# 以 RFM 資料分析為基礎建立檔案分群機制 提升雲端運算的服務效能


| 王順生 | 嚴國慶* | 王淑卿* | 陳聖中 |
|---|---|---|---|
| 朝陽科技大學 | 朝陽科技大學 | 朝陽科技大學 | 朝陽科技大學 |
| 工業工程與管理系副教授 | 企業管理系教授 | 資訊管理系教授 | 資訊管理系研究生 |
| sswang@cyut.edu.tw | kqyan@cyut.edu.tw | scwang@cyut.edu.tw | s10014609@cyut.edu.tw |

*: 聯絡人



## 摘要

隨著時代的演進，電腦科技技術的成長，越來越多人使用雲端運算中的服務。而在使用者使用雲端運算服務的同時，使用者也會將其檔案儲存在雲端儲存(Cloud Storage)中。然而在雲端運算環境中，提供使用者高且可用的服務，是提供雲端運算服務時必須考慮的重要因子之一。為達到此目的，本研究分析檔案被使用的特性，並透過檔案的分群，將相同類型的資源集中，使具有相似屬性的檔案及服務對應到相似屬性的資源中。在本研究所提出的分群機制中，首先依檔案的大小透過 File 或 Block Level 進行檔案的儲存，並依檔案的熱門程度因子進行屬性分析，最後以改良過的 K-means 分群演算法進行檔案的分群，使得相似的檔案存放在相同的群組中。使用者在存取雲端儲存中的檔案時，則因在同一群組中進行存取，所以可以得到較快的回應，藉此得以提升使用者在雲端運算中使用檔案的效能。



**關鍵詞**：雲端運算、雲端儲存、分散式檔案系統、熱門程度、分群



## Abstract

With the evolution of information technology, more and more users are using cloud-computing services. When users use the cloud computing services, their files will be stored in the Cloud Storage. However, the availability of services in the cloud-computing environment is one of the important factors that must be considered when providing cloud-computing services. To achieve this goal, the characteristics of the file are analyzed in this study. However, through the clustering of files, the same type of resources is concentrated, and then the resources can be corresponded to the similar attributes of files and services. According to the file size, in the proposed clustering mechanism, the files are stored by file or block level. Then, the attributes of file are analyzed by the factor of file popularity. Finally, the modified K-means clustering algorithm is used to cluster files and the similar files are stored in the same cluster. When users access the similar files in same cluster, users can get a faster response of file operation, and the performance of cloud computing can be improved.



**Keywords**: Cloud Computing, Cloud Storage, Distributed File System, Popularity, Clustering


## 1. 前言

由於網際網路的蓬勃發展、電腦硬體效能的快速成長與網路頻寬的增加，現今網際網路的使用者對於網際網路的服務越來越依賴，因此越來越多的資源投入雲端運算(Cloud Computing)中，也使得雲端運算相關的應用蓬勃發展[14]。其中，雲端運算的運算能力是由分散式運算方式提供，儲存能力則是建構在分散式架構下的分散式檔案系統(Distributed File System)。

由於目前在雲端運算環境中，大都是使用分散式的檔案系統儲存資料。而分散式檔案系統，由邏輯的角度分析則可視為是一個階層式的檔案系統[13]。當系統依據檔案的階層進行複製後，分散式檔案系統可以達到檔案的可靠性與使用效能[7]。而在進行大量資料處理時，增加檔案的複製備份數目，不只可以增加檔案的可靠性，也可以加快存取時間[1,20]。換言之，由於分散式檔案系統可以透過複製備份來提高系統的可靠度，但過多的副本檔案則將會浪費儲存空間。

除此之外，如果檔案在進行儲存前未能事先規劃存放位置，則可能會出現所有的檔案都

存放在相同的目錄下，並且無法充分利用雲端運算環境中的各種資源，因此造成日後使用者存取檔案時的問題。

由於過去在分散式檔案的架構下，大多是使用 Block Level(區塊層級)的方式進行儲存[24]。但在處理小於一個區塊大小的檔案時，若使用 Block Level 的方式進行儲存，不僅會浪費許多多餘的空間，並且在進行檔案存取時所需的時間也將等同於存取一個 Block 一樣。在過去相關的研究中，有學者將各個小檔案壓縮在一個區塊中進行儲存[25-27]，但也會因為藉由索引進行檔案的儲存，導致降低存取的速度。因此在本研究中，將小於一個 Block 大小的檔案歸類為小檔案，並且使用 File Level(檔案層級)進行儲存。

