

瞭解 DB2 9 自主運算功能的優點

瞭解 DB2 9 的革命性功能如何能夠改變您的業務

級別：中級

雷曼兄弟公司 (Lehman Brothers, Inc) DB2 資料庫專家 [Mohankumar Saraswatipura \(mohankumarsp@gmail.com\)](#)

IBM 軟體工程師 [Swetha Prasad \(swetha.prasad@in.ibm.com\)](#)

2007 年 11 月 29 日

自調式記憶體管理 (STMM) 是 IBM® DB2® 9 最先推出的一種革命性記憶體微調功能。STMM 可透過為大部分的記憶體配置參數 (包括緩衝池、套件快取、鎖定記憶體、排序資料堆以及總計資料庫共用記憶體) 自動設定最佳的數值，來減輕記憶體配置的工作。啓用 STMM 後，記憶體微調器就會在不同的記憶體耗用單位之間動態分配可用的記憶體。本文將解釋 STMM 的功能，並教您啓用該功能的方法，同時也會討論 STMM 如何能夠為您的商業環境帶來實質的利益。

前言

為了因應現今的商業挑戰，企業不得不透過最佳化資料庫，盡可能取得最佳效能。無論 DBA 的技術層級為何，資料庫引擎都必須具備高度可調性並可輕鬆配置，才能取得最佳效能。為了面對這些挑戰，DB2 9 具備了一組可適當準確微調配置參數的自主功能；自調式記憶體管理程式 (STMM) 就是可提供這類功能的管理程式。本文將討論目前的產業挑戰，並說明 STMM 功能如何協助您克服這些挑戰。

目前的產業挑戰

為了滿足目前產業的需求資料庫技術不斷地成長，在資料庫中內嵌更精準的自主運算功能，就成為不可或缺的一環。現代的資料庫引擎非常複雜，雖然複雜性提供了許多豐富的功能以及更大的彈性，但也可能造成配置上的挑戰。

這些商業挑戰中，主要影響資料庫微調的部分如下：

- 由於今日對資料庫業務的需求愈來愈龐大，因此資料庫引擎的複雜度也必須快速增加，才能滿足這些需求。
- 隨著資料庫引擎的複雜度不斷增加，與微調資料庫相關的參數數目也越來越多。
- 在大多數情況下，記憶體參數都是互相連結的，變更某個參數可能會影響另一個相關的參數，進而影響資料庫的效能。記憶體配置對於資料庫管理系統 (DBMS) 的效能會有非常重大的影響。
- 隨著資料庫引擎的複雜度不斷增加，產業需要經驗豐富的 DBA 才能管理資料庫伺服器。
- 從嘗試和錯誤中所得到的經驗來微調記憶體，通常是既困難又費時的。

- 在許多企業實務中，資料庫伺服器的工作量變化極大。在這種情況下，對於任何熟練的 DBA 來說，資料庫微調是一項困難的工作，而靜態配置並不一定是最好的。

DB2 9 如何克服難題

自調式記憶體管理 (STMM)

如果可以根據工作量的需求來適當微調記憶體，對於系統效能就會有幫助。若要讓效能發揮到極致，就必須隨著工作量變動來配置記憶體。

使用 DB2 9 中全新的自調式記憶體管理 (STMM) 功能，來微調 DB2 資料庫記憶體 (DATABASE_MEMORY) 和緩衝池以達到最佳效能，就可以毫不費力。這項功能可以自動配置大部分的記憶體設定，並在執行時期加以調整，以最佳化效能。更棒的是，STMM 在根據工作量變化微調記憶體參數的過程中，完全不需要 DBA 的介入。

STMM 如何運作？

DB2 9 中的 STMM 可以簡化在資料庫啟動和執行時期期間設定數個記憶體配置參數的工作。STMM 有智慧型的控制機制，可以記錄不同工作量的特性、每個記憶體參數所耗用的記憶體，以及資料庫中許多共用記憶體資源的需求。它會根據需求，動態調整資源的記憶體用量。

