

DB2 10 for z/OS Beta
DBA Improvements

Roger Miller
IBM Silicon Valley Lab
millerrl@us.ibm.com

Session Code: A01

May 11, 2010 12:30 PM – 01:30 PM
Platform: DB2 for z/OS

This presentation will give an overview of key DBA improvements in the DB2 10 for z/OS beta. Roger will cover enhancements in productivity for DBAs: more work performed automatically, fewer interruptions, more online schema changes; new database design options; functions to reduce need for REORG; data compression on the fly; backup and recovery enhancements. This session will focus on the improvements for database administrators, and broaden a bit to include SQL, XML and security improvements. It will also include improvements in performance, scalability, and availability to make the jobs of database administration, system administration, systems programming, and application programming more productive.

Objectives:

Easier performance improvements

Reduced virtual storage management


More Online Schema Capabilities

New database design options (e.g. Hash, Member Cluster)

Improved productivity

2

2010 IDUG North America



Disclaimer/Trademarks

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.

The information on the new product is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information on the new product is for informational purposes only and may not be incorporated into any contract. The information on the new product is not a commitment, promise, or legal obligation to deliver any material, code or functionality. The development, release, and timing of any features or functionality described for our products remains at our sole discretion. * DB2 10 for z/OS beta is announced, not the product.

This information may contain 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.

Trademarks The following terms are trademarks or registered trademarks of other companies and have been used in at least one of the pages of the presentation:

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both: AIX, AS/400, DataJoiner, DataPropagator, DB2, DB2 Connect, DB2 Extenders, DB2 OLAP Server, DB2 Universal Database, Distributed Relational Database Architecture, DRDA, eServer, IBM, IMS, iSeries, MVS, NetData, OS/390, OS/400, PowerPC, pSeries, RS/6000, SQL/400, SQL/DS, Tivoli, VisualAge, VM/ESA, VSE/ESA, WebSphere, z/OS, zSeries

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

Intel and Pentium are trademarks of Intel 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.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

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

At this point, the beta announcement for DB2 10 was made February 9, 2010. The general availability announcement has not been made, so this is not an available product. We have used the code name DB2 X for z/OS up to this point.

See the web for current information, the announcement, ...

<http://www.ibm.com/software/data/db2/zos/db2-10/>

http://www.ibm.com/common/ssi/rep_ca/5/877/ENUSZP10-0015/ENUSZP10-0015.PDF

<http://www.ibm.com/support/docview.wss?uid=swg27017960>

<http://it.toolbox.com/blogs/db2zos/db2-10-for-zos-beta-announced-today-36790>

<http://www.ibm.com/developerworks/spaces/db2zos>

<http://davebeulke.com/?p=625>

<http://community.solutionscenter.techweb.com/community/mainframe/blog/2010/02/09/db2-10-for-zos-beta-starts-today>

<http://www.triton.co.uk/blog/?p=415>

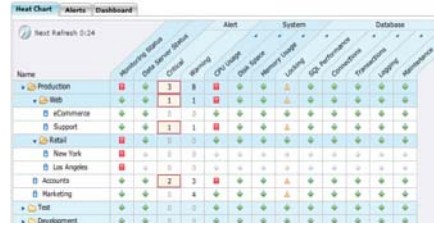
DB2 10 for z/OS sessions at IDUG NA 2010

S01 Overview	Jeff Josten
A01: DBA Improvements	Roger Miller
B02: What's new from the optimizer	Terry Purcell
A03: Performance Preview	Akiko Hoshikawa
A06: DB2 and System z Synergy	Chris Crone
A08: Availability Enhancements	Haakon Roberts

Six sessions will be discussing DB2 10 for z/OS. Others will discuss the performance, availability, optimization and synergy. This session will focus on the improvements for database administrators, but broaden a bit to include SQL, XML and security improvements. It will also include improvements in performance, scalability, and availability to make the jobs of database administration, system administration, systems programming, and application programming more productive. Most of the detail for use of utilities is in A08. Usability for access paths is in B02, although I will note the need to REBIND.

DB2 10: Productivity – Doing More with Less!

- Auto statistics collection
- Easier scaling, simpler memory management
- Reduce contention, more online processing
- Access path stability
- Reduced need for REORG
 - Build compression dictionary on the fly
 - Index list prefetch enhancements
- Configure IBM UDFs and stored procedures
- Allow one SDSNEXIT data set for many subsystems
- Monitoring enhanced
 - Timeout / deadlock diagnostics
 - Identify SQL statements



Manual invocation of

- RUNSTATS
- COPY/BACKUP SYSTEM
- QUIESCE
- MODIFY RECOVERY
- REORG

Most of the improvements come within DB2 for z/OS. Improvements in SQL and XML improve productivity for those who develop new applications and for those who are porting from other platforms. Some of the improvements remove complexity from application tasks, while others reduce the length of the task list or the number of interruptions. DB2 has a strong focus on making DB2 easier to use by automating tasks and eliminating tasks where possible. Avoiding the manual invocations can also help avoid problems for running the function too often or not often enough. Where the task cannot be eliminated, the frequency and monitoring can be reduced, such as the need to reorganize. The improvements for virtual storage and for availability also help DBA productivity.

Allowing tailored names for DSNHDECP will permit many subsystems to share the SDSNEXIT data set. Configuring the IBM-supplied user defined functions and stored procedures makes the install, migration, and service processes faster and more reliable.

