DB2 9 self-tuning memory management

Manage your business, not your database

Skill Level: Intermediate

Paul Read (paul_read@uk.ibm.com)
Product Introduction Manager
IBM

Eva Angelika Billich (bch@de.ibm.com)
Senior DB2 Consultant for Strategic IBM Business Partners in Europe, Middle East, and Africa
IBM

30 Nov 2006

Starting in IBM® DB2® 9, a new memory-tuning feature, self-tuning memory management, simplifies the task of memory configuration by automatically setting values for several memory configuration parameters. When enabled, the memory tuner dynamically distributes available memory resources among several memory consumers, including sorts, the package cache, the lock list, and buffer pools. In this tutorial, walk through a series of exercises to understand and learn how to administer this new feature.

Section 1. Before you start

Introduction

This tutorial is based on the IBM DB2 9 self-tuning memory management feature. The tuner works within the memory limits defined by the database_memory configuration parameter. The value of database_memory can itself be automatically tuned on Windows(R) and AIX(R). When self-tuning is enabled for database_memory (when it is set to AUTOMATIC), the tuner determines the overall memory requirements for the database and increases or decreases the amount of memory allocated for database shared memory depending on the current database requirements. For example, if the current database requirements are high, and there is sufficient free memory on the system, more memory will be consumed by database shared memory. Once the database memory requirements drop, or the amount of free memory on the system drops too low, some database shared memory is released.

This tutorial provides hands-on experience in the five basic areas:

1. Activating self-tuning memory
2. Determining which memory consumers are enabled for self tuning
3. Determining the current size of DB2 memory usage
4. Automatic tuning in a production environment
5. Deactivating self-tuning memory

Further description of the feature can be found in the article " Self-tuning memory in DB2 9" (developerWorks, June 2006)..

About this tutorial

The following exercises enable you to work with the DB2 self-tuning memory feature and to have available memory resources dynamically distributed among several memory consumers, including sorts, the package cache, the lock list, and buffer pools.

They have been developed to demonstrate one or more tasks found within each of the areas.

Objectives

This tutorial helps familiarize you with concepts and capabilities of self tuning in DB2 9:

1. Self-tuning memory overview and logon
2. Self-tuning memory and SORTHEAP
3. Self-tuning memory and PCKCACHESZ
4. Self-tuning memory and LOCKLIST
5. Self-tuning memory and health checking

System requirements

To run the examples in this tutorial, you need the following installed on your machine:

- DB2 9 Data Server
- Microsoft Windows 2000 or later and an account with administrator privileges or Linux (Validated edition) with root access
- Java Runtime Environment 1.4.2 or later
- Hardware requirements referred to on the DB2 9 system requirements page

DB2 9 Express-C is available from the DB2 9 Data Server link above. See the tutorial "DB2 XML evaluation guide (developerWorks, June 2006) for steps on installing DB2. Unless the DB2 configuration is altered, then DB2 will automatically start after installation.

Use the samples scripts and data provided in the stmm_zipped.zip file (see Downloads section) to demonstrate the concepts in this tutorial. Extract the contents into a subdirectory called stmm_scripts (C:\stmm_scripts or home/userid/stmm_scripts, for example). This directory will be referred to simply as stmm_scripts throughout the tutorial. This tutorial assumes that you have used the default directories for the DB2 installation and all the DB2 exercises are performed by a database administrator id.

Section 2. Self-tuning memory overview and logon

Introduction

The following exercise will set the environment for the following sections. The DB2 self-tuning memory feature uses the available memory resources dynamically. It distributes the resources among several memory consumers, including sorts, the package cache, the lock list, and buffer pools.

Key points

The following exercise guides you through the following steps:

1. Log on to your machine and DB2
2. Set archival logging
3. Create test table LINEITEM and associated objects
4. Reorganize the table and create statistics

5. Deactivate self-tuning memory, and set the initial values for the database

Instructions

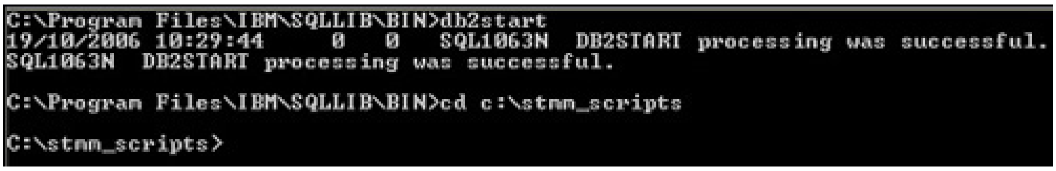
1. Log on to your machine using a suitable id (with DBA authority). For the purpose of these exercises in this tutorial, we have used:
   - UID: db2inst1
   - PW: password
2. Open a terminal window (or DB2 command window on Windows), and start the DB2 Control Center.
3. Start DB2 by issuing the following command:

```
db2start
```

4. Change the current directory to stmm_scripts:

```
cd \stmm_scripts
```

Figure 1. Command window



```
C:\Program Files\IBM\SQLLIB\BIN>db2start
19/10/2006 10:29:44     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.

C:\Program Files\IBM\SQLLIB\BIN>cd c:\stmm_scripts

C:\stmm_scripts>
```

5. Change from circular logging to archival logging to have enough log space for the import operation, and take a backup using the following commands:

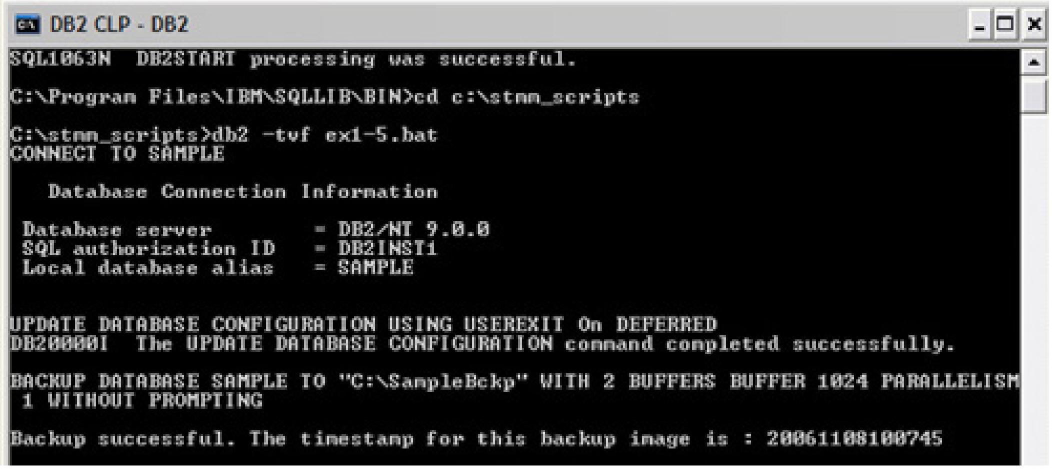
Listing 1. Backup database sample

```
CONNECT TO SAMPLE;
UPDATE DATABASE CONFIGURATION USING USEREXIT On DEFERRED ;
BACKUP DATABASE SAMPLE
        TO "/SampleBckp"
        WITH 2 BUFFERS BUFFER 1024
        PARALLELISM 1
        WITHOUT PROMPTING;
```

Or execute the following batch script:

```
db2 –tvf EX1-5.bat
```

Figure 2. Command window



```
SQL1063N  DB2START processing was successful.

C:\Program Files\IBM\SQLLIB\BIN>cd c:\stmm_scripts

C:\stmm_scripts>db2 -tvf ex1-5.bat
CONNECT TO SAMPLE

   Database Connection Information

 Database server        = DB2/NT 9.0.0
 SQL authorization ID   = DB2INST1
 Local database alias   = SAMPLE


UPDATE DATABASE CONFIGURATION USING USEREXIT On DEFERRED
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

BACKUP DATABASE SAMPLE TO "C:\SampleBckp" WITH 2 BUFFERS BUFFER 1024 PARALLELISM
 1 WITHOUT PROMPTING

Backup successful. The timestamp for this backup image is : 20061108100745
```

6. Create table LINEITEM and associated indexes, and populate the table with data using the following commands:

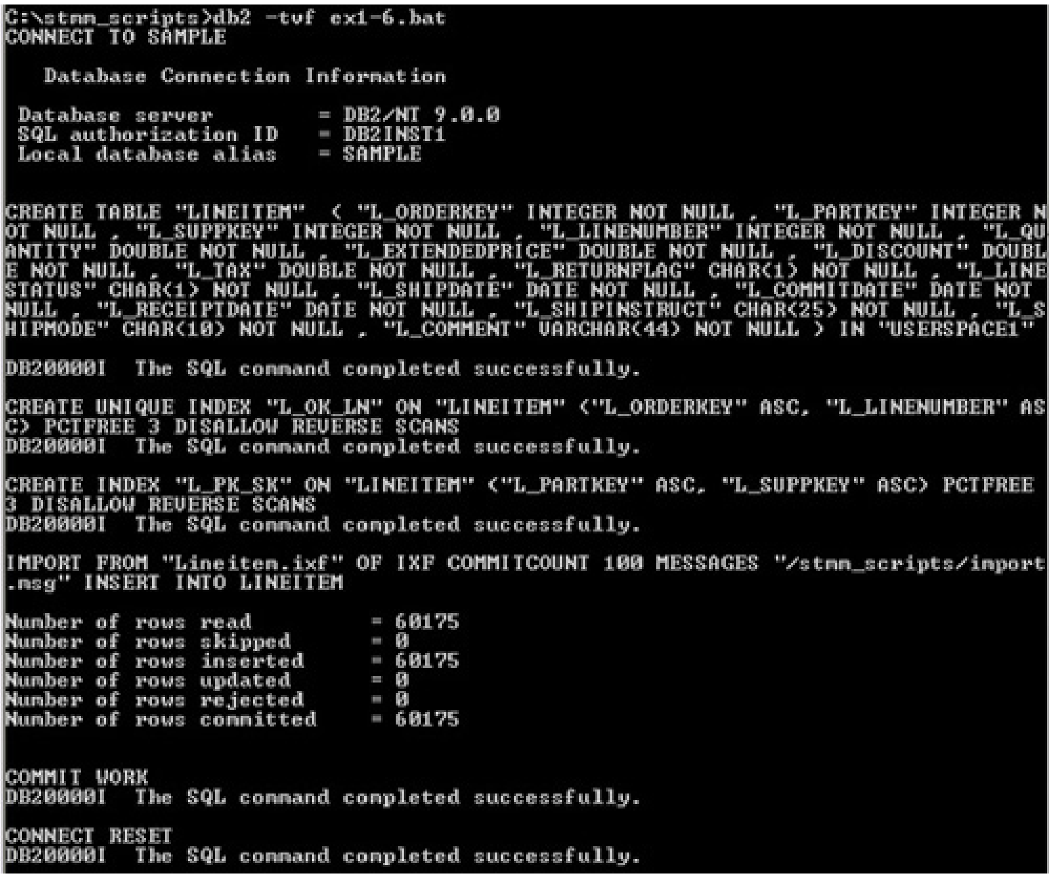
Listing 2. Create table LINEITEM and associated indexes; populate table

```
CREATE TABLE "LINEITEM"  (
                  "L_ORDERKEY" INTEGER NOT NULL ,
                  "L_PARTKEY" INTEGER NOT NULL ,
                  "L_SUPPKEY" INTEGER NOT NULL ,
                  "L_LINENUMBER" INTEGER NOT NULL ,
                  "L_QUANTITY" DOUBLE NOT NULL ,
                  "L_EXTENDEDPRICE" DOUBLE NOT NULL ,
                  "L_DISCOUNT" DOUBLE NOT NULL ,
                  "L_TAX" DOUBLE NOT NULL ,
                  "L_RETURNFLAG" CHAR(1) NOT NULL ,
                  "L_LINESTATUS" CHAR(1) NOT NULL ,
                  "L_SHIPDATE" DATE NOT NULL ,
                  "L_COMMITDATE" DATE NOT NULL ,
                  "L_RECEIPTDATE" DATE NOT NULL ,
                  "L_SHIPINSTRUCT" CHAR(25) NOT NULL ,
                  "L_SHIPMODE" CHAR(10) NOT NULL ,
                  "L_COMMENT" VARCHAR(44) NOT NULL )
                 IN "USERSPACE1" ;
CREATE UNIQUE INDEX "L_OK_LN" ON "LINEITEM"
                ("L_ORDERKEY" ASC,
                 "L_LINENUMBER" ASC)
                PCTFREE 3 DISALLOW REVERSE SCANS;
CREATE INDEX "L_PK_SK" ON "LINEITEM"
                ("L_PARTKEY" ASC,
                 "L_SUPPKEY" ASC)
                PCTFREE 3 DISALLOW REVERSE SCANS;
IMPORT FROM "Lineitem.ixf" OF IXF
                COMMITCOUNT 100
                MESSAGES "/stmm/import.msg"
                INSERT INTO LINEITEM;
```