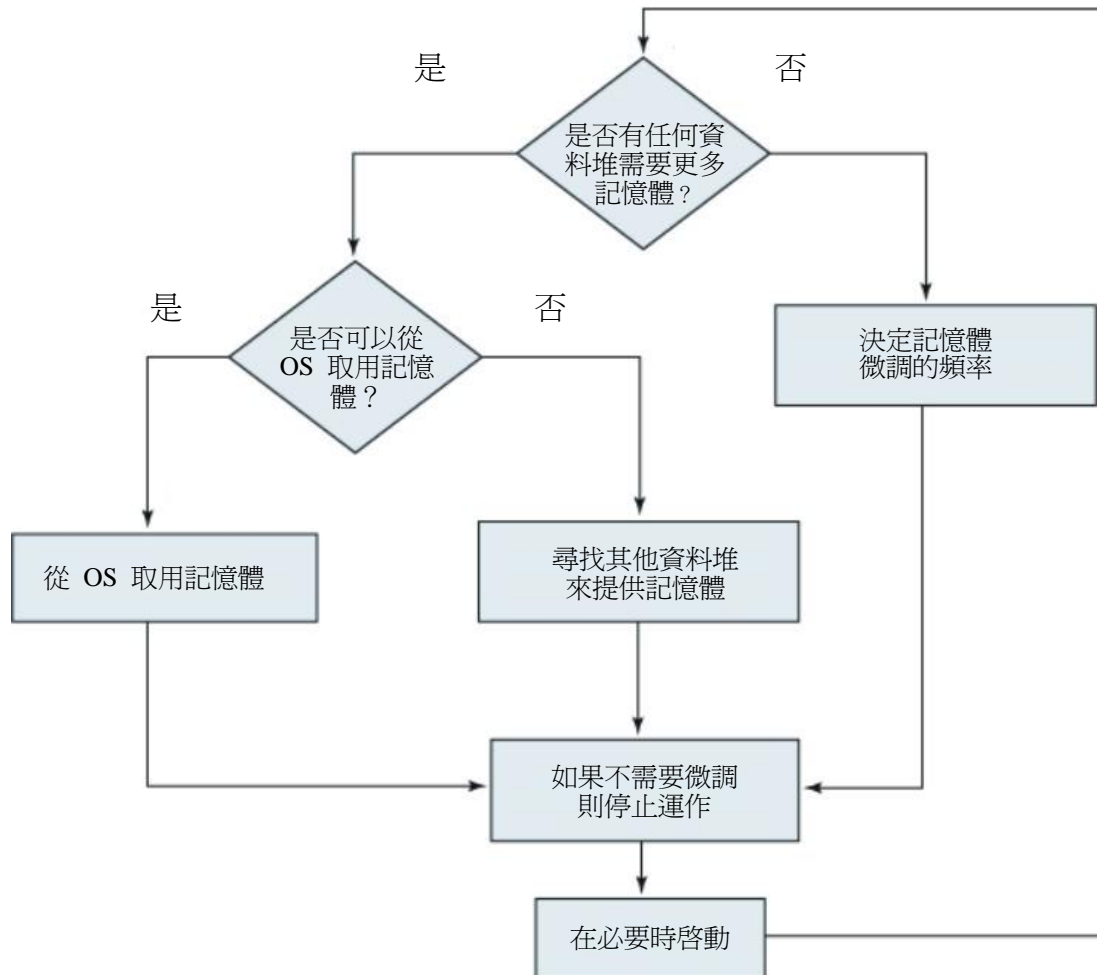
只有在將系統設定為可微調資料庫記憶體的總耗用量時，STMM 才會不斷地監控系統，以利用所有可用的作業系統記憶體。