DB2 for z/OS: The most robust and cost effective data server



Efficiency

- Deep synergy with System z
- HW Compression
- Consolidation



Resilience

- Unmatched availability
- Unparalleled security
- Industry leading reliability



Growth

- Near-linear scalability
- Optimized for SOA
- Flexible development
- Warehousing capabilities

DB2

- 20%-30% Utility CPU savings
- Compress indexes, save 50% disk
- More CPU on specialty engines

DB2 9

- Flexible context and role security
- Expanded online schema changes
- Volume level backup & recovery

DB2 10

- Save up to 20% CPU batch & transactions
- On-the-fly data Compression
- Temporal data support
- Skip-level migration

- Ten times more concurrent users
- More online schema changes
- More granular access control

- Seamless integration of XML and relational
- Improved SQL
- Partition by growth
- OLAP expressions

Beta Announced:
Feb 9, 2010

Customers have come to know DB2 as the most robust and cost effective data server. With every version of DB2, we are focused on the needs of our customers to operate efficiently, to be up and running 24x7, and to grow with their business. With DB2 9, customers get CPU and disk savings as well as a boost in application productivity with the new pureXML technology. For the next DB2 version, a lot of customers are getting excited. We are putting a lot of focus on out-of-the-box performance improvements and productivity improvements such as online schema, temporal data support, and fine-grain security controls. DB2 continues to be the choice for mission critical business data and we continue to make it easier for customers to keep data on the platform.

DB2 9: One of the key initiatives of V8 was online schema evolution, and that theme is expanding and changing to be data definition on demand. These are key improvements for resilience. One of the important changes is to be able to replace one table quickly with another. Another is to be able to rename a column or an index. A new type of table space combines the attributes of segmented and partitioned, without a partitioning key. Rebuild index can be run with much less disruption. Online table space reorganization for a few partitions is improved a lot, removing the BUILD2 phase for all types of secondary indexes. Table space and index logging can be altered.

Many other improvements help with performance, with scalability and with availability. Index on an expression can be combined with caseless comparisons to improve text search. Improved insert rates can result from improved latching of the log data. Significant reductions in cpu usage are provided with new utilities.

Today's complex applications include both transactions and reporting, so performing both well is imperative. The key improvements for reporting are optimization enhancements to improve query and reporting performance and ease of use. More queries can be expressed in SQL with new SQL enhancements. Improved data is provided for the optimizer, with improved algorithms. Improved cpu and elapsed times can be achieved with the FETCH FIRST clause specified on a subquery. The INTERSECT and EXCEPT clauses make SQL easier to write.

DB2 10: DB2 10 for z/OS provides the best reduction in CPU for transactions, queries, and batch for over 20 years, since V2R1. We expect most customers to reduce CPU times between 5% and 10% as soon as DB2 10 is out of the box.

Applications which can take advantage of additional benefits, such as hash access, index include columns, inline large objects, parallel index updates, faster single row retrievals, work file in-memory, index list prefetch, 64 bit memory enhancements, use of the System z10 1 megabyte page size, buffer pools in memory, access path enhancements, member clustering for universal table spaces, efficient caching of dynamic SQL statements with literals, improved large object streaming, and SQL procedure language performance can have additional CPU and memory reductions. As always with performance, individual customer experiences will vary, and individual workloads will vary more.

Scalability is the second major benefit, with the ability to run five to ten times as many threads in a single subsystem by moving 80% to 90% of the virtual storage above the bar. Schema evolution or data definition on demand enhancements improve availability, by using an ALTER where the only prior option was DROP and recreate. Improved concurrency to DB2 catalog access and utilities extends the scaling. Security is enhanced with better granularity for administrative privileges, masking for data, and new audit capabilities.

SQL, pureXML, and web services improvements extend usability and application portability to the System z, z/OS and DB2 for z/OS platform. Temporal or versioned data improves productivity for applications in a wide range of industries. Applications ranging from SAP to warehousing see benefits from every category and item.

The net result is productivity improvements in DB2 10 for application developers, for database administrators, and for systems administrators that are very important as data grows in scale and complexity.

Top 10 in DB2 10 for z/OS



In 2010, with DB2 10, what could be better than a top 10 items to show how DB2 10 can be a perfect 10 for improving DBA productivity.

Top 10 in DB2 10 for z/OS

1. Productivity improved for database & systems administrators, systems and application programmers
2. CPU reductions for transactions, queries, & batch
3. Ten times more users by avoiding memory constraints
4. More concurrency for catalog, utilities, and SQL
5. More online changes for data definition, utilities and subsystems
6. Improved security with more granularity
7. Temporal or versioned data
8. SQL enhancements improve portability
9. pureXML performance and usability enhancements
10. Hash, index include columns, access path stability, skip migration, ... Pick your favorite.



DB2 10 for z/OS provides the best reduction in CPU for transactions and batch for over 20 years, since V2R1. We expect most customers to reduce CPU times between 5% and 10% as soon as DB2 10 is out of the box. Applications which can take advantage of additional benefits, such as hash access, index include columns, inline large objects, parallel index updates, faster single row retrievals, work file in-memory, index list prefetch, 64 bit memory enhancements, use of the System z10 1 megabyte page size, buffer pools in memory, access path enhancements, member clustering for universal table spaces, efficient caching of dynamic SQL statements with literals, improved large object streaming, and SQL procedure language performance can have additional CPU and memory reductions. As always with performance, individual customer experiences will vary, and individual workloads will vary more.

Scalability is the second major benefit, with the ability to run five to ten times as many threads in a single subsystem by moving 80% to 90% of the virtual storage above the bar. Schema evolution or data definition on demand enhancements improve availability, by using an ALTER where the only prior option was DROP and recreate. Improved concurrency to DB2 catalog access and utilities extends the scaling. Security is enhanced with better granularity for administrative privileges, masking for data, and new audit capabilities.

SQL, pureXML, and web services improvements extend usability and application portability to the System z, z/OS and DB2 for z/OS platform. Temporal or versioned data improves productivity for applications in a wide range of industries.

The net result is productivity improvements in DB2 10 for application developers, for database administrators, and for systems administrators that are very important as data grows in scale and complexity.