Or execute the following batch script:

```
db2 –tvf EX1-6.bat
```

Figure 3. Command window



```
C:\stmm_scripts>db2 -tvf ex1-6.bat
CONNECT TO SAMPLE

   Database Connection Information

 Database server        = DB2/NT 9.0.0
 SQL authorization ID   = DB2INST1
 Local database alias   = SAMPLE


CREATE TABLE "LINEITEM"  ( "L_ORDERKEY" INTEGER NOT NULL , "L_PARTKEY" INTEGER N
OT NULL , "L_SUPPKEY" INTEGER NOT NULL , "L_LINENUMBER" INTEGER NOT NULL , "L_QU
ANTITY" DOUBLE NOT NULL , "L_EXTENDEDPRICE" DOUBLE NOT NULL , "L_DISCOUNT" DOUBL
E NOT NULL , "L_TAX" DOUBLE NOT NULL , "L_RETURNFLAG" CHAR(1) NOT NULL , "L_LINE
STATUS" CHAR(1) NOT NULL , "L_SHIPDATE" DATE NOT NULL , "L_COMMITDATE" DATE NOT
NULL , "L_RECEIPTDATE" DATE NOT NULL , "L_SHIPINSTRUCT" CHAR(25) NOT NULL , "L_S
HIPMODE" CHAR(10) NOT NULL , "L_COMMENT" VARCHAR(44) NOT NULL ) IN "USERSPACE1"

DB20000I  The SQL command completed successfully.

CREATE UNIQUE INDEX "L_OK_LN" ON "LINEITEM" ("L_ORDERKEY" ASC, "L_LINENUMBER" AS
C) PCTFREE 3 DISALLOW REVERSE SCANS
DB20000I  The SQL command completed successfully.

CREATE INDEX "L_PK_SK" ON "LINEITEM" ("L_PARTKEY" ASC, "L_SUPPKEY" ASC) PCTFREE
3 DISALLOW REVERSE SCANS
DB20000I  The SQL command completed successfully.

IMPORT FROM "Lineitem.ixf" OF IXF COMMITCOUNT 100 MESSAGES "/stmm_scripts/import
.msg" INSERT INTO LINEITEM

Number of rows read         = 60175
Number of rows skipped      = 0
Number of rows inserted     = 60175
Number of rows updated      = 0
Number of rows rejected     = 0
Number of rows committed    = 60175


COMMIT WORK
DB20000I  The SQL command completed successfully.

CONNECT RESET
DB20000I  The SQL command completed successfully.
```

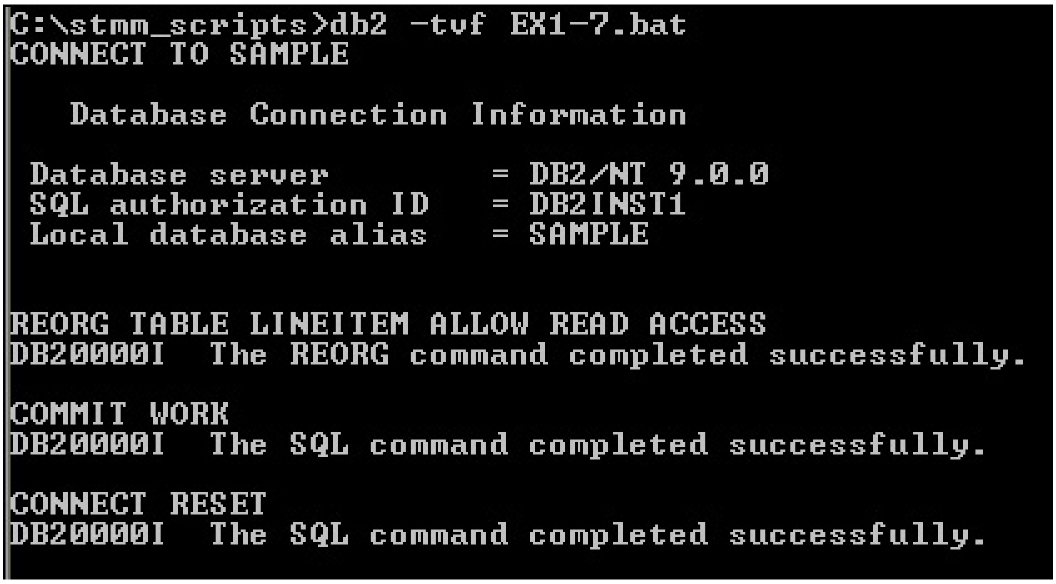
7. Reorganize table LINEITEM using the following command:

```
REORG TABLE LINEITEM ALLOW READ ACCESS;
```

Or execute the following batch script:

```
db2 –tvf EX1-7.bat
```

Figure 4. Command window



```
C:\stmm_scripts>db2 -tvf EX1-7.bat
CONNECT TO SAMPLE

   Database Connection Information

 Database server        = DB2/NT 9.0.0
 SQL authorization ID   = DB2INST1
 Local database alias   = SAMPLE


REORG TABLE LINEITEM ALLOW READ ACCESS
DB20000I  The REORG command completed successfully.

COMMIT WORK
DB20000I  The SQL command completed successfully.

CONNECT RESET
DB20000I  The SQL command completed successfully.
```

8. Create new statistics on table LINEITEM using the RUNSTATS command:

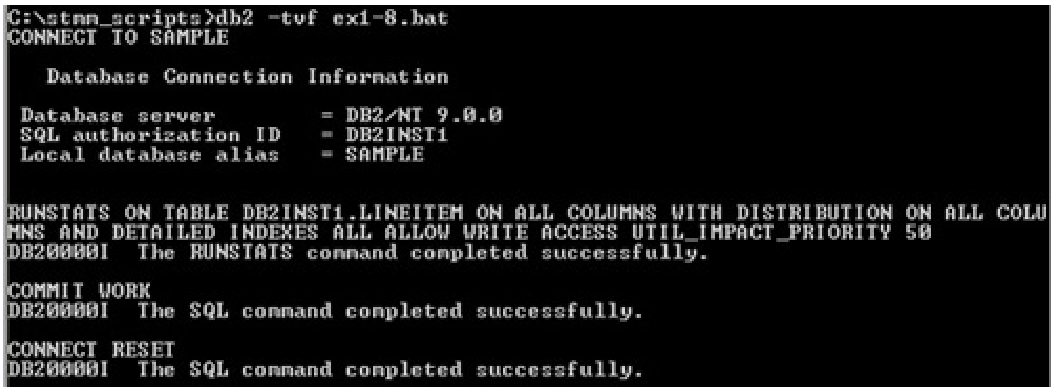
Listing 3. Create new statistics on table LINEITEM

```
RUNSTATS ON TABLE DB2INST1.LINEITEM
        ON ALL COLUMNS WITH DISTRIBUTION
        ON ALL COLUMNS AND DETAILED INDEXES ALL
        ALLOW WRITE ACCESS
        UTIL_IMPACT_PRIORITY 50;
```

Or execute the following batch script:

```
db2 –tvf EX1-8.bat
```

Figure 5. Command window



```
C:\stmm_scripts>db2 -tvf ex1-8.bat
CONNECT TO SAMPLE

   Database Connection Information

 Database server        = DB2/NT 9.0.0
 SQL authorization ID   = DB2INST1
 Local database alias   = SAMPLE


RUNSTATS ON TABLE DB2INST1.LINEITEM ON ALL COLUMNS WITH DISTRIBUTION ON ALL COLU
MNS AND DETAILED INDEXES ALL ALLOW WRITE ACCESS UTIL_IMPACT_PRIORITY 50
DB20000I  The RUNSTATS command completed successfully.

COMMIT WORK
DB20000I  The SQL command completed successfully.

CONNECT RESET
DB20000I  The SQL command completed successfully.
```

9. Deselect the parameter option AUTOMATIC for all instance and database parameters, as follows:

Listing 4. Deselect the parameter option AUTOMATIC for all instance and database parameters

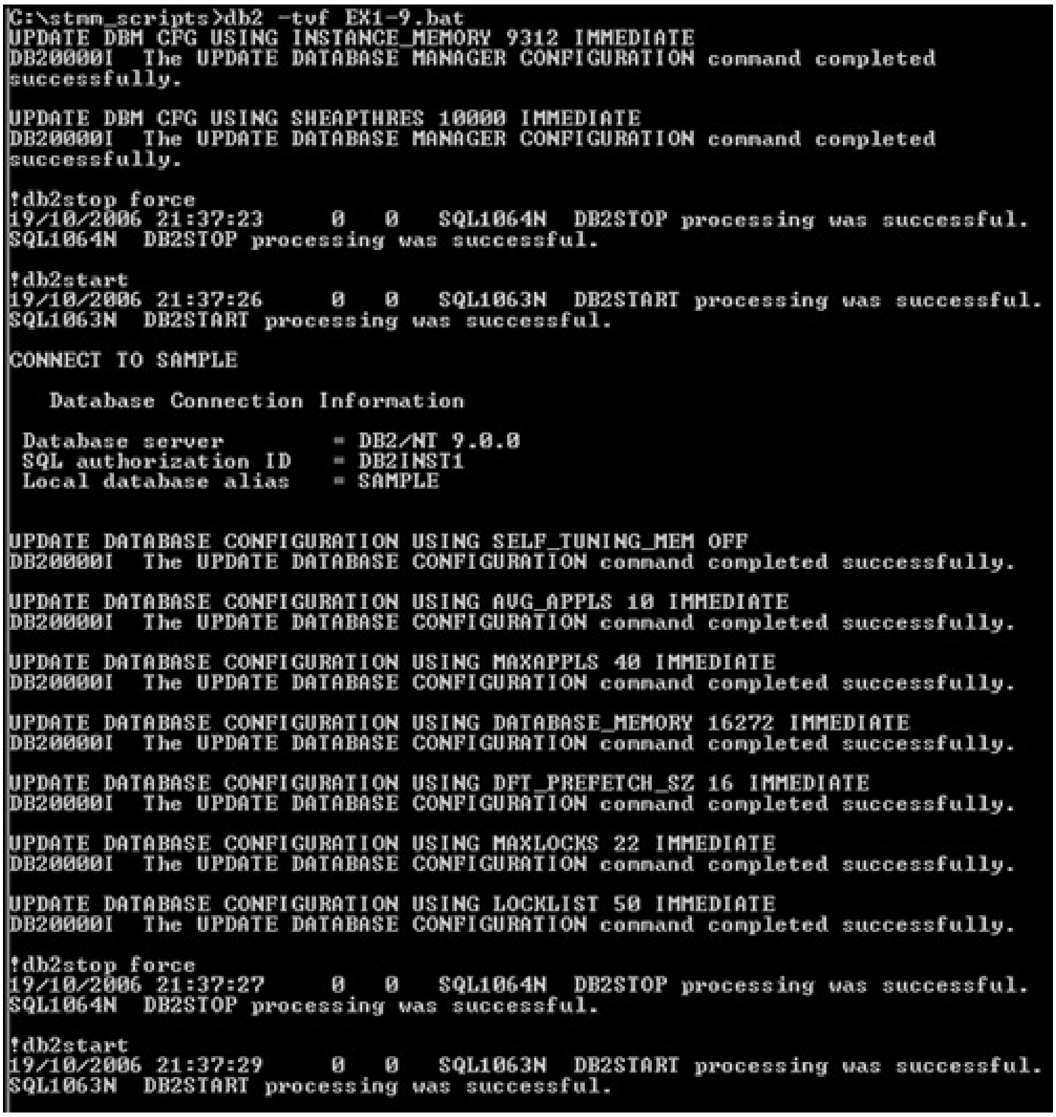
```
UPDATE DBM CFG USING INSTANCE_MEMORY 9312 IMMEDIATE;
UPDATE DBM CFG USING SHEAPTHRES 10000 IMMEDIATE;
```

```
!db2stop force;
!db2start;
CONNECT TO SAMPLE;
UPDATE DATABASE CONFIGURATION USING SELF_TUNING_MEM OFF;
UPDATE DATABASE CONFIGURATION USING AVG_APPLS 10 IMMEDIATE ;
UPDATE DATABASE CONFIGURATION USING MAXAPPLS 40 IMMEDIATE ;
UPDATE DATABASE CONFIGURATION USING DATABASE_MEMORY 16272 IMMEDIATE ;
UPDATE DATABASE CONFIGURATION USING DFT_PREFETCH_SZ 16 IMMEDIATE ;
UPDATE DATABASE CONFIGURATION USING MAXLOCKS 22 IMMEDIATE ;
UPDATE DATABASE CONFIGURATION USING LOCKLIST 50 IMMEDIATE ;
!db2stop force;
!db2start;
```

Or execute the following batch script:

```
db2 –tvf EX1-9.bat
```

Figure 6. Command window



```
C:\stmm_scripts>db2 -tvf EX1-9.bat
UPDATE DBM CFG USING INSTANCE_MEMORY 9312 IMMEDIATE
DB20000I  The UPDATE DATABASE MANAGER CONFIGURATION command completed
successfully.

UPDATE DBM CFG USING SHEAPTHRES 10000 IMMEDIATE
DB20000I  The UPDATE DATABASE MANAGER CONFIGURATION command completed
successfully.

!db2stop force
19/10/2006 21:37:23     0   0   SQL1064N  DB2STOP processing was successful.
SQL1064N  DB2STOP processing was successful.

!db2start
19/10/2006 21:37:26     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.

CONNECT TO SAMPLE

   Database Connection Information

 Database server        = DB2/NT 9.0.0
 SQL authorization ID   = DB2INST1
 Local database alias   = SAMPLE


UPDATE DATABASE CONFIGURATION USING SELF_TUNING_MEM OFF
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

UPDATE DATABASE CONFIGURATION USING AVG_APPLS 10 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

UPDATE DATABASE CONFIGURATION USING MAXAPPLS 40 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

UPDATE DATABASE CONFIGURATION USING DATABASE_MEMORY 16272 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

UPDATE DATABASE CONFIGURATION USING DFT_PREFETCH_SZ 16 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

UPDATE DATABASE CONFIGURATION USING MAXLOCKS 22 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

UPDATE DATABASE CONFIGURATION USING LOCKLIST 50 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

!db2stop force
19/10/2006 21:37:27     0   0   SQL1064N  DB2STOP processing was successful.
SQL1064N  DB2STOP processing was successful.

!db2start
19/10/2006 21:37:29     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.
```

Section 3. Self-tuning memory and SORTHEAP

Introduction

The sort heap (SORTHEAP) determines the maximum number of memory pages that can be used for each sort. The sort heap area is allocated in the agent private memory if intra-partition parallelism is disabled. However, if intra-partition parallelism is enabled, a sort operation is processed in parallel, and the sort heap area is allocated in the agent private memory or the database global memory, depending on which type of sort -- a private sort or a shared sort -- is performed. For a private sort, a sort heap area is allocated independently for each parallel agent in the private agent memory. For a shared sort, a sort heap area is allocated in the database global memory, and each parallel agent shares this sort heap. The SORTHEAP parameter is used by the optimizer to determine if the sorting can be performed in memory or on disk. DB2 always attempts to perform the sort in memory.