它會反覆運作，為下列記憶體參數配置最佳記憶體：排序 (sheaphres_shr 和 sortheap)、套件快取 (pckcachesz)、鎖定清單 (locklist 和 maxlocks)，以及緩衝池。STMM 採用反覆式的方法，試圖避免系統的不穩定狀態。

圖 1 顯示 STMM 的工作流程，該工作流程包含下列各項：

1. 啟動微調器 (1) 並查看記憶體參數是否需要更多記憶體 (2)。
2. 如果答案是「是」，它就會檢查可用的作業系統記憶體數量 (如果將 DATABASE_MEMORY 設定為自動的話) (5)。如果作業系統層次有足夠的實體記憶體可用，它就會將這些記憶體的一部分，分配給有需要的資料堆 (6)。如果沒有可用的記憶體，它就會尋找其他可以提供記憶體的資料堆 (7)。
3. 這個程序會不斷地進行，直到沒有記憶體可以移動為止。
4. 然後，微調器便會停止運作，直到有必要時才會再度啟動 (4)。啟動的頻率會依工作量的特性來決定 (3)。

圖 1. STMM 的工作流程



STMM 的作業模式

STMM 可以四種不同的模式運作

- **模式 1**：當 DATABASE_MEMORY = AUTOMATIC 時
在此情況下，必要時會從作業系統取得並歸還所需的記憶體。隨著時間的經過，DB2 使用的記憶體總量會不斷地增加，而且只會受到作業系統記憶體的可用性限制。
附註：只有在 AIX 和 Windows 中才支援 DATABASE_MEMORY = AUTOMATIC。
- **模式 2**：當 DATABASE_MEMORY = <NUMERIC VALUE> AUTOMATIC 時
在此情況下，仍會進行記憶體微調，但資料庫使用的記憶體總量會從 NUMERIC 值開始，並會隨著時間經過不斷地增加，而且只會受到作業系統的可用記憶體

限制。

附註：只有在 AIX 和 Windows 中才支援 DATABASE_MEMORY = AUTOMATIC。

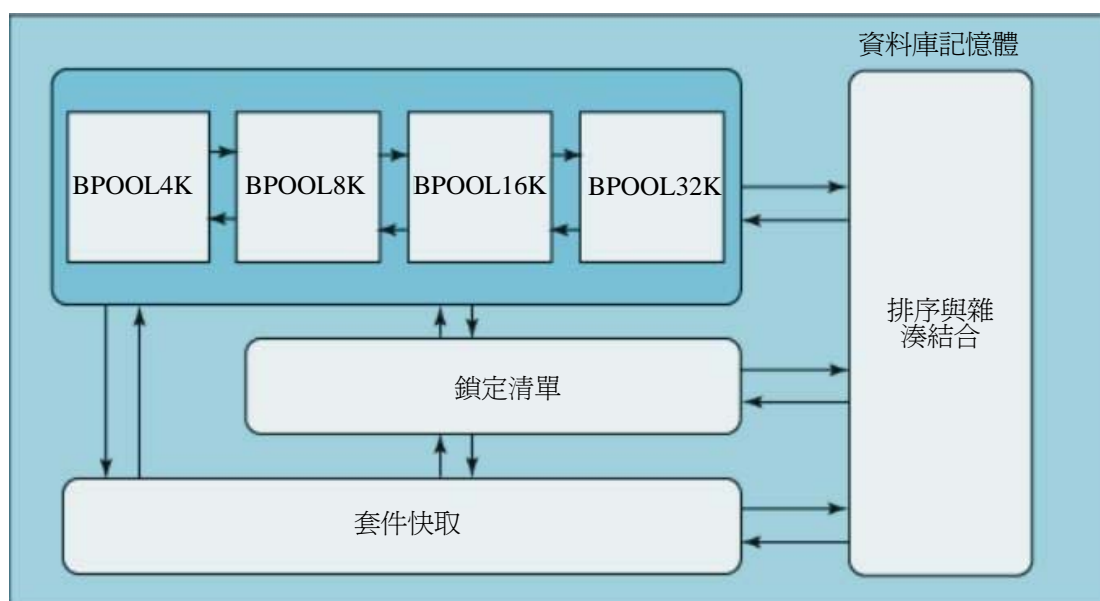
- **模式 3**：當 DATABASE_MEMORY=<NUMERIC VALUE> 時 DB2 會在啓動期間配置某個數量的記憶體，而且這個記憶體設定是靜態的。它無法隨需從作業系統取得或歸還記憶體。
- **模式 4**：當 DATABASE_MEMORY=COMPUTED 時 資料庫記憶體會根據資料庫啓動期間，資料庫記憶體資料堆的起始值總數來計算。在此情況下，不會啓用 DATABASE_MEMORY 來進行自動微調。