當使用者在雲端運算環境中使用其儲存於雲儲存(Cloud Storage)的檔案時，如何確保該檔案的高可用性，是在雲端運算環境中進行檔案維護時必須探討的重要議題之一。因此在本研究中，將探討在雲端運算環境中檔案儲存的方式，以及透過檔案屬性的分析，將檔案以分群的方式儲存。藉由達到相似檔案之群聚，使得使用者在相似檔案的檢索上可以達到更高的效率，使雲端運算環境成為一個友善的平台。

本文第 2 節為文獻探討，將說明雲端運算、分散式檔案系統中檔案的儲存方式及分群方法的相關研究；第 3 節將說明本研究所提出的方法；第 4 節為方法的實例說明；最後一節為結論與未來研究。

## 2. 文獻探討

在本章節中將說明雲端運算、分散式檔案系統中檔案的儲存方式以及分群方法的相關研究。

### 2.1 雲端運算

雲端運算不是一項新興技術，而是一種過去就有的分散式運算形式，雲端運算與代表多台電腦同時進行運算與叢集運算(Cluster Computing)的概念類似，皆是指透過整合大量電腦的運算資源來處理運算需求[7,18]。

「雲」即為網際網路；「端」則是指使用者端(Client)或泛指使用者運用網路來完成服務[1]。其最基本的概念是透過網際網路將龐大的運算處理程序(Process)，自動分拆成無數個較小的子程序(Sub-Process)，再交由多部伺服器(Multi-Server)所組成的龐大系統，透過搜尋與運算分析之後，再將處理結果回傳給使用者端[18]。透過這項技術，網路服務提供者可以在數秒之內，處理數以千萬計甚至億計的資訊，達到和「超級電腦」同樣強大效能的網路服務[16]。雲端運算是繼 1980 年大型電腦到使用者端-伺服器的大轉變之後的又一種巨變。使用者不再需要了解「雲端」中基礎設施的細節，不必具有相對應的專業知識，也無需直接進行控制。雲端運算描述了一種基於網際網路及資訊技術所提供的新型服務、使用和交付模式，通常涉及透過網際網路來提供動態易擴充功能，而且經常是虛擬化的資源[22]。典型的雲端運算服務提供者往往提供通用的網路應用服務，使用者可以透過瀏覽器等軟體或者其他 Web 服務來存取儲存在伺服器上的軟體和資料[9,19]。除此之外，雲端運算的關鍵要素，還包括個性化的使用者體驗。整體而言，雲端運算讓網路上不同的電腦同時提供使用者端進行所需的服務，大幅增進網路服務的處理速度。

雲端運算包括 3 個層次的服務，軟體即服務(Software as a Service，SaaS)、平台即服務(Platform as a Service，PaaS)和基礎設施即服務(Infrastructure as a Service，IaaS)[9]。對應的產業三級分層則為：雲端軟體、雲端平台和雲端設備。上層分級為雲端軟體(SaaS)，提供使用者可以透過瀏覽器存取雲端運算的服務。中層分級為雲端平台(PaaS)，打造程式開發平台與作業系統平台，除了讓開發人員可以透過網路撰寫程式與服務，雲端運算需求使用者也可透過開發人員掛載的相關應用程式得到所要的服務。下層分級為雲端設備(IaaS)，是將 IT 系統、資料庫等基礎設備內部功能做整合。

整體而言，雲端運算的優點包括：(1)強大的運算能力；(2)高容錯能力；(3)高可靠性；(4)可行動化；及(5)降低運算成本等[21]。因此，透過雲端運算可以提升現行系統的效能。在目前的生活中，已經有許多相關的應用透過雲端運算，服務提供給消費者，其中包括 Gmail 與 Youtube 等，讓使用者只要可以與網路連結就能使用雲端服務而不被使用平台所限制。

### 2.2 儲存方式

在分散式檔案系統中進行檔案儲存的儲存方式可以分為兩類[8,10]，分別是 Block Level(區塊層級)以及 File Level(檔案層級)。而在進行檔案實體儲存時，相關的技術也從

DAS(Direct Attached Store；直接連接儲存)演進為 SAN(Storage Attachment Network；儲存區域網路)到 NAS(Network Attached Storage；網路附接儲存)至 iSCSI(inter SCSI)，隨著以上幾種實體儲存方式的提出與使用，其所儲存的格式還是採用 File Level 或 Block Level 兩種。因此，在本小節中將分別說明 File Level 或 Block Level 的儲存方式。