Performance Enhancements with Few Changes (CM)

- SQL runtime improved efficiency
- Address space, memory changes to 64 bit, some REBINDs
- Faster single row retrievals via open / fetch / close chaining
- Distributed thread reuse High Performance DBATs
- DB2 9 utility enhancements in CM8
- Parallel index update at insert
- Workfile in-memory enhancements
- Index list prefetch
- Solid State Disk use
- Buffer pool enhancements
 - Utilize z10 1MB page size
 - “Fully in memory” option (ALTER BUFFERPOOL)

These are the improvements which we expect almost every customer to see as soon as DB2 10 is running, even in conversion mode. Many processes reduce CPU time and leverage better 64-bit memory capabilities. This results in better SQL performance even for many existing SQL access paths. Memory improvements provide immediate relief for all memory constrained systems. With some REBINDs, memory usage is reduced, allowing you to use memory more effectively for example in buffer pools for performance in your environment.

Better handling of singleton Cursor Selects through chaining, combines the open fetch and close work ONLY once across the network improving network bandwidth efficiency and overall performance.

Also for DDF transactions, there is increased DDF thread reuse. This enhancement starts to handle and reuse DDF threads through the same methods that we have had for a long time through CICS thread interfaces for robust consistent type transactions.

For people that are directly coming to DB2 10 through migrations from Version 8 they immediately get long list of the Version 9 enhancements especially all the Utility performance improvements. Some are experiencing 20% CPU time savings in DB2 9.

Updates to index columns are done in parallel in DB2 10 improving insert performance. This out of the box enhancement along with DB2 10 List Pre-fetch capabilities improve all existing applications that use list pre-fetch activities across all existing accesses paths and especially when using indexes that may be a little disorganized and in need of a reorg.

DB2 can now use solid state disk devices. These devices are great for the workfiles, GTTs and other high performance table spaces within your environment.

Also, the enhanced way DB2 10 uses in-memory Workfiles and the improvements related to RID pool overflows helps all application systems avoid the deadly table space scan at the peak processing times.

DB2 can utilize the new bigger 1MB page size on z10 and provide additional buffer pool options to put a table fully in memory with an easy simple table space ALTER.

CPU times are reduced for SQL running transactions and batch which are generally the peak customer workload. These techniques take very little change, but the buffer pool enhancements do need an ALTER BUFFERPOOL command.

Performance Enhancements requiring REBIND (CM)

- Most access path enhancements
- SQL paging performance enhancements
 - Single index access for complex OR predicates:
- IN list performance
 - Optimized Stage1 processing (single or multiple IN lists)
 - Matching index scan on multiple IN lists
- Query parallelism improvements
- More stage 2 predicates can be pushed down to stage 1
- More aggressive merge of views and table expressions
 - Avoid materialization of views
- REBIND enables further SQL runtime improvements
- If migrate from V8, get new RUNSTATS before mass rebind

Rebind is required for a long list of improvements in optimization and parallelism. The key improvements from REBIND in CM include SQL inlist improvements, SQL paging enhancements, query parallelism improvements, and more aggressive view and table expression merge.

Improvements with WHERE OR clauses that have columns that all reference the same index can be optimized for a single Stage 1 evaluation instead of being evaluated multiple times and then retrieved multiple times. This consolidates and improves performance dramatically for processing.

IN-List predicates are now evaluated through Stage 1 processing and provide a matching index access of multiple IN-list Where clause criteria. This is common and great for applications that have cursor pagination within their application.

More SQL query parallelism offloads precious CPU main engine cycles and pushes more processing into your specialty zIIP engines.

REBINDS also let DB2 push down more predicate evaluations from Stage 2 to Stage 1 during data retrieval. This helps reduce the rows evaluated in each step of the SQL access path improving performance significantly for complex access paths.

This also comes into play with View and expression materialization. Since the amount of data through the materialized steps can be sometimes big, cutting down the amount of data in Stage 1 processing interim result sets can really help performance of these complex SQL statement and their processing situations.

If you are migrating from DB2 V8, then you will want to get improved statistics for cluster ratio, data repeat factor and high cardinality non-uniform distribution of data by running RUNSTATS before you REBIND.

Performance Enhancements requiring NFM

- Efficient caching of dynamic SQL statements with literals
- Most utility enhancements
- LOB streaming between DDF and rest of DB2
- Faster fetch and insert, lower virtual storage consumption
- SQL Procedure Language performance improvements
- Workfile spanned records, PBG
- Insert improvement for UTS

Some of the performance improvements require new function mode and some work by database administrators to tune the database design and often to REBIND. Efficient caching for literals needs a rebind in NFM.

One of the enhancements that will help everyone with large dynamic SQL applications are the improved processes that handle the Dynamic Statement Cache. They now consolidate SQL statements that are the same but have different literals. This reduces the SQL statement space used in the Dynamic Statement Cache and reuses the security and object verification and access path already developed for the SQL statement. This dramatically improves Dynamic Statement Cache cache hits, reduces the duplicate SQL statement previously held, makes room for more SQL improving the overall workload performance.

Also all the concurrency and performance enhancements through the DB2 10 Utilities improvements are available for more on-line ALTER and maintenance activities.

Better streaming and minimized LOB default sizes helps improve when LOB materialization is happening with the system and this is especially important for DDF type applications because of the network impact of large objects.

Small WORKFILES are now available for simple predicate evaluation for improving performance DB2 provides native support for the SQL procedural language eliminating the cumbersome requirement to generate a C program from the SQL procedure that would then execute as an external stored procedure. DB2 10 SQL procedures are better optimized to execute more efficiently more common constructs are optimized within the DB2 code making SQL procedures very efficient for performance within the SQL procedure language.

WORKFILE can have expanded records up to 65K so larger Joins and answers set can be generated from DB2.

DB2 10 supports partition-by-growth table spaces in the WORKFILE database and provides in-memory work file enhancements in the WORKFILE database.

In the WORKFILE database, DB2 supports simple predicate evaluation for work files. This enhancements reduces the CPU time for workloads that execute queries that require the use of small work files.