STMM 摘要

- 如果將 DATABASE_MEMORY 設定為 AUTOMATIC 或 AUTOMATIC 與某數值的組合，STMM 就會微調資料庫記憶體的總耗用量。
- 如果將其設定為 AUTOMATIC，則當資料庫有需求時，就會從作業系統取得或歸還記憶體。
- 如果將其設定為某個數值，則仍會在設定為 AUTOMATIC 的記憶體資料堆之間進行微調。唯一的限制就是需要將一個以上的記憶體資料堆設定為 AUTOMATIC，才能微調記憶體。
- 如果將其設定為 COMPUTED，就不會進行 DATABASE_MEMORY 微調。

圖 2 顯示 STMM 記憶體結構：

圖 2. STMM 記憶體結構



STMM 會處理資料庫共用記憶體參數。這些參數包括：

- **緩衝池**是由 ALTER BUFFERPOOL 和 CREATE BUFFERPOOL 陳述式所控制。資料庫緩衝儲存池是一個記憶體片段，可在讀取或更新磁碟中所選表格索引和資料頁面時，用來快取這些索引和頁面。
- **套件快取**是由 pckcachesz 配置參數所控制。這個參數是由資料庫共用記憶體所配置，並且用於緩衝靜態 SQL、動態 SQL 和 XQuery 陳述式的區段。
- **鎖定記憶體**是由 locklist 和 maxlocks 配置參數所控制。*Locklist* 參數可指出配置給鎖定清單的儲存體數量。每個資料庫都有一個 locklist，其中包含所有同時連接至資料庫的應用程式所保留的鎖定數。*Maxlocks* 參數定義由應用程式所保留，而且必須在資料庫管理程式執行擴大鎖定之前，先填入資料的鎖定清單百分比。
- **排序記憶體**是由 sheapthres_shr 和 sorheap 配置參數所控制。所配置的共用排序記憶體的大小總計是根據資料庫啟動期間的 SHEAPTHRES_SHR 參數值而定，而且 sorheap 會決定可用於每個排序的記憶體頁數上限。
- **資料庫共用記憶體總計**是由 database_memory 配置參數所控制。這個參數可指定保留給資料庫共用記憶體區域的共用記憶體數量。

啓用 STMM

如何啓用 STMM

您可能會想瞭解如何啓用 DB2 9 的 STMM。首先，請使用「清單 1」中的指令，將資料庫配置參數 SELF_TUNING_MEM 更新為 ON：

清單 1. 將資料庫配置參數更新為 ON

```
db2 "connect to OLTPDB"  
db2 "update database configuration for OLTPDB using  
SELF_TUNING_MEM ON"
```

或透過「控制中心」中的「變更資料庫配置參數」視窗設定該參數。

接著，若要啓用記憶體配置參數所控制的記憶體區自我微調功能，請使用 UPDATE DATABASE CONFIGURATION 指令，或透過控制中心中的「變更資料庫配置參數」視窗，將相關的配置參數設定為 AUTOMATIC。清單 2 顯示這個指令：