**(1) File Level**

在使用一般個人電腦時，檔案的儲存是以 File Level 為主，而在 File Level 下，使用者是使用單一的目錄進行儲存。同時，在進行檔案實體儲存時，則是採用 DAS 技術存取檔案。由於是使用 File Level 儲存檔案，因此讀取時必須等到整份檔案讀取完畢才能回應給使用者。其主要是因為在使用 DAS 技術時，受限於網路傳輸的速度，使得讀取檔案的速度最快則是與網路速度相同。由過去相關的研究可知，當檔案的大小小於 Block Level 中的 Block 大小時，以 File Level 進行檔案的儲存，其儲存速度會快於 Block Level，且其優點為直接存取、架構單純、更為經濟等[10]。

**(2) Block Level**

在分散式的檔案架構下，大多數的儲存架構都是以 Block Level 建置。在 Block Level 中，由於檔案是以 Block Level 儲存，因此進行檔案存取時，可以藉由不同的 Block 同時回應給使用者。由於以 Block Level 建置的儲存架構，可以透過多個 Block 支援同一份檔案的傳輸，因此不會受限於單一網路傳輸速度。換言之，因為 Block Level 所建置的儲存架構具有高效率、容量大與擴充容易等優點，因此使用 Block Level 進行檔案儲存是較有效率。

## 2.3 RFM 因子分析

在過去相關的顧客關係管理之研究領域中，已知有許多方法可以幫助企業瞭解其顧客的相關資訊[5]，運用這些相關資訊進而了解顧客的消費行為並制訂滿足其顧客需求的行銷策略。其中，Kahan 主張 RFM(Recently、Frequency、Monetary)資料分析技術由於能提供企業每個顧客的交易資訊，因此是比一般認知分析(Cognitive Analysis)更為有用的行為分析(Behavioral Analysis)技術[12,15]。並且在許多藉以瞭解顧客相關資訊的方法中，以 RFM 資料分析技術最為被廣泛運用[15]。

RFM 資料分析技術是由 Arthur Hughes 在 1994 年所定義的，所謂的 RFM 資料分析技術分別為最近購買時間(Recently；R)、購買頻率(Frequency；F)與購買金額(Monetary；M)的縮寫。其中，R 是指顧客最近的購買時間，即所謂顧客最近一次購買的時間與現在時間的距離天數，用來衡量顧客再次購買的可能性。當購買時間距離愈近，則表示該顧客再次購買程度愈高；若最近購買日期距離愈遠，則表示該顧客購買意願降低、或購買行為改變、或是因其他因素而導致至他處消費。F 是指在某段期間內購買該企業產品的總次數，此期間可定義為一個月、一季、或是任何可衡量的時間長度，可用來衡量顧客在購買行為中與企業的互動程度。當購買頻率愈高，則表示顧客的熱衷程度愈高。M 是指在某段期間內購買該企業產品的總金額，可做為用來評價顧客對該企業的貢獻度及顧客價值。當顧客的購買金額愈高時，則表示顧客的價值較高。RFM 資料分析技術主要是在分析及衡量顧客的消費行為，透過顧客過去的歷史交易資訊進行顧客的區隔，以作為衡量顧客的忠誠度與貢獻度之依據。

Arthur Hughes 認為 RFM 資料分析技術在衡量一個顧客的重要性程度是一致的[2,3]，透過顧客最近一次消費可以呈現顧客最近才來購買商品或者服務，很有可能近期再來購買。因為吸引最近才來購買的顧客，比吸引很久沒有來購買的顧客來的容易。消費頻率是顧客在同一期間內購買的次數，也可以說最常購買的顧客，也就是對商品或者服務是最滿意的，因此相對的消費頻率高也就是忠誠度高。而在消費金額則是透過顧客在同一期間內消費的總金額，透過這三個要素分析消費者的價值。

傳統使用於顧客關係管理的 RFM 資料分析技術，可以以顧客過去的歷史交易資訊進行顧客的區隔，透過衡量顧客的忠誠度與貢獻度後，提供深度經營之參考依據。而由於使用雲端運算相關服務的使用者越來越多，當使用者將其個人的檔案儲存在雲端運算環境時，雲端運算系統的可使用性就成為提供雲端服務時必須考慮的重要因素。而為了維護雲端運算環境中檔案的可使用性，在雲端運算環境中常藉由複製備份的方式，以增加雲端運算中資料檔案的可使用性[11,20]。