Key points

The following exercise guides you through enabling the sort heap for self tuning and running a workload, which demonstrates how the sort heap is tuned automatically by DB2 over time.

Instructions

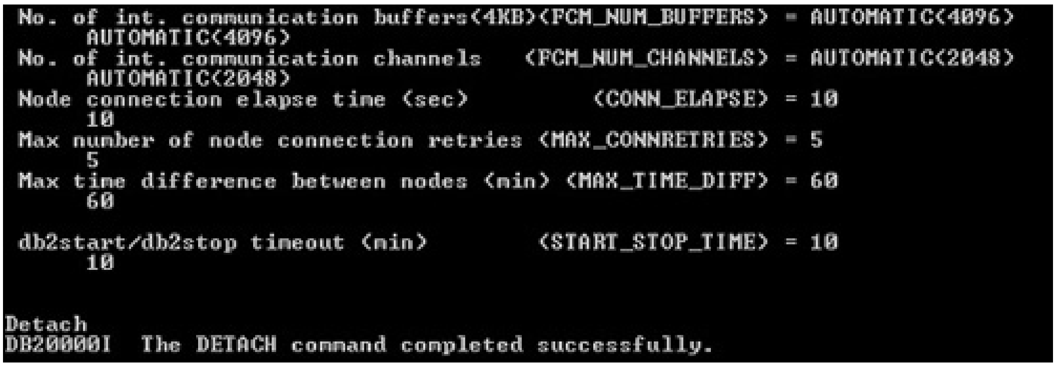
1. Determine the current value of the instance parameter sheapthres:

```
db2 attach to db2
db2 get dbm cfg show detail
```

Or execute the following batch script:

```
db2 -tvf EX2-1.bat
```

Figure 7. Command window



```
 No. of int. communication buffers(4KB)(FCM_NUM_BUFFERS) = AUTOMATIC(4096)
      AUTOMATIC(4096)
 No. of int. communication channels   (FCM_NUM_CHANNELS) = AUTOMATIC(2048)
      AUTOMATIC(2048)
 Node connection elapse time (sec)         (CONN_ELAPSE) = 10
      10
 Max number of node connection retries (MAX_CONNRETRIES) = 5
      5
 Max time difference between nodes (min) (MAX_TIME_DIFF) = 60
      60

 db2start/db2stop timeout (min)        (START_STOP_TIME) = 10
      10


Detach
DB20000I  The DETACH command completed successfully.
```

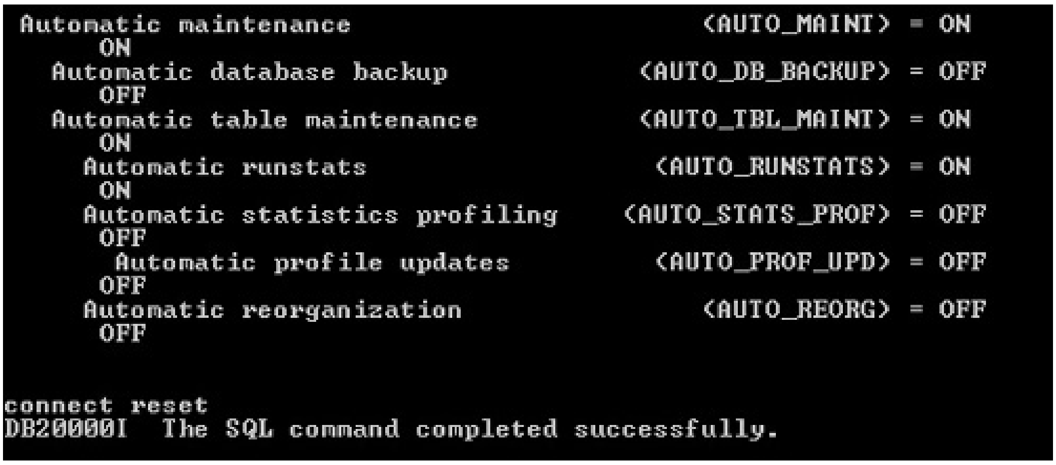
2. Determine the current value of the database parameters self_tuning_mem, sheapthres_shr, and sheap:

```
db2 connect to sample
db2 get db cfg for sample show detail
db2 disconnect current
```

Or execute the following batch script:

```
db2 -tvf EX2-2.bat
```

Figure 8. Command window



```
 Automatic maintenance                              (AUTO_MAINT) = ON
      ON
   Automatic database backup                    (AUTO_DB_BACKUP) = OFF
      OFF
   Automatic table maintenance                  (AUTO_TBL_MAINT) = ON
      ON
     Automatic runstats                        (AUTO_RUNSTATS) = ON
      ON
     Automatic statistics profiling           (AUTO_STATS_PROF) = OFF
      OFF
       Automatic profile updates               (AUTO_PROF_UPD) = OFF
      OFF
     Automatic reorganization                     (AUTO_REORG) = OFF
      OFF


connect reset
DB20000I  The SQL command completed successfully.
```

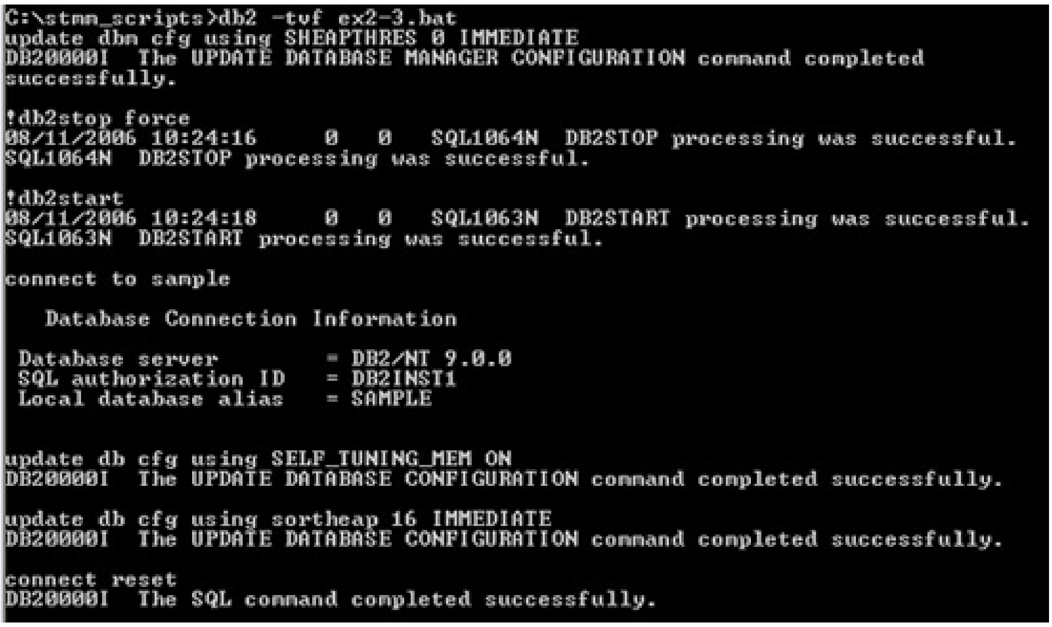
3. Set the SORTHEAP to 16 pages and SHEAPTHRES to zero:
   Listing 5. Set SORTHEAP to 16 pages and SHEAPTHRES to zero

```
db2 update dbm cfg using SHEAPTHRES 0 IMMEDIATE
db2stop force
db2start
db2 connect to sample
db2 update db cfg using SELF_TUNING_MEM ON
db2 update db cfg using SORTHEAP 16 IMMEDIATE
```

Or execute the following batch script:

```
db2 -tvf EX2-3.bat
```

Figure 9. Command window



```
C:\stmm_scripts>db2 -tvf ex2-3.bat
update dbm cfg using SHEAPTHRES 0 IMMEDIATE
DB20000I  The UPDATE DATABASE MANAGER CONFIGURATION command completed
successfully.

!db2stop force
08/11/2006 10:24:16     0   0   SQL1064N  DB2STOP processing was successful.
SQL1064N  DB2STOP processing was successful.

!db2start
08/11/2006 10:24:18     0   0   SQL1063N  DB2START processing was successful.
SQL1063N  DB2START processing was successful.

connect to sample

   Database Connection Information

 Database server        = DB2/NT 9.0.0
 SQL authorization ID   = DB2INST1
 Local database alias   = SAMPLE


update db cfg using SELF_TUNING_MEM ON
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

update db cfg using sortheap 16 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

connect reset
DB20000I  The SQL command completed successfully.
```

4. Stop and start the database SAMPLE by opening the DB2 Control Center -- right-click on the database SAMPLE node, STOP and START. Make sure self tuning is enabled and sortheap is enabled for self tuning:

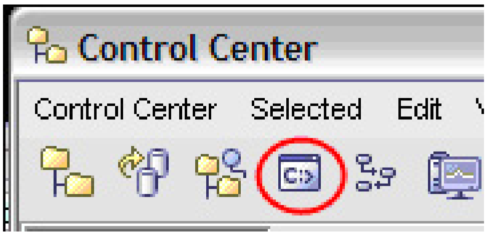
```
db2 get dbm cfg
db2 get db cfg for sample show detail
```

5. . Open a command line prompt, and issue the following command:

```
db2mtrk -d -v -r 1 > mtrk_sh
```

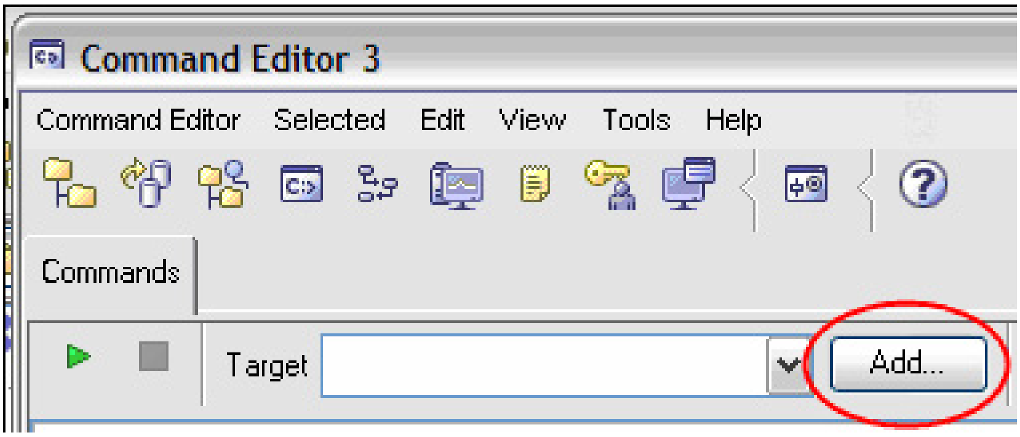
6. Open the DB2 Command Editor from the Control Center:

Figure 10. Command Editor



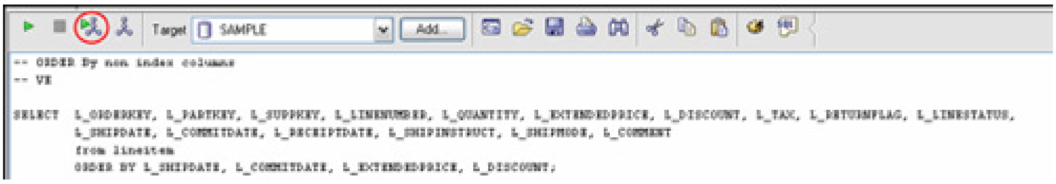
7. Connect to the SAMPLE database by clicking the **Add** button and selecting the database SAMPLE:

Figure 11. Command Editor



8. Issue the following query (Query_03.txt, which is included in the tutorial's download) by clicking the radio button **Execute and Access Plan**:

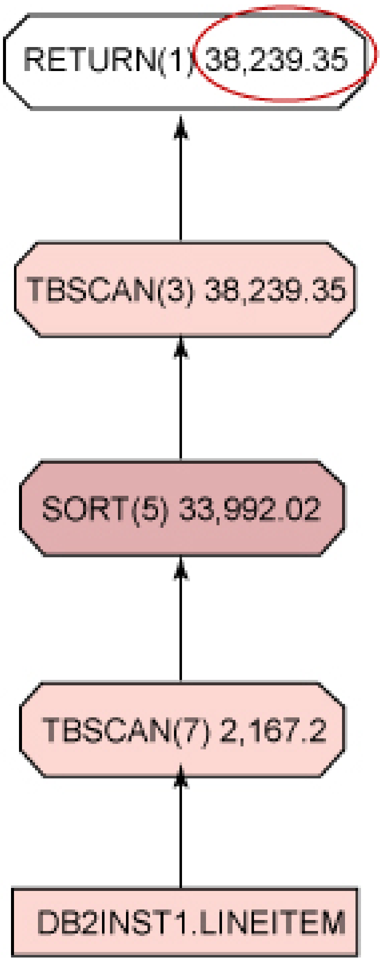
Figure 12. Command Editor



```
-- ORDER By non index columns
-- VE

SELECT  L_ORDERKEY, L_PARTKEY, L_SUPPKEY, L_LINENUMBER, L_QUANTITY, L_EXTENDEDPRICE, L_DISCOUNT, L_TAX, L_RETURNFLAG, L_LINESTATUS,
        L_SHIPDATE, L_COMMITDATE, L_RECEIPTDATE, L_SHIPINSTRUCT, L_SHIPMODE, L_COMMENT
        from lineitem
        ORDER BY L_SHIPDATE, L_COMMITDATE, L_EXTENDEDPRICE, L_DISCOUNT;
```

Note the total amount of timerons for the execution of the query:

Figure 13. Total amount of timerons

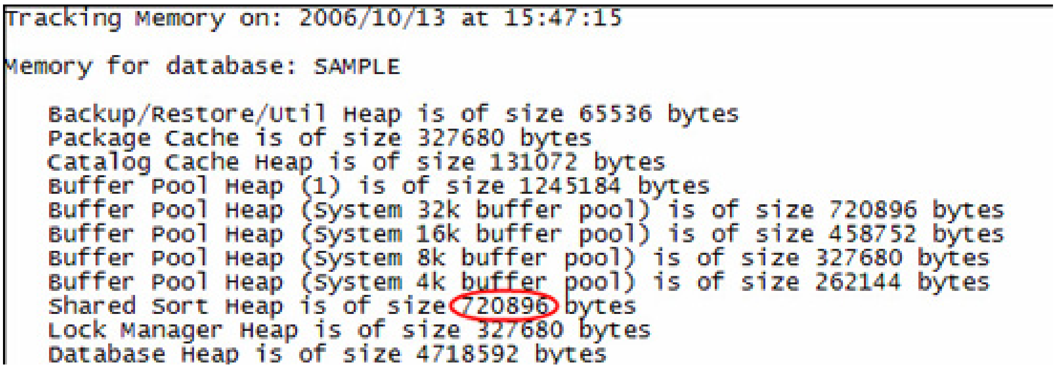


9. When the query is finished, stop the db2mtrk in the command line prompt by pressing the **Ctrl** + **C** keys.

```
notepad mtrk.sh
```

Note: While 16 pages had been defined in the database configuration, DB2 automatically increased the SortHeap up to 176 pages (720896 bytes) on my system.

