



Best Practices

Cost Reduction Strategies with DB2

Mark Wilding
*DB2 Cloud Computing and DB2 Health
Advisor Service*

Burt Vialpando
Senior IT Specialist

Saurabh Jain
DB2 Evangelist

Last updated: 2009-10-23

Executive summary 4

DB2 databases and total cost of ownership..... 5

 Standardization 5

 Business strategy 5

 Cost reduction features and technology 6

 DB2 database cost reduction features at a glance 6

Planning your system..... 7

Installing DB2 systems 7

Configuring DB2 systems 8

 The DB2 Configuration Advisor 8

 Detecting the DB2 Configuration Advisor11

 Improving buffer pool usage using prefetchers and page cleaners 11

 Detecting the prefetcher and page cleaner automatic settings12

The DB2 Design Advisor 12

 Tips for using the DB2 Design Advisor13

 DB2 Design Advisor for partitioned databases14

Maintaining DB2 systems 15

 The self-tuning memory manager 15

 Detecting the self-tuning memory manager16

 Frequently asked questions17

 Automatic storage..... 18

 Storage management: Automatic resizing19

 Detecting automatic storage use for a table space.....19

 Frequently asked questions20

 Automatic maintenance 21

 Using automatic maintenance.....21

 Maintaining statistics using automatic RUNSTATS and real-time statistics 22

 Detecting automatic statistics updates25

Hardware cost reduction strategies 25

Troubleshooting cost reduction strategies 26

Conclusion 28

Further reading..... 29

Contributors.....	29
Notices	30
Trademarks	31

Executive summary

The DB2 development and planning teams have long recognized the need for low maintenance costs. This recognition has led to investments in many industry-leading features such as the Configuration Advisor, the Design Advisor, and autonomic features, such as the self-tuning memory manager, that can help reduce the human costs associated with DB2 maintenance. Hardware cost-reducing features such as data compression and support for virtualization are also part of the lost-cost evolution.

Not only does the DB2 database incorporate features designed to help lower costs, it continuously remains at the top of industry performance benchmarks. The DB2 Performance Benchmarks page at <http://www.ibm.com/software/data/highlights/db2leads.html> documents this performance leadership.

With industry leading features designed to help reduce costs, performance leadership, and an open standard, the DB2 database is a leader in the low-cost evolution.

This document takes a holistic look at cost reduction strategies and provides guidance on a wide range of topics including:

- Planning
- Installation
- Setup and configuration
- Maintenance and operations
- Reducing hardware costs
- Reducing troubleshooting costs

Some sections include pointers to other best-practice papers that describe the specific technology or best practice applicable to the topic. Other sections highlight key considerations for the topic.

Lastly, this document makes it easy to address cost reduction strategies in a corporate standard by giving you convenient methods to detect features designed to help reduce costs in a DB2 environment. Detection is important to help ensure that the cost reduction features can be monitored across hundreds or thousands of systems using remote monitoring technology. This helps extend your cost reduction strategies enterprise-wide.

DB2 databases and total cost of ownership

This paper is written primarily for database administrators (DBAs) and focuses on rapid cost reduction practices for the DB2 database. The goal of the paper is to free up DBAs to put more focus, time, and effort into moving the business forward. In the spirit of lowering costs, we'll skip a long introduction and jump right in.

There are three key aspects of enterprise cost reduction that we'll discuss first:

- IT practices including standardization
- Business strategies such as master data management that can consolidate and/or streamline your business
- Cost reduction features and technology

We'll highlight each briefly and then go into depth on the last topic as it is the primary focus of this paper.

Standardization

Standardization is important to reduce the complexity and uniqueness of individual systems and DB2 instances. Uniqueness is only valuable to a business if it drives business value. Uniqueness for any other reason creates unnecessary complexity. For example, having 100 ad-hoc systems and DB2 instances can increase maintenance costs since each system has its own schedule, its own unique characteristics, and its own unique challenges or opportunities. One hundred standardized systems, on the other hand, have a much better chance of sharing schedules, principles, best practices, and known improvements. Enterprise standardization provides increased consistency, improved predictability, and reduced risk, all of which can help to reduce your business costs. For warehouse requirements, an IBM Balanced Warehouse™ provides a standardized platform for use across an organization.

Business strategy

There are several business strategies such as consolidation, virtualization, master data management (MDM), and streamlined development. IBM has industry leading technology for each of these including its MDM solution, Data Studio, and a wide range of technologies to streamline your business. Ask your local sales representative for more information on how these can help save your business money.

For a good standardization practice in your business, IBM has consultants for both business and technology who are happy to help streamline your operation to help you achieve lower costs. To contact either group, refer to IBM Global Technology Services at <http://www.ibm.com/services/us/index.wss>.

Cost reduction features and technology

The DB2 database is a performance leader in the industry. Coupled with industry-leading features designed to help reduce costs such as compression, the DB2 database is the best choice to help you reduce the costs of your IT department. Both performance and features such as compression help you make the most of your hardware capital costs. Refer to the following best practices for pointers on how to leverage the world renowned performance of a DB2 database and how it can help you achieve hardware cost savings:

- Writing and Tuning Queries for Optimal Performance - <http://www.ibm.com/developerworks/db2/bestpractices/querytuning/>
- Tuning and Monitoring Database System Performance - <http://www.ibm.com/developerworks/db2/bestpractices/systemperformance/>
- Deep Compression - <http://www.ibm.com/developerworks/db2/bestpractices/deepcompression/>
- Improving Data Server Utilization and Management through Virtualization - <http://www.ibm.com/developerworks/db2/bestpractices/virtualization/>

Besides hardware costs, human costs are also a significant portion of the average IT budget. The more that can be automated, the less time has to be spent on mundane tasks. This frees up time to focus on moving the business forward (not to mention giving some DBAs much needed breathing space). DBAs and businesses alike are realizing that teams who take advantage of cost saving technology can reinvest their time into positive growth for the company that can have a real impact on the company’s bottom line, as well as a DBA’s value to a company. In fact, the benefit is two-fold: lower day-to-day costs and increased capacity for moving the business forward. What else would you want your best people doing?