除此之外，在雲端運算中提供快速的服務

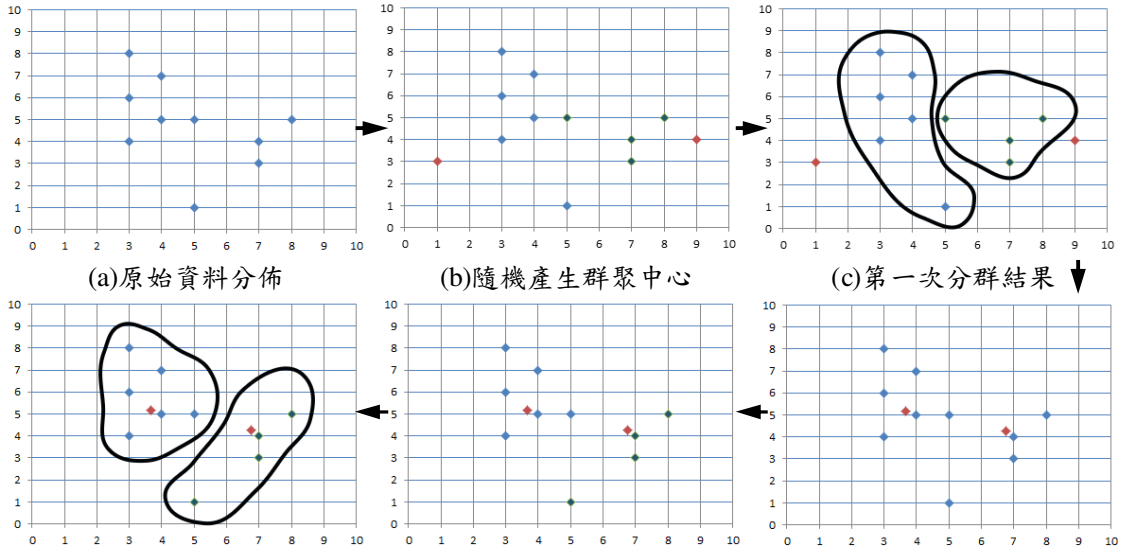
回應機制，亦是在雲端儲存環境中必須探討的重要議題之一。而由於在雲端儲存環境中，檔案被存取的行為資訊與顧客的消費行為之交易資訊有類似的特性，因此本研究將改良傳統之 RFM 因子，並運用於檔案儲存時儲存資料的價值分析。

## 2.4 K-mean 分群演算法

群組分析在許多應用領域中是一種基礎工具，其主要探討的是如何將資料或物件予以分群或分類的方法[6]。其中資料或物件的呈現方式最常用的就是一組特徵向量，而它的主要目的就是將這些多維的特徵向量分成若干個群組，而屬於同一群組中的向量與其它群組中的向量相較時，則同一群組中的向量彼此會較為類似。因此透過分群的技術，將這些看似毫無規則可循的向量，依其特性分成好幾個群組，使每個群組中的元素有最大的相似性，而和其它群組的元素有最大的不相似性[17]。

現有的分群演算法種類繁多，各種方法皆有其優缺點以及適用的範圍。然而分群演算法的成功與否，除了群組的數目及其中心位置能否得知外，資料的幾何特性亦屬關鍵。多數的分群演算法大都採用歐基里德距離(Euclidean Distance)來量測資料間的相似程度[4]。在眾多的分群演算法中，比較有名也較常被使用的就是 K-means 分群演算法[23]。K-means 是 MacQueen 於 1967 年所提出的分群演算法，必須事前設定群集的數量 K，然後找尋所設定公式的極大值，以達到分群的最佳化之目的。

K-means 演算法是屬於切割式的分群演算法(Partition Clustering Algorithms)，其主要的精神是以重心點或中心點(Mean)為基礎的方式，將資料群體進行分群。K-means 是採重心基礎的切割式分群演算法[23]，因為各群體的代表點不一定是群組中的一點，所以可以在多數的情況下找到最佳群組。圖 1 是個 K-means 演算法的範例，其中圖 1 紅點所指出的就是各群組族群的中心點，在整個計算的過程中，左右兩個群組的資料，因對中心點距離的不同，而有所修正。



