

Enable Automatic Storage for Your SAP Database and Table Spaces



Applies to:

SAP systems using IBM DB2 9.7

For more information, visit the [Landscape Design and Architecture homepage](#).

Summary

This document describes how to take advantage of the new feature in DB2 9.7 to convert an existing non-automatic storage database to use automatic storage in a SAP/DB2 environment. Two alternative methods to fully enable automatic storage are described in this paper. SAP customers running on IBM DB2 9.7 can use this document as a step by step guide to enable automatic storage for their existing non-automatic storage enabled system.

Author: Cong Lin

Company: IBM Canada

Created on: 10 May 2009

Author Bio



Cong Lin is a member of the IBM SAP Integration and Support Centre at the IBM Toronto Lab. Her current activities include testing of SAP applications with DB2 LUW and helping customers with problem analysis and troubleshooting. She is also a customer advocate, providing support for large customer accounts running SAP and DB2 LUW.

Table of Contents

1. Overview	3
2. Enable Automatic Storage for Database and Table Spaces	3
2.1 Preparation	3
2.1.1 Check the current status of your database	3
2.1.2 Check the Current Status of Your Table Spaces	4
2.1.3 Prepare the New Storage Paths	5
2.2 Enable Automatic Storage for Database	5
2.3 Enable Automatic Storage for Table Spaces Online	5
2.4 Enable Automatic Storage for Table Spaces via Redirect Restore	11
2.5 Take a Database Backup	12
3. Other Consideration	13
Related Content	13
Disclaimer and Liability Notice	14

1. Overview

Automatic storage is a DB2 feature initially introduced in DB2 V8.2 FP9 and intended to make storage management easier. Rather than managing storage at the table space level using explicit container definitions, storage is managed at the database level and the responsibility of creating, extending and adding containers is taken over by the database manager.

In order to take fully advantage of this feature, automatic storage must be enabled at both database level and table space level. In prior releases, a database could only be enabled for automatic storage within the CREATE DATABASE command when it was first created. In V8.2 FPs starting from FP9, automatic storage could be enabled for a database at database creation time using the automatic storage yes clause. Starting from V9.1, the database manager creates all databases as automatic storage databases by default (unless **automatic storage no** was specified when issuing the CREATE DATABASE command). Databases that are enabled for automatic storage have a set of one or more storage paths associated with it. Table spaces created in an automatic storage enabled database could either be normal DMS/SMS table spaces or automatic storage enabled table spaces. Table spaces defined to use automatic storage will have their containers assigned by DB2 based on those storage paths.

In one of the DB2 9.7 enhancements, existing databases that were not enabled for automatic storage can now be converted to use automatic storage. Likewise, existing DMS table spaces can now be converted to use automatic storage.

2. Enable Automatic Storage for Database and Table Spaces

2.1 Preparation

Depending on which DB2 and SAP release your initial SAP database was created, your current database could be in one of following three statuses:

- Both the database and table spaces are already automatic storage enabled
- Automatic storage enabled database with DMS table spaces not managed by automatic storage.
- Neither database nor the table spaces are automatic storage enabled.

2.1.1 Check the current status of your database

The value "0" for "Number of Storage Paths" in db2pd or database snapshot output indicates a non automatic storage database. Following example shows a database not yet enabled for automatic storage:

```
saplp4: db2ax4> db2pd -db AX4 -storagepaths
Database Partition 0 -- Database AX4 -- Active -- Up 0 days 00:49:18
Database Storage Paths:
Number of Storage Paths    0
```

Following example shows a database enabled for automatic storage, with 4 storage paths:

```
saplp4:db2bwl 58> db2pd -db BWL -storagepaths

Database Partition 0 -- Database BWL -- Active -- Up 0 days 00:00:11
Database Storage Paths:
Number of Storage Paths      4

Address      PathID  PathState PathName
0x000004002887C720 0      InUse    /db2/BWL/sapdata1
0x000004002887C420 1      InUse    /db2/BWL/sapdata2
0x000004002887C120 2      InUse    /db2/BWL/sapdata3
0x000004002887BE20 3      InUse    /db2/BWL/sapdata4
```

Following example shows the same information with database snapshot:

```
saplp4:db2bwl 60> db2 get snapshot for database on BWL | grep "automatic storage"

Number of automatic storage paths      = 4
```

2.1.2 Check the Current Status of Your Table Spaces

The table space snapshot has the information for each individual table space to show if it is automatic storage enabled or not. Following example shows the automatic storage status for table space SYSCATSPACE using a tablespace snapshot:

```
saplp4:db2bwl 60> db2 get snapshot for tablespaces on BWL

Tablespace Snapshot

First database connect timestamp      = 02/13/2009 14:16:37.520216
...
Tablespace name                       = SYSCATSPACE
Tablespace ID                         = 0
...
Using automatic storage              = No
```

2.1.3 Prepare the New Storage Paths

If all the table spaces in your database are already enabled for automatic storage, you don't need to do anything. Otherwise, please have the new storage paths created if you wish to enable the automatic storage for your database and table spaces. Make sure there is enough disk space in the new storage paths. For example, if the size of your database is about 1 TB, then make sure you have 1 TB free space on the new storage path(s). If you have a multi-partition database, make sure the new storage paths exist on all partitions.