Enhancements requiring NFM + DBA work

- Hash access path Alter + Reorg + rebind
- Index include columns Alter + Rebuild + rebind
- Inline LOBs Alter (need UTS and RRF)
- MEMBER CLUSTER for UTS Alter + Reorg
- DEFINE NO for LOB and XML columns
- Alter to UTS, page size, DSSIZE, SEGSIZE Alter + Reorg
- Online reorg all catalog & directory table spaces

11

Some of the performance improvements require new function mode and some work by database administrators to tune the database design and often to REBIND. The DBAs can improve on their direct keys access and implement the new Hash access table space type reducing a key access to potentially a single I/O. This is great for the customer or product id lookouts that are done millions or billions of times a day. The Index include non-unique columns within a UNIQUE index definition provides great way to eliminate indexes and consolidate other indexes for better optimization and overall performance. This is great also for better access paths because all the columns are now in the single index for better access paths and fewer indexes to reference.

Inline LOBs use the reorder row format and handle the LOB better for overall streaming and application performance. This along with the DEFINE NO feature allows the row to be used and the data set for the LOB not to be defined. The application can still reference the row and get the proper results. The system doesn't define the LOB data set until the LOB is saved which can save a lot of space for tables that only sometimes had LOB entries.

The MEMBER CLUSTER setting can help eliminate contention and the clustering requirements within a data sharing environment table. This boosts performance and relieves contention across your data-sharing environment.

And now almost any attribute within the definition of the table space can be Altered within DB2 and then applied through an on-line reorg. This is great for application availability to keep your business rolling with DB2.

Converting to Hash Access

Evaluate potential candidate tables:

Check if they have **unique keys**

used for single fetch queries

*Tables that are fairly **static** in size*

Evaluate existing index that can be replaced with the hash access

If the index is used for range queries, it cannot be dropped

Estimate number of rows, average row size and space needed

ALTER ADD organization-clause

REORG AUTOESTSPACE YES

Rebind applications with fully qualified equal predicates on hash key
To pick up Hash access.

Followup:

Check to see if hash access was chosen

Check that RTS for appropriate space specifications

Monitor index last-used RTS info to see if index can be dropped

This is an example of the process needed to convert to hash access from index access. The first step is to see if the key conditions are met.

The process for converting includes the ALTER, a REORG and REBIND for the applications.

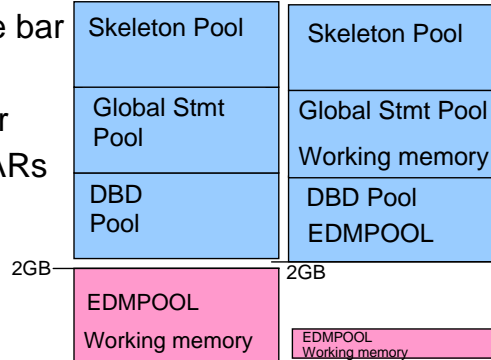
Then you need to be sure that the benefits are delivered, checking that hash access is used, space is accurate and the performance improvements are provided. At some point in the future, you can see if the index can be dropped or if some processes need the index.

DB2 10: 64 bit Virtual Storage Management Relief

DB2 9 helped (~ 10% – 15%)

DB2 10: 5 to 10 times more threads, up to 20,000

- Move 80% - 90% above bar
- More concurrent work
- Reduce need to monitor
- Able to consolidate LPARs
- Reduced cost
- Easier to manage
- Easier to grow



Scalability: Virtual storage constraint is an important issue for many DB2 customers prior to DB2 10.

Virtual storage is most common constraint for large customers. Virtual storage can limit the number of concurrent threads for a single member or subsystem and can take substantial time to monitor and manage.

DB2 9 virtual storage was usually 10-15% relief. The DB2 10 measurements show 80% to 90% of the DBM1 address space. We expect the result to be the ability to run much more concurrent work, with an estimate 5 to 10 times more threads.

Storage monitoring should be drastically reduced. Customers are consolidating LPARs. Sometimes they need to have more than one DB2 subsystem on an LPAR, costing real storage and CPU. With these changes, work can run in one DB2 subsystem, rather than needing more members.

The net for this change is expected to be reduced cost, improved productivity, easier management, and the ability to grow DB2 use much more easily.

Increasing the number of concurrent threads will expose the next tier of constraints. DB2 10 will address a number of the next items, such as utility locking, catalog concurrency.

Other System Scaling Improvements

- Other bottlenecks can emerge in extremely heavy workloads
 - several improvements reduce latching and other system serialization contention
 - new option to for readers to avoid waiting for inserters
 - eliminate UTSERIAL lock contention for utilities
 - Use 64-bit common storage to avoid ECSA constraints
- Concurrent DDL / BIND / Prepare processes can contend with one another
 - restructure parts of DB2 catalog to avoid the contention
- SPT01 64GB limit can be a constraint, especially if package stability is enabled
 - Allow many more packages by using LOBs
- Improved accounting rollup, compress SMF options

14

Increasing the number of concurrent threads will expose the next tier of constraints. DB2 10 will address a number of the next items, such as utility locking and catalog concurrency.

The UTSERIAL lock means that scheduling 20 concurrent REORGs for hundreds of partitions in each one will result in deadlocks too often. Reducing the granularity by removing this lock means that the jobs run. DB2 10 eliminates the use of UTSERIAL by DB2 utilities. This enhancement prevents the majority of timeouts on the global UTSERIAL lock resource.

Improving the catalog structure to allow row level locking can improve concurrency substantially.

The DB2 catalog structure is changed to move most of the large fields with repeating rows of data into LOB columns, eliminating the 64 GB limit and making the information more readable by separating character from binary data. The LOB columns are inline for improved performance.

