_2)$
$r_1, r_2, r_3, r_4$	9	10
$r_5, r_4, r_1, r_7$	6	7
$r_7, r_3, r_9, r_6$	7	8
$r_8, r_1, r_3, r_6,$	7	8
$r_4, r_5, r_9, r_8, r_{10}$	8	6
加總	37	39

(二)、我們再以 Client 10 為一組，分別是  $C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9, C_{10}$ ，最小交易支持度是 3，透過 Arpioir 演算法得到 large itemsets 為  $\{r_4, r_5, r_9\}$ 。將這 large itemsets 再和 Server 端  $D=20$  對應，再和 Random 做比較，看其順序長度是否為各 Client 所需求的長度，則該 large itemsets 即為廣播順序。

Database D

TID	Items
$C_1$	$r_1, r_2, r_3, r_4, r_{18}$
$C_2$	$r_5, r_4, r_1, r_7, r_{17}$
$C_3$	$r_7, r_3, r_9, r_6, r_{15}$
$C_4$	$r_8, r_1, r_3, r_6, r_{16}$
$C_5$	$r_4, r_5, r_9, r_8, r_{14}$
$C_6$	$r_1, r_2, r_9, r_6, r_{11}$
$C_7$	$r_7, r_3, r_8, r_{10}, r_{19}$
$C_8$	$r_5, r_9, r_4, r_6, r_{13}$
$C_9$	$r_4, r_5, r_7, r_9, r_{10},$
$C_{10}$	$r_2, r_3, r_4, r_6, r_{12}, r_{20}$

## 10 個 client 端的實驗

Server端以20個資料欄位為基礎，以10個Client產生10組不一樣的資料欄位，在Client端所需要的資料欄位也是經由隨機亂數產生，資料欄位最多20個，套入我們的方法和隨機亂數作資料庫廣播時的資料欄位比較。結果是我們的方法比隨機亂數加總起來的長度還要短，相差20。

## 10 個 client 我們的方法和隨機亂數產生的資料順序作比較

Client	我們的方法	隨機亂數 $R_1$
	( $r_4, r_5, r_9, r_2, r_3, r_6, r_1, r_7, r_8, r_{10}$ ) ( $r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16}, r_{17}, r_{18},$ $r_{19}, r_{20}$ )	( $r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}$ ) ( $r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16}, r_{17}, r_{18},$ $r_{19}, r_{20}$ )
$r_1, r_2, r_3, r_4, r_{18}$	9	18
$r_5, r_4, r_1, r_7, r_{17}$	17	17
$r_7, r_3, r_9, r_6, r_{15}$	13	15
$r_8, r_1, r_3, r_6, r_{16}$	12	16
$r_4, r_5, r_9, r_8, r_{14}$	14	14
$r_1, r_2, r_9, r_6, r_{11}$	9	11
$r_7, r_3, r_8, r_{10}, r_{19}$	15	19
$r_5, r_9, r_4, r_6, r_{13}$	13	13
$r_4, r_5, r_7, r_9, r_{10},$	10	10
$r_2, r_3, r_4, r_6, r_{12}, r_{20}$	20	19
加總	132	152

(三)我們再以 Client 15 為一組，分別是  $C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}, C_{15}$ ，最小交易支持度是 3，透過 Arpioir 演算法得到 large itemsets 為  $\{r_6, r_9, r_{15}\}$ 。將這 large itemsets 再和 Server 端  $D=30$  對應，再和 Random 做比較，看其順序長度是否為各 Client 所需求的長度，則該 large itemsets 即為廣播順序。

Database D

TID	Items
$C_1$	$r_5, r_2, r_3, r_4, r_{18}, r_{26}$
$C_2$	$r_5, r_4, r_{11}, r_7, r_{17}, r_{25}$
$C_3$	$r_7, r_3, r_9, r_6, r_{15}, r_{27}$
$C_4$	$r_8, r_1, r_3, r_6, r_{16}$
$C_5$	$r_4, r_{15}, r_9, r_8, r_{14}, r_{24}$
$C_6$	$r_1, r_2, r_9, r_6, r_{11}, r_{15}, r_{28}$
$C_7$	$r_7, r_3, r_4, r_8, r_{10}, r_{19}$
$C_8$	$r_{15}, r_9, r_{14}, r_6, r_{13}, r_{29}$
$C_9$	$r_4, r_5, r_7, r_9, r_{10}, r_{30}$
$C_{10}$	$r_2, r_3, r_4, r_6, r_{12}, r_{20}$
$C_{11}$	$r_1, r_3, r_{14}, r_6, r_{12}, r_{29}$
$C_{12}$	$r_1, r_{12}, r_{19}, r_6, r_{28}$
$C_{13}$	$r_7, r_{13}, r_{18}, r_{10}, r_{27}$
$C_{14}$	$r_8, r_1, r_{13}, r_6, r_{19}$
$C_{15}$	$r_4, r_6, r_7, r_9, r_{15}$

### 15個client端的實驗

Server端以15個資料欄位為基礎，以15個Client產生15組不一樣的資料欄位，在Client端所需要的資料欄位也是經由隨機亂數產生，資料欄位最多30個，套入我們的方法和隨機亂數作資料庫廣播時的資料欄位比較。結果是我們的方法比隨機亂數加總起來的長度還要短，相差50。



15 個 client 我們的方法和隨機亂數產生的資料順序作比較

Client	我們的方法 ( $r_6, r_9, r_{15}, r_1, r_3, r_4, r_7, r_5, r_{12}, r_{10}, r_2, r_8, r_{13}, r_{14}, r_{19}, r_{11}, r_{18}, r_{16}, r_{17}, r_{20}, r_{30}, r_{29}, r_{28}, r_{27}, r_{26}, r_{25}, r_{24}, r_{23}, r_{22}, r_{21}$ )	隨機亂數 $R_1$ ( $r_1, r_2, r_3, r_4, r_5, r_6, r_2, r_8, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16}, r_{17}, r_{18}, r_{19}, r_{20}, r_{21}, r_{22}, r_{23}, r_{24}, r_{25}, r_{26}, r_{27}, r_{28}, r_{29}, r_{30}$ )
$r_5, r_2, r_3, r_4, r_{18}, r_{26}$	20	25
$r_5, r_4, r_{11}, r_7, r_{17}, r_{25}$	20	22
$r_7, r_3, r_9, r_6, r_{15}, r_{27}$	23	25
$r_8, r_1, r_3, r_6, r_{16}$	18	16
$r_4, r_{15}, r_9, r_8, r_{14}, r_{24}$	25	21
$r_1, r_2, r_9, r_6, r_{11}, r_{15}, r_{28}$	22	28
$r_7, r_3, r_4, r_8, r_{10}, r_{19}$	11	17
$r_{15}, r_9, r_{14}, r_6, r_{13}, r_{29}$	21	24
$r_4, r_5, r_7, r_9, r_{10}, r_{30}$	20	27
$r_2, r_3, r_4, r_6, r_{12}, r_{20}$	21	19
$r_1, r_3, r_{14}, r_6, r_{12}, r_{29}$	21	29
$r_1, r_{12}, r_{19}, r_6, r_{28}$	22	28
$r_7, r_{13}, r_{18}, r_{10}, r_{27}$	17	21
$r_8, r_1, r_{13}, r_6, r_{19}$	15	19
$r_4, r_6, r_7, r_9, r_{15}$	7	12
加總	283	333

(四)我們再以 Client 20 為一組，分別是  $C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}, C_{15}, C_{16}, C_{17}, C_{18}, C_{19}, C_{20}$ ，最小交易支持度是 3，透過 Arpioir 演算法得到 large itemsets 為  $\{r1, r6, r28\}$ ， $\{r1, r6, r35\}$ 。將這 large itemsets 再和 Server 端  $D=40$  對應，再和 Random 做比較，看其順序長度是否為各 Client 所需求的長度，則該 large itemsets 即為廣播順序。