2.2 Enable Automatic Storage for Database

Note: If your database is already automatic storage enabled, please skip this step and go directly to either section 2.3 or 2.4 to convert the table spaces.

Please issue following SQL statement to convert a non-automatic storage database. The paths specified in the SQL statement will be used as the storage paths for the database. Once the transaction is committed the database will be automatic storage enabled. This step should be fast.

```
ALTER DATABASE <DatabaseName> ADD STORAGE ON <Path1> [,<Path2>,...,<Pathn>]
```

Example:

```
db2 alter database ax4 add storage on '/db2/AX4/sapdata7', '/db2/AX4/sapdata8';
```

Note: If you receive SQL1051N error at this step, make sure instance owner has all permission on the path(s) you specified.

2.3 Enable Automatic Storage for Table Spaces Online

There are two ways to enable automatic storage at table space level. The online method could be a better choice for you if you cannot afford to take the database offline. As a tradeoff, this method involves a rebalance operation that takes time to move data from the non-automatic storage containers to the new automatic storage containers.

1. Convert all DMS table spaces to use automatic storage

Following SQL statement will convert a DMS tablespace to automatic storage:

```
ALTER TABLESPACE <TablespaceName> MANAGED BY AUTOMATIC STORAGE
```

You can run the following command to find all non-automatic DMS table spaces and create batch file to convert all none-automatic DMS table spaces to automatic storage:

```
db2 -x "select 'alter tablespace ' || CHR(34) || TBSP_NAME || CHR(34) || ' managed by automatic storage;'
from SYSIBMADM.SNAPTbsp where TBSP_USING_AUTO_STORAGE != 1 and TBSP_TYPE = 'DMS' and
TBSP_CONTENT_TYPE in ('ANY', 'LARGE')" > alt_tbs.sql;
```

Execute: `db2 -tvf alt_tbs.sql`

After this step, all the DMS tablespaces become automatic storage table spaces. This will be reflected in the table space snapshot. It is important to note that all containers that existed prior to executing this statement will still exist but they can no longer be altered. Any new growth will automatically be handled by automatic storage on a new stripe set with new containers that are defined using the database storage paths.

Following is an example on how a table space snapshot looks like after this step:

```

Tablespace name           = SYSCATSPACE
Tablespace ID             = 0
Tablespace Type           = Database managed space
Using automatic storage   = Yes
...
Number of containers      = 6
Container Name            = /db2/AX4/sapdata1/NODE0000/SYSCATSPACE.container000
Container Name            = /db2/AX4/sapdata2/NODE0000/SYSCATSPACE.container001
Container Name            = /db2/AX4/sapdata3/NODE0000/SYSCATSPACE.container002
Container Name            = /db2/AX4/sapdata4/NODE0000/SYSCATSPACE.container003
Container Name            = /db2/AX4/sapdata7/db2ax4/NODE0000/AX4/T0000000/C0000000.CAT
Container Name            = /db2/AX4/sapdata8/db2ax4/NODE0000/AX4/T0000000/C0000001.CAT
...
Table space map:

Range      Stripe      Stripe  Max    Max    Start  End    Adj.  Containers
Number     Set        Offset  Extent Page  Stripe Stripe
[ 0]      [ 0]        0       95995 191991 0      23998 0     4 (0,1,2,3)
[ 1]      [ 1]      23999   97017 194035 23999 24509 0     2 (4,5)

```

There are six containers. The first four are existing containers and the last two are new containers created and managed by the database automatically. The table space map shows two stripe sets, the first one is for the existing data in existing containers and the second one is the new stripe set which will handle the future growth in this table space.

2. Rebalance the table spaces and move data off the old containers to new ones

There is no need to do this step right away. However to have all regular or large table spaces rebalanced will improve the I/O parallelism. To rebalance the containers, issue the following statement for each DMS table space:

```
ALTER TABLESPACE <TablespaceName> REBALANCE;
```

Though the command line for this SQL statement will return right away, the job will run asynchronous in the background. Since rebalancing will move all data in the tablespace it will have an impact on your system performance, please plan this step carefully and try to run it when system is not very busy.