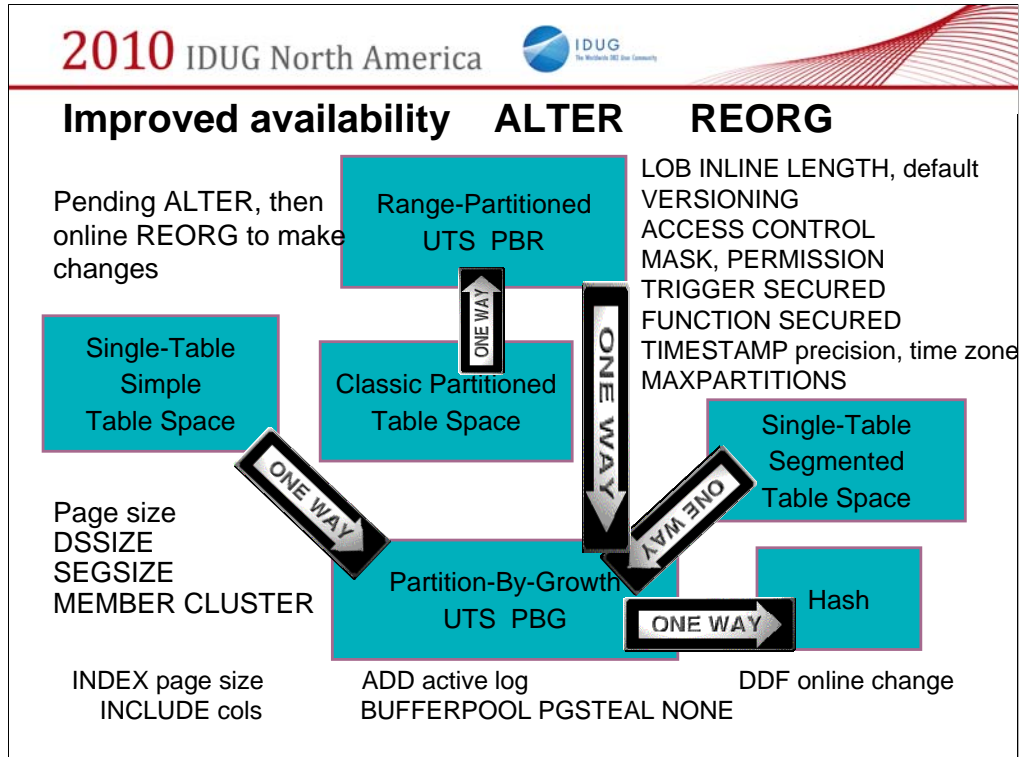
Major changes in DB2 10 catalog & directory

- Improve availability and productivity
- Increase maximum size substantially
- Reduce contention: BIND, DDL, utilities
- Allow SELECT from SYSLGRNX
- Catalog changes: Remove links
 - Many more table spaces, partition by growth
 - Row level locking
 - CLOB and BLOB columns for long strings
 - Online reorganization and check
 - More automatic: DB2-managed SMS-controlled

The DB2 catalog and directory are restructured in DB2 10 to improve productivity and availability. The current size limits are increased substantially and contention among process like BIND, dynamic SQL, data definition and utilities is reduced.

The primary techniques are changes in the DB2 catalog to remove links and the special structures for the catalog. These table spaces change from many tables to one table per table space in a partition by growth table space defined as DSSIZE 64 GB and MAXPART 1. Row level locking is used in place of page level locking. The new catalog tables use a partition by growth universal table space structure. Each table space holds a single table, so many more table spaces are needed. Rather than repeating columns with parts of long strings, the catalog will use CLOB and BLOB columns to store the data, expanding maximum sizes. The new structure allows more standard processes, so that all catalog tables can be reorganized and checked online.

The DB2 catalog changes from using manual definition and extension to DB2 managed data sets under SMS control. The changes improve productivity and availability, but take time to set up.



DB2 10 brings many new options for ALTER BUFFERPOOL, TABLE, INDEX, and TABLE SPACE. These are the changes in table space type in diagram form, adding the ability to change from single table segmented, simple or partitioned table spaces to universal table spaces. This release also adds the ability to modify some new attributes, the page size, the dataset size, and the segment size. These attributes are pending changes when the ALTER is performed, then the changes take place when the online REORG occurs. If a mistake is made before the REORG, then DROP PENDING CHANGES allows you to start again. More alters are provided for universal table spaces, adding the ability to change to MEMBER CLUSTER and the ability to ALTER inline length for LOB columns.

Indexes can now be altered to add INCLUDE columns and index page sizes can be altered, as a pending change. Bufferpools can be altered to PGSTEAL NONE, meaning that they stay resident.

What is not done? Change from multi-table segmented table space. Change back to classic simple, segmented and partitioned. The strategic choice for table space type is the universal table space. Simple table spaces are deprecated, and this version provides a migration path. The ability to add a new active log data set is included. Many online REORG restrictions are removed, to allow more online operations. If you need more improvements in table spaces, then universal table spaces – either partition by range or partition by growth should be your choice.

Online Schema – Some are still drop and recreate

- Change of table - or index space attributes require an outage
- Change of table space attributes
 - Unload data
 - Drop table space
 - Recreate table space, tables, indexes, views
 - Re-establish authorization & RI
 - Reload data
- Change of index space attributes
 - Alter index
 - Index placed in RBDP
 - Rebuild index
- Undo of DDL changes
 - Same as above

Online Schema - Solution

- Execute ALTER statement
- Changes are cached & materialized by next REORG
 - SHRLEVEL REFERENCE|CHANGE
- Undo of DDL changes if not materialized
 - ALTER TABLESPACE... DROP PENDING CHANGES
 - All pending changes are removed
- Undo of DDL changes if materialized
 - Perform compensating ALTER & schedule REORG
 - Assumes no dependencies on prior ALTER have evolved

Online Schema - What Attributes are ALTERable?