DB2 database cost reduction features at a glance

The following table lists the DB2 features designed to help organizations reduce costs and indicates which release each feature is in

	DB2 UDB Version 8.2	DB2 Version 9.1	DB2 Version 9.5	DB2 Version 9.7
Configuration Advisor	X	X	X	X
Design Advisor	X	X	X	X
Self-tuning memory manager		X	X	X
Automatic backup	X	X	X	X
Automatic statistics collection	X	X	X	X
Workload management			X	X
Data compression		X	X	X

Automatic storage	X	X	X	X
Automatic reorganization	X	X	X	X
Non-root install			X	X
pureXML®		X ¹	X	X
Low footprint threaded architecture			X	X
Self-tuning backup	X	X	X	X
Self-tuning load	X	X	X	X
Query rewrite	X	X	X	X
Database administration tools (Data Studio)			X	X

This paper provides guidance on several of the key features mentioned in the table above.

Planning your system

Capacity planning is a great place to start implementing cost reduction practices. A system that is properly sized for a workload and its anticipated growth experiences fewer problems than an undersized system. An oversized system does not cause problems, but it wastes valuable hardware resources. Striking a balance is an important goal and one that comes with a set of best practices.

If the system and instance is up and running, you can collect various Key Performance Indicators (KPIs) using a technology such as a DB2 Performance Expert historical archive. DB2 Performance Expert can give projections about the growth of a workload over time and the impact this growth has on the system resources as well as the optimal tuning for the DB2 system. On the other hand, if you're starting out fresh with no prior system metrics to leverage, it's best to talk to an expert who has experience with other systems to help you with the initial sizing and picking the necessary hardware. The Tuning and Monitoring Database System Performance best practice at <http://www.ibm.com/developerworks/db2/bestpractices/systemperformance/> covers some aspects of capacity planning under "Hardware Configuration".

Installing DB2 systems

For best practices for installation, see the Deploying IBM DB2 Products best practice at <http://www.ibm.com/developerworks/db2/bestpractices/deployingdb2/>. The paper covers topics such as install planning, installation methods, and mass deployment from subject matter experts at IBM. Mass deployment methods are particularly important for enterprise environments with hundreds or thousands of DB2 instances.

¹ pureXML in 9.1 only supports UNICODE databases

Configuring DB2 systems

This section outlines the key DB2 cost reduction features that help to configure and tune your instance and database. It includes best practices for using each feature and how to detect whether they are used. The latter is to make it easier to monitor a standard practice and identify potential cost reduction opportunities across your organization

The DB2 Configuration Advisor

The extremely diverse DB2 customer base makes it impossible for any hard-coded default configuration to be suitable for more than a tiny percentage of systems. The DB2 database includes a feature called the Configuration Advisor that can be used to point and click through a simple set of steps to configure a new or existing database in a minimal amount of time. The Configuration Advisor can help achieve dramatic performance improvements over the default configuration values.

When run automatically on a new database, the Configuration Advisor makes some assumptions such as the mix between OLTP and DSS workloads on the database. If you run the Configuration Advisor after a database is created and has been in use for some time, you can supply information about the workload which is now known, which results in an even better configuration.

The Configuration Advisor does not continuously change configuration parameters to respond to a changing workload. If you want this behavior, use the self-tuning memory manager (STMM) instead (this is described later in this paper). However, the Configuration Advisor does tune some parameters that the STMM does not. The following table shows which parameters are tuned by the configuration advisor and which are tuned by STMM:

Parameter	Type	Configuration Advisor	STMM
AGENT_STACK_SZ	DBM	✓	
ASLHEAPSZ	DBM	✓	
FCM_NUM_BUFFERS	DBM	✓	
INTRA_PARALLEL	DBM	✓	
MAX_QUERYDEGREE	DBM	✓	
MAXAGENTS	DBM	✓	
NUM_POOLAGENTS	DBM	✓	
NUM_INITAGENTS	DBM	✓	
PRIV_MEM_THRESH	DBM	✓	
RQRIOLBK	DBM	✓	
SHEAPTHRES	DBM	✓	
APP_CTL_HEAP_SZ	DB	✓	
APPGROUP_MEM_SZ	DB	✓	
APPLHEAPSZ	DB	✓	

CATALOGCACHE_SZ	DB	✓	✓①
CHNGPGS_THRESH	DB	✓	
DBHEAP	DB	✓	✓①
DFT_DEGREE	DB	✓	
DFT_EXTENT_SZ	DB	✓	
DFT_PREFETCH_SZ	DB	✓	
DFT_QUERYOPT	DB	✓	
LOCKLIST	DB	✓	✓
LOGBUFSZ	DB	✓	
LOGFILSIZ	DB	✓	
LOGPRIMARY	DB	✓	
LOGSECOND	DB	✓	
MAXAPPLS	DB	✓	
MAXLOCKS	DB	✓	✓
MINCOMMIT	DB	✓	
NUM_IOCLEANERS	DB	✓	
NUM_IOSERVERS	DB	✓	
PCKCACHESZ	DB	✓	✓
SOFTMAX	DB	✓	
SORTHEAP	DB	✓	✓
STMTHEAP	DB	✓	
STAT_HEAP_SZ	DB	✓	
UTIL_HEAP_SZ	DB	✓	✓①
SELF_TUNING_MEM	DB	✓	
AUTO_RUNSTATS	DB	✓	
AUTO_MAINT	DB	✓	
AUTO_TBL_MAINT	DB	✓	
SHEAPTHRES_SHR	DB	✓	✓
DATABASE_MEMORY	DB		✓
(Buffer pool sizes)	Buffer pool	✓	✓

①The DB2 database will tune the amount of free database memory that can be used when any of these heaps run out of memory

Starting with DB2 Version 9, AUTOCONFIGURE (the command line invocation of the Configuration Advisor utility) is activated by default for a new database. You can also run the AUTOCONFIGURE command to use the Configuration Advisor on an existing database.