Figure 14. Memory for database SAMPLE



```
Tracking Memory on: 2006/10/13 at 15:47:15

Memory for database: SAMPLE

   Backup/Restore/Util Heap is of size 65536 bytes
   Package Cache is of size 327680 bytes
   Catalog Cache Heap is of size 131072 bytes
   Buffer Pool Heap (1) is of size 1245184 bytes
   Buffer Pool Heap (System 32k buffer pool) is of size 720896 bytes
   Buffer Pool Heap (System 16k buffer pool) is of size 458752 bytes
   Buffer Pool Heap (System 8k buffer pool) is of size 327680 bytes
   Buffer Pool Heap (System 4k buffer pool) is of size 262144 bytes
   Shared Sort Heap is of size 720896 bytes
   Lock Manager Heap is of size 327680 bytes
   Database Heap is of size 4718592 bytes
```

The exercise can be enhanced by dropping the index "L_OK_LN," deactivating the database SAMPLE, activating the database SAMPLE, and repeating the instructions from step 5 to step 10. Compare the two results.

Section 4. Self-tuning memory and PCKCACHESZ

Introduction

Sometimes, the complete statement is not known at application development time. In this case, the compiler is invoked during program execution to generate an access plan for the query that can be used by the database manager to access the data. This type of SQL statement is called a dynamic SQL statement. The access plans for a dynamic SQL statement are not stored in the system catalogs. They are temporarily stored in memory (known as the global package cache). The compiler will not be invoked if the access plans for the dynamic SQL statements already exist in the package cache. Access plans are cached for static and dynamic SQL statements in the package cache.

Key points

The following exercise guides you through the steps from enabling the package cache for self tuning, how it is automatically increased by DB2, and how you are using the DB2 Memory Tracker tool to follow the cache changes during an SQL execution.

Instructions

1. Repeat step 3 from the "Self-tuning memory and SORTHEAP" section and add the following command:

```
db2 update db cfg using PCKCACHESZ 32 IMMEDIATE  AUTOMATIC
```

   Or edit the script EX2-3.bat (included in the tutorial's download) to include the command, and run:

```
db2 -tvf EX2-3.bat
```

2. Make sure self tuning is enabled and sortheap, sheapthres_thr, and pckcachesz are enabled for self tuning:

```
db2 connect to sample
db2 get db cfg for sample show detail
db2 disconnect current
```

   Or execute the following batch script:

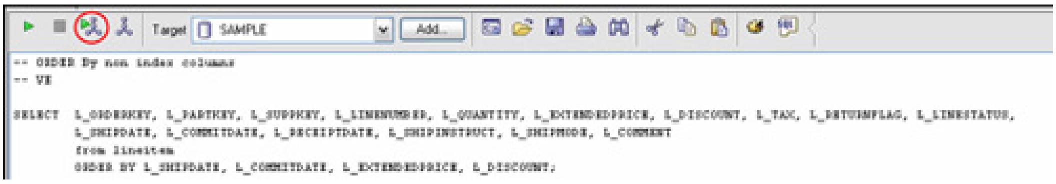
```
db2 -tvf EX2-2.bat
```

3. Determine the current sort heap value using the DB2 memory tracker. Open a command line prompt, and issue the following command:

```
db2mtrk –d –v –r 1 > mtrk_pc
```

4. Open the DB2 Command Editor from the Control Center, then connect to the SAMPLE database by clicking on the **Add** button and selecting the database SAMPLE.

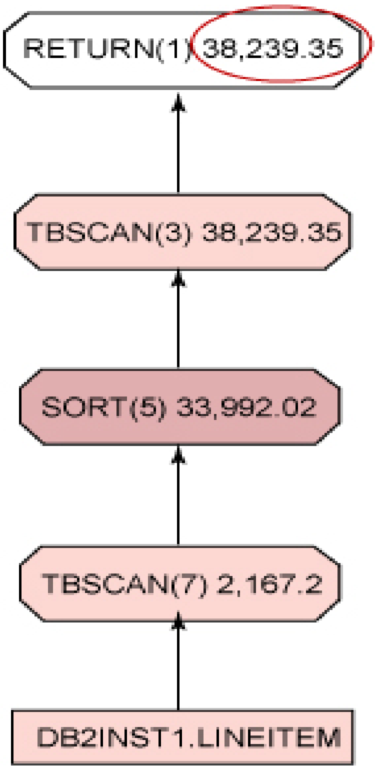
5. Issue the following query (Query_03.txt, which is included in the tutorial's download) by selecting the radio button **Execute and Access Plan**:
   Figure 15. Command Editor



Target: SAMPLE

```
-- ORDER By non index columns
-- VE

SELECT  L_ORDERKEY, L_PARTKEY, L_SUPPKEY, L_LINENUMBER, L_QUANTITY, L_EXTENDEDPRICE, L_DISCOUNT, L_TAX, L_RETURNFLAG, L_LINESTATUS,
        L_SHIPDATE, L_COMMITDATE, L_RECEIPTDATE, L_SHIPINSTRUCT, L_SHIPMODE, L_COMMENT
        from lineitem
        ORDER BY L_SHIPDATE, L_COMMITDATE, L_EXTENDEDPRICE, L_DISCOUNT;
```

   Note the total amount of timerons for the execution of the query:
   Figure 16. Total amount of timerons



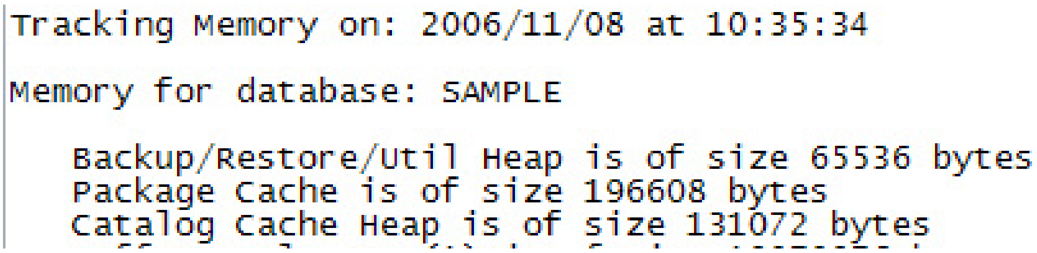
6. When the query is finished, stop the db2mtrk in the command line prompt by pressing the **Ctrl** + **C** keys.

7. Examine the file mtrk_pc by looking for the packcachesz values over execution time:

```
notepad mtrk.pc
```

Note: While 32 pages had been defined in the database configuration, DB2 automatically increased the SortHeap to 48 pages (196608 bytes) on my system:

Figure 17. Memory for database SAMPLE



```
Tracking Memory on: 2006/11/08 at 10:35:34

Memory for database: SAMPLE

   Backup/Restore/Util Heap is of size 65536 bytes
   Package Cache is of size 196608 bytes
   Catalog Cache Heap is of size 131072 bytes
```

8. Reset pckcachesz:

```
db2 update db cfg using PCKCACHESZ -1 IMMEDIATE
```

9. Stop and start the database SAMPLE by opening the DB2 Control Center -- right-click on the database SAMPLE, then STOP and START.

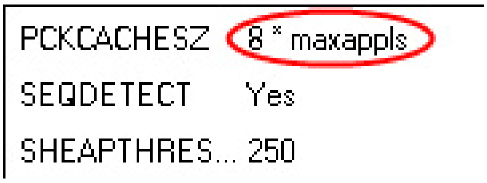
10. Determine the value of the pckcachesz using the DB2 Control Center.

    1. Open the Control Center using the following command:

```
db2cc
```

    2. Right-click on the database node SAMPLE/Configure parameters.

Figure 18. Value of the pckcachesz



PCKCACHESZ	8 * maxappls
SEQDETECT	Yes
SHEAPTHRES...	250

Section 5. Self-tuning memory and LOCKLIST

Introduction

DB2 is a multi-user database product. As users request data, the DB2 locking mechanism attempts to avoid resource conflicts, yet still provide full data integrity. As SQL statements are processed, the transactions may obtain locks. The locks are released when the resource is no longer required at the end of the transaction. The locks are stored in memory on the database server (in a structure known as the locklist). DB2 supports two types of locks -- table locks and row locks. The locking strategy used by DB2 during transaction processing is specified using an isolation level, as defined when binding the application. The database configuration parameter MAXLOCKS defines the percentage of the total locklist permitted to be allocated to a single application. If an application exceeds the percentage of the locklist, as defined by the MAXLOCKS configuration parameter, or many applications connected to the database fill the locklist by acquiring a large number of locks, DB2 will attempt to free memory by obtaining a table lock and releasing row locks.

Key points

The following exercise guides you through the steps of how to enable the locklist for self tuning and to follow the locklist usage over time during a running workload.

Instructions

1. Repeat step 3 from the "Self-tuning memory and SORTHEAP" section, and add the following commands:

```
db2 update db cfg using LOCKLIST 10 IMMEDIATE
db2 update db cfg using MAXLOCKS 98 IMMEDIATE
```

Or edit the script EX2-3.bat (included with the tutorial's download) to include the commands for this exercise, and run the following command:

```
db2 -tvf EX2-3.bat
```

2. Make sure self tuning is enabled and sortheap, sheapthres_shr, locklist, and maxlocks are enabled for self tuning:

```
db2 connect to sample
db2 get db cfg for sample show detail
db2 disconnect current
```

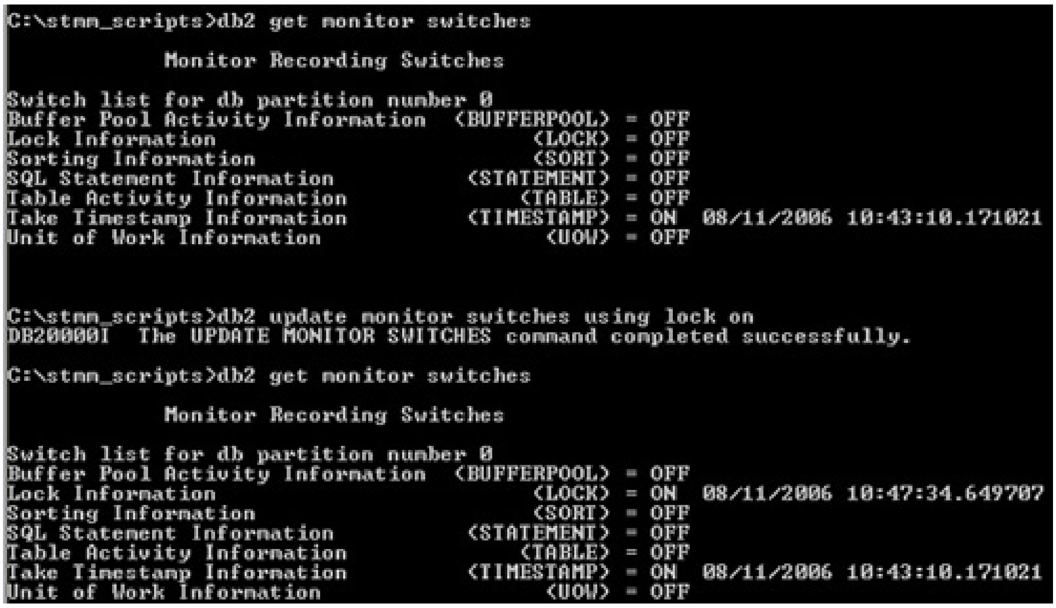
Or execute the following batch script:

```
db2 -tvf EX2-2.bat
```

3. Determine the current setting of the monitor switches, and set snapshot monitor for locklist:

```
db2 get monitor switches
db2 update monitor switches using lock on
```