In our test, moving data from 4 existing containers of a 20GB table space to two new automatic storage containers took about 2hour 30min. The time needed to rebalance a 20GB table space on your system

might vary in your particular environment. Please start with a relative small table space and use the result as a reference to estimate how long it will take to rebalance your large table spaces. See the next steps on how to check the progress for table space rebalancing.

Alternatively, you could also use the throttling utility to limit the performance impact of rebalancing on your system.

The database manager configuration parameter `util_impact_lim` sets the limit on the impact throttled utilities can have on the overall workload of the machine. By default, parameter `util_impact_lim` is set to 10, which means all utilities combined can have no more than a 10% average impact upon the total workload as judged by the throttling algorithm. Value 100 indicates no throttling.

Once `util_impact_lim` is set to a value (0-99), you could use `SET UTIL_IMPACT_PRIORITY` command to throttle a table space rebalancing process. To do this you need obtain the utility ID with the `LIST UTILITIES` command and set the priority to a value. Here is the command:

```
SET UTIL_IMPACT_PRIORITY FOR <utility_id> TO <priority>;
```

Here is an example:

```

augzburg2:db2ax4 68> db2 list utilities show detail

ID                = 4
Type              = REBALANCE
Database Name     = AX4
Partition Number  = 0
Description       = Tablespace ID: 0
Start Time        = 05/28/2009 09:43:43.303254
State            = Executing
Invocation Type   = User
Throttling:
  Priority      = Unthrottled
Progress Monitoring:
  Estimated Percentage Complete = 8
  Total Work                   = 49333 extents
  Completed Work                = 3884 extents
  Start Time                   = 05/28/2009 09:43:43.303275

```

```
augzburg2:db2ax4 70> db2 list utilities show detail

ID                = 4
Type              = REBALANCE
Database Name     = AX4
Partition Number  = 0
Description       = Tablespace ID: 0
Start Time        = 05/28/2009 09:43:43.303254
State             = Executing
Invocation Type   = User
Throttling:
  Priority       = 50
Progress Monitoring:
  Estimated Percentage Complete = 17
  Total Work           = 49333 extents
  Completed Work       = 8150 extents
  Start Time          = 05/28/2009 09:43:43.303275
```

Note: In DPF environment, a rebalance should be initiated on every database partition in which the tablespace resides.

3. Check the status for tablespace rebalancing

To monitor the status, you could either run the “db2list utilities show detail” or take a table space snapshot.

```

Db2 list utilities show detail

Here is an example output:

ID                = 6219
Type              = REBALANCE
Database Name     = AX4
Partition Number  = 0
Description       = Tablespace ID: 37
Start Time        = 04/27/2009 21:37:37.932471
State             = Executing
Invocation Type   = User
Throttling:
  Priority         = Unthrottled

Progress Monitoring:
  Estimated Percentage Complete = 15
  Total Work        = 22366 extents
  Completed Work    = 3318 extents
  Start Time        = 04/27/2009 21:37:37.932500

ID                = 6216
Type              = REBALANCE
Database Name     = AX4
Partition Number  = 0
Description       = Tablespace ID: 36
Start Time        = 04/27/2009 21:37:37.634176
State             = Executing
Invocation Type   = User
Throttling:
  Priority         = Unthrottled

Progress Monitoring:
  Estimated Percentage Complete = 4
  Total Work        = 103440 extents
  Completed Work    = 4404 extents
  Start Time        = 04/27/2009 21:37:37.634202

```

In above example, the table spaces with ID 37 and 36 are still being rebalanced.

When the rebalance is done for a tablespace, the “Rebalancer Mode” in tablespace snapshot for this table space shows “No Rebalancing”. Otherwise you will see “DMS rebalancer is active” under that table space in table space snapshot.

In following example, after rebalance table space < SYSCATSPACE > it only has 2 containers (both managed by automatic storage) and one stripe set in the table space map. All data which existed on the old containers has been moved to the new automatic storage containers.

```

Tablespace name          = SYSCATSPACE
Rebalancer Mode         = No Rebalancing
Number of containers    = 2
Container Name          =
/db2/AX4/sapdata7/db2ax4/NODE0000/AX4/T0000000/C0000000.CAT
Container Name          =
/db2/AX4/sapdata8/db2ax4/NODE0000/AX4/T0000000/C0000001.CAT

Table space map:
Range      Stripe  Stripe  Max    Max    Start  End    Adj.  Containers
Number    Set    Offset Extent Page   Stripe Stripe
[ 0]      [ 0]    0      57269 114539 0      28634 0      2 (0,1)