Database	D
TID	Items
$C_1$	r5, r2, r3, r4, r18, r22, r26, r28, r34, r38
$C_2$	r5, r4, r21, r7, r17, r25, r39,
$C_3$	r7, r23, r9, r6, r15, r27, r36,
$C_4$	r8, r1, r3, r6, r16, r25, r24, r28
$C_5$	r4, r15, r9, r8, r14, r24, r37, r26
$C_6$	r1, r2, r9, r6, r11, r25, r28, r37
$C_7$	r7, r3, r4, r8, r10, r29, r23, r36
$C_8$	r15, r9, r14, r26, r23, r29, r33
$C_9$	r4, r5, r17, r9, r10, r30, r37
$C_{10}$	r2, r23, r4, r6, r12, r25, r26, r35
$C_{11}$	r1, r3, r14, r16, r12, r29, r34,
$C_{12}$	r1, r12, r19, r6, r21, r28, r35
$C_{13}$	r7, r30, r13, r18, r27, r31, r23
$C_{14}$	r11, r6, r8, r13, r26, r19, r38
$C_{15}$	r24, r6, r7, r9, r15, r27, r32,
$C_{16}$	r1, r3, r14, r6, r12, r29, r35,
$C_{17}$	r1, r12, r19, r6, r28, r33, r38, r40
$C_{18}$	r1, r22, r5, r6, r11, r15, r28, r35
$C_{19}$	r1, r3, r4, r8, r10, r19, r40
$C_{20}$	r1, r5, r7, r9, r10, r30, r32

#### 20個client端的實驗

Server端以20個資料欄位為基礎，以20個Client產生20組不一樣的資料欄位，在Client端所需要的資料欄位也是經由隨機亂數產生，資料欄位

最多40個，套入我們的方法和隨機亂數作資料庫廣播時的資料欄位比較。結果是我們的方法比隨機亂數加總起來的長度還要短，相差75。

20 個 client 我們的方法和隨機亂數產生的資料順序作比較

Client	我們的方法 ( $r_1, r_6, r_{28}, r_{35}, r_3, r_9, r_{15}, r_1$ $9, r_4, r_8, r_{29}, r_5, r_{10}, r_{26}, r_{11},$ $r_{25}, r_7, r_{23}, r_{27}, r_{14}, r_2, r_{12},$ $r_{30}, r_{38}, r_{13}, r_{16}, r_{17}, r_{18}, r_{22},$ $r_{24}, r_{32}, r_{33}, r_{34}, r_{36}, r_{37},$ $r_{40}, r_{21}, r_{31}, r_{39}, r_{20}$ )	隨機亂數 $R_1$ ( $r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9,$ $r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16},$ $r_{17}, r_{18}, r_{19}, r_{20}, r_{21}, r_{22}, r_{23},$ $r_{24}, r_{25}, r_{26}, r_{27}, r_{28}, r_{29}, r_{30},$ $r_{31}, r_{32}, r_{33}, r_{34}, r_{35}, r_{36}, r_{37},$ $r_{38}, r_{39}, r_{40}$ )
$r_5, r_2, r_3, r_4, r_{18}, r_{22}, r_{26}, r_{28}, r_{34}, r_{38}$	31	37
$r_5, r_4, r_{21}, r_7, r_{17}, r_{25}, r_{39},$	31	36
$r_7, r_{23}, r_9, r_6, r_{15}, r_{27}, r_{36},$	33	31
$r_8, r_1, r_3, r_6, r_{16}, r_{25}, r_{24}, r_{28}$	29	28
$r_4, r_{15}, r_9, r_8, r_{14}, r_{24}, r_{37}, r_{26}$	29	34
$r_1, r_2, r_9, r_6, r_{11}, r_{25}, r_{28}, r_{37}$	35	37
$r_7, r_3, r_4, r_8, r_{10}, r_{29}, r_{23}, r_{36}$	29	34
$r_{15}, r_9, r_{14}, r_{26}, r_{23}, r_{29}, r_{33}$	26	25
$r_4, r_5, r_{17}, r_9, r_{10}, r_{30}, r_{37}$	29	34
$r_2, r_{23}, r_4, r_6, r_{12}, r_{25}, r_{26}, r_{35}$	20	34
$r_1, r_3, r_{14}, r_{16}, r_{12}, r_{29}, r_{34},$	33	34
$r_1, r_{12}, r_{19}, r_6, r_{21}, r_{28}, r_{35}$	37	35
$r_7, r_{30}, r_{13}, r_{18}, r_{27}, r_{31}, r_{23}$	21	25
$r_{11}, r_6, r_8, r_{13}, r_{26}, r_{19}, r_{38}$	23	31
$r_{24}, r_6, r_7, r_9, r_{15}, r_{27}, r_{32},$	29	27
$r_1, r_3, r_{14}, r_6, r_{12}, r_{29}, r_{35},$	22	35
$r_1, r_{12}, r_{19}, r_6, r_{28}, r_{33}, r_{38}, r_{40}$	36	40
$r_1, r_{22}, r_5, r_6, r_{11}, r_{15}, r_{28}, r_{35}$	29	35
$r_1, r_3, r_4, r_8, r_{10}, r_{19}, r_{40}$	36	40
$r_1, r_5, r_7, r_9, r_{10}, r_{30}, r_{32}$	31	32
加總	589	664

(五)我們再以 Client 25 為一組，分別是  $C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}, C_{15}, C_{16}, C_{17}, C_{18}, C_{19}, C_{20}, C_{21}, C_{22}, C_{23}, C_{24}, C_{25}$ ，最小交易支持度是 3，透過 Arpioir 演算法得到 large itemsets 為  $\{r1, r6, r28, r31\}$ 、 $\{r7, r23, r31, r36\}$ 。將這 large itemsets 再和 Server 端  $D=50$  對應，再和 Random 做比較，看其順序長度是否為各 Client 所需求的長度，則該 large itemsets 即為廣播順序。