- ALTER TABLESPACE

- Page size (not XML) (BUFFERPOOL)
- DSSIZE
- SEGSIZE
- Table space type
 - Single table simple -> PBG (inherit MC)
 - Single table segmented -> PBG
 - Classic partitioned -> PBR (inherit MC)
- MEMBER CLUSTER

ALTER TABLESPACE ... MAXPARTITIONS m



- ALTER INDEX

- Page size (BUFFERPOOL)
 - In DB2 9 this was immediate with RBDP set

ALTER TABLESPACE ... SEGSIZE s



Online Schema - Optimizations

- Undefined table or index spaces
 - ALTERs take immediate effect
- ALTER BUFFERPOOL (no pagesize change)
 - ALTERs take immediate effect
 - Unless other pending operations exist

Universal Table Space Enhancements - Requirement

- **No MEMBER CLUSTER support**
- **No pre-allocation of partitions for PBG**
- **DSN1COPY difficult since user cannot explicitly create PBG partitions**
- **Need drop and create to change table space type**

UTS Enhancements - Solution

- **MEMBER CLUSTER support for UTS**
 - ALTER support
 - Pending ALTER
 - New column MEMBER CLUSTER in SYSTABLESPACE
 - Value in existing TYPE column used to populate new column on DB2 10 migration

UTS Enhancements - Solution

- Provide ability to add new partitions to PBG
 - ALTER TABLE ... ADD PARTITION;
 - Aux objects implicitly created
 - Single partition at a time
- Provide ability to create multiple partitions on create of PBG

- Primarily to support hash access

CREATE TABLESPACE ...

MAXPARTITIONS 10

NUMPART 10

IN ...

UTS Enhancements - Solution

- **Deprecate classic partitioned table spaces**
 - Create of new classic partitioned table spaces requires explicit specification of **SEGSIZE 0**
 - Old syntax will create PBR
 - Default **SEGSIZE** for UTS in 10 is 32

Logging Enhancements - Solution

- Dynamic add of active logs
 - New –SET LOG NEWLOG option
 - New active log must be IDCAMS defined & preformatted by DSNJLOGF
 - Only a single log dataset at a time
 - Issue command twice for dual logging
 - Limit is still 93 active log pairs
 - No dynamic delete of active logs
- Pre-emptable backout
 - Pre-DB2 10, abort/backout schedules non-preemptable SRB
 - On single CPU system may give impression of DB2 hang
 - DB2 10: Create enclave at restart for preemptable SRB backout processing

DB2 10: Security & Compliance

- Protect sensitive data from privileged users & improve productivity
 - SYSADM & DBADM without data access
 - Usability: DBADM for all DB
 - Revoke without cascade
- Separate authorities to perform security related tasks, e.g. security administrator, EXPLAIN, performance monitoring and management
- Audit privileged users
- Row and column access control
 - Allow masking of value
 - Restrict user access to individual cells



Customers are being pressed for a wide range of improved security and compliance. Data retention is a growing need. Protecting sensitive data from the privileged users and administrators is required. Separation of authority for security, access, and some common tasks, like EXPLAIN will help. Auditing for privileged users can also make compliance simpler. Access control is refined in several ways with better granularity for the administrative privileges and with finer grained access control at the row and column level, including the ability to mask access to some fields. Auditing is also enhanced.

DB2 10 Security Benefits

- More flexible authorization
- Separation of duties
- Do job without access to data
- Policies for audit
- Simpler control
- Tighter security
- Avoid cascade delete
- Avoid views and application security logic
- Allow more tools
- Evolve security policies
- Easier to manage security policy
- Improved productivity & tighter security



Access control is refined in several ways with better granularity for the administrative privileges and row and column access, including the ability to mask access to some fields. Auditing is also enhanced.

New security administrative authorities allow a flexible solution to allow a company to not use SYSADM by defining separate administrative authorities as required by company security policies

Performance analysts can perform their tasks without having access to user data

New audit capability allows security administrator to audit all use of new administrative authorities as well as define a set of security policies to audit the DB2 security policies

Using SQL to provide tighter security directly on tables with column masking and row filtering.

Eliminating views and security logic in applications

Preventing access from authorities including SYSADM and DBADM

Allowing usage of ad-hoc query tools, report generation tools

Making the evolution of security policies easy to deal with

Relieving the difficulties of managing security policies

Autonomics and DBA Productivity...

- Checkpoint intervals based on both time and log records
- Run 'must complete' backout under pre-emptable SRB
- Identify unused packages
- SQL Statement level monitoring
 - Statement ID introduced
 - Trace records & messages extended to include statement ID
 - New trace class for statement detail
 - GetPages, Locks, I/Os, cpu/elapsed time at statement level
- Manage max threads, connections, idle thread timeout on an application basis
 - Warning or exceptions issued when threshold is hit

This release improves the support for monitoring within DB2 for z/OS by providing additional performance and diagnostic monitoring capabilities. Version 10 enhances performance monitoring support and monitoring support for problem determination for both static and dynamic SQL. This new support uses the Instrumentation Facility Interface (IFI) to capture and externalize monitoring information for consumption by tooling.

In order to facilitate the collection and correlation of enhanced monitoring data, this release introduces a unique statement execution identifier (STMTID). The statement ID is defined at the DB2 for z/OS server, returned to the DRDA application requester, and captured in IFCID records for both static and dynamic SQL. Through DRDA, the statement ID is returned to the client drivers, along with a compilation source identifier and a compilation time.

To support problem determination, the statement ID is provided in several existing messages, including messages related to deadlocks and timeouts. In these messages, the STMTID is associated with thread information. You can use this thread information to correlate the statement execution on the server with the client application on whose behalf the server is executing the statement.

To support performance monitoring, some existing trace records that deal with statement-level information are modified to capture the new statement ID and new statement-level performance metrics. Also, this release introduces new trace records that provide access to performance monitoring statistics in real time, and allow tooling to retrieve monitoring data without requiring disk access.