Figure 19. Command window

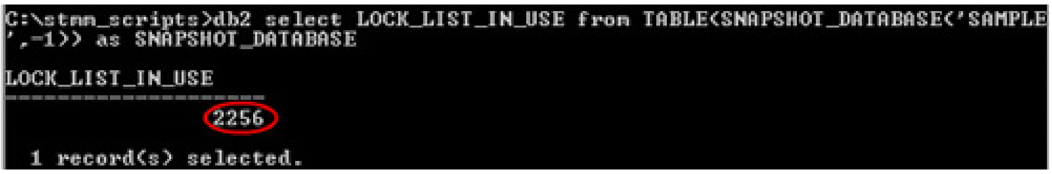


```
C:\stmm_scripts>db2 get monitor switches

            Monitor Recording Switches

Switch list for db partition number 0
Buffer Pool Activity Information  (BUFFERPOOL) = OFF
Lock Information                        (LOCK) = OFF
Sorting Information                     (SORT) = OFF
SQL Statement Information          (STATEMENT) = OFF
Table Activity Information             (TABLE) = OFF
Take Timestamp Information         (TIMESTAMP) = ON  08/11/2006 10:43:10.171021
Unit of Work Information                 (UOW) = OFF


C:\stmm_scripts>db2 update monitor switches using lock on
DB20000I  The UPDATE MONITOR SWITCHES command completed successfully.

C:\stmm_scripts>db2 get monitor switches

            Monitor Recording Switches

Switch list for db partition number 0
Buffer Pool Activity Information  (BUFFERPOOL) = OFF
Lock Information                        (LOCK) = ON  08/11/2006 10:47:34.649707
Sorting Information                     (SORT) = OFF
SQL Statement Information          (STATEMENT) = OFF
Table Activity Information             (TABLE) = OFF
Take Timestamp Information         (TIMESTAMP) = ON  08/11/2006 10:43:10.171021
Unit of Work Information                 (UOW) = OFF
```

4. Determine the current value of the database configuration parameter locklist using the snapshot monitor:

```
db2 select LOCK_LIST_IN_USE
    from TABLE(SNAPSHOT_DATABASE('SAMPLE',-1 ))
    as SNAPSHOT_DATABASE
```

Figure 20. Command window



```
C:\stmm_scripts>db2 select LOCK_LIST_IN_USE from TABLE(SNAPSHOT_DATABASE('SAMPLE
',-1)) as SNAPSHOT_DATABASE

LOCK_LIST_IN_USE
--------------------
                2256

  1 record(s) selected.
```

5. Determine the current locklist value using the DB2 memory tracker. Open a command line prompt, and issue the following command:

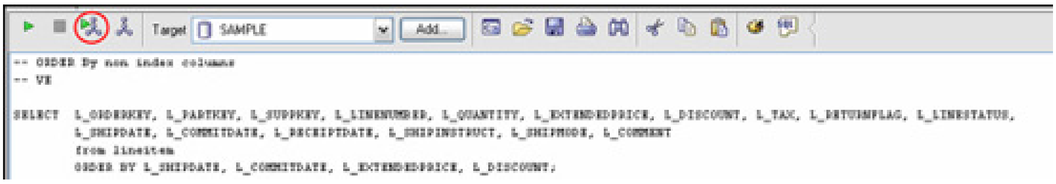
```
db2mtrk -d -v -r 1 > mtrk_ll
```

6. Open the DB2 Command Editor from the Control Center.

7. Connect to the SAMPLE database by clicking on the **Add** button and selecting the database SAMPLE.

8. Issue the following query (Query_03.txt, which is included in the tutorial's download) by selecting the radio button **Execute and Access Plan**:

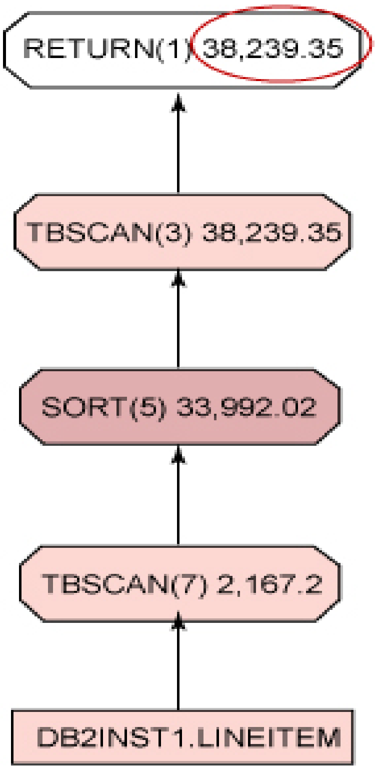
Figure 21. Command Editor



```
-- ORDER By non index columns
-- VE

SELECT  L_ORDERKEY, L_PARTKEY, L_SUPPKEY, L_LINENUMBER, L_QUANTITY, L_EXTENDEDPRICE, L_DISCOUNT, L_TAX, L_RETURNFLAG, L_LINESTATUS,
        L_SHIPDATE, L_COMMITDATE, L_RECEIPTDATE, L_SHIPINSTRUCT, L_SHIPMODE, L_COMMENT
        from lineitem
        ORDER BY L_SHIPDATE, L_COMMITDATE, L_EXTENDEDPRICE, L_DISCOUNT;
```

Note the total amount of timerons for the execution of the query:

Figure 22. Total amount of timerons

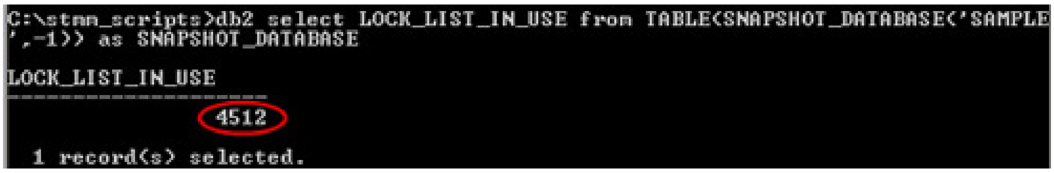


9. When the query is finished, stop the db2mtrk in the command line prompt by pressing the **Ctrl** + **C** keys.

10. Determine the current value of the database configuration parameter locklist using the snapshot monitor:

```
db2 select LOCK_LIST_IN_USE
    from TABLE(SNAPSHOT_DATABASE('SAMPLE',-1 ))
    as SNAPSHOT_DATABASE
```

Figure 23. Command window



```
C:\stmm_scripts>db2 select LOCK_LIST_IN_USE from TABLE(SNAPSHOT_DATABASE('SAMPLE
',-1)) as SNAPSHOT_DATABASE

LOCK_LIST_IN_USE
--------------------
                4512

  1 record(s) selected.
```

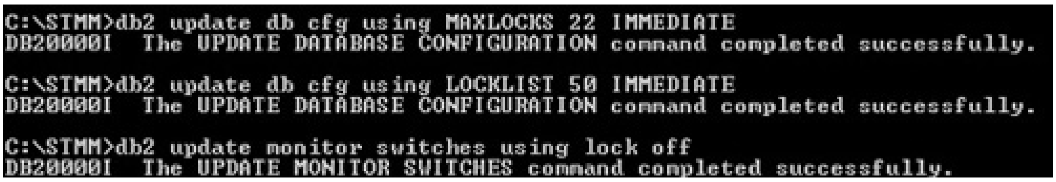
11. Examine the file mtrk_ll by looking for the packcachesz values over execution time:

```
notepad mtrk_ll
```

12. Reset the locklist, maxlocks, and monitor switches:

```
db2 update db cfg using MAXLOCKS 22 IMMEDIATE
db2 update db cfg using LOCKLIST 50 IMMEDIATE
db2 update monitor switches using lock off
```

Figure 24. Command window



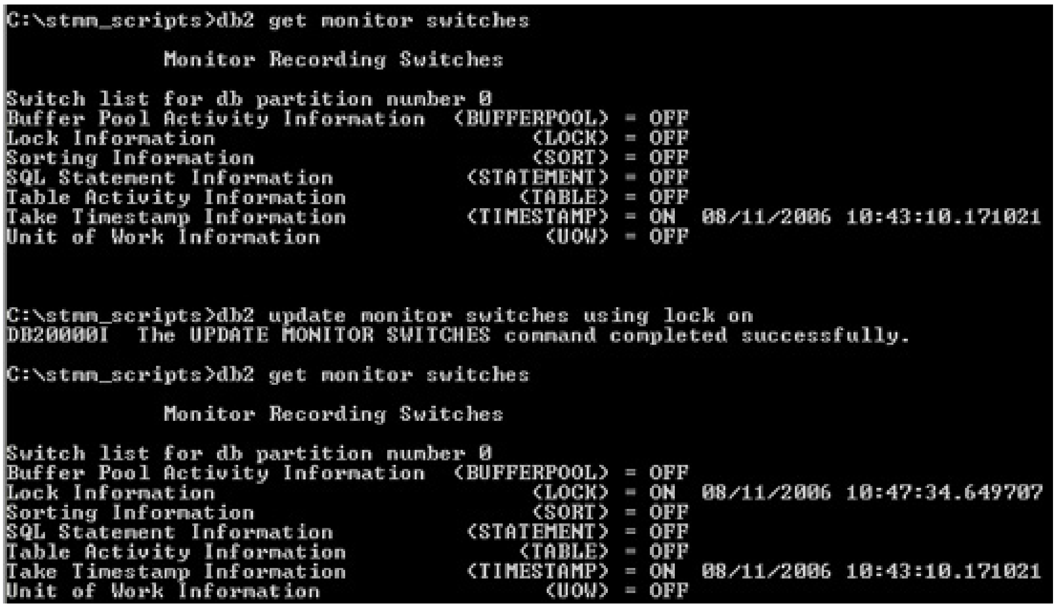
```
C:\STMM>db2 update db cfg using MAXLOCKS 22 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

C:\STMM>db2 update db cfg using LOCKLIST 50 IMMEDIATE
DB20000I  The UPDATE DATABASE CONFIGURATION command completed successfully.

C:\STMM>db2 update monitor switches using lock off
DB20000I  The UPDATE MONITOR SWITCHES command completed successfully.
```

13. Stop and start the database SAMPLE.

14. Determine the values of the locklist, maxlocks, and monitor switches using the following command:

```
db2 get db cfg for sample show detail
db2 get monitor switches
```

Figure 25. Values of the locklist, maxlocks, and monitor switches



```
C:\stmm_scripts>db2 get monitor switches

            Monitor Recording Switches

Switch list for db partition number 0
Buffer Pool Activity Information  (BUFFERPOOL) = OFF
Lock Information                        (LOCK) = OFF
Sorting Information                     (SORT) = OFF
SQL Statement Information          (STATEMENT) = OFF
Table Activity Information             (TABLE) = OFF
Take Timestamp Information         (TIMESTAMP) = ON  08/11/2006 10:43:10.171021
Unit of Work Information                 (UOW) = OFF


C:\stmm_scripts>db2 update monitor switches using lock on
DB20000I  The UPDATE MONITOR SWITCHES command completed successfully.

C:\stmm_scripts>db2 get monitor switches

            Monitor Recording Switches

Switch list for db partition number 0
Buffer Pool Activity Information  (BUFFERPOOL) = OFF
Lock Information                        (LOCK) = ON  08/11/2006 10:47:34.649707
Sorting Information                     (SORT) = OFF
SQL Statement Information          (STATEMENT) = OFF
Table Activity Information             (TABLE) = OFF
Take Timestamp Information         (TIMESTAMP) = ON  08/11/2006 10:43:10.171021
Unit of Work Information                 (UOW) = OFF
```

Section 6. Self-tuning memory and bufferpool size

Introduction

Buffer pools are database objects used to cache data pages in memory. Once a data page is placed in a buffer pool, physical I/O access to disk can be avoided. Buffer pools can be assigned to cache only a particular table space. Buffer pools are a very important tuning area in DB2 performance tuning. The way that you configure the system's real and virtual memory, DB2's internal memory usage, and the DB2 buffer pools greatly influences the performance of the system. The more real memory available, the better the overall performance is. Balance this against the number of processors in the system. Not enough real memory will cause excessive paging in the operating system, which affects all applications, including DB2. The situation can arise where even though a page of DB2 data is sitting in the buffer pool, because the operating system does not have enough memory to keep that page in its real memory, it has to write it out to disk temporarily. This situation can have a severe performance effect on DB2.

Key points

The following exercise guides you through the steps starting with enabling the bufferpool size for self tuning and demonstrating how the buffer pool is automatically tuned by DB2 during a special workload.