清單 2. 啓用記憶體區自我微調功能

```
db2 "connect to OLTPDB"
db2 "update database configuration for oltpdb using PCKCACHESZ
AUTOMATIC"
db2 "update database configuration for oltpdb using LOCKLIST
AUTOMATIC"
db2 "update database configuration for oltpdb using MAXLOCKS
automatic"
db2 "update database configuration for oltpdb using SORTHEAP
AUTOMATIC"
db2 "update database configuration for oltpdb using SHEAPTHRES_SHR
AUTOMATIC"
db2 "update database configuration for oltpdb using
DATABASE_MEMORY AUTOMATIC"
```

最後，若要啓用緩衝池的自我微調功能，請將緩衝池的大小設定爲 `AUTOMATIC`。您可以針對現有的緩衝池使用 `ALTER BUFFER POOL` 陳述式，或者針對新的緩衝池使用 `CREATE BUFFER POOL` 陳述式來完成這項工作。清單 3 顯示這個指令：

清單 3. 啓用緩衝池的自我微調功能

```
db2 "connect to OLTPDB"
db2 "create bufferpool bpool8k size AUTOMATIC pagesize 8 k"
```

如果緩衝池已存在，請使用下列程式碼：`db2 "alter bufferpool bpool4k size automatic"`

重要注意事項：

- 自我微調功能會在不同的記憶體參數之間重新分配記憶體。將 `database_memory` 設爲數值時，只需將另一個記憶體耗用單元設爲 `AUTOMATIC`，就會進行微調。
- 爲了啓用 `locklist` 配置參數以進行自我微調，也必須啓用 `maxlocks`。
- 爲了啓用 `sheapthres_shr` 配置參數以進行自我微調，也必須啓用 `sortheap`。
- 只有將資料庫管理程式配置參數 `sheapthres` 設爲 0 時，才允許自動微調 `sheapthres_shr` 或 `sortheap`。

- 自我微調記憶體只會在 HADR 主要伺服器上執行。在 HADR 配對上啓動自我微調記憶體功能時，它只會在主要伺服器上執行，而不會在備用伺服器上執行。如果在目前的備用機器上執行 HADR TAKEOVER 指令，自我微調記憶體作業也會切換至新的主機上。

如何在作用中及非作用中模式中啓用 STMM

您可以執行 `db2 get database configuration` 指令搭配 `show detail` 來查看目前的 STMM 設定是處於作用中或非作用中狀態。

清單 4. 檢視目前的 STMM 設定

```
Self Tuning Memory
(SELF_TUNING_MEM) = OFF
Self Tuning Memory
(SELF_TUNING_MEM) = ON (Active)
Self Tuning Memory
(SELF_TUNING_MEM) = ON (Inactive)
```

如果參數顯示 "ON (Active)"，就代表記憶體微調器正在主動微調系統上的記憶體。如果參數顯示 "ON (Inactive)"，則代表雖然參數設為 ON，但並未進行自我微調，因為啓用來進行自我微調的記憶體參數在兩個以下。

企業實務

許多微調不良的緩衝池存在時就會顯現 STMM 的重要性

在所有的生產資料庫中，由於緩衝池微調不良，造成資料庫效能降低是意料中的事。隨著緩衝池的數目不斷地增加，要根據記憶體需求來管理緩衝池，是相當困難的事。本文將採用類似 OLTP 的工作量來測試緩衝池微調功能。該系統具有下列規格：