Database D

TID	Items
$C_1$	$r_5, r_2, r_3, r_4, r_{18}, r_{22}, r_{26}, r_{28}, r_{33}, r_{38}$
$C_2$	$r_5, r_4, r_{21}, r_7, r_{17}, r_{25}, r_{32}, r_{39}, r_{40}$
$C_3$	$r_7, r_{23}, r_9, r_6, r_{15}, r_{27}, r_{31}, r_{36}, r_{48}$
$C_4$	$r_8, r_1, r_3, r_6, r_{16}, r_{25}, r_{24}, r_{28}, r_{31}, r_{50}$
$C_5$	$r_4, r_{15}, r_9, r_8, r_{14}, r_{24}, r_{33}, r_{37}, r_{26}, r_{46}$
$C_6$	$r_1, r_2, r_9, r_6, r_{11}, r_{25}, r_{28}, r_{36}, r_{37}$
$C_7$	$r_7, r_3, r_4, r_8, r_{10}, r_{29}, r_{23}, r_{31}, r_{36}, r_{47}$
$C_8$	$r_{15}, r_9, r_{14}, r_{26}, r_{23}, r_{29}, r_{32}, r_{33}, r_{39}$
$C_9$	$r_4, r_5, r_{17}, r_9, r_{10}, r_{18}, r_{30}, r_{31}, r_{37}, r_{39}$
$C_{10}$	$r_2, r_{23}, r_4, r_6, r_{12}, r_{25}, r_{26}, r_{35}, r_{38}, r_{41}$
$C_{11}$	$r_1, r_3, r_{14}, r_{16}, r_{12}, r_{29}, r_{32}, r_{34}, r_{40}$
$C_{12}$	$r_1, r_{12}, r_{19}, r_6, r_{21}, r_{28}, r_{31}, r_{33}, r_{35}, r_{42}$
$C_{13}$	$r_7, r_{30}, r_{13}, r_{18}, r_{27}, r_{31}, r_{23}, r_{36}, r_{40}$
$C_{14}$	$r_6, r_8, r_{11}, r_{13}, r_{26}, r_{19}, r_{37}, r_{38}, r_{43}$
$C_{15}$	$r_2, r_6, r_7, r_9, r_{15}, r_{27}, r_{31}, r_{33}, r_{37}, r_{39}$
$C_{16}$	$r_1, r_3, r_{14}, r_6, r_{12}, r_{29}, r_{35}, r_{36}, r_{39}, r_{44}$
$C_{17}$	$r_1, r_{12}, r_{19}, r_6, r_{28}, r_{31}, r_{33}, r_{38}, r_{40}, r_{49}$
$C_{18}$	$r_1, r_5, r_6, r_{11}, r_{15}, r_{22}, r_{28}, r_{32}, r_{35}$
$C_{19}$	$r_1, r_3, r_4, r_8, r_{10}, r_{19}, r_{36}, r_{40}, r_{45}$
$C_{20}$	$r_1, r_5, r_7, r_9, r_{10}, r_{18}, r_{30}, r_{32}$
$C_{21}$	$r_7, r_{20}, r_{23}, r_{28}, r_{29}, r_{34}, r_{36}, r_{46}$
$C_{22}$	$r_3, r_6, r_7, r_9, r_{14}, r_{17}, r_{23}, r_{35}, r_{47}$
$C_{23}$	$r_1, r_5, r_{19}, r_{28}, r_{34}, r_{36}, r_{37}, r_{39}, r_{46}, r_{50}$
$C_{24}$	$r_{16}, r_{18}, r_{21}, r_{23}, r_{26}, r_{29}, r_{32}, r_{35}, r_{44}$
$C_{25}$	$r_{15}, r_{24}, r_{25}, r_{27}, r_{37}, r_{39}, r_{42}, r_{45}, r_{48}$

## 25個client端的實驗

Server端以20個資料欄位為基礎，以25個Clinet產生25組不一樣的資料欄位，在Clinet端所需要的資料欄位也是經由隨機亂數產生，資料欄位最多50個，套入我們的方法和隨機亂數作資料庫廣播時的資料欄位比較。結果是我們的方法比隨機亂數加總起來的長度還要短，相差52。

## 25 個 client 我們的方法和隨機亂數產生的資料順序作比較

Client	我們的方法 ( ) Γ <sub>1</sub> ,Γ <sub>6</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>7</sub> ,Γ <sub>23</sub> ,Γ <sub>36</sub> Γ <sub>15</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub> ,Γ <sub>12</sub> ,Γ <sub>35</sub> ,Γ <sub>19</sub> ,Γ <sub>9</sub> ,Γ <sub>33</sub> ,Γ <sub>27</sub> ,Γ <sub>3</sub> ,Γ <sub>8</sub> ,Γ <sub>14</sub> ,Γ <sub>40</sub> ,Γ <sub>11</sub> , Γ <sub>25</sub> ,Γ <sub>2</sub> ,Γ <sub>4</sub> ,Γ <sub>10</sub> ,Γ <sub>29</sub> ,Γ <sub>32</sub> ,Γ <sub>26</sub> , Γ <sub>18</sub> ,Γ <sub>5</sub> ,Γ <sub>17</sub> ,Γ <sub>21</sub> ,Γ <sub>24</sub> ,Γ <sub>34</sub> ,Γ <sub>38</sub> ,Γ <sub>13</sub> ,Γ <sub>16</sub> ,Γ <sub>22</sub> ,Γ <sub>30</sub> ,Γ <sub>50</sub> ,Γ <sub>48</sub> , Γ <sub>47</sub> ,Γ <sub>46</sub> ,Γ <sub>45</sub> ,Γ <sub>44</sub> ,Γ <sub>42</sub> ,Γ <sub>49</sub> , Γ <sub>41</sub> ,Γ <sub>43</sub> ,Γ <sub>20</sub>	隨機亂數R <sub>1</sub> ( Γ <sub>1</sub> ,Γ <sub>2</sub> ,Γ <sub>3</sub> ,Γ <sub>4</sub> ,Γ <sub>5</sub> ,Γ <sub>6</sub> ,Γ <sub>7</sub> ,Γ <sub>8</sub> ,Γ <sub>9</sub> ,Γ <sub>10</sub> ,Γ <sub>11</sub> ,Γ <sub>12</sub> ,Γ <sub>13</sub> ,Γ <sub>14</sub> ,Γ <sub>15</sub> ,Γ <sub>16</sub> ,Γ <sub>17</sub> ,Γ <sub>18</sub> ,Γ <sub>19</sub> ,Γ <sub>20</sub> ,Γ <sub>21</sub> ,Γ <sub>22</sub> ,Γ <sub>23</sub> ,Γ <sub>24</sub> ,Γ <sub>25</sub> ,Γ <sub>26</sub> ,Γ <sub>27</sub> ,Γ <sub>28</sub> ,Γ <sub>29</sub> ,Γ <sub>30</sub> ,Γ <sub>31</sub> ,Γ <sub>32</sub> ,Γ <sub>33</sub> ,Γ <sub>34</sub> ,Γ <sub>35</sub> ,Γ <sub>36</sub> ,Γ <sub>37</sub> ,Γ <sub>38</sub> ,Γ <sub>39</sub> ,Γ <sub>40</sub> ,Γ <sub>41</sub> ,Γ <sub>42</sub> ,Γ <sub>43</sub> ,Γ <sub>44</sub> ,Γ <sub>45</sub> ,Γ <sub>46</sub> ,Γ <sub>47</sub> ,Γ <sub>48</sub> ,Γ <sub>49</sub> ,Γ <sub>50</sub> )
Γ <sub>5</sub> ,Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>4</sub> ,Γ <sub>18</sub> ,Γ <sub>22</sub> ,Γ <sub>26</sub> ,Γ <sub>28</sub> ,Γ <sub>33</sub> ,Γ <sub>38</sub>	35	37
Γ <sub>5</sub> ,Γ <sub>4</sub> , Γ <sub>2</sub> , Γ <sub>1</sub> , Γ <sub>7</sub> ,Γ <sub>17</sub> ,Γ <sub>25</sub> , Γ <sub>32</sub> ,Γ <sub>39</sub> ,Γ <sub>40</sub> ,	22	37
Γ <sub>7</sub> ,Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>9</sub> , Γ <sub>6</sub> ,Γ <sub>15</sub> ,Γ <sub>27</sub> ,Γ <sub>31</sub> ,Γ <sub>36</sub> ,Γ <sub>48</sub>	39	43
Γ <sub>8</sub> ,Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>6</sub> ,Γ <sub>16</sub> ,Γ <sub>25</sub> , Γ <sub>24</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>50</sub>	40	50
Γ <sub>4</sub> ,Γ <sub>15</sub> , Γ <sub>9</sub> , Γ <sub>8</sub> ,Γ <sub>14</sub> Γ <sub>24</sub> ,Γ <sub>33</sub> ,Γ <sub>37</sub> ,Γ <sub>26</sub> ,Γ <sub>46</sub>	35	43
Γ <sub>1</sub> ,Γ <sub>2</sub> , Γ <sub>9</sub> , Γ <sub>6</sub> ,Γ <sub>11</sub> ,Γ <sub>25</sub> , Γ <sub>28</sub> ,Γ <sub>36</sub> ,Γ <sub>37</sub>	20	37
Γ <sub>7</sub> ,Γ <sub>3</sub> , Γ <sub>4</sub> , Γ <sub>8</sub> , Γ <sub>10</sub> ,Γ <sub>29</sub> ,Γ <sub>23</sub> ,Γ <sub>31</sub> ,Γ <sub>36</sub> ,Γ <sub>47</sub>	38	45
Γ <sub>15</sub> ,Γ <sub>9</sub> , Γ <sub>1</sub> , Γ <sub>4</sub> , Γ <sub>2</sub> , Γ <sub>6</sub> ,Γ <sub>23</sub> , Γ <sub>29</sub> , Γ <sub>32</sub> ,Γ <sub>33</sub> ,Γ <sub>39</sub>	22	31
Γ <sub>4</sub> ,Γ <sub>5</sub> ,Γ <sub>1</sub> , Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>10</sub> , Γ <sub>18</sub> , Γ <sub>30</sub> ,Γ <sub>31</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub>	35	36
Γ <sub>2</sub> ,Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>4</sub> , Γ <sub>6</sub> ,Γ <sub>12</sub> , Γ <sub>25</sub> ,Γ <sub>26</sub> ,Γ <sub>35</sub> ,Γ <sub>38</sub> ,Γ <sub>41</sub>	47	40
Γ <sub>1</sub> ,Γ <sub>3</sub> , Γ <sub>14</sub> , Γ <sub>1</sub> , Γ <sub>6</sub> ,Γ <sub>12</sub> ,Γ <sub>29</sub> , Γ <sub>32</sub> ,Γ <sub>34</sub> ,Γ <sub>40</sub> ,	37	40
Γ <sub>1</sub> ,Γ <sub>1</sub> , Γ <sub>2</sub> , Γ <sub>19</sub> , Γ <sub>6</sub> , Γ <sub>21</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>33</sub> ,Γ <sub>35</sub> ,Γ <sub>42</sub>	46	41
Γ <sub>7</sub> ,Γ <sub>30</sub> , Γ <sub>13</sub> , Γ <sub>18</sub> ,Γ <sub>27</sub> ,Γ <sub>31</sub> ,Γ <sub>23</sub> ,Γ <sub>36</sub> ,Γ <sub>40</sub> ,	35	34
Γ <sub>6</sub> ,Γ <sub>8</sub> , Γ <sub>1</sub> , Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>2</sub> , Γ <sub>6</sub> , Γ <sub>19</sub> ,Γ <sub>37</sub> ,Γ <sub>38</sub> ,Γ <sub>43</sub>	48	38
Γ <sub>2</sub> ,Γ <sub>6</sub> ,Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>15</sub> ,Γ <sub>27</sub> ,Γ <sub>31</sub> ,Γ <sub>33</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub> ,	21	38
Γ <sub>1</sub> ,Γ <sub>3</sub> , Γ <sub>14</sub> , Γ <sub>6</sub> ,Γ <sub>12</sub> ,Γ <sub>29</sub> ,Γ <sub>35</sub> ,Γ <sub>36</sub> ,Γ <sub>39</sub> ,Γ <sub>44</sub> ,	45	44
Γ <sub>1</sub> ,Γ <sub>1</sub> , Γ <sub>2</sub> , Γ <sub>19</sub> , Γ <sub>6</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>33</sub> ,Γ <sub>38</sub> ,Γ <sub>40</sub> ,Γ <sub>49</sub>	47	49
Γ <sub>1</sub> , Γ <sub>5</sub> , Γ <sub>6</sub> ,Γ <sub>11</sub> ,Γ <sub>15</sub> , Γ <sub>2</sub> , Γ <sub>2</sub> , Γ <sub>28</sub> , Γ <sub>32</sub> ,Γ <sub>35</sub>	38	35

Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>4</sub> , Γ <sub>8</sub> , Γ <sub>10</sub> , Γ <sub>19</sub> , Γ <sub>36</sub> , Γ <sub>40</sub> , Γ <sub>45</sub>	44	45
Γ <sub>1</sub> , Γ <sub>5</sub> , Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>10</sub> , Γ <sub>18</sub> , Γ <sub>30</sub> , Γ <sub>32</sub>	39	32
Γ <sub>7</sub> , Γ <sub>20</sub> , Γ <sub>23</sub> , Γ <sub>28</sub> , Γ <sub>29</sub> , Γ <sub>34</sub> , Γ <sub>36</sub> , Γ <sub>46</sub> ,	48	40
Γ <sub>3</sub> , Γ <sub>6</sub> , Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>14</sub> , Γ <sub>17</sub> , Γ <sub>23</sub> , Γ <sub>35</sub> , Γ <sub>47</sub>	40	45
Γ <sub>1</sub> , Γ <sub>5</sub> , Γ <sub>9</sub> , Γ <sub>28</sub> , Γ <sub>34</sub> , Γ <sub>36</sub> , Γ <sub>37</sub> , Γ <sub>39</sub> , Γ <sub>46</sub> , Γ <sub>50</sub>	43	50
Γ <sub>16</sub> , Γ <sub>18</sub> , Γ <sub>21</sub> , Γ <sub>23</sub> , Γ <sub>26</sub> , Γ <sub>29</sub> , Γ <sub>32</sub> , Γ <sub>35</sub> , Γ <sub>44</sub>	39	29
Γ <sub>15</sub> , Γ <sub>24</sub> , Γ <sub>25</sub> , Γ <sub>27</sub> , Γ <sub>37</sub> , Γ <sub>39</sub> , Γ <sub>42</sub> , Γ <sub>45</sub> , Γ <sub>48</sub> ,	38	34
加總	941	993

#### 第四節 實驗結果--排序方法

在 Apriori 無線廣播的排序方法中，把 large itemset 的這些資料集中，因為它是最多 Client 端要的資料欄位，所以在考慮排序條件是 1. 長度由長至短排列，2. 以不破壞前面 large-itemsets 順序為原則，可得最佳廣播資料的順序。換句話說，廣播順序排序由最長的 large-itemsets 頻率高的先考慮，到短的頻率低的，依序選擇。如果長度會被破壞，則短的低頻率廣播封包，則被排列在依最小限制支持度大小的位置。調整中間的順序以符合最佳廣播排序。以下是各組實驗結果：

(一) 經實驗結果 5 個 Client 端和我們的方法和隨機亂數產生的資料順序作比較，我們的方法所算出的值=37，Random 值=39。廣播最佳排序為如下表 1：

(表 1) 5 個 Client，資料欄位 10 與 Random 比較

Client	我們的方法 (Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>4</sub> , Γ <sub>5</sub> , Γ <sub>6</sub> , Γ <sub>7</sub> , Γ <sub>8</sub> , Γ <sub>9</sub> , Γ <sub>2</sub> , Γ <sub>10</sub> )	隨機亂數 R <sub>1</sub> (Γ <sub>1</sub> , Γ <sub>7</sub> , Γ <sub>3</sub> , Γ <sub>10</sub> , Γ <sub>5</sub> , Γ <sub>6</sub> , Γ <sub>4</sub> , Γ <sub>8</sub> , Γ <sub>9</sub> , Γ <sub>2</sub> )
Γ <sub>1</sub> , Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>4</sub>	9	10
Γ <sub>5</sub> , Γ <sub>4</sub> , Γ <sub>1</sub> , Γ <sub>7</sub>	6	7
Γ <sub>7</sub> , Γ <sub>3</sub> , Γ <sub>9</sub> , Γ <sub>6</sub>	7	8
Γ <sub>8</sub> , Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>6</sub> ,	7	8
Γ <sub>4</sub> , Γ <sub>5</sub> , Γ <sub>9</sub> , Γ <sub>8</sub> , Γ <sub>10</sub>	8	6
加總	37	39

(二) 經實驗結果 10 個 Client 端和我們的方法和隨機亂數產生的資料順序作比較，我們的方法所算出的值=132，Random 值=152。廣播最佳排序為如下表 2：