Logging Enhancements - Solution

- Provide ability to checkpoint based on both time and number of log records
 - Meaning of CHKFREQ is unchanged
 - Minimum # of log records raised from 200 to 1000
 - New ZPARAMs to control new behavior
 - CHKLOGR – number of log records between checkpoints
 - 1000 – 99,999,999
 - CHKMINS – number of minutes between checkpoints
 - 1-1439
 - CHKTYPE SINGLE|BOTH – govern old/new
 - Set by dynamic ZPARAM or –SET LOG command
 - -SET LOG change does not persist across restart
 - -DIS LOG command indicates settings and if mode is SINGLE or BOTH

REORG INDEX Avoidance - Requirement

- REORG of indexes required to improve application performance
- REORG needed to implement compression
- But REORG requires an outage and is costly in terms of resource

REORG INDEX Avoidance - Solution

- Ability to list prefetch index leaf pages based on index non-leaf information for range scans
 - May greatly reduce sync I/O waits for queries using disorganized indexes
 - REORG INDEX, CHECK INDEX, RUNSTATS expected to benefit
- Improved caching of non-leaf pages
 - Reduce getpages for root page
- Enable sequential detection & index look-aside for parent key lookup on RI insert
- New IFCID359 to track leaf page splits
- All available in DB2 10 CM

REORG Enhancements - Solution

- Introduce new AUX keyword for REORG
 - UTS or classic partitioned
 - Allows movement of base rows by REORG even though LOB columns exist
 - Essential for PBG
 - Allows REBALANCE even though LOB columns exist
 - Would allow pruning of PBGs even though LOB columns exist...
 - Allows DISCARD to delete associated LOB values
 - Default is AUX NO unless:
 - Multi-part REORG of PBG with LOB columns
 - REBALANCE of PBR/classic partitioned with LOB columns
 - REORG of PBR/classic partitioned with multiple parts in REORP
 - No mapping table change
 - Restrictions
 - No XML column support

REORG Enhancements - Solution

- REORG & LISTDEF support for multiple part ranges
 - REORG TABLESPACE... PART 1,23:48,596,3042:3800
 - Retrofit REORG support to DB2 9 in PK87762
- Allow REORG to cancel threads
 - Option to cancel all or just read claimers to ensure drain succeeds
 - FORCE(NO|READERS|ALL)
- Support REORG SHRLEVEL REFERENCE or CHANGE if REORP
 - Previously SHRLEVEL NONE was only option after alter of limitkey
 - Provides restartability
- Support REORG SHRLEVEL CHANGE for REBALANCE
- Reduce outage by updating inline stats after drain released in UTILTERM

REORG Enhancements - Solution

- REORG SHRLEVEL CHANGE for LOB page sets
 - No mapping table required
 - No access to base table, but not permitted if base is NOT LOGGED
- REORG SHRLEVEL NONE for LOBs deprecated in DB2 10 NFM
 - Will run but is a noop with a message saying nothing done

Backup/Recovery Enhancements - Requirement

- Improve COPY/RECOVER performance & reduce overhead
- Faster PIT recovery
- Allow creation of consistent copies with no outage
- COPY problems with CHANGELIMIT if no copy taken
- COPY problems with incrementals if no pages changed

FlashCopy Support - Solution

- Dataset level FlashCopy for utilities
 - COPY
 - REORG inline copy
 - LOAD inline copy
 - FlashCopy of indexes for LOAD, REORG, REORG INDEX, REBUILD INDEX
- Can combine with sequential copy if required
- ZPARAMs for global settings & utility parms for local settings
- FlashCopy backups can be used as input to:
 - RECOVER
 - COPYTOCOPY
 - Create sequential copies from FlashCopy
 - DSN1COPY, DSN1PRNT
 - Remove performance issue with DSN1COPY of inline copies
 - Cannot unload from FlashCopy
 - Use COPYTOCOPY and unload from that

FlashCopy Support - Solution

- REORG, REBUILD, LOAD SHRLEVEL NONE always produce consistent copies
- COPY, LOAD SHRLEVEL NONE produce consistent copies if FLASHCOPY CONSISTENT specified
 - Copy made consistent by backing out uncommitted updates against copy as shadow
 - Do not use just for fun – recovery can be elongated
- FlashCopies are dataset level but may be copied to single dataset to create sequential copy

COPY - Solution

- **COPY CHANGELIMIT**
 - Delay allocating output dataset until CHANGELIMIT checked
 - &ICTYPE in template will no longer be a "C", instead will reflect the correct type of image copy
 - Use RTS to decide between incremental or full
- **Incremental copies**
 - Delay allocating output dataset until pages to be copied are found
 - Insert dummy SYSCOPY record to register empty IIC

PIT Recovery - Solution

- New BACKOUT option on RECOVER
 - Roll back on log from current point instead of restoring recovery base and rolling forward
 - Works with PIT consistency, so changes prior to logpoint may be backed out
 - Can only be done once for a given log range

Autonomic Statistics - Requirement

- Hard to know what stats to gather
 - Too much is wasteful
 - Too little impacts application performance and is wasteful

Autonomic Statistics - Solution

- Automate process
- Autonomic Statistics is implemented through a set of Stored Procedures
 - ADMIN_UTL_MODIFY
 - ADMIN_UTL_EXECUTE
 - ADMIN_UTL_MONITOR
 - Identifies out-of-date/missing/inconsistent stats, updates profile
- SP's run automatically according to a predetermined schedule
- Working together, these SP's
 - Determine what stats to collect
 - Determine when stats need to be collected
 - Schedules and performs the stats collection
 - Records activity for later review

Autonomic Statistics - Solution

- RUNSTATS
 - New options to SET / UPDATE / USE a statistics profile
 - RUNSTATS ... TABLE tbl COLUMN(C1)... **SET PROFILE**
 - RUNSTATS ... TABLE tbl COLUMN(C5)... **UPDATE PROFILE**
 - RUNSTATS ... TABLE tbl **USE PROFILE**

Autonomic Statistics – Use Case

- **Step 1: Add RUNSTATS maintenance window to SYSAUTOTIMEWINDOWS**
 - E.g. Sunday noon-midnight
- **Step 2: Schedule STATS_MONITOR in DB2 Scheduler for administrative tasks**
 - Call ADMIN_TASK_ADD, e.g. Using options DBNAME NOT IN...
 - Optionally schedule other instances of STATS_MONITOR with other options, e.g. different DBs on every first day of the month
- **Step 3: Schedule cleanup**
 - Call ADMIN_TASK_ADD to schedule ADMIN_UTL_MODIFY to clean up history
- **Done**
 - History log in SYSIBM.SYSAUTORUNS_HIST & SYSIBM.SYSAUTOALERTS


Autonomic Statistics – Other Schedulers

- The SYSAUTO* catalog tables are designed to be an extendable, “open” interface
 - DB2 will add new procedures as it automates more functions
 - Users may add their own procedures to “roll their own” autonomic processes
 - The ADMIN_PLANNER function only “knows” about the DB2 Admin Scheduler, so it is coded to use this Scheduler only
- The design is that “DB2-managed” automated processes will use the DB2 Admin Scheduler, while other processes can continue using the non-IBM Scheduler
 - Schedulers can co-exist and complement each other
 - Each scheduler can be configured independently so that they don’t conflict with each other

DB2 10 for z/OS: Skip-Level Migration

May move from V8 to DB2 10,
but just because you can, doesn't mean you always should....