Note: The Configuration Advisor does not run automatically during upgrades so you need to manually use it if desired.

Research shows that non-automated or non-default features are used 20 times less than automated and default features². If this holds true for your organization, manually running the Configuration Advisor after a database is created is unlikely to happen without a corporate standard and encouragement. This could result in many untuned systems or having to spend extra time tuning each system manually. Make sure that your organization knows the value of running the Configuration Advisor and that the recommendations are considered at least initially.

Here are some rules of thumb for using the Configuration Advisor:

- If you use the Data Partitioning Feature, disable AUTOCONFIGURE with the registry variable. Each node should have the same tuning settings. You can run AUTOCONFIGURE on one node intentionally to get suggested tuning for that node and then replicate those settings manually over the other nodes (partitions).
- If you do not run AUTOCONFIGURE intentionally during database creation of a brand new database, go through the Configuration Advisor GUI when the database is relatively new as soon as possible. The GUI is a nice walkthrough to remind you of the things you need to know about your database regarding parameter tuning. Eventually, you should write the answers down about the ten questions about your database presented in the GUI so that you can schedule a task to do this automatically for you later.
- Once the database has run with a representative workload, or once it is in production, use a scheduled task to run the AUTOCONFIGURE command with the APPLY NONE keywords once a month. Make sure you use all of the ten input keywords to this command so that the scheduled task reminds you and other DBAs about what this database is being tuned for. Since many DBAs inherit databases by default from others, a monthly AUTOCONFIGURE task run against each production database gives those new to the environment a chance to know what that database is expected to be doing. This does not actually make configuration changes to the database but gives a reasonable set of suggestions for configuration changes. Review the changes and apply the ones with the largest benefit at the appropriate time.
- Set the AUTOCONFIGURE command ISOLATION parameter to the highest isolation used by any of the queries in your applications using that database. This affects things like the **locklist** configuration parameter that are related to concurrency, so setting this conservatively by choosing the highest isolation level used is recommended. The STMM can always adjust the **locklist** configuration parameter later if the **locklist** configuration parameter is set to AUTOMATIC.
- The BP_RESIZABLE parameter determines if the AUTOCONFIGURE command is allowed to suggest changes to the sizes of your buffer pools. Setting BP_RESIZABLE to YES will recommend the best initial sizes for each buffer pool. The STMM can always adjust these later if they are set to automatic. The exceptions to allowing AUTOCONFIGURE and STMM changing your buffer pools are databases using DPF or databases that an ISV has required you do not change.

² Making DB2 Products Self-Managing: Strategies and Experiences - <http://www.cs.brown.edu/courses/cs227/archives/2008/Papers/IEEE-DataEngineeringBulletin/Lohman.pdf>

For more information about the Configuration Advisor, see the DB2 Information Center topic on Configuration Advisor at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.dbobj.doc/doc/c0052481.html>.

Detecting the DB2 Configuration Advisor

To detect whether the Configuration Advisor is active (on DB2 Version 9.1 and above), check the DB2_ENABLE_AUTOCONFIG_DEFAULT registry variable. As long as the DB2_ENABLE_AUTOCONFIG_DEFAULT registry variable is not set to NO, the Configuration Advisor is invoked by default during a database creation even if the AUTOCONFIGURE keyword is not specified. In scripts, you can check for AUTOCONFIGURE APPLY NONE, which can be used during a CREATE DATABASE command to turn off the Configuration Advisor. If AUTOCONFIGURE is not turned off with the registry variable, using APPLY NONE in a script runs the AUTOCONFIGURE utility and produce a recommendation report; however, the parameter and buffer pool recommendations are not applied.

If you want to disable the configuration advisor, you can either set a registry variable:

```
db2set DB2_ENABLE_AUTOCONFIG_DEFAULT=NO
```

Or, during CREATE DATABASE, you can specify "AUTOCONFIGURE APPLY NONE"

```
db2 CREATE DB db_name AUTOCONFIGURE APPLY NONE
```

Note: While the STMM automatically tunes important memory parameters, it does not provide all the configuration options that are related to the performance of your environment. Refer to the self tuning memory section for more details on STMM.

Improving buffer pool usage using prefetchers and page cleaners

The prefetch size was once set to a constant default value that would likely be incorrect for all but a very specific environment. As of DB2 Version 8.2, the prefetch size is set automatically to the following algorithm:

```
prefetch size = (number of containers) X
                (number of physical disks per container) X
                extent size
```

This makes the value much more dynamic and accurate for a wide range of configurations. There is usually no need to tune this further.

In versions prior to DB2 Version 9, you had to manually set the number of prefetchers and page cleaners. If the number was too low or high, performance would suffer.

Starting with DB2 Version 9, the number of prefetchers (NUM_IOSERVERS) and page cleaners (NUM_IOCLEANERS) is set automatically during database activation by the configuration parameter value of AUTOMATIC (the default value). There is a lot of documentation available on prefetchers (IOSERVERS) and page cleaners (IOCLEANERS), so details of what each does and how they work will not be described here. For information, see the following topics in the DB2 Information Center:

- num_iocleaners - Number of asynchronous page cleaners configuration parameter - <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.config.doc/doc/r0000332.html>
- num_ioservers - Number of I/O servers configuration parameter - <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.config.doc/doc/r0000333.html>

Excluding any manual, in-depth tuning of these parameters, NUM_IOCLEANERS and NUM_IOSERVERS are set to values that are much better than any default. Using the automatic settings for the prefetchers and page cleaners is a great option for a corporate cost-reduction standard.

Detecting the prefetcher and page cleaner automatic settings

To detect the prefetcher and page cleaner settings, use the GET DATABASE CONFIGURATION command:

```
db2 get database configuration for db_name show detail
```

Look for NUM_IOCLEANERS and NUM_IOSERVERS in the output:

```
...
Number of asynchronous page cleaners (NUM_IOCLEANERS) = AUTOMATIC(1)
Number of I/O servers (NUM_IOSERVERS) = AUTOMATIC(3)
...
```

When the value for both is set to AUTOMATIC, the number of page cleaners and prefetchers is selected automatically on database activation.