(a)原始資料分佈　(b)隨機產生群聚中心　(c)第一次分群結果

(f)分群異動後的新群組　(e)計算中心距離　(d)分群群組中心

**圖 1 K-means 演算法的群聚過程圖[23]**

K-means 演算法的執行過程如下[23]：

Step 1. 決定 K 個分類群組數，並決定每個群組的初始中心點 $Z_1(1), Z_2(1), \ldots, Z_K(1)$

Step 2. 逐一比較每個點，將每個點歸類到所屬的群組，其比較方式為：

$$\text{If } \|X-Z_i(N)\| < \|X-Z_j(N)\| \text{ then } X \in S_i(N)$$
$$\text{for } i, j = 1, 2, \ldots, k$$

Step 3. 每個群組的中心點座標重新計算，其計算公式為

$$Z_i(N+1) = 1/M \sum_{X \in S_i(N)} X, i = 1,2,\ldots,k$$

Step 4. 舊中心點座標和新中心點座標比較，當兩者相同時整個過程就停止，若兩個中心點座標有一定程度上的差異時，就重新進行 Step 2。不過當兩者的差距越來

越小呈收斂情況時，也可以視收斂程度來設定停止條件。

符號定義如下：

$Z_i(N)$：代表第 $i$ 個分類群組，第 $N$ 次中心點

$S_i(N)$：代表第 $i$ 個分類群組的群集

$X$：具有屬性值的資料點

$M$：$S_i(N)$群組所屬 $X$ 的個數

亦即，K-means 透過四個步驟進行：(1)隨機指派群集中心；(2)產生初始群集；(3)產生新的質量中心；(4)變動群集邊界。反覆以上四個步驟直到分群結果不變，達到分群結果。

# 3. 以 RFM 資料分析為基礎建立檔案分群機制

為提升雲端運算中儲存檔案的使用效能，本研究提出「以 RFM 資料分析為基礎建立檔案分群機制(K-means based on RFM data analysis；RFM K-means)」，藉此提升雲端運算的服務效能。RFM K-means 機制中，首先依檔案的大小分別以 File Level 或 Block Level 儲存檔案，接著以本研究所提出的熱門分析 RFM 因子對檔案進行熱門程度的分析，並以改良過的 K-means 分群演算法將檔案進行分群，使得特性或屬性相似的檔案存放在相同的群組之中，提升檔案在雲端運算中的使用效能。

## 3.1 儲存方式

由於過去在分散式的架構下大多是使用 Block Level 儲存[4,24]，但在處理小於一個區塊大小的檔案時，則會浪費多餘的空間，並且存取速度與一個區塊一樣。有研究將各個小檔案壓縮在一個區塊中進行儲存[25-27]，但也會因為藉由索引進行儲存，導致降低存取速度。

因此在本研究中，在進行儲存檔案時，RFM K-means 依據檔案的大小分別採用不同的儲存方式，大檔案使用區塊層級(Block Level)，小檔案則使用檔案層級(File Level)。換言之，RFM K-means 將小於一個 Block 大小的檔案歸類為小檔案並且使用 File Level 進行儲存。在更新傳播方式中，RFM K-means 則使用兩種更新的方式，包括多點循序傳播與單點同時傳播的方式，藉以充分使用 Block Level 的平行處裡特色，與 File Level 的直接存取處理的特色。

在雲端運算環境中，由於統整了許多異質的設備，因此在檔案進入到雲端運算環境時 RFM K-means 首先判斷檔案大小是否大於一個 Block 的大小，若檔案大於一個 Block，則使用 Block Level 進行檔案儲存；若檔案小於或等於一個 Block 的大小，則使用 File Level 進行檔案儲存。RFM K-means 透過兩種檔案的儲存方式，以使檔案在雲端運算中的使用達到更有效率。

## 3.2 RFM 因子分析

在本研究所提出的 RFM K-means 中，以熱門度 RFM 分析因子對檔案進行熱門程度的分析。本研究所提出的熱門度 RFM 分析因子，R 為 Recently 是檔案最近一次被存取的時間；F 為 Frequency 是檔案被存取的頻率；M 為 aMounts 是檔案存取所需的時間長度，其與檔案的大小成正相關。

透過熱門度 RFM 分析因子對檔案進行分析後，再將分析後的資料儲存至 Metadata Server 中。當檔案透過 RFM 分析後，將可以分析出每一個檔案的熱門程度。而針對每一個檔案的 R、F、與 M 都會有一個介於 0 至 1 的熱門程度，且此熱門程度將會影響檔案的更新速度。當檔案透過 RFM 分析後，將可以分析出各個檔案的價值，進而再透過價值分群達到資料分群的效果。