(表 2) 10 個 Client，資料欄位 20 與 Random 比較

Client	我們的方法 ( $r_4, r_5, r_9, r_2, r_3, r_6, r_1, r_7, r_8, r_{10}$ ) ( $r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16}, r_{17}, r_{18},$ $r_{19}, r_{20}$ )	隨機亂數 $R_1$ ( $r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}$ ) ( $r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16}, r_{17}, r_{18},$ $r_{19}, r_{20}$ )
$r_1, r_2, r_3, r_4, r_{18}$	9	18
$r_5, r_4, r_1, r_7, r_{17}$	17	17
$r_7, r_3, r_9, r_6, r_{15}$	13	15
$r_8, r_1, r_3, r_6, r_{16}$	12	16
$r_4, r_5, r_9, r_8, r_{14}$	14	14
$r_1, r_2, r_9, r_6, r_{11}$	9	11
$r_7, r_3, r_8, r_{10}, r_{19}$	15	19
$r_5, r_9, r_4, r_6, r_{13}$	13	13
$r_4, r_5, r_7, r_9, r_{10},$	10	10
$r_2, r_3, r_4, r_6, r_{12}, r_{20}$	20	19
加總	132	152

(三) 經實驗結果 15 個 Client 端和我們的方法和隨機亂數產生的資料順序作比較，我們的方法所算出的值=283，Random 值=333。廣播最佳排序為如下表 3：

(表 3) 15 個 Client，資料欄位 30 與 Random 比較

Client	我們的方法 ( $r_6, r_9, r_{15}, r_1, r_3, r_4, r_7, r_5, r_{12}, r_{10}, r_2, r_8, r_{13}, r_{14}, r_{19}, r_{11}, r_{18}, r_{16}, r_{17}, r_{20}, r_{30}, r_{29}, r_{28}, r_{27}, r_{26}, r_{25}, r_{24}, r_{23}, r_{22}, r_{21}$ )	隨機亂數 $R_1$ ( $r_1, r_2, r_3, r_4, r_5, r_6, r_2, r_8, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16}, r_{17}, r_{18}, r_{19}, r_{20}, r_{21}, r_{22}, r_{23}, r_{24}, r_{25}, r_{26}, r_{27}, r_{28}, r_{29}, r_{30}$ )
$r_5, r_2, r_3, r_4, r_{18}, r_{26}$	20	25
$r_5, r_4, r_{11}, r_7, r_{17}, r_{25}$	20	22
$r_7, r_3, r_9, r_6, r_{15}, r_{27}$	23	25
$r_8, r_1, r_3, r_6, r_{16}$	18	16
$r_4, r_{15}, r_9, r_8, r_{14}, r_{24}$	25	21
$r_1, r_2, r_9, r_6, r_{11}, r_{15}, r_{28}$	22	28
$r_7, r_3, r_4, r_8, r_{10}, r_{19}$	11	17
$r_{15}, r_9, r_{14}, r_6, r_{13}, r_{29}$	21	24
$r_4, r_5, r_7, r_9, r_{10}, r_{30}$	20	27
$r_2, r_3, r_4, r_6, r_{12}, r_{20}$	21	19
$r_1, r_3, r_{14}, r_6, r_{12}, r_{29}$	21	29
$r_1, r_{12}, r_{19}, r_6, r_{28}$	22	28
$r_7, r_{13}, r_{18}, r_{10}, r_{27}$	17	21
$r_8, r_1, r_{13}, r_6, r_{19}$	15	19
$r_4, r_6, r_7, r_9, r_{15}$	7	12
加總	283	333

(四) 經實驗結果 20 個 Client 端和我們的方法和隨機亂數產生的資料順序作比較，我們的方法所算出的值=589，Random 值=664。廣播最佳排序為如下表 4：



(表 4) 20 個 Client，資料欄位 40 與 Random 比較

Client	我們的方法 ( $r_1, r_6, r_{28}, r_{35}, r_3, r_9, r_{15}, r_{19}, r_4, r_8, r_{29}, r_5, r_{10}, r_{26}, r_{11}, r_{25}, r_7, r_{23}, r_{27}, r_{14}, r_2, r_{12}, r_{30}, r_{38}, r_{13}, r_{16}, r_{17}, r_{18}, r_{22}, r_{24}, r_{32}, r_{33}, r_{34}, r_{36}, r_{37}, r_{40}, r_{21}, r_{31}, r_{39}, r_{20}$ )	隨機亂數 $R_1$ ( $r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}, r_{11}, r_{12}, r_{13}, r_{14}, r_{15}, r_{16}, r_{17}, r_{18}, r_{19}, r_{20}, r_{21}, r_{22}, r_{23}, r_{24}, r_{25}, r_{26}, r_{27}, r_{28}, r_{29}, r_{30}, r_{31}, r_{32}, r_{33}, r_{34}, r_{35}, r_{36}, r_{37}, r_{38}, r_{39}, r_{40}$ )
$r_5, r_2, r_3, r_4, r_{18}, r_{22}, r_{26}, r_{28}, r_{34}, r_{38}$	31	37
$r_5, r_4, r_{21}, r_7, r_{17}, r_{25}, r_{39}, r_7, r_{23}, r_9, r_6, r_{15}, r_{27}, r_{36}, r_8, r_1, r_3, r_6, r_{16}, r_{25}, r_{24}, r_{28}$	31	36
$r_4, r_{15}, r_9, r_8, r_{14}, r_{24}, r_{37}, r_{26}$	33	31
$r_1, r_2, r_9, r_6, r_{11}, r_{25}, r_{28}, r_{37}$	29	28
$r_7, r_3, r_4, r_8, r_{10}, r_{29}, r_{23}, r_{36}$	29	34
$r_{15}, r_9, r_{14}, r_{26}, r_{23}, r_{29}, r_{33}$	29	37
$r_4, r_5, r_{17}, r_9, r_{10}, r_{30}, r_{37}$	29	34
$r_2, r_{23}, r_4, r_6, r_{12}, r_{25}, r_{26}, r_{35}$	26	34
$r_1, r_3, r_{14}, r_{16}, r_{12}, r_{29}, r_{34}, r_1, r_{12}, r_{19}, r_6, r_{21}, r_{28}, r_{35}$	33	34
$r_7, r_{30}, r_{13}, r_{18}, r_{27}, r_{31}, r_{23}$	37	35
$r_{11}, r_6, r_8, r_{13}, r_{26}, r_{19}, r_{38}$	21	25
$r_{24}, r_6, r_7, r_9, r_{15}, r_{27}, r_{32}, r_1, r_3, r_{14}, r_6, r_{12}, r_{29}, r_{35}, r_1, r_{12}, r_{19}, r_6, r_{28}, r_{33}, r_{38}, r_{40}$	23	31
$r_1, r_{22}, r_5, r_6, r_{11}, r_{15}, r_{28}, r_{35}$	29	27
$r_1, r_3, r_4, r_8, r_{10}, r_{19}, r_{40}$	22	35
$r_1, r_5, r_7, r_9, r_{10}, r_{30}, r_{32}$	36	40
加總	589	664