Instructions

1. Repeat step 3 from the "Self-tuning memory and SORTHEAP" section,

and add the following commands:

```
db2 alter bufferpool IBMDEFAULTBP IMMEDIATE SIZE 50 AUTOMATIC
db2 update monitor switches using lock on
```

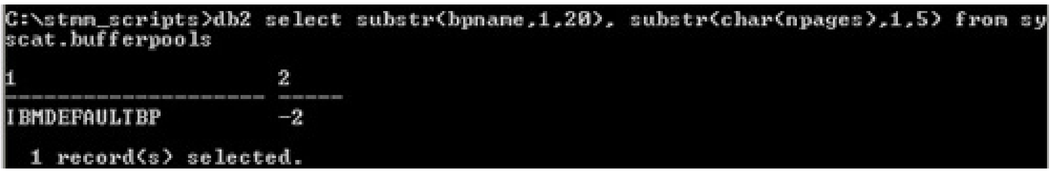
Or edit the script EX2-3.bat (included in the tutorial's download) to include the commands, for this exercise, and run the following command:

```
db2 -tvf EX2-3.bat
```

2. Determine the current size of the buffer pool and if it is enabled for self tuning:

```
db2 select substr(bpname,1,20), substr(char(npages),1,5)
    from syscat.
```

Figure 26. Command window



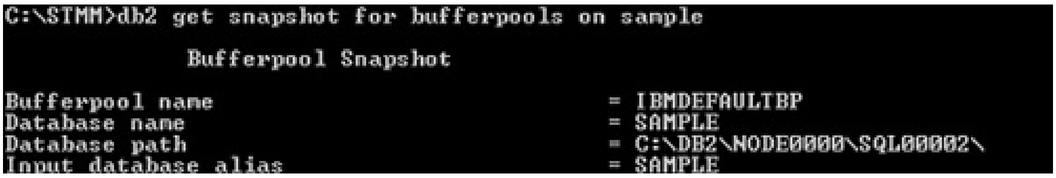
```
C:\stmm_scripts>db2 select substr(bpname,1,20), substr(char(npages),1,5) from sy
scat.bufferpools

1                    2
-------------------- -----
IBMDEFAULTBP         -2

  1 record(s) selected.
```

Note: "-2" indicates automatic bufferpools

3. Determine the current size of the buffer pool IBMDEFAULTBP using the snapshot monitor:

```
db2 get snapshot for bufferpools on sample
```

Figure 27. Command window



```
C:\STMM>db2 get snapshot for bufferpools on sample

                Bufferpool Snapshot

Bufferpool name                              = IBMDEFAULTBP
Database name                                = SAMPLE
Database path                                = C:\DB2\NODE0000\SQL00002\
Input database alias                         = SAMPLE
```

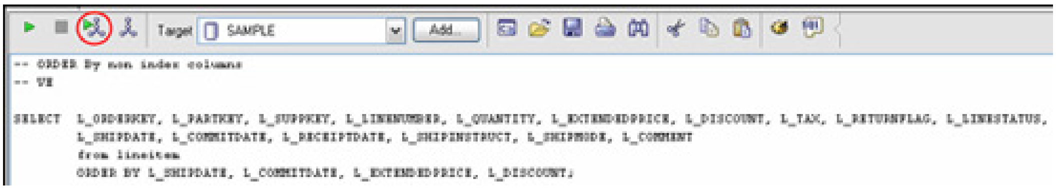
4. Determine the current value of the buffer pool size over time using the DB2 memory tracker:

```
db2mtrk –d –v –r 1 > mtrk_bp
```

5. Open the DB2 Command Editor from the Control Center.

6. Connect to the SAMPLE database by clicking the **Add** button and selecting the database SAMPLE.

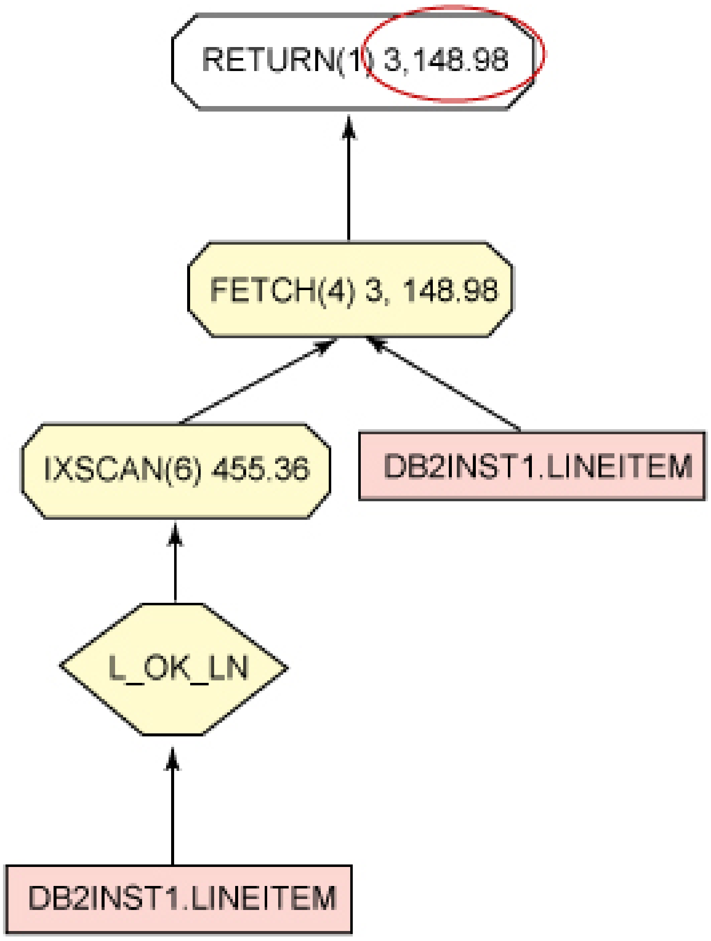
7. Issue the following query (Query_02.txt, which is included in the tutorial's download) by selecting the radio button **Execute and Access Plan**.
Figure 28. Command Editor



```
-- ORDER By non index columns
-- VE

SELECT  L_ORDERKEY, L_PARTKEY, L_SUPPKEY, L_LINENUMBER, L_QUANTITY, L_EXTENDEDPRICE, L_DISCOUNT, L_TAX, L_RETURNFLAG, L_LINESTATUS,
        L_SHIPDATE, L_COMMITDATE, L_RECEIPTDATE, L_SHIPINSTRUCT, L_SHIPMODE, L_COMMENT
        from lineitem
        ORDER BY L_SHIPDATE, L_COMMITDATE, L_EXTENDEDPRICE, L_DISCOUNT;
```

Note the total amount of timerons for the execution of the query:

Figure 29. Total amount of timerons



8. When the query is finished, stop the db2mtrk in the command line prompt by pressing the **Ctrl** + **C** keys.

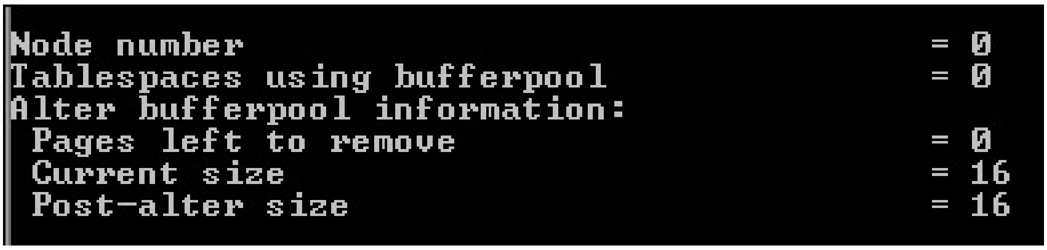
9. Examine the file mtrk_bp file, and look for the buffer pool size values over execution time:

```
notepad mtrk_bp
```

10. Determine the current size of the buffer pool IBMDEFAULTBP using the snapshot monitor:

```
db2 get snapshot for bufferpools on sample
```

Figure 30. Command window

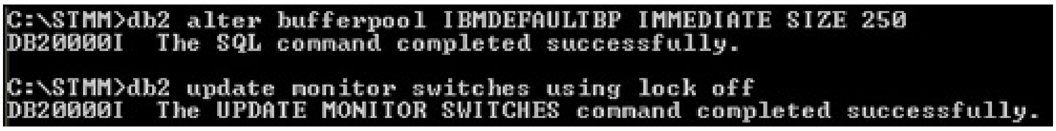


```
Node number                                   = 0
Tablespaces using bufferpool                  = 0
Alter bufferpool information:
 Pages left to remove                         = 0
 Current size                                 = 16
 Post-alter size                              = 16
```

11. Reset the buffer pool size and monitor switch for the buffer pool:

```
db2 alter bufferpool IBMDEFAULTBP IMMEDIATE SIZE 250
db2 update monitor switches using lock off
```

Figure 31. Command window



```
C:\STMM>db2 alter bufferpool IBMDEFAULTBP IMMEDIATE SIZE 250
DB20000I  The SQL command completed successfully.

C:\STMM>db2 update monitor switches using lock off
DB20000I  The UPDATE MONITOR SWITCHES command completed successfully.
```

12. Stop and start the database SAMPLE.

Section 7. Self-tuning memory and health checking

Introduction

IBM has an autonomic computing initiative that is focusing on, creating an IT infrastructure that can self-manage, self-diagnose and self-heal itself. DB2 provides a variety of tools to enable the database to manage itself. Self-managing and resource-tuning (SMART) database technology allows greater automation with respect to configuring, tuning, and managing database operations. The Heath Center is a server-side tool that can monitor the health of a DB2 instance, even without user interaction. When defined thresholds are exceeded (log space is not sufficient, in other words) alerts are activated. Alert notifications can be sent using e-mail or a pager system; alternatively, a preconfigured set of actions can be taken that has been defined in a script or task. The Health Monitor runs by default on the instance you are setting up. It watches all DB2-related data within the instance. The Health Monitor automatically tracks performance in the database environment.

Key points

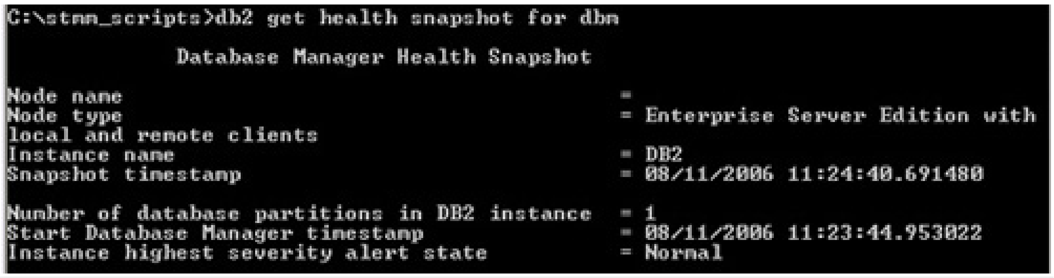
The following exercise guides you through all necessary steps for setting up the health indicators and monitoring the health in your DB2 environment and provides one example of how the DB2 Health Monitor reacts when a problem occurs.

Instructions

1. Determine the current setting of the health indicators on the instance and database level:

```
db2 get health snapshot for dbm
```

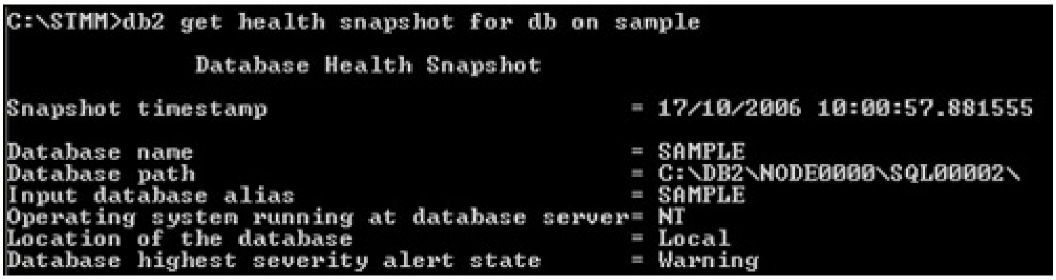
Figure 32. Command window



```
C:\stmm_scripts>db2 get health snapshot for dbm

             Database Manager Health Snapshot

Node name                                      =
Node type                                      = Enterprise Server Edition with
local and remote clients
Instance name                                  = DB2
Snapshot timestamp                             = 08/11/2006 11:24:40.691480

Number of database partitions in DB2 instance  = 1
Start Database Manager timestamp               = 08/11/2006 11:23:44.953022
Instance highest severity alert state          = Normal
```

```
db2 get health snapshot for db on sample
```

Figure 33. Command window



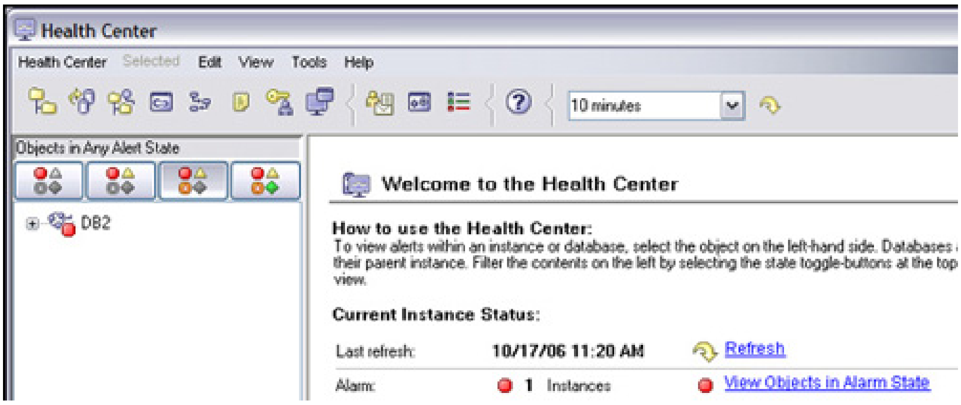
```
C:\STMM>db2 get health snapshot for db on sample

             Database Health Snapshot

Snapshot timestamp                             = 17/10/2006 10:00:57.881555

Database name                                  = SAMPLE
Database path                                  = C:\DB2\NODE0000\SQL00002\
Input database alias                           = SAMPLE
Operating system running at database server= NT
Location of the database                       = Local
Database highest severity alert state          = Warning
```

2. Define how the DBA should be informed when alerts occur:

   a. Issue the following command:

```
db2hc
```

Figure 34. Health Center



Welcome to the Health Center

How to use the Health Center:
To view alerts within an instance or database, select the object on the left-hand side. Databases
their parent instance. Filter the contents on the left by selecting the state toggle-buttons at the top
view.