The DB2 Design Advisor

You can use the DB2 Design Advisor to improve workload performance. When you provide the Design Advisor with the set of SQL statements in a workload, it suggests indexes, Materialized Query Tables (MQTs), clustering dimensions, or data partitions to create for a complex workload. You can have the Design Advisor implement its recommendations immediately or later.

The Design Advisor can be used with a new database, or an existing one. For a new database, the Design Advisor can reduce the amount of time you need to spend setting up or improving performance. For an existing database, in many situations, you can potentially dramatically increase performance.

If you are designing a single-partition database, you can use the Design Advisor to generate design alternatives in a test environment. You can also use it to evaluate indexes, MQTs, multidimensional clustering (MDC) tables, or database partitioning strategies that you have generated.

Tips for using the DB2 Design Advisor

Use the Design Advisor to create recommendations from your test system. If you do not want to run the Design Advisor against your production system, run it against your test system instead. The Design Advisor needs three things to do its calculations:

- The database structures that currently exist
- The workload that goes against these structures
- The statistics.

If your test environment contains all the same database structures as your production environment, you can run the Design Advisor against it. However, because your test environment is usually not completely filled with data, it is unlikely that the test statistics will be the same as the production statistics, even if you do a RUNSTATS on everything in the test environment. You can either mimic the production statistics (or copy them from your production system) then run the Design Advisor on your test system. To do this, use the following DB2LOOK utility command to mimic the production statistics.

```
db2look -d db_name -a -m -c;
```

This command gives you the appropriate UPDATE commands to run against the updateable views in your test database. After updating the views, run the Design Advisor against your test database. The Design Advisor then gives you the same recommendations as if you ran it against the production database. You can then implement the Design Advisor recommendations on your production database.

There is one exception: test tables for which you want MDC recommendations. For the Design Advisor to be able to make MDC recommendations, the test tables must have a good representation of data, even if you copy production statistics to the test environment. Without a good sampling of data for a table, statistics are not enough to recommend a possible MDC value. To ensure that the Design Advisor can make the best MDC recommendations for specific tables, try to populate them fully (if possible, using production data).

- Get MQT recommendations for improving workload performance instead of refresh performance

The Design Advisor does not recommend incrementally refreshed MQTs because it does not consider the refresh of the MQT to be part of the workload. This observation also applies to indexes recommended on MQTs, and for the same reason. The index is for the performance of the MQT and the workload using it, not necessarily for the refresh of the MQT.

- You must determine if you want to create staging tables for incrementally refreshed MQTs. You will also have to determine if the refresh will be adversely affected by recommended indexes.
- Starting in DB2 Version 9.5, the Design Advisor can also accept input from DB2 Workload Manager. Simply capture specific workloads of interest using COLLECT ACTIVITY DATA WITH DETAILS or COLLECT ACTIVITY DATA WITH DETAILS AND VALUES options and load the results into the Design Advisor using the following command:

```
db2advis -d db_name -wlm DB2ACTIVITIES
```

For more information, refer to the DB2 Information Center topic Importing activity information into the Design Advisor at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/index.jsp?topic=/com.ibm.db2.luw.admin.wlm.doc/doc/t0052477.html>.

DB2 Design Advisor for partitioned databases

If you are designing a partitioned database, you can use the Design Advisor to:

- Determine the partitioning strategy before loading data into the database.
- Assist in migrating from a single-partition database to a multiple-partition database.
- Assist in migrating from another database product to a multiple-partition DB2 database.

After your database is set up, you can use the Design Advisor to:

- Help improve the performance of a particular statement or workload.
- Help improve general database performance, using the performance of a sample workload as a gauge.
- Help improve performance of the most frequently executed queries, for example, as identified by the Activity Monitor, application snapshots, or application administrative views.
- Determine how to optimize the performance of a new key query.

- Respond to Health Center or Data Studio Administration Console recommendations regarding shared-memory utility or sort-heap problems in a sort-intensive workload.
- Find objects that are not used in a workload.

For information on running the Design Advisor, see the DB2 Information Center topic Using the Design Advisor at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.pdf.doc/doc/t0021648.html>.

Include the Design Advisor as part of your cost reduction standard as it can help you greatly reduce the amount of time required to design database objects and improve workload performance.

Maintaining DB2 systems

This section describes the features and practices that you can use to potentially reduce the cost that is related to the ongoing maintenance and operation of the instance and database.

The self-tuning memory manager

Whether you are junior or a senior DBA, the self-tuning memory manager (STMM) can help save you hours or days of tuning time by leveraging the expertise provided by some of the best DB2 performance experts.

The STMM is one of the most advanced self-tuning technologies in the industry. Unlike the Configuration Advisor (which is only invoked at database creation time or by the user), the STMM can adapt to actual workloads and changes in the environment and adjust the DB2 memory configuration accordingly. The Configuration Advisor and the STMM are complementary tools. The Configuration Advisor tunes all performance related parameters (including many that are not related to memory), but does so only when requested. The STMM tunes the biggest memory areas of the DB2 database, and does it on an ongoing basis, but it does not address other performance related parameters.

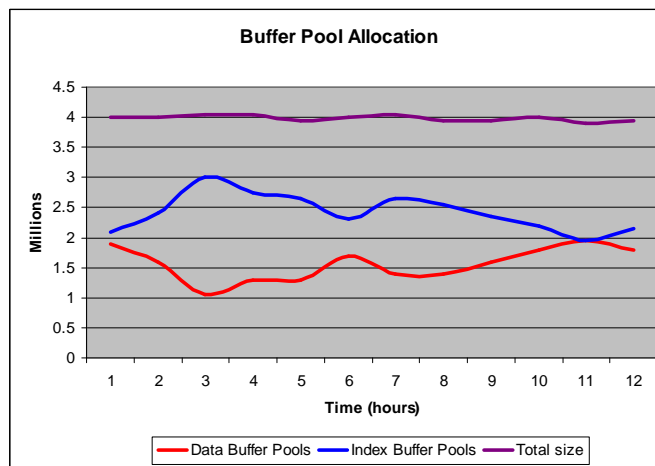
Self-tuning memory is especially valuable for workloads that are dynamic or difficult to predict. The STMM is always working to improve memory utilization which frees you from having to make frequent changes and also from closely monitoring the tuning in the first place.