(五) 經實驗結果 25 個 Client 端和我們的方法和隨機亂數產生的資料順序作比較，我們的方法所算出的值=941，Random 值=993。廣播

最佳排序為如下表 5：

(表 5) 25 個 Client，資料欄位 50 與 Random 比較

Client	我們的方法 Γ <sub>1</sub> ,Γ <sub>6</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>7</sub> ,Γ <sub>23</sub> ,Γ <sub>36</sub> Γ <sub>15</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub> ,Γ <sub>12</sub> ,Γ <sub>35</sub> ,Γ <sub>19</sub> ,Γ <sub>9</sub> ,Γ <sub>33</sub> ,Γ <sub>27</sub> ,Γ <sub>3</sub> ,Γ <sub>8</sub> ,Γ <sub>14</sub> ,Γ <sub>40</sub> ,Γ <sub>11</sub> , Γ <sub>25</sub> ,Γ <sub>2</sub> ,Γ <sub>4</sub> ,Γ <sub>10</sub> ,Γ <sub>29</sub> ,Γ <sub>32</sub> ,Γ <sub>26</sub> , Γ <sub>18</sub> ,Γ <sub>5</sub> ,Γ <sub>17</sub> ,Γ <sub>21</sub> ,Γ <sub>24</sub> ,Γ <sub>34</sub> ,Γ <sub>38</sub> ,Γ <sub>13</sub> ,Γ <sub>16</sub> ,Γ <sub>22</sub> ,Γ <sub>30</sub> ,Γ <sub>50</sub> ,Γ <sub>48</sub> , Γ <sub>47</sub> ,Γ <sub>46</sub> ,Γ <sub>45</sub> ,Γ <sub>44</sub> ,Γ <sub>42</sub> ,Γ <sub>49</sub> , Γ <sub>41</sub> ,Γ <sub>43</sub> ,Γ <sub>20</sub>	隨機亂數R <sub>1</sub> ( Γ <sub>1</sub> ,Γ <sub>2</sub> ,Γ <sub>3</sub> ,Γ <sub>4</sub> ,Γ <sub>5</sub> ,Γ <sub>6</sub> ,Γ <sub>7</sub> ,Γ <sub>8</sub> ,Γ <sub>9</sub> ,Γ <sub>10</sub> ,Γ <sub>11</sub> ,Γ <sub>12</sub> ,Γ <sub>13</sub> ,Γ <sub>14</sub> ,Γ <sub>15</sub> ,Γ <sub>16</sub> ,Γ <sub>17</sub> ,Γ <sub>18</sub> ,Γ <sub>19</sub> ,Γ <sub>20</sub> ,Γ <sub>21</sub> ,Γ <sub>22</sub> ,Γ <sub>23</sub> ,Γ <sub>24</sub> ,Γ <sub>25</sub> ,Γ <sub>26</sub> ,Γ <sub>27</sub> ,Γ <sub>28</sub> ,Γ <sub>29</sub> ,Γ <sub>30</sub> ,Γ <sub>31</sub> ,Γ <sub>32</sub> ,Γ <sub>33</sub> ,Γ <sub>34</sub> ,Γ <sub>35</sub> ,Γ <sub>36</sub> ,Γ <sub>37</sub> ,Γ <sub>38</sub> ,Γ <sub>39</sub> ,Γ <sub>40</sub> ,Γ <sub>41</sub> ,Γ <sub>42</sub> ,Γ <sub>43</sub> ,Γ <sub>44</sub> ,Γ <sub>45</sub> ,Γ <sub>46</sub> ,Γ <sub>47</sub> ,Γ <sub>48</sub> ,Γ <sub>49</sub> ,Γ <sub>50</sub> )
Γ <sub>5</sub> ,Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>4</sub> ,Γ <sub>18</sub> ,Γ <sub>22</sub> ,Γ <sub>26</sub> ,Γ <sub>28</sub> ,Γ <sub>33</sub> ,Γ <sub>38</sub>	35	37
Γ <sub>5</sub> ,Γ <sub>4</sub> , Γ <sub>2</sub> , Γ <sub>1</sub> , Γ <sub>7</sub> ,Γ <sub>17</sub> ,Γ <sub>25</sub> , Γ <sub>32</sub> ,Γ <sub>39</sub> ,Γ <sub>40</sub> ,	22	37
Γ <sub>7</sub> ,Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>9</sub> , Γ <sub>6</sub> ,Γ <sub>15</sub> ,Γ <sub>27</sub> ,Γ <sub>31</sub> ,Γ <sub>36</sub> ,Γ <sub>48</sub>	39	43
Γ <sub>8</sub> ,Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>6</sub> ,Γ <sub>16</sub> ,Γ <sub>25</sub> , Γ <sub>24</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>50</sub>	40	50
Γ <sub>4</sub> ,Γ <sub>15</sub> , Γ <sub>9</sub> , Γ <sub>8</sub> ,Γ <sub>14</sub> Γ <sub>24</sub> ,Γ <sub>33</sub> ,Γ <sub>37</sub> ,Γ <sub>26</sub> ,Γ <sub>46</sub>	35	43
Γ <sub>1</sub> ,Γ <sub>2</sub> , Γ <sub>9</sub> , Γ <sub>6</sub> ,Γ <sub>11</sub> ,Γ <sub>25</sub> , Γ <sub>28</sub> ,Γ <sub>36</sub> ,Γ <sub>37</sub>	20	37
Γ <sub>7</sub> ,Γ <sub>3</sub> , Γ <sub>4</sub> , Γ <sub>8</sub> , Γ <sub>10</sub> ,Γ <sub>29</sub> ,Γ <sub>23</sub> ,Γ <sub>31</sub> ,Γ <sub>36</sub> ,Γ <sub>47</sub>	38	45
Γ <sub>15</sub> ,Γ <sub>9</sub> , Γ <sub>1</sub> , Γ <sub>4</sub> , Γ <sub>2</sub> , Γ <sub>6</sub> ,Γ <sub>23</sub> , Γ <sub>29</sub> , Γ <sub>32</sub> ,Γ <sub>33</sub> ,Γ <sub>39</sub>	22	31
Γ <sub>4</sub> ,Γ <sub>5</sub> ,Γ <sub>1</sub> , Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>10</sub> , Γ <sub>18</sub> , Γ <sub>30</sub> ,Γ <sub>31</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub>	35	36
Γ <sub>2</sub> ,Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>4</sub> , Γ <sub>6</sub> ,Γ <sub>12</sub> , Γ <sub>25</sub> ,Γ <sub>26</sub> ,Γ <sub>35</sub> ,Γ <sub>38</sub> ,Γ <sub>41</sub>	47	40
Γ <sub>1</sub> ,Γ <sub>3</sub> , Γ <sub>14</sub> , Γ <sub>1</sub> , Γ <sub>6</sub> ,Γ <sub>12</sub> ,Γ <sub>29</sub> , Γ <sub>32</sub> ,Γ <sub>34</sub> ,Γ <sub>40</sub> ,	37	40
Γ <sub>1</sub> ,Γ <sub>1</sub> , Γ <sub>2</sub> , Γ <sub>19</sub> , Γ <sub>6</sub> , Γ <sub>21</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>33</sub> ,Γ <sub>35</sub> ,Γ <sub>42</sub>	46	41
Γ <sub>7</sub> ,Γ <sub>30</sub> , Γ <sub>13</sub> , Γ <sub>18</sub> ,Γ <sub>27</sub> ,Γ <sub>31</sub> ,Γ <sub>23</sub> ,Γ <sub>36</sub> ,Γ <sub>40</sub> ,	35	34
Γ <sub>6</sub> ,Γ <sub>8</sub> , Γ <sub>1</sub> , Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>2</sub> , Γ <sub>6</sub> , Γ <sub>19</sub> ,Γ <sub>37</sub> ,Γ <sub>38</sub> ,Γ <sub>43</sub>	48	38
Γ <sub>2</sub> ,Γ <sub>6</sub> ,Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>15</sub> ,Γ <sub>27</sub> ,Γ <sub>31</sub> ,Γ <sub>33</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub> ,	21	38
Γ <sub>1</sub> ,Γ <sub>3</sub> , Γ <sub>14</sub> , Γ <sub>6</sub> ,Γ <sub>12</sub> ,Γ <sub>29</sub> ,Γ <sub>35</sub> ,Γ <sub>36</sub> ,Γ <sub>39</sub> ,Γ <sub>44</sub> ,	45	44
Γ <sub>1</sub> ,Γ <sub>1</sub> , Γ <sub>2</sub> , Γ <sub>19</sub> , Γ <sub>6</sub> ,Γ <sub>28</sub> ,Γ <sub>31</sub> ,Γ <sub>33</sub> ,Γ <sub>38</sub> ,Γ <sub>40</sub> ,Γ <sub>49</sub>	47	49
Γ <sub>1</sub> , Γ <sub>5</sub> , Γ <sub>6</sub> ,Γ <sub>11</sub> ,Γ <sub>15</sub> , Γ <sub>2</sub> , Γ <sub>2</sub> , Γ <sub>28</sub> , Γ <sub>32</sub> ,Γ <sub>35</sub>	38	35
Γ <sub>1</sub> , Γ <sub>3</sub> , Γ <sub>4</sub> , Γ <sub>8</sub> , Γ <sub>10</sub> ,Γ <sub>19</sub> ,Γ <sub>36</sub> ,Γ <sub>40</sub> ,Γ <sub>45</sub>	44	45
Γ <sub>1</sub> , Γ <sub>5</sub> ,Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>10</sub> , Γ <sub>18</sub> , Γ <sub>30</sub> ,Γ <sub>32</sub>	39	32
Γ <sub>7</sub> ,Γ <sub>20</sub> , Γ <sub>23</sub> , Γ <sub>28</sub> ,Γ <sub>29</sub> ,Γ <sub>34</sub> ,Γ <sub>36</sub> ,Γ <sub>46</sub> ,	48	40
Γ <sub>3</sub> ,Γ <sub>6</sub> ,Γ <sub>7</sub> , Γ <sub>9</sub> , Γ <sub>14</sub> , Γ <sub>17</sub> , Γ <sub>23</sub> ,Γ <sub>35</sub> ,Γ <sub>47</sub>	40	45
Γ <sub>1</sub> ,Γ <sub>5</sub> , Γ <sub>1</sub> , Γ <sub>9</sub> , Γ <sub>2</sub> , Γ <sub>8</sub> ,Γ <sub>34</sub> Γ <sub>36</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub> ,Γ <sub>46</sub> ,Γ <sub>50</sub>	43	50
Γ <sub>16</sub> ,Γ <sub>18</sub> , Γ <sub>2</sub> , Γ <sub>3</sub> , Γ <sub>2</sub> , Γ <sub>6</sub> , Γ <sub>29</sub> ,Γ <sub>32</sub> ,Γ <sub>35</sub> ,Γ <sub>44</sub>	39	29