- 系統配置：AIX，8 GB 的 RAM，220 GB 的儲存空間
- 資料庫名稱：OLTPDB
- 資料庫大小：830 MB

- **暫存表格空間**：表格空間 TEMP4K、TEMP8K、TEMP16K、TEMP32K，其頁面大小分別為 4K、8K、16K 及 32K
- **一般系統管理的表格空間**：表格空間 SMS_4K、SMS_8K、SMS_16K、SMS_32K，其頁面大小分別為 4K、8K、16K 及 32K
- **一般資料庫管理的表格空間**：表格空間 DMS_4K、DMS_8K、DMS_16K、DMS_32K，其頁面大小分別為 4K、8K、16K 及 32K
- **索引表格空間**：表格空間 INDEX_4K、INDEX_8K、INDEX_16K、INDEX_32K，其頁面大小分別為 4K、8K、16K 及 32K
- **大型表格空間**：表格空間 LARGE_32K，其頁面大小為 32K
- **緩衝池**：我們建立了 4 個緩衝池 BPOOL_4K、BPOOL_8K、BPOOL_16K 及 BPOOL_32K，其頁面大小分別為 4K、8K、16K 及 32K

對於所有經驗豐富的 DBA 而言，要監控所有緩衝池，並在執行時期根據需求微調這些緩衝池，都是非常困難的事。當每個工作量所需的記憶體都不相同時，這幾乎是不可能的任務。

STMM 可以處理多個緩衝池，而不論其頁面大小為何，它可以在許多緩衝池之間交換記憶體。

在這個案例中，為了設定啓用 STMM 的資料庫，我們採用了下列 DB cfg 參數：

清單 5. 用於設定啓用 STMM 資料庫的參數

```
db2 "update database configuration for OLTPDB using
SELF_TUNING_MEM ON"
db2 "update database configuration for OLTPDB using PCKCACHESZ
AUTOMATIC"
db2 "update database configuration for OLTPDB using LOCKLIST
AUTOMATIC"
db2 "update database configuration for OLTPDB using MAXLOCKS
AUTOMATIC"
db2 "update database configuration for OLTPDB using SORTHEAP
AUTOMATIC"
db2 "update database configuration for OLTPDB using SHEAPTHRES_SHR
AUTOMATIC"
db2 "update database configuration for OLTPDB using
DATABASE_MEMORY AUTOMATIC"
db2 "connect to OLTPDB"
db2 "alter bufferpool bpool_4k size automatic"
db2 "alter bufferpool bpool_8k size automatic"
```



```
db2 "alter bufferpool bpool_16k size automatic"
db2 "alter bufferpool bpool_32k size automatic"
```

爲了測試 STMM 對緩衝池所採取的動作，我們建立了預設大小爲 1000 個頁面的緩衝池。一開始，我們在資料伺服器上執行了會產生 100 個以上用戶端的並行應用程式，並使用 iwatch 測量每秒的交換量 (TPS)。

圖 3 和圖 4 顯示 STMM 對我們測試的不同緩衝池所採取的動作：

圖 3. STMM 對微調不良的緩衝池所採取的動作

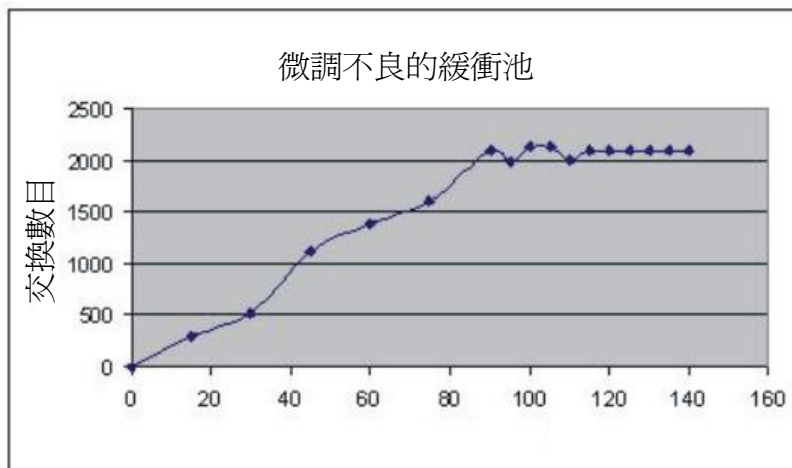
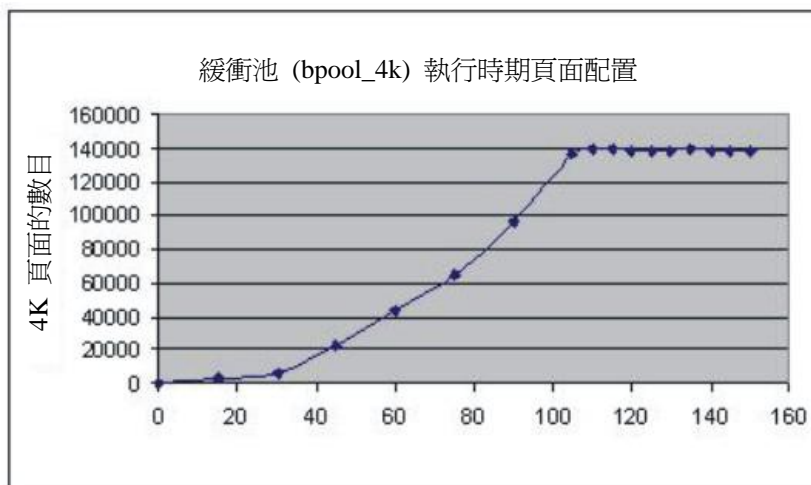