Self-tuning memory works so well that it has been used in recent benchmarks to help tune the database for top performance. The STMM was designed by some of the top performance tuning experts at the IBM Toronto Lab. In DB2 Version 9, the STMM is switched on by default for new single-partition databases. For upgrades from DB2 UDB Version 8, it remains off unless manually switched on. For instructions on enabling the

STMM after upgrading, see the developerWorks® article at <http://www.ibm.com/developerworks/data/library/techarticle/dm-0709saraswatipura/index.html>.

In partitioned database environments, the STMM is off by default. Self-tuning memory management has some special considerations in this type of environment. For example, the STMM treats all database partitions as homogeneous. The STMM can be effective if all the database partitions are similar or when it is enabled for a specific set of database partitions that are similar. For more information about STMM in a partitioned database environment, see the DB2 Information Center topic Using self-tuning memory in partitioned database environments at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/index.jsp?topic=/com.ibm.db2.luw.admin.perf.doc/doc/c0021628.html>.

This following table shows how the STMM dynamically tunes the DB2 database as the workload changes, all while keeping within a given amount of memory. Doing this ongoing tuning manually is not practical, which is why self-tuning memory can be a great way to both save time and to improve performance.



Detecting the self-tuning memory manager

Use the GET DATABASE CONFIGURATION command to detect the STMM:

```
db2 get database configuration for db_name
```

Look for the following entry:

```
Self tuning memory (SELF_TUNING_MEM) = ON
```

In addition, two or more of the STMM configuration parameters must be set to AUTOMATIC in order for the STMM to be enabled.

If you need to disable the STMM, you can do so in part or completely. For more information, see the DB2 Information Center topic Disabling self-tuning memory at

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.pdf.doc/doc/t0021634.html>.

If you still want to hand-tune memory, you can run the STMM under a workload for 12-24 hours to make its best guess at the memory parameters before turning it off, then implement the recommendations that you think are appropriate. This practice is sometimes used in partitioned instances as well to get a good recommended configuration for hand tuning the environment.

Frequently asked questions

Question #1: Does self-tuning memory work with AIX® Virtualization?

Yes. There are several white papers that cover this including this one from developerWorks:

Implementing System p virtualization with DB2 and WebSphere using IBM Enterprise Workload Management available at <http://www.ibm.com/developerworks/systems/library/es-pvirtualizationewlm/index.html>.

Question #2: Does the STMM work on systems with a small amount of memory (4GB or 8GB)

Yes, the STMM was designed to work on large or small systems.

Question #3: What is the overhead of using the STMM in a production environment?

The overhead is very small and given the large potential gains in performance, it is usually overwhelmingly offset by the performance gains delivered by the STMM.

Question #4: What are these STMM diagnostic log records I keep seeing?

```
2008-01-03-10.32.55.634423-300 I27429A459 LEVEL: Event
PID : 798770 TID : 1 PROC : db2stmm (SAMPLE) 0
INSTANCE: db2inst1 NODE : 000 DB : SAMPLE
APPHDL : 0-203 APPID: *LOCAL.DB2.080151576421
AUTHID : WILDING
FUNCTION: DB2 UDB, config/install, sqlfLogUpdateCfgParam,
probe:20
CHANGE : STMM CFG DB SAMPLE: "Database_memory" From:
"592000"
<automatic> To: "615200" <automatic>
```

These are the automatic configuration changes taking place as STMM tunes your system.

Question #5: Does STMM work for multiple databases under the same LPAR?

STMM is database specific and will work even if there are multiple databases on the same computer or LPAR.

For more information on STMM, see the following links:

- DB2 Information Center topic Self-tuning memory overview - <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/index.jsp?topic=/com.ibm.db2.luw.admin.perf.doc/doc/c0024366.html>
- Understanding the advantages of DB2 9 autonomic computing features - <http://www.ibm.com/developerworks/data/library/techarticle/dm-0709saraswatipura/index.html>
- The Self-Tuning Memory Manager (STMM): A Technical White Paper - <ftp://ftp.software.ibm.com/software/data/pubs/papers/stmm.pdf>
- Adaptive Self-Tuning Memory in DB2 - <http://www.vldb.org/conf/2006/p1081-storm.pdf>
- Introducing DB2 9, Part 3: Self-tuning memory in DB2 9 - <http://www.ibm.com/developerworks/db2/library/techarticle/dm-0606ahuja/>

Automatic storage

Storage is one of the four key resources (CPU, memory, network, and storage) and a fundamental part of reducing costs. An automatic storage database has table spaces that are completely managed by the DB2 database manager. At the most basic level, databases that are enabled for automatic storage have a set of one or more storage paths associated with them. A table space can be defined as `MANAGED BY AUTOMATIC STORAGE` and its containers are assigned and allocated by the data server based on those storage paths. In DB2 Version 9.5, automatic storage is the default for non-DPF systems.

For databases created with DB2 9.7, automatic storage can be enabled at any time. With an automatic storage enabled database, any table space can also be moved into an automatic storage pool providing additional flexibility. For DB2 versions prior to 9.7, automatic storage must be enabled at database creation time and table spaces cannot be moved into an automatic storage pool.

The command line processor (CLP) provides a way to disable automatic storage during database creation by explicitly using the `AUTOMATIC STORAGE NO` clause. If you are using the create database wizard in the Control Center, you can disable automatic storage by clicking the **I want to manage my storage manually** button. If you are using

the Data Studio Administration Console, uncheck the box **Manage storage automatically with DB2**.

Storage management: Automatic resizing

You can also enable automatic storage table spaces for automatic resizing to allow the database manager to handle full file system conditions automatically by adding a new stripe set of containers.