Γ <sub>15</sub> ,Γ <sub>24</sub> , Γ <sub>25</sub> , Γ <sub>27</sub> ,Γ <sub>37</sub> ,Γ <sub>39</sub> , Γ <sub>42</sub> ,Γ <sub>45</sub> ,Γ <sub>48</sub> , 加總	38 941	34 993
--	-----------	-----------

本研究實驗結果 Client 端與 Random 比較，如表 6 值與圖 7

(表 6) 資料長度與 Random 比較值

C,S	我們的值	Ran	我們的方法比 Ran 好之百分比
5,10	37	39	5%
10,10	74	83	10%
15,10	103	120	6%
20,10	126	162	28%
25,10	154	202	31%
10,20	132	152	13%
15,30	283	333	15%
20,40	589	664	11%
25,50	941	993	5%

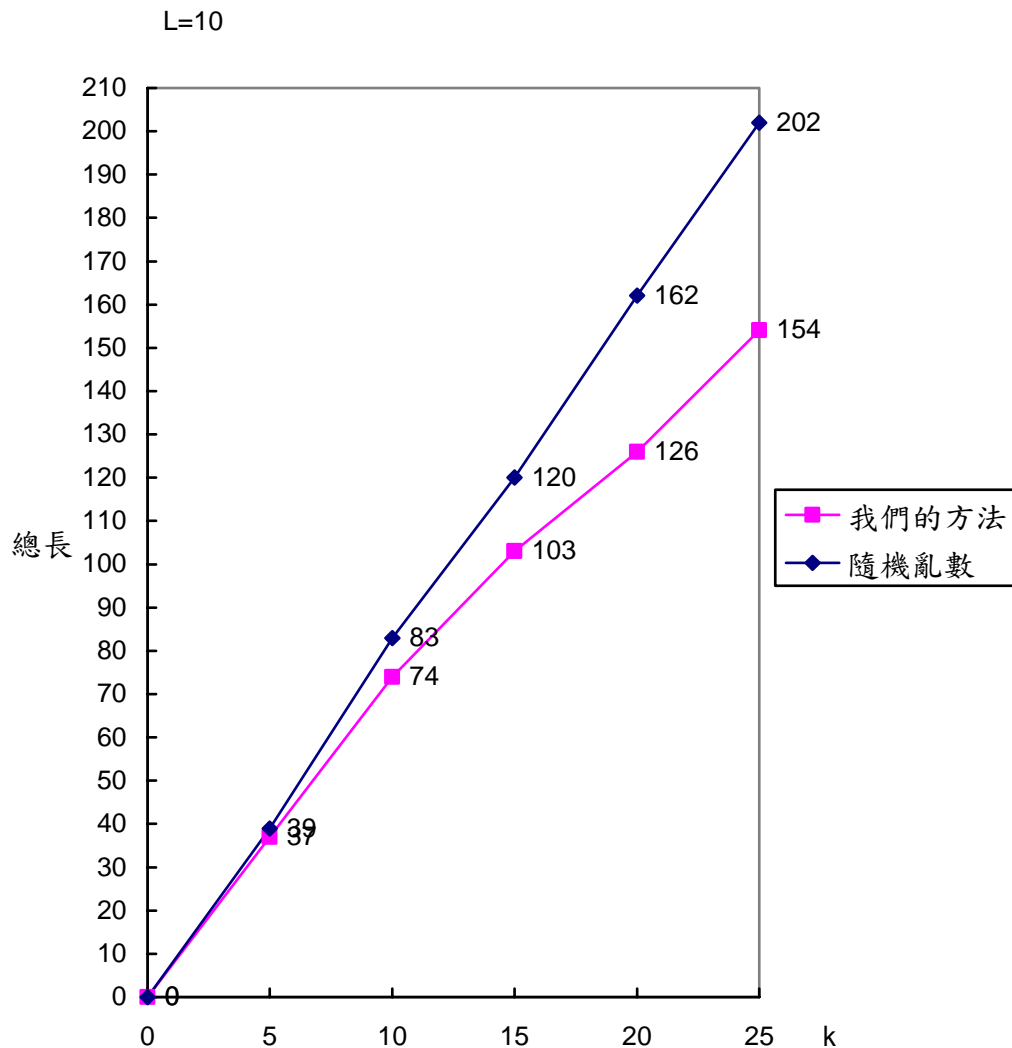


圖 5, L=10, 我們的方法與Random比較效率曲線圖

經多次實驗計算我們的方法與 Random 計算驗證結果，我們實驗計算個數為：長度 L=10，有 5 組 Client、10 組 Client、15 組 Client、20 組 Client、25 組 Client，等實驗組數。另外又計算 5 組 Client 對應 10 筆 Server 資料欄位，10 組 Client 對應 20 筆 Server 資料欄位，15 組 Client 對應 30 筆 Server 資料欄位，20 組 Client 對應 40 筆 Server 資料欄位，25 組 Client 對應 50 筆 Server 資料欄位。可以提供 Server 廣播出來的資料欄位順序，所需要的資料。Client 資料存取，可以找到一個最好的順序，對所有的資料欄位查詢越短越好，是本篇論文最主要的研究結果。

## 第六章 結論

本研究利用了資料探勘中的 Apriori 演算法來解決在無線廣播之中的節目排列問題，我們利用 Apriori 演算法的基本特性，依照每一個收聽者或接收訊息者所選擇節目的習慣、順序，用 Apriori 演算法來幫助收聽者尋找最佳解，對於收聽者而言也就是可以幫收聽者快速找到他所想要收聽的頻道，讓總接收的時間達到最短和最快速。