圖 4. STMM 對緩衝池執行時期頁面配置所採取的動作



當我們開始執行應用程式時，起始 TPS 很低，然後隨著時間經過，它也開始增加。STMM 會負責微調緩衝池，最後使得效能獲得改善。

當一個實例中有許多資料庫時

一個實例中有多個資料庫是很平常的事，微調多個資料庫是代價高昂的工作。有了 STMM 的協助，管理者就可以輕鬆微調多個資料庫。STMM 會在不同的資料庫之間交換記憶體，讓系統記憶體的使用最佳化。

在實例 DB2INST1 中，我們有兩個資料庫 OLTPDB（來自上一節）和 SAMPLE，而且這兩個資料庫都已啓用 STMM，如下所示：

```
SELF_TUNING_MEM = ON
DATABASE_MEMORY = AUTOMATIC
```

這個個案研討採用具有典型 OLTP 工作量的線上銀行系統作為範例。該系統具有下列規格：

- 系統配置：AIX，8 GB 的 RAM，220 GB 的儲存空間
- 資料庫：OLTPDB 及 SAMPLE
- 資料庫大小：OLTPDB 830 MB 及 SAMPLE 600 MB

圖 5 和圖 6 顯示 STMM 對 OLTPDB 和 SAMPLE DB 所採取的動作：

圖 5. STMM 對 OLTPDB 所採取的動作

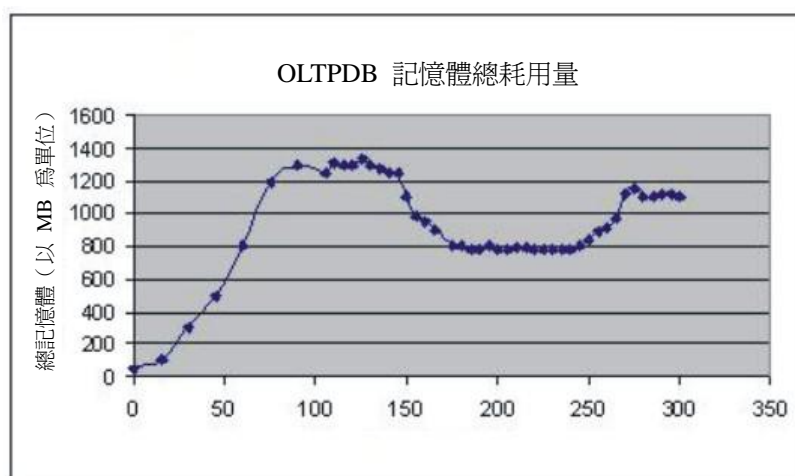
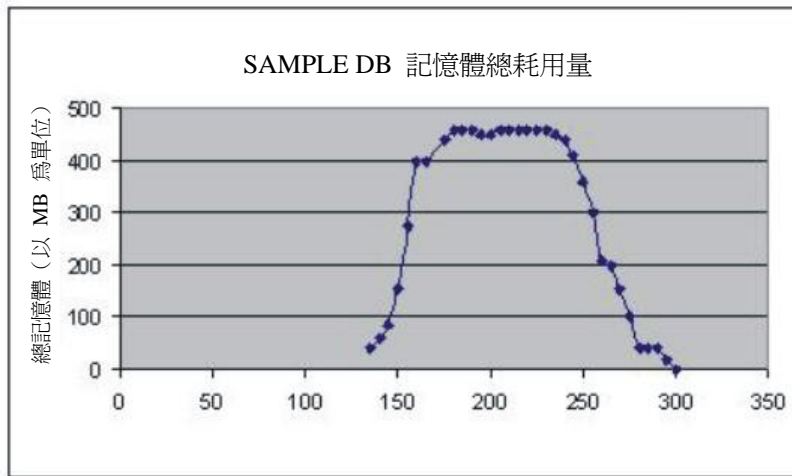