You can enable automatic storage using a CREATE DATABASE command or through the Control Center Wizard by clicking **Let DB2 manage my storage**. In DB2 Version 9, databases are created with automatic storage as the default, but in DB2 UDB Version 8, you had to specify automatic storage when you created the database.

The following are some examples of automatic storage being enabled either explicitly or implicitly:

```
CREATE DATABASE DB1

CREATE DATABASE DB4 ON D:\PathToStorage DBPATH ON C:

CREATE DATABASE DB2 AUTOMATIC STORAGE YES ON E:

CREATE DATABASE DB3 ON /data/pathA, /data/pathB
```

Detecting automatic storage use for a table space

To detect whether automatic storage is in use for a table space, take a snapshot. The snapshot will show the automatic storage path that is being used.

Use the following command to get a snapshot for table spaces and look for “Using automatic storage” in the command output:

```
db2 get snapshot for tablespaces on db_name
```

Where *db_name* is the name of the target database.”. For example, on UNIX or Linux systems you could use the grep utility to filter the output as follows:

```
db2 get snapshot for tablespaces on db_name | grep 'Tablespace
name|Using automatic storage'
```

Another way to look at automatic storage is to get a snapshot for the database and look for “Automatic storage”

```
db2 get snapshot for database on db_name | grep -i 'automatic
storage'
```

```
Number of automatic storage paths = 1
Automatic storage path = /data/db2/db2inst1
```

The resulting lines show the storage paths for the automatic storage feature.

Lastly, the **db2pd** command can also be used to detect whether automatic storage is turned on using the **db2pd -tablespaces** command. Specifically look at the output for tablespace autoresize statistics. More information can be found in the DB2 Information Center topic at

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.trb.doc/doc/c0054595.html>.

Note: When a database has been enabled with automatic storage, automatic storage does not have to be used for every table space in that database. Automatic storage is used for each table space that has been created either by specifying **MANAGED BY AUTOMATIC STORAGE** or by leaving out the **MANAGED BY DATABASE** or **MANAGED BY SYSTEM** keywords during table space creation.

For DPF instances, the storage paths for automatic storage are homogeneous by default. For example, the following command assumes paths /path1, /path2 and /db1 on all physical systems in the instance:

```
CREATE DATABASE mydb ON /path1,/path2, DBPATH ON /db1
```

To create heterogeneous paths, use the \$N partition identifier as follows:

```
CREATE DATABASE mydb ON /p$N,/p$N,DBPATH ON /db1
```

This assumes paths /p1, /p2, /p3, and so on where 1, 2, and 3 are the actual partition numbers.

Frequently asked questions

Question #1: How can I find the maximum size for a tablespace in the DB2 catalog?

DB2 Version 9 and later provides an administrative view you can query with SQL.

The name of the view is **SYSIBMADM.TBSP_UTILIZATION** and it contains the following columns: **TBSP_MAX_SIZE**, **TBSP_INCREASE_SIZE**, and **TBSP_LAST_RESIZE_TIME**. You can also use this view to see the percentage of used pages and the current size to determine how close you are to the maximum size.

Question #2: How do I convert from SMS to DMS storage?

Use **db2look** to recreate the DDL that was used to create the table spaces and the tables in them. Recreate the table spaces (and tables) using **DMS**, then use **db2move** to move the data from the **SMS** to the **DMS** storage. Alternatively, you can use **EXPORT** and **IMPORT** to move the data.

Note: You can also use a table space backup with a re-directed restore.

Question #3: What happens to an application when a DMS file system fills up using automatic storage?

The application will wait (pause) for the resizing to occur.

Question #4: How can I set an alert on tablespace full when the DB2 database waits until the file system is full before?

Instead of monitoring for “table space full” trends, monitor the free space and ensure that the table space has autoresize turned on. Use the administrative view called SYSIBMADM.TBSP_UTILIZATION to see the % of used pages.

Question #5: Can I grow the storage pool by increasing the amount of disk space under a file system?

Yes. Extend the file system or file systems as appropriate keeping the container sizes equivalent.

For more information on automatic storage, see:

- DB2 Automatic Storage: The Future of Storage Management in DB2 for Linux, UNIX, and Windows - <ftp://ftp.software.ibm.com/software/data/db2/9/labchats/20061114-slides.pdf>
- DB2 Information Center topic Automatic re-sizing of DMS table spaces at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.dboj.doc/doc/c0012277.html>

Automatic maintenance

The DB2 database includes several automatic maintenance features to make database management easier. These features help DBAs manage backups, gathering database statistics, and handling database reorganization.

Using automatic maintenance

To see the automatic table maintenance settings, simply look at the database configuration parameters using the following command:

```
db2 get database configuration
```

Look for the following automatic maintenance entries:

Automatic maintenance	(AUTO_MAINT) = ON
Automatic database backup	(AUTO_DB_BACKUP) = OFF
Automatic table maintenance	(AUTO_TBL_MAINT) = ON
Automatic runstats	(AUTO_RUNSTATS) = ON
Automatic statement statistics	(AUTO_STMT_STATS) = OFF

Automatic statistics profiling	(AUTO_STATS_PROF) = OFF
Automatic profile updates	(AUTO_PROF_UPD) = OFF
Automatic reorganization	(AUTO_REORG) = OFF

There is a hierarchy to the configuration parameters as shown by the indentation of the entries above. The AUTO_MAINT configuration parameter sets the highest level, The AUTO_TBL_MAINT configuration parameter covers the table maintenance options

Note: Automatic backup and automatic reorganization are typically used for smaller systems and instances, especially where a high degree of automation is required. For enterprise customers, it might be preferable to perform REORGs as needed and to automate backup through other means. We focus on automatic RUNSTATS and real-time statistics.

Maintaining statistics using automatic RUNSTATS and real-time statistics

The DB2 optimizer uses catalog statistics to determine the most efficient access plan for any given query. Accurate and complete database statistics are critical to efficient data access and optimal workload performance. You can use the DB2 automatic statistics collection feature to update and maintain relevant database statistics. You can optionally enhance this functionality in environments where a single database partition operates on a single processor (serial environment) by collecting query data and generating statistics profiles that help the DB2 data server automatically collect the exact set of statistics that is required by your workload.