Current Instance Status:

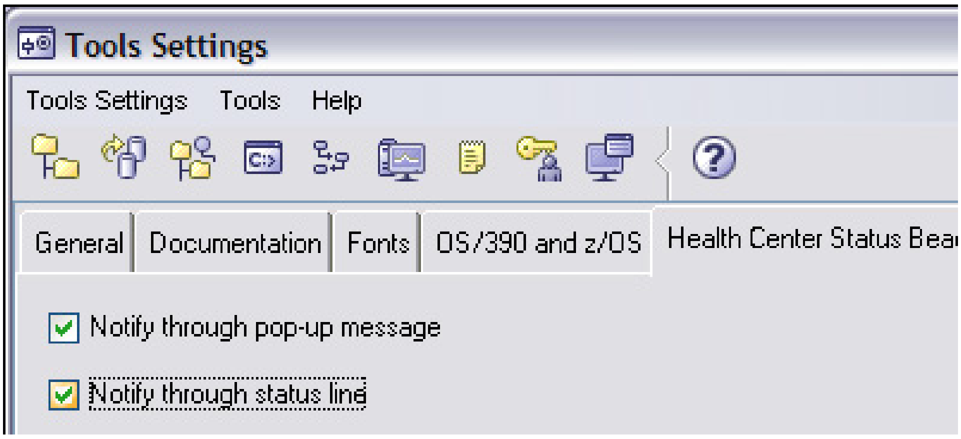
Last refresh: 10/17/06 11:20 AM

Alarm: 1 Instances

   b. Select the menu item **Tools/Tools Settings** in the DB2 Health Center.

c. Change to the tab **Health Center Status Beacon**, and mark both check boxes.

Figure 35. Health Center



☑ Notify through pop-up message

☑ Notify through status line

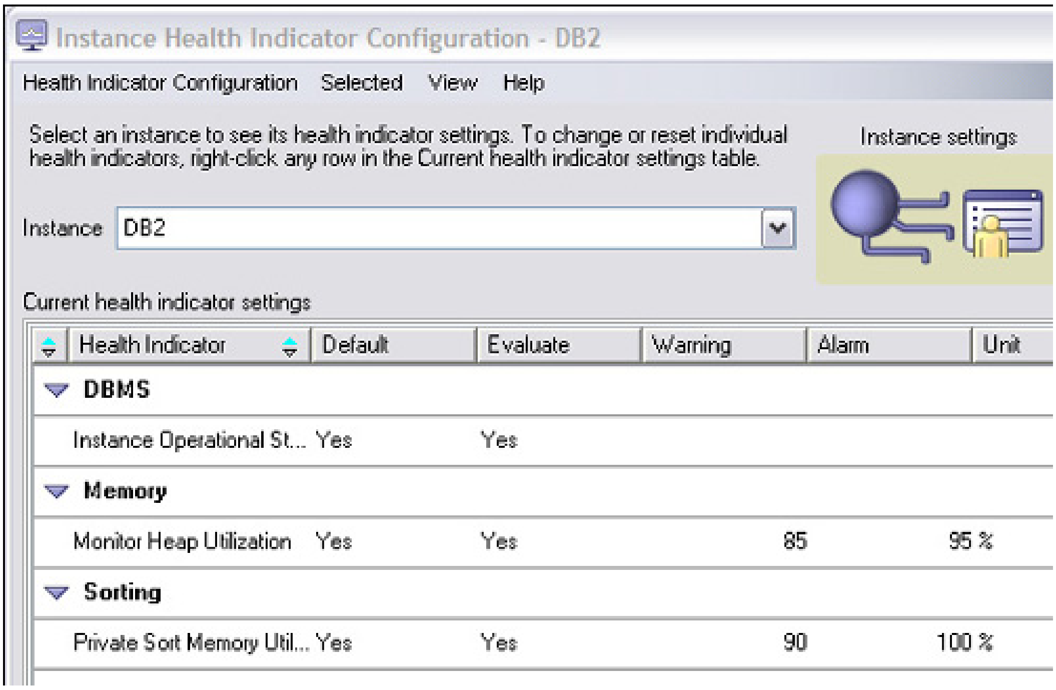
3. Customize the health indicators using the DB2 Health Center

a. Select menu item **Health Center/Configure/Health Indicator Settings**.

b. Click on the**Instance Settings** button.

c. Select **DB2** in the Instance entry field:

Figure 36. Health Center


Instance Health Indicator Configuration - DB2
Health Indicator Configuration
Selected
View
Help
Instance settings

Select an instance to see its health indicator settings. To change or reset individual health indicators, right-click any row in the Current health indicator settings table.

Instance DB2

Current health indicator settings

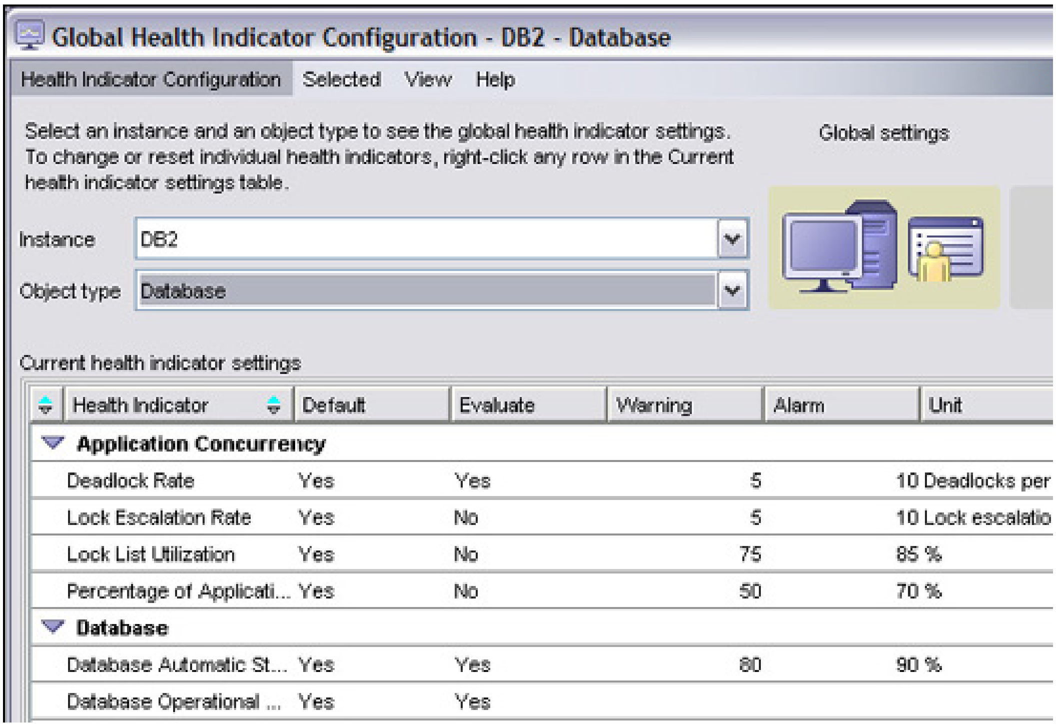
Health Indicator	Default	Evaluate	Warning	Alarm	Unit
DBMS					
Instance Operational St...	Yes	Yes			
Memory					
Monitor Heap Utilization	Yes	Yes	85	95	%
Sorting					
Private Sort Memory Util...	Yes	Yes	90	100	%

Note: The private sort memory is enabled for health checking. The DBA will get a warning from the DB2 Health Monitor if 90% of the private sort memory is used and an alarm if it reaches 100%.

a. Close the Health Indicator Configuration window.

b. Click on the **Global Settings** button in the Health Indicator Configuration Launchpad, and select the instance **DB2** and the object type **Database**.

Figure 37. Health Center



Global Health Indicator Configuration - DB2 - Database

Select an instance and an object type to see the global health indicator settings. To change or reset individual health indicators, right-click any row in the Current health indicator settings table.

Instance: DB2

Object type: Database

Current health indicator settings

Health Indicator	Default	Evaluate	Warning	Alarm	Unit
Application Concurrency					
Deadlock Rate	Yes	Yes	5	10	Deadlocks per
Lock Escalation Rate	Yes	No	5	10	Lock escalatio
Lock List Utilization	Yes	No	75	85	%
Percentage of Applicati...	Yes	No	50	70	%
Database					
Database Automatic St...	Yes	Yes	80	90	%
Database Operational ...	Yes	Yes			

Note: All current health indicator settings are listed. Except for the "Package and Catalog Caches" and "Workspaces" parameters, all listed parameters are checked by default for system health. By double-clicking on a parameter, the values of the thresholds can be changed or an action can be defined, which is executed when a threshold is reached.

a. Close the Global Health Configuration Indicator Configuration window.

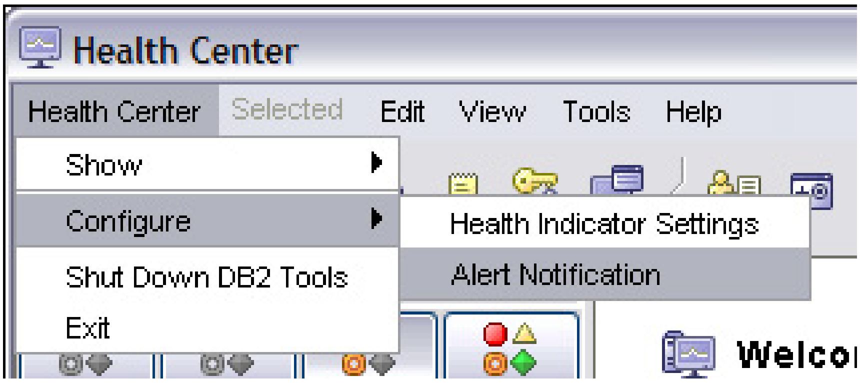
b. Click on the **Object Settings** button in the Health Indicator Configuration Launchpad.

c. Click on the **Object** select list, select the database **SAMPLE**, and press the **OK** button.
Note: All current health indicator settings are listed. Except for the "Package and Catalog Caches" and "Workspaces" parameters, all listed parameters are checked by default for system health. By double-clicking on a parameter, the values of the thresholds can be changed or an action can be defined which is executed when a threshold is reached.

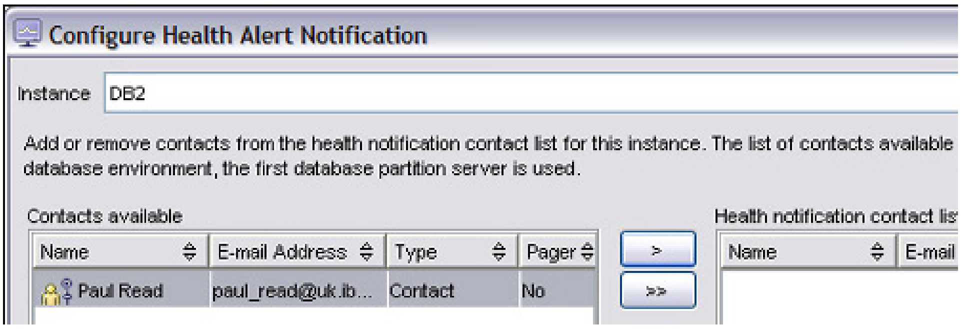
d. Close the Health Indicator Configuration Launchpad.

4. Define who should be notified:

a. Select menu item **Health Center/Configure/Alert Notification**:
Figure 38. Health Center



b. Select the instance **DB2**:
Figure 39. Health Center


Configure Health Alert Notification

Instance DB2

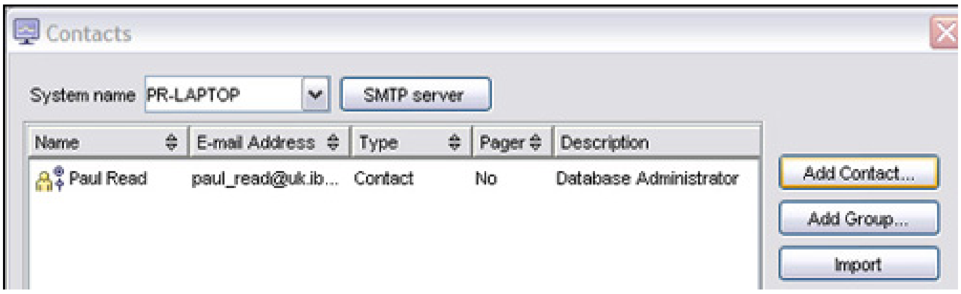
Add or remove contacts from the health notification contact list for this instance. The list of contacts available database environment, the first database partition server is used.

Contacts available

Name	E-mail Address	Type	Pager
Paul Read	paul_read@uk.ib...	Contact	No

>
>>
Health notification contact lis
Name
E-mail

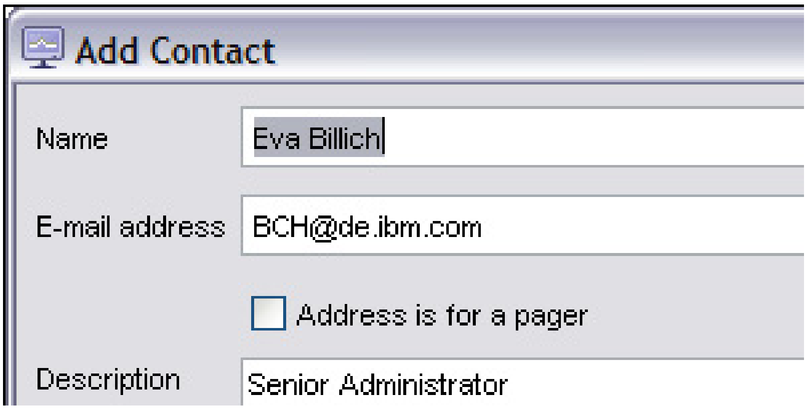
c. Click on the **Manage Contacts** button:
Figure 40. Health Center



System name PR-LAPTOP

Name	E-mail Address	Type	Pager	Description
Paul Read	paul_read@uk.ib...	Contact	No	Database Administrator

d. Add a contact to the contact list by clicking on the **Add Contact** button:
Figure 41. Health Center



Add Contact

Name: Eva Billich

E-mail address: BCH@de.ibm.com

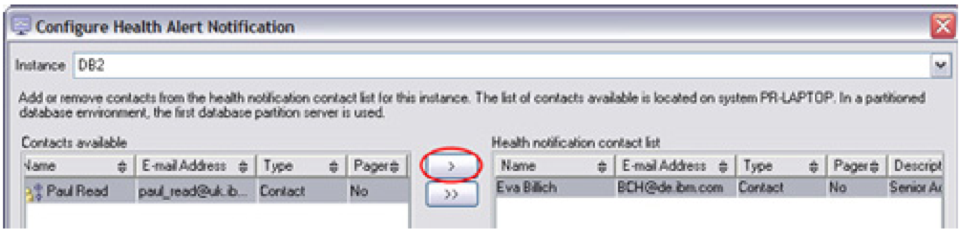
□ Address is for a pager

Description: Senior Administrator

e. Close the "Add Contact" and the "Manage Contacts" windows

f. Select the contact in the "Contacts available" window, and add it to the "Health notification contact list" window by pressing the **>** button:

Figure 42. Health Center



Configure Health Alert Notification

Instance: DB2

Add or remove contacts from the health notification contact list for this instance. The list of contacts available is located on system PR-LAPTOP. In a partitioned database environment, the first database partition server is used.