圖 6. STMM 對 SAMPLE DB 所採取的動作



當第一個資料庫 OLTPDB 啟動後，它會耗用大部分可用的系統記憶體。1 小時 15 分鐘以後，第二個資料庫 SAMPLE 也啟動了，然後 STMM 就在這兩個資料庫之間交換資料庫記憶體。4 小時 54 分鐘以後，SAMPLE 停止了；STMM 就將所有的記憶體配置給資料庫 OLTPDB。

附註：這項測試已利用測試工作量在測試伺服器上執行過。雖然工作量與 OLTP 類似，但它們並非真正的 OLTP 應用程式。測試結果可能無法反映出 OLTP 即時系統上實際的測試結果。

商業效益

如您所見，因為 STMM 可以執行重要但通常非常耗時的記憶體管理工作，因此對企業極具效益。當資料庫中有許多緩衝池時，它會在不同的緩衝池之間，自動配置記憶體空間。STMM 可以每秒為基礎來微調資料庫，並可在實例中有一個以上的資料庫時，輕鬆地分配記憶體。STMM 也可以在所有的實例中微調資料庫，有了 DB2 9 的自調式記憶體管理功能企業可以因此而受益，該功能可自動執行需要耗費 DBA 數小時及許多心力才能完成的工作。

結語

DB2 9 的自調式記憶體管理功能，可以不斷地協助 DB2 資料庫管理者有效管理資料伺服器。

在大部分的企業運算中，DB2 資訊管理軟體都是處於核心地位。DB2 9 負責儲存及擷取各種資訊，在記憶體需求不同的情況下，DB2 9 本身會負責管理記憶體，而無須 DBA 的介入。當工作量不斷地大幅變更，而且資料庫管理者發現很難有效管理的情況下，這項功能也非常實用。

STMM 可說是 DB2 9 中一項可自我微調記憶體的革命性功能。

致謝

特別感謝 Adam Storm (STMM 的首席開發人員) 協助我們更佳瞭解 STMM 的概念以及修飾本文。

作者簡介

Mohan 是 DB2 資料庫專家，專精於 DB2 Q-Replication、DB2 效能以及 DB2 管理。在印度的 IBM 軟體實驗室工作時，他和 High Performance On Demand Solutions (HiPODS) 團隊共同合作，以提供效能解決方案給客戶。之前他則是印度 DB2 產品開發團隊的一員。Mohan 是通過 IBM 認證的 DB2 高階資料庫管理者、DB2 應用程式開發人員，以及 DB2 問題診斷大師。

Swetha Prasad 是印度 IBM 軟體實驗室的 DB2 應用程式開發人員。她和 DB2 ID 範例開發小組共同合作開發 DB2 範例應用程式。