Automatic statistics collection calls the RUNSTATS utility on the tables that you want, using the statistics refresh interval that you require. After you automate your statistics collection, the RUNSTATS utility automatically updates statistics about a table, its associated indexes, or statistical views. The updated information includes the number of records, the number of pages, and the average record length for the object. The optimizer then uses this information to determine the best access path to the data.

Automatic RUNSTATS does a good job of keeping the statistics up to date but sometimes even more current statistics are desired. In DB2 Version 9.5, real-time statistics collection was introduced that collects statistics synchronously (that is, whenever a query is run). Real-time statistics collection is enabled using the AUTO_STMT_STATS dynamic configuration parameter. There is a query sensitivity analysis that determines whether a particular statement needs more up-to-date statistics.

There is a limit placed on the time spent collecting statistics; five seconds is the default. If the collection time takes longer than the limit, an asynchronous statistics collection takes place as a background request. For more information, see the DB2 Information Center topic at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.pe rf.doc/doc/c0011762.html>.

If current statistics are not available, the optimizer might choose an inefficient access plan based on inaccurate default statistics.

You should:

- Automate statistics collection on all tables and indexes that might be accessed by a query.
- Automate statistics collection on tables that have significant update activity, or that have had new indexes created since the last time that the statistics were updated.
- Rebind applications (and optionally re-explain their statements) after the statistics are updated, or after you change configuration parameters. The query optimizer might choose a different access plan if it has new statistics or configuration values change.

Following these practices provides the optimizer with the most accurate information with which to determine the best access plan.

To accelerate the statistics update (and to save disk space used to store statistics), consider specifying only the columns for which data distribution statistics should be collected.

If statistics collection is done manually, ensure that the index statistics are synchronized with the table, update table and index statistics at the same time. If the table has been modified extensively since the last time its table statistics were gathered, gathering only the index statistics for that table leaves the two sets of statistics out of synchronization.

Starting with DB2 Version 9, automatic statistics is enabled by default. You should include automatic statistics updates as part of your cost reduction procedures as having more accurate statistics helps the optimizer to select the best available access path to the data for queries and applications.

To enable automatic statistics collection, configure your database instance by using the Configure Automatic Maintenance tool or the command line:

- To use the Configure Automatic Maintenance tool:
 - a. Open the tool either from the Control Center by right-clicking a database object or from the Health Center by right-clicking a database instance.
 - b. Select **Configure Automatic Maintenance** from the pop-up window. Within this tool you can enable automatic statistics collection, specify which tables to collect statistics for, and the maintenance window for the statistics update.
- To use the command line, set each of the following database configuration parameters to ON using the UPDATE DATABASE CONFIGURATION command:

```
DB2 UPDATE DB CFG FOR db_name USING AUTO_MAINT ON
```

```
DB2 UPDATE DB CFG FOR db_name USING AUTO_TBL_MAINT ON
DB2 UPDATE DB CFG FOR db_name USING AUTO_RUNSTATS ON
```

- To set the policy via the command line, use the supplied stored procedures AUTOMAINT_GET_POLICY, AUTOMAINT_GET_POLICYFILE, AUTOMAINT_SET_POLICY, and AUTOMAINT_SET_POLICYFILE. If you have already successfully configured a database using the Configure Automatic Maintenance tool and are happy with the configuration, you use these stored procedures to set this same policy across all the other similar databases with a line command method. For more information on these stored procedures, see the DB2 Information Center topic at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.s ql.rtn.doc/doc/r0023485.html>.
- Optional: If you want to enable automatic statistics profile generation, set the AUTO_STATS_PROF and AUTO_PROF_UPD database configuration parameters to ON.
- Optional: If you want to enable real-time statistics gathering, set the AUTO_STMT_STATS database configuration parameter to ON. When this configuration parameter is set to ON, table statistics are automatically compiled at statement compilation time, whenever they are needed to optimize a query.

Note: Automatic statistics collection is not available in partitioned database environments, certain federated environments, or environments in which intra-partition parallelism is enabled. In these environments, you should call RUNSTATS manually to regularly update the statistics. (You can also call RUNSTATS from the ADMIN_CMD procedure.)

When you collect statistics for a table in a partitioned database environment, RUNSTATS collects statistics only for tables on the database partition from which you execute it. The RUNSTATS results from this database partition are extrapolated to the other database partitions. If the database partition from which you execute RUNSTATS does not contain a portion of a particular table, the request is sent to the first database partition in the database partition group that contains a portion of the table.

Because RUNSTATS collects statistics only on a single database partition, the statistics are less accurate if the data is not distributed consistently across all database partitions. If you suspect that there is skewed data distribution, you might want to redistribute the data across database partitions using the REDISTRIBUTE DATABASE PARTITION GROUP command before executing the RUNSTATS command.

Manually invoking RUNSTATS on a production system during high activity might negatively impact the performance of the production workload. You can

throttle the RUNSTATS utility to limit the performance impact of RUNSTATS execution during high levels of database activity.

For information about statistics, see the DB2 Information Center topic Catalog statistics at <http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/topic/com.ibm.db2.luw.admin.perf.doc/doc/c0005067.html>.

Detecting automatic statistics updates

To detect whether you have automatic statistics update enabled, use the GET DATABASE CONFIGURATION command:

```
db2 get database configuration for db_name
```

The command returns information such as the following information:

```
Automatic maintenance                (AUTO_MAINT) = ON
      Automatic table maintenance      (AUTO_TBL_MAINT) = ON
      Automatic runstats                (AUTO_RUNSTATS) = ON
```

If you are using automatic statistics profile generation, you also see:

```
Automatic statistics profiling        (AUTO_STATS_PROF) = OFF
      Automatic profile updates        (AUTO_PROF_UPD) = OFF
```

To turn off automatic statistics collection, use the UPDATE DATABASE CONFIGURATION command to turn off AUTO_TBL_MAINT and AUTO_RUNSTATS.