Contacts available

Name	E-mail Address	Type	Pager
Paul Read	paul_read@uk.ib...	Contact	No

Health notification contact list

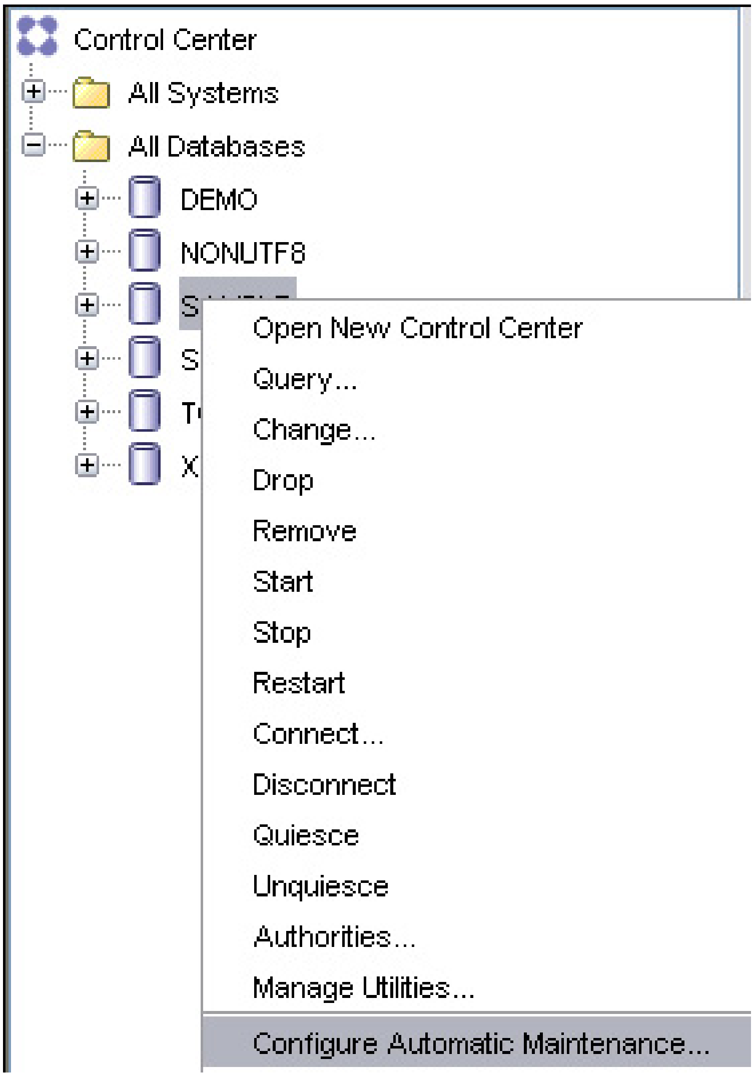
Name	E-mail Address	Type	Pager	Descript
Eva Billich	BCH@de.ibm.com	Contact	No	Senior Ad

g. Press the **OK** button in the "Configure Health Alert Notification" window.

5. Configure automatic maintenance:

a. Right-click on the SAMPLE database note, and select menu item **Configure Automatic Maintenance**:

Figure 43. Control Center

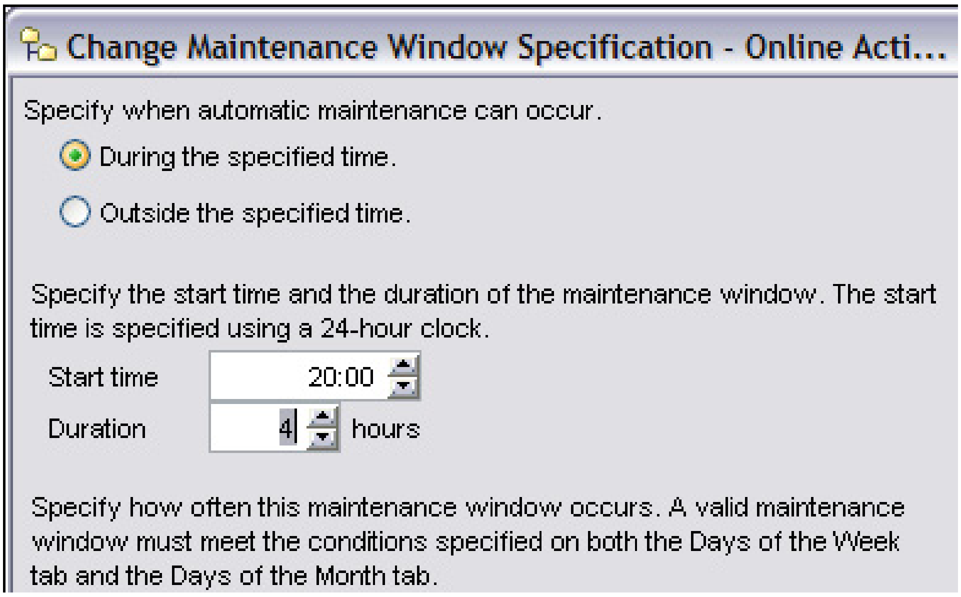


b. On the "1. Introduction" panel, click the **Next** button.

c. n the "2. Type" panel, select the **Next**.

d. In the "3. Timing" panel, click on the **Change** button, and specify the time frame 20:00 for 4 hours.

Figure 44. Control Center



Change Maintenance Window Specification - Online Acti...

Specify when automatic maintenance can occur.

● During the specified time.

○ Outside the specified time.

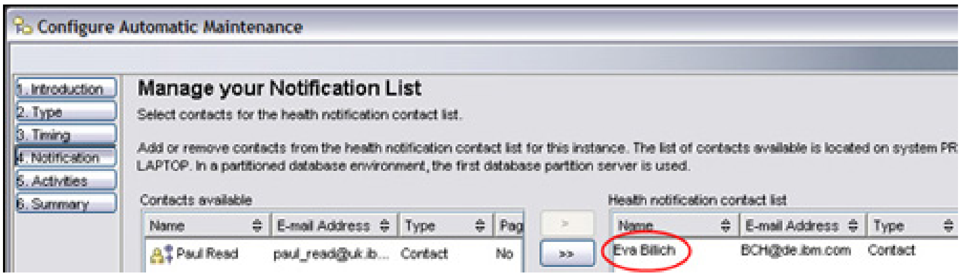
Specify the start time and the duration of the maintenance window. The start time is specified using a 24-hour clock.

Start time 20:00

Duration 4 hours

Specify how often this maintenance window occurs. A valid maintenance window must meet the conditions specified on both the Days of the Week tab and the Days of the Month tab.

e. Click the **OK** button, and then click the **Next** button.
In the "4. Notification" panel, your previous contact will be listed on the right side:
Figure 45. Control Center



Configure Automatic Maintenance

Manage your Notification List

Select contacts for the health notification contact list.

Add or remove contacts from the health notification contact list for this instance. The list of contacts available is located on system PR LAPTOP. In a partitioned database environment, the first database partition server is used.

Contacts available

Name	E-mail Address	Type	Pag
Paul Read	paul_read@uk.ib...	Contact	No

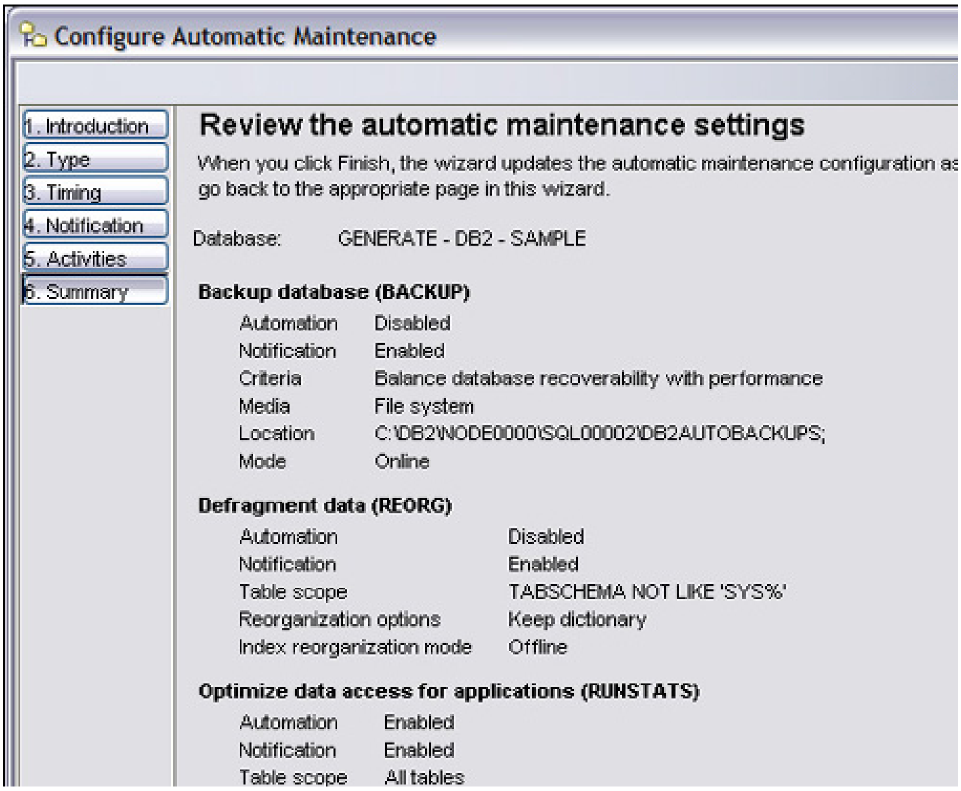
Health notification contact list

Name	E-mail Address	Type
Eva Billich	BCH@de.ibm.com	Contact

f. Click the>**Next** button.

g. In the "5. Activities" panel, click the **Next** button.

h. Review the automatic maintenance settings, and press the **Finish** button:
Figure 46. Control Center



Review the automatic maintenance settings

When you click Finish, the wizard updates the automatic maintenance configuration as
go back to the appropriate page in this wizard.

Database: GENERATE - DB2 - SAMPLE

Backup database (BACKUP)

Automation	Disabled
Notification	Enabled
Criteria	Balance database recoverability with performance
Media	File system
Location	C:\DB2\NODE0000\SQL00002\DB2AUTOBACKUPS;
Mode	Online

Defragment data (REORG)

Automation	Disabled
Notification	Enabled
Table scope	TABSCHEMA NOT LIKE 'SYS%'
Reorganization options	Keep dictionary
Index reorganization mode	Offline

Optimize data access for applications (RUNSTATS)

Automation	Enabled
Notification	Enabled
Table scope	All tables

A DB2 Message Box appears with the message, "Command completed successfully."

6. Deactivate DB2 self tuning, issuing the following command:

```
db2 update db cfg using SELF_TUNING_MEM OFF
```

Section 8. Conclusion

This tutorial is based on the IBM DB2 9 self-tuning memory management feature. You have completed exercises with hands-on experience in the following areas:

- Activating self-tuning memory
- Determining which memory consumers are enabled for self tuning
- Determining the current size of DB2 memory usage
- Automatic tuning in a production environment
- Deactivating self tuning memory

With the exercises, you worked with the DB2 self-tuning memory feature to manage available memory resources dynamically. It distributed the resources among several memory consumers including sorts, the package cache, the lock list, and buffer pools.

Downloads

Description	Name	Size	Download method
Sample SQL scripts and queries for this tutorial	stmm_zipped.zip	2408KB	HTTP

Information about download methods

Resources

Learn

- DB2 Express-C: Learn about the no-charge version of DB2 Express Edition for the community.
- "IBM DB2 'Viper' revs XML engine" (*InfoWorld*, May 2006): Find a description of many of the new features of DB2 9.
- Read more articles about DB2 9 on developerWorks.
- developerWorks Information Management zone: Expand your skills on IBM Information Management products.
- developerWorks DB2 for Linux, UNIX, and Windows resource page: Read articles and tutorials, and link to documentation, support, learning resources, and more.
- Stay current with developerWorks technical events and Webcasts

Get products and technologies

- Download IBM DB2 9 evaluation versions to try out the features described in this article.
- Build your next development project with IBM trial software, available for download directly from developerWorks.

Discuss

- Participate in the discussion forum for this content.
- Participate in the DB2 9 discussion forum
- Check out developerWorks blogs, and get involved in the developerWorks community.

About the authors

Paul Read
Paul Read is a Product Introduction Manager in the Global BetaWorks team based in the IBM Lab in Hursley, Hampshire, United Kingdom. He is responsible for Beta and Early Support Programs for Information Management products on Linux, UNIX and Windows.

Eva Angelika Billich
Eva Billich is a Senior DB2 Consultant in the Business Partner Technical Enablement Team based in the IBM Lab in Boeblingen, Germany. She is responsible for enabling

strategic IBM business partners in EMEA to accelerate their application projects with DB2 on Linux, Unix and Windows from project start until "Go to market."