使用 Apriori 演算法的模式，套用在無線廣播頻道之中，很切合實際，因為廣播的帶狀型態就類似 Apriori 演算法之中的排序組合，如此再運用於廣播中去解決問題的是最佳的方法。

## 參考文獻

- [1] <http://liny.csie.nctu.edu.tw/mc.html>
- [2] <http://www.ibm.com>
- [3] G. Lee, and S.-C. Lo, "Broadcast Data Allocation for Efficient Access of Multiple Data Items in Mobile Environments," ACM Mobile Networks and Applications ( MONET ) , to appear.
- [4] G. Lee, M. S. Y, S.-C. Lo, and A.L.P. Chen, "A strategy for Efficient Access of Multiple Data Items in Mobile Environments," Proc. Int'l Conf. Mobile Data Management, pp. 71-78, 2002. [5] H. Toivonen, "Sampling Large Databases for Association Rules," VLDB, pp.134-145, 1996.
- [5]R. Agrawal, T. Imilienski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proceedings of the ACM SIGMOD Int'l Conf.on Management of Data, pp. 207-216, May 1993.
- [6] L. Breiman, J. Friedman, R. Olshen, and C. Stone, "Classification of Regression Trees," Wadsworth, 1984.
- [7] Agrawal, R., Srikant, R., "Fast Algorithms for Mining Association Rules", Proceedings of the 20<sup>th</sup> International Conference on Very Large Databases, Sep.1994,pp.487-499.
- [8] R. Agrawal and R. Srikant. "Fast Algorithms for Mining Association Rules". Proc. of the 20th VLDB Conference, 1994.
- [9] Jiawei Han, Jian Pei, and Yiwen Yin, "Mining Frequent Patterns without Candidate Generation," Proc. of 2000 ACM Int. Conf. on Management of Data (SIGMOD'00), pp. 1-12, Dallas, TX, May 2000.
- [10] R.J. Bayardo Jr., "Efficiently Mining Long Patterns from Databases," Proc. of ACM SIGMOD Conf. on Management of Data, pp. 85-93, ACM Press, 1998.
- [11] J. Han, G. Dong, and Y. Yin, "Efficient Mining of Partial Periodic

Patterns in Time Series Database,” Proceedings of Fifteenth International Conference on Data Engineering, Sydney, Australia, IEEE Computer Society, pp.106-115, 1999.

- [12] S. Acharya, R. Alonso, M. Franklin, and S. Zdonik, “Broadcast Disks : Data Management for Asymmetric Communication Environments,” Proc. ACM SIGMOD Conf., pp. 199-210, May 1995
- [13] D. Aksoy and M. J. Franklin, “Scheduling for Large-Scale On Demand Data Broadcasting.” Proc. IEEE INFOCOM Conf., pp. 651-659, 1998.
- [14] D. Adolphson and T. C. Hu, “Optimal Linear Ordering.” SIAMJ. Applied Math, vol. 25, pp. 403-423, 1973
- [15] A. Bar-Noy, J. Naor, and B. Schieber, “Pushing Dependent Data in Clients-Providers-Servers Systems,” Proc. MOBICOM Conf., pp. 222-230, 2000.
- [16] A. Bar-Noy and Y. Shilo, “Optimal Broadcasting of Two Files over an Asymmetric Channel,” Proc. IEEE INFOCOM Conf., pp. 267-274, 1999.
- [17] Y.C. Chehadeh, A.R. Hurson, and M. Kavehrad, “Object Organization on a Single Broadcast Channel in the Mobile Computing Environment,” Multimedia Tools and Applications, vol. 9, no. 1, pp. 69-94 July 1999.
- [18] Y.D. Chung and M.H. Kim, “QEM : A Scheduling Method for Wireless Broadcast Data,” Proc. Sixth Int’l Conf. Database Systems for Advanced Applications
- [19] M.R. Garey and D.S. Johnson, Computers and Intractability : A Guide to the Theory of NP-Completeness. Freeman Publishing, 1976.
- [20] V. Gondhalekar, R. Jain, and J. Werth, “Scheduling on Airdisks : Efficient Access to Personalized Information Services via Periodic Wireless Data Broadcast,” Proc. Int’l Conf. Comm. ( ICC ) , pp. 1276-1280, 1997
- [21] J. Gray, P. Sundaresan, S. Englert, K. Baclawski, and P.J. Weinberger, “Quickly Generating Billion-Record Synthetic Databases,” Proc. ACM SIGMOD Conf., pp.243-252, May 1994
- [22] A.R. Hurson, Y.C. Chehadeh, and J. Hannan, ”Object Organization on

- Parallel Broadcast Channels in a Global Information Environment,” Proc. IEEE Int’1 Conf. Performance, Computing, and Comm., pp. 347-353, 2000.
- [23] G. Herman, G. Gopal, K.C. Lee, and A. Weinrib, “The Datacycle Architecture for Very High Throughput Database Systems,” ACM SIGMOD Record, pp.97-103, 1987.
- [24] C.H. Hsu, G. Lee, and A.L.P. Chen, “Index and Data Allocation on Multiple Broadcast Channels Considering Data Access Frequencies,” Proc. Int’1 Conf. Mobile Data Management, pp.87-93, 2002.
- [25] K. Hoffman and R. Kunze, LINEAR ALGEBRA. Prentice Hall, 1971.
- [26] T. Imiehnski, S. Viswanathan, and B.R. Badrinath, “Energy Efficient Indexing on Air,” Proc. ACM SIGMOD C., . 25-36, May 1994.
- [27] T. Imiehnski, S. Viswanathan, and B.R. Badrinath, “Data on Air : Organization and Access,” IEEE Trans. Knowledge and Data Eng., vol. 9, no. 3, pp. 353-372, May/June 1997.
- [28] S.C. Lo and A.L.P. Chen, “Optimal Index and Data Allocation in Multiple Broadcast Channels,” Proc. 16<sup>th</sup> IEEE Int’1 Conf. Data Eng., PP. 293-302, 2000.
- [29] S.C. Lo and A.L.P. Chen, “An Adaptive Access Method for Broadcast Data under an Error-Prone Mobile Environment,” IEEE Trans. Knowledge and Data Eng., vol. 12, no 4, PP. 609-620, July/Aug. 2000.
- [30] G. Lee, S.-C. Lo, and A.L.P. Chen, “Data Allocation on Wireless Broadcast Channels for Efficient Query Processing.” Technical report, <http://mckm.csie.ndhu.edu.tw/lee-research.htm>, 2002.
- [31] G. Lee, and S.-C. Lo, “Broadcast Data Allocation for Efficient Access of Multiple Data Items in Mobile Environments,” ACM Mobile Networks and Applications ( MONET ) , to appear.
- [32] G. Lee, M. S. Y, S.-C. Lo, and A.L.P. Chen, “A strategy for Efficient Access of Multiple Data Items in Mobile Environments,” Proc. Int’1 Conf. Mobile Data Management, pp. 71-78, 2002.



- [33] W.C. Lee and D.L. Lee, "Using Signature Techniques for Information Filtering in Wireless and Mobile Environments," *Distributed and Parallel Databases*, vol. 4, no. 3, pp.205-227, July 1996.
- [34] A. Si and H.v. Leong, "Query Optimization for Broadcast Database," *Data and Knowledge Eng.*, vol 29, no. 3, pp.351-380, Mar.1999.
- [35] K. Thulasiraman and M.N.S. Swamy, *Graphs : Theory and Algorithms*. Wiley-Interscience, 1992.
- [36] N. Vaidya and S. Hameed, "Scheduling Data Broadcast in Asymmetric Communication Environments," *ACM/ Baltzer Wireless Networks*, vol. 5, no. 3, pp.171-182,1999.
- [37] K.H. Yeung and T.S. Yum, "Selective Broadcast Data Distribution Systems," *IEEE Trans. Computers*, vol. 46, no. 1, pp. 100-104, Jan.1997.