## 3.3 資料分群

由於在雲端運算環境中，其所提供的資源型態不同。因此若能將相同型態的資源集結一起並且提供相對應的服務，則不僅可以對使用者的服務需求達到較好的服務回應結果，更可使雲端資源的使用達較佳的狀態。

分群是將資料分成不同的群組，將具有相同傾向與樣式(Pattern)的資料加以群組的方法，其主要的目的是讓性質相似的資料群聚在一起。對於需要將雲儲存上的檔案劃分成多個不同群組的情況，正好適合資料探勘中的分群分析，因此本研究以分群分析中的 K-means 演算法，對檔案依其熱門程度屬性進行區隔，先區隔出主要的幾類檔案群組，再依檔案的屬性執行儲存，藉此提升雲端運算的服務效能。

RFM K-means 分群的應用架構如圖 2 所示，首先將檔案被儲存或擷取的歷史資訊轉化並儲存至資料庫中，再將雲端使用者的使用紀錄導入分群演算法中。接著，依雲端使用者的

使用紀錄之屬性特質，區隔成若干不同屬性的檔案區隔群組。

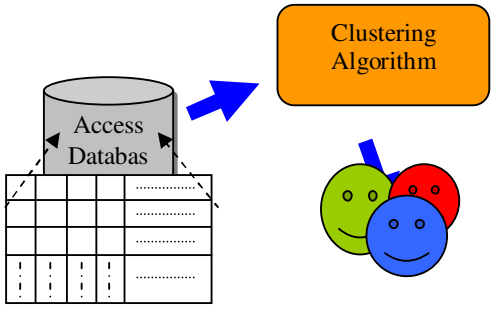



**圖 2 RFM K-means 分群分析的應用過程**

在檔案進入雲端運算環境中進行儲存時，RFM K-means 首先進行資料的前置處理。RFM K-means 將檔案依其大小，分別以 Block/File Level 進行區隔儲存。接著，透過 Metadata Server 記錄檔案被存取的資訊，並進行 RFM 熱門因子分析。RFM K-means 透過 RFM 熱門因子分析，將檔案依其特性分群到適當的群組存放。

RFM K-means 資料分群是改良 K-means 方法而來，將透過以下四個步驟進行，如圖 3 所示：

[步驟1] 收集資料。
[步驟2] RFM 因子分析。
[步驟3] 進行分群。
[步驟4] 紀錄分群結果，並重新設定群間距。

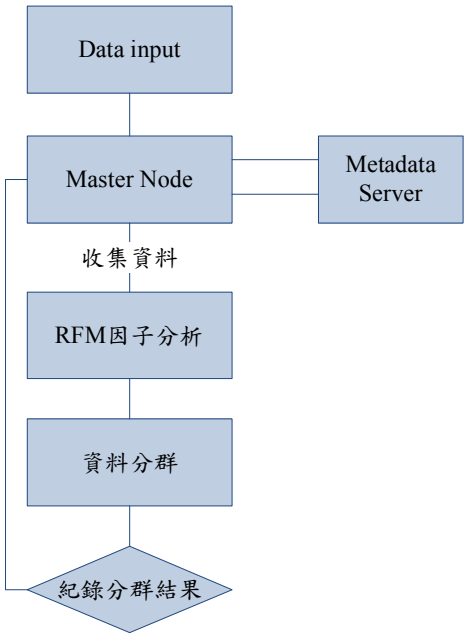



**圖 3 RFM K-means 分群流程圖**

使用 RFM K-means，首先在[步驟 1]收集前一時間階段檔案被使用的所有歷史資料；[步驟 2]則將檔案被使用的歷史資料正規化為 RFM 熱門因子並進行分析；[步驟 3]將檔案依 RFM 熱門因子分析的結果，分群到相符合的 RFM 群組中；[步驟 4]將檔案分群結果紀錄，並檢測各個群組之間距，若該群組負擔過大，則從新調整 RFM 熱門因子分群之間距，待下一時間階段時再將檔案重新分配到其他群組。

## 4. 實例說明

為說明本研究所提出的方法，確實可以使檔案依據 RFM 資料分析為基礎進行檔案分群。因此，本節中將以實例說明。

假設在雲端運算環境中，將 RFM 群組設定為 8 等分(2*2*2)，則當檔案進入雲端運算環境時，將先依預設 RFM 熱門因子分析之值分配至該群組。此例中假設雲端運算環境中已有 8 組群組如表 1 所示。其中，表中的 RFM 值表示為基底群組，檔案之 RFM 值則在 0 到 1 之間。檔案將依群間距分配至靠近其 RFM 值之群組中，RFM K-means 在初始狀態則依 0.5 進行分配。