```

The rebalance events are logged in the db2diag.log. You could calculate the time actual used for rebalance a table space based on these log entries.

```

2009-04-07-17.59.31.665152-240 E1342A392    LEVEL: Info
PID           : 1298682          TID          : 4379       PROC        : db2sysc 0
INSTANCE      : db2ax4          NODE         : 000
EDUID        : 4379            EDUNAME      : db2rebal (AX4) 0
FUNCTION      : DB2 UDB, buffer pool services, sqlb_rev_rebalance, probe:10
MESSAGE       : ADM6058I Rebalancer for table space "AX4#ES700I" (ID "21") was
                started.
...

2009-04-07-19.24.48.030229-240 E6339A396    LEVEL: Info
PID           : 1298682          TID          : 4379       PROC        : db2sysc 0
INSTANCE      : db2ax4          NODE         : 000
EDUID        : 4379            EDUNAME      : db2rebal (AX4) 0
FUNCTION      : DB2 UDB, buffer pool services, sqlb_rebalance, probe:2876
MESSAGE       : ADM6062I Rebalance for table space "AX4#ES700I" (ID "21") has been completed.

```

4. Create new automatic temp table spaces to replace the existing ones

Temporary table spaces and table spaces managed by system (SMS) cannot be converted to use automatic storage via "alter tablespace". To convert temporary table spaces to use automatic storage, you'll need to create a new automatic storage managed temporary table space and drop the old one.

In a standard SAP database, there are two SMS temp tablespaces: PSAPTEMP16 and SYSTOOLSTMPSPACE. Following is an example script to rename the existing temp table spaces, recreate the new temp table spaces with original names and drop the old temp table spaces. Please run the recreate_tmptbs.sql first and then drop_oldtmp.sql.

Content of recreate_tmptbs.sql:

```
rename tablespace PSAPTEMP16 to oldTEMP16;
rename tablespace SYSTOOLSTMPSPACE to oldTOOLSTMPSPACE;
create temporary tablespace PSAPTEMP16
  in nodegroup IBMTEMPGROUP pagesize 16k extentsize 2 prefetchsize automatic
  no file system caching
  dropped table recovery off;
create user temporary tablespace SYSTOOLSTMPSPACE
  in nodegroup IBMCATGROUP pagesize 16k extentsize 2 prefetchsize automatic
  no file system caching
  dropped table recovery off;
```

Content of drop_oldtmp.sql

```
drop tablespace oldTEMP16;
drop tablespace oldTOOLSTMPSPACE ;
```

Note: If you couldn't drop one of your old temp table spaces, it might still be used. Try to drop it at a later time.

2.4 Enable Automatic Storage for Table Spaces via Redirect Restore

This method doesn't need to rebalance the table spaces since the data would be automatically managed by containers created by the database at the time of restore. However, this is an offline process.

1. Take a database backup.
2. Generate the restore script with the RESTORE command from the backup
3. Enable automatic storage for all table spaces using a redirected restore.

Prepare your redirect restore script and execute it.

Following is an example:

```
db2 restore db <dbname> from <backup_location> redirect;  
db2 set tablespace containers for <tablespace_id1> using automatic storage;  
db2 set tablespace containers for <tablespace_id2> using automatic storage;  
...  
db2 set tablespace containers for <tablespace_idn> using automatic storage;  
db2 restore database <dbname> continue;  
db2 rollforward db <dbname> to end of the logs;
```

Note: You could generate a template script for redirect restore and obtain all table space IDs using “RESTORE DB ... REDIRECT GENERATE SCRIPT” command. Here is an example:

```
db2 restore db <dbname> from <backup_location> taken at <timestamp> redirect generate script restore.clp
```

Please modify the restore.clp to include the “set tablespace container for tbs_id using automatic storage” for each table spaces listed in restore.clp.

2.5 Take a Database Backup

Now your database and table spaces are fully enable to use automatic storage. Make a database backup to have a clean starting point in case you need to restore you database in the future.

3. Other Consideration

- In this document, it is suggested that all of your table spaces in your SAP database should be enabled for automatic storage to simplify your storage management tasks. However, it is possible to have only some of the DMS tablespaces enabled for automatic storage should you wish to do so for a specific reason.
- Restrictions: There is no way to disable automatic storage for a database after it has been created or enabled.
- Restrictions: For partitioned databases, the same set of storage paths must be used on each database partition (unless database partition expression is used).

Related Content

[IBM DB2 Database for Linux, UNIX, and Windows Information Center](#)

For more information, visit the [Landscape Design and Architecture homepage](#).

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.