More information is available in the developerWorks article Automatic table maintenance in DB2, Part 1 at <http://www.ibm.com/developerworks/db2/library/techarticle/dm-0706tang/>.

Hardware cost reduction strategies

This section outlines features and practices to help you reduce hardware costs. Hardware costs are obviously a part of the total cost of ownership. The good news is that hardware is getting less expensive, faster, and more reliable. Of course, this doesn't negate the need to focus on further reducing hardware costs. There are several different other best practices worth reading to help further cost reduction:

- Improving Data Server Utilization and Management through Virtualization - <http://www.ibm.com/developerworks/db2/bestpractices/virtualization/>

- Deep Compression -
<http://www.ibm.com/developerworks/db2/bestpractices/deepcompression/>
- Tuning and Monitoring Database System Performance -
<http://www.ibm.com/developerworks/db2/bestpractices/systemperformance/>

Troubleshooting cost reduction strategies

Depending on the platform, there are extremely valuable tools and technologies available to help diagnose a problem should the need arise. For AIX, Solaris, and HP-UX operating systems, most of the problem determination tools come with the operating system but there are some very valuable freeware tools as well. These include:

- **lsof:** The lsof tool lists all of the open files on the operating system (OS). When a file is open, the OS returns a numeric file descriptor to the process to use. This tool lists all of the open files on the OS with their respective process IDs and file descriptors.
- A debugger of some type. A debugger can be very helpful for more advanced diagnostics.

For Linux operating systems, there is a long list of powerful and no-charge troubleshooting tools. Using these tools can help prevent a simple problem from turning into a long and painful ordeal that can affect a business and your productivity. Here are some of the more useful diagnostic tools:

- **strace:** The strace tool traces the system calls, special functions that interact with the operating system. You can use this for many types of problems, especially those that relate to the operating system.
- **ltrace:** The ltrace tool traces the functions that a process calls. This is similar to strace, but the called functions provide more detail.
- **lsof:** The lsof tool lists all of the open files on the operating system (OS). When a file is open, the OS returns a numeric file descriptor to the process to use. This tool lists all of the open files on the OS with their respective process IDs and file descriptors.
- **top:** This tool lists the “top” processes that are running on the system. By default it sorts by the amount of current CPU being consumed by a process.
- **traceroute/tcptraceroute:** These tools can be used to trace a network route (or at least one direction of it).
- **ping:** Ping simply checks whether a remote system can respond. Sometimes firewalls block the network packets ping uses, but it is still very useful.

- **hexdump (or equivalent)**: This is simply a tool that can display the raw contents of a file.
- **tcpdump (or ethereal)**: Used for network problems, these tools can display the packets of network traffic.
- **GDB**: This is a powerful debugger that can be used to investigate some of the more difficult problems.
- **readelf**: This tool can read and display information about various sections of an Executable and Linking Format (ELF) file.

All of these tools might not be used, but are very useful to have installed just in case the need arises. For more details on troubleshooting Linux systems in general by one of the authors of this paper, see the book "Self-Service Linux: Mastering the Art of Problem Determination", ISBN: 013147751X.

Windows operating systems also have some valuable diagnostic tools including the following tools from Microsoft® TechNet at <http://technet.microsoft.com/en-us/sysinternals/default.aspx>:

- TCPView for Windows operating systems
- Process Explorer

Be sure to read about the various tools so that you can use them when needed. This knowledge can be tied together with good investigation practices to further improve investigation efforts.

For DB2 software, be sure to check out the Frequently Asked Questions from Problem Management Reports at <http://www.ibm.com/developerworks/db2/bestpractices/FAQs/>.

Conclusion

DB2 is a low-cost, high-performance database. It has many best practices and features for helping to reduce the total cost of ownership and is consistently at the top of the industry performance benchmarks. By becoming familiar with the various cost reduction technologies and strategies, you can potentially save time and effort, both of which can be reinvested into moving your business forward.

Further reading

- IBM DB2 Best Practices
<http://www.ibm.com/developerworks/db2/bestpractices/>
- IBM DB2 9.7 for Linux, UNIX, and Windows Information Center
<http://publib.boulder.ibm.com/infocenter/db2luw/v9r7/index.jsp>
- Making DB2 Products Self-Managing: Strategies and Experiences
<http://www.cs.brown.edu/courses/cs227/archives/2008/Papers/IEEE-DataEngineeringBulletin/Lohman.pdf>
- The Self-Tuning Memory Manager (STMM): A Technical White Paper (PDF)
<ftp://ftp.software.ibm.com/software/data/pubs/papers/stmm.pdf>
- Adaptive Self-Tuning Memory in DB2 (PDF)
<http://www.vldb.org/conf/2006/p1081-storm.pdf>
- DB2 Automatic Storage: The Future of Storage Management in DB2 for Linux, UNIX, and Windows (PDF)
<ftp://ftp.software.ibm.com/software/data/db2/9/labchats/20061114-slides.pdf>
- Understanding the advantages of DB2 9 autonomic computing features.
<http://www.ibm.com/developerworks/data/library/techarticle/dm-0709saraswatipura/index.html>
- Implementing System p virtualization with DB2 and WebSphere using IBM Enterprise Workload Management
<http://www.ibm.com/developerworks/systems/library/es-pvirtualizationewlm/index.html>
- Introducing DB2 9, Part 3: Self-tuning memory in DB2 9
<http://www.ibm.com/developerworks/db2/library/techarticle/dm-0606ahuja/>
- Automatic table maintenance in DB2, Part 1: Automatic statistics collection in DB2 for Linux, UNIX, and Windows
<http://www.ibm.com/developerworks/db2/library/techarticle/dm-0706tang/>

Contributors

Tim Vincent
Chief Architect DB2 LUW

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

Without limiting the above disclaimers, IBM provides no representations or warranties regarding the accuracy, reliability or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any recommendations or techniques herein is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Anyone attempting to adapt these techniques to their own environment do so at their own risk.

This document and the information contained herein may be used solely in connection with the IBM products discussed in this document.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual

results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.