**表 1 群組分配**

| | R | F | M |
|---|---|---|---|
| 群組 1 | =<0.5 | =<0.5 | =<0.5 |
| 群組 2 | >0.5 | =<0.5 | =<0.5 |
| 群組 3 | =<0.5 | >0.5 | =<0.5 |
| 群組 4 | >0.5 | >0.5 | =<0.5 |
| 群組 5 | =<0.5 | =<0.5 | >0.5 |
| 群組 6 | >0.5 | =<0.5 | >0.5 |
| 群組 7 | =<0.5 | >0.5 | >0.5 |
| 群組 8 | >0.5 | >0.5 | >0.5 |

假設有一個檔案 A 將被新增至雲端運算環境中，首先判斷檔案 A 的大小應該儲存為 File 或 Block Level 再進行群組選擇。由於沒有對檔案 A 過去的存取資料，因此將檔案 A 先存放至群組 1 中。RFM K-means 將隨著一個時間階段的進行，再進行檔案分群的修正。[步驟 1]先進行收集檔案被使用的所有歷史資料，[步驟 2]將過去檔案被使用的歷史資料進行 RFM 熱門因子分析，並且將資料正規化為 0 至 1 之間的數值。由於經 RFM 熱門因子分析後，檔

案 A 的 RFM 因子分析數值如表 2 所示，所以經執行[步驟 3]，符合 R=0.7、F=0.4、且 M=0.6 的群組為 6，因此將檔案 A 分配到群組 6。

**表 2 RFM 因子分析數值**

| | R | F | M |
|---|---|---|---|
| 檔案 A | 0.7 | 0.4 | 0.6 |

最後執行[步驟 4]，檢測群組 1 至 8 各個群組的群間情況。由於本實例透過 RFM 熱門因子將檔案分配至 8 個群組，因此計算雲端環境中所有 RFM 因子分配情況。以檔案最近存取時間(R)為例，將依照存取時間的分配情況，若前 50%數量檔案的 R 都聚集在 0.7~1 之間，則下一個時間階段 R 會依照 0.7 分配，高於 0.7 的 R 之檔案會搬移到本實例中的群組 2、4、6、或 8 之一，接著再依照 F 與 M 之間的價值存入。使得雲端運算環境各群組的檔案儲存量，得以，獲得負載平衡。

## 5.結論與未來研究

本研究所提出的 RFM K-mean，首先依檔案的大小分別以 File Level 或 Block Level 儲存檔案，接著以本研究所提出的熱門分析 RFM 因子對檔案進行熱門程度的分析，並以改良過的 K-means 分群演算法將檔案進行分群。由於 RFM K-means 是透過熱門分析 RFM 因子的分析，因此可以達到資料分群與負載平衡，使得使用者在使用相似的檔案時可以達到較快的回應。由於在本研究中以 RFM 熱門因子為基礎進行檔案的分析，因此在群組分配上的速度將可大幅增加。而 RFM K-means 在分群中的分群次數則以多個時間階段取代，因此雲端環境建立越久，則將使得特性或屬性相似的檔案存放在相同的群組之中，因此 RFM K-means 分群可以分得越精確，藉以提升檔案在雲端運算中的使用效能。

然而在本研究中，並未考慮群組間可容忍的負載量，而是依據 RFM 熱門因子分析的結果將檔案分配到各個群組中。因此，在雲端運算環境提供服務需求時，若有系統狀態呈現不忙碌的空閒時，將無法達到較快的回應時間。

## 致謝

這篇論文是國科會計畫(NSC101-2221-E-324-032 與 101-2221-E-324-034)研究成果的一部份，在此我們感謝國科會經費支持這個計畫的研究。

## 參考文獻

[1] 中華民國資訊軟體協會，***雲端運算 Cloud Computing 的概念與應用***，e 化部落，2010。

[2] 林陽助，***服務行銷***，鼎茂出版社，2003。

[3] 陳彤生，"運用改良 RFM 提升行銷效益的實證研究，"***第七屆人工智慧與應用研討會論文集***，2002。

[4] 陳惠良，***顧客關係管理於電子商務應用之互動與相關係研究***，碩士論文，國立台北科技大學生產系統工程與管理研究所，臺北，2001。

[5] 連惟謙，***應用資料分析技術進行顧客流失與顧客價值之研究***，碩士論文，中原大學資訊管理研究所，**2003**。

[6] 趙景明，***移動式網格之分散式資料分群技術***，東吳大學資訊科學系碩士論文，2006。

[7] Buyya, R. Yeo, C.S. and Venugopal, S., "Market-oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities," ***Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications***, pp. 5-13, 2009.

[8] Cai, B., Xie, C. and G. Zhu, "EDRFS: An Effective Distributed Replication File System for Small-File and Data-Intensive Application," ***2007 2nd International Conference on Communication Systems Software and Middleware***, pp. 1-7, 2007.

[9] Grossman, R.L., "The Case for Cloud Computing," ***IT Professional***, Vol. 11, No. 2, pp. 23-27, 2009.

[10] Gwertzman, J. and Seltzer, M., "The Case for Geographical Push-caching" ***Proceedings of the 5th Annual Workshop on Hot Operating Systems***, pp. 51-55, 1995.

[11] He, Q., Li, Z. and Zhang, X., "Data Deduplication Techniques" ***Proceedings of the 2010 International Conference on Future Information Technology and Management Engineering*** (FITME), pp. 430-433, 2010.

[12] Hughes, A.M., "Boosting Response with RFM" ***Marketing Tools 5***, pp. 4-10, 1996.

[13] Islam, M.A., Vrbsky, S.V. and Hoque, M.A.,

"Performance Analysis of a Tree-Based Consistency Approach for Cloud Databases," ***Proceedings of the 2012 International Conference on Computing, Networking and Communications*** *(ICNC)*, pp. 39-44, 2012.

[14] Mathur, P. and Nishchal, N., "Cloud Computing New Challenge to the Entire Computer Industry," ***Parallel Distributed and Grid Computing*** *(PDGC)*, pp. 232-228, 2010.

[15] Kahan, R., "Using Database Marketing Techniques to Enhance Your One-to-One Marketing Initiatives," ***Journal of Consumer Marketing***, Vol. 15, pp. 491-493, 1998.

[16] Luo, Y., "Network I/O Virtualization for Cloud Computing," ***IT Professional***, Vol. 12, No. 5, pp. 36-41, 2010.

[17] MacQueen, J., "Some Methods for Classification and Analysis of Multivariate Observations" ***Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability***, No. 2, pp. 281-286, 1965.

[18] Rimal, B.P., Choi, E. and Lumb, I., "A Taxonomy and Survey of Cloud Computing," ***Proceedings of The NCM2009 5th International Joint Conference on INC, IMS and IDC***, pp. 44-51, 2009.

[19] Ranganathan, K. and Foster, I., "Identifying Dynamic Replication Strategies for a High-Performance Data Grid," ***Proceedings of the International Workshop on Grid Computing***, pp. 75-86, 20001.

[20] Potlog, A.D., Xhafa, F., Pop, F. and Cristea, V., "Evaluation of Optimistic Replication Techniques for Dynamic files in P2P System," ***Proceedings of the 2011 International Conference on P2P, Grid Cloud and Internet Computing*** (3PGCIC), pp. 259-265, 2011.

[21] Vouk, M.A., "Cloud Computing- Issues, Research and Implementations," ***Information Technology Interfaces***, pp. 31-40, 2008.

[22] Wang, G. and Ng, T.S.E., "The Impact of Virtualization on Network Performance of Amazon EC2 Data Center," ***Proceedings of the 29th IEEE Conference on Computer Communications*** (IEEE INFOCOM), pp. 1-9, 2010.

[23] Wu, H.H., Chang, E.C. and Lo, C.F, "Applying RFM Model and K-means Method in Customer Value Analysis of an outfitter" ***Global Perspective for Competitive Enterprise, Economy and Ecology Advanced Concurrent Engineering***, pp.665-672, 2009.

[24] The Hadoop Distributed File System, http://hadoop.apache.org/hdfs/，擷取日期 2011 年 10 月 18 日。

[25] The Sequence File Layout, http://www.hadoop.tw/2008/12/hadoop uncompressed -sequencefi.html，擷取日期 2011 年 10 月 18 日。

[26] The MapFile File, http://hadoop.apache.org/common/docs/current/api/org/apache/hadoop /io/MapFile.html，擷取日期 2011 年 10 月 18 日。

[27] The Hadoop Archive, http://hadoop.apache.org/core/docs/r0.20.0/hadoop_archives.html，擷取日期 2011 年 10 月 18 日。