Key considerations:

- 
- Risk/reward analysis
 - What's the risk? Tolerance level?
 - How will you do it? What's your mitigation plan? Are ISVs ready?
 - What workloads do you need to test and can you test them properly?
 - Am I missing out on DB2 9 value in the meantime?
 - May not see large migration cost savings
 - Expect 20% to 25% cost savings versus two migrations
 - Larger migration project, longer migration timeline, more risk
 - Applications and ISVs may not be ready

If you are on V7 or earlier, go to V8

If you plan to migrate in 2010 or 2011, go to DB2 9

If you are on V8 for over 2 years, go to DB2 9

DB2 10 supports migration from DB2 9 NFM or from V8 NFM. Customers not yet running V8 or DB2 9 should plan to migrate to DB2 for z/OS V8 NFM as preparation for a migration to DB2 10. We estimate that about one customer in five migrated using a skip version technique for V5 to V7, and we'll see a similar fraction this time. Normal migration is moving one version at a time every three years. For customers who have gotten behind, the ability to skip a migration cycle will be attractive, but this ability is not "something for nothing". Customers need to consider the tradeoffs and challenges that we know about in skip version migration. Most customers who migrate to new versions by three years after GA are already on DB2 9. The project for skipping is larger. While the testing and rollout are only a little larger than a single version migration, the education and remediation work is roughly double the normal size. Most project plans estimate 150%. Consider the timing carefully. Improvements in DB2 9 are delayed for 2 to 4 or more years with a skip plan. You may need to have extended service on V8.

Improvements to installation & migration information based on customer feedback

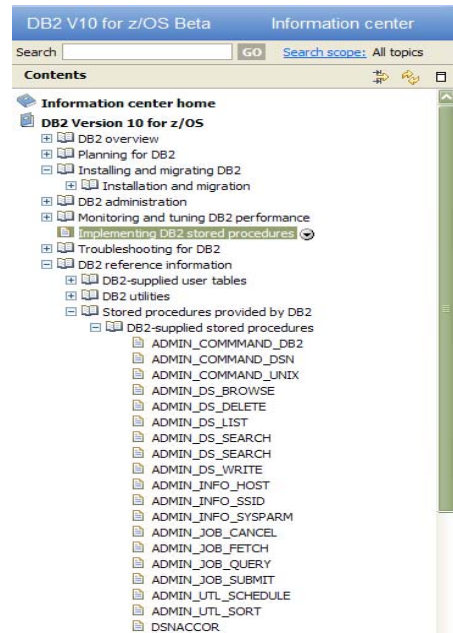
- Pre-migration and migration checklists were added to help plan for and keep track of the migration process. The checklists include links to the pre-migration and migration steps. Access the checklists from the following locations:
 - Introduction to migration from DB2 Version 8:
http://publib.boulder.ibm.com/infocenter/ecsimzic/v1r0/topic/com.ibm.db2.z10.doc.inst/db2z_intro2migfromv8.htm
 - Introduction to migration from DB2 Version 9.1:
http://publib.boulder.ibm.com/infocenter/ecsimzic/v1r0/topic/com.ibm.db2.z10.doc.inst/db2z_intro2migfromv9.htm
- Installation and migration steps clearly define tasks to complete. Related concept and reference information are included as links from the tasks.
- Several new jobs simplify the setup and installation of DB2-supplied stored procedures and user-defined functions. The process is documented in [Installation step 20: Set up DB2-supplied routines \(optional\)](#) and [Migration step 25: Set up DB2-supplied routines \(optional\)](#).

For the V10 development cycle, the information development team focused on addressing solutions to top customer pain points. Based on customer feedback, one of the areas we focused on improving is the installation and migration information.

Our team will be asking you specifically about your migration experience with V10 to gather feedback on the success of these improvements.

Improvements to stored procedure information based on customer feedback

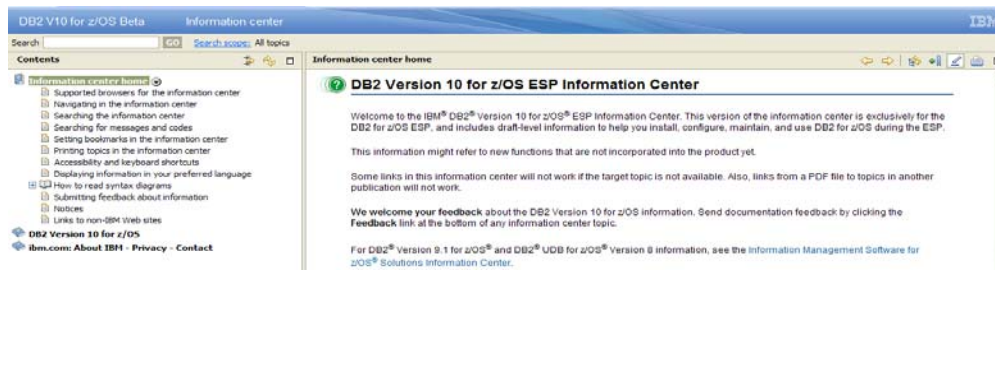
- We made the stored procedure information easier to find.
 - “Implementing stored procedures” explains all of the tasks involved in setting up, creating, and calling stored procedures with links to the detailed steps for each task.
- We added all of the stored procedure reference topics to the “DB2 reference information” section.
- We added missing information, such as setting up WLM environments, granting privileges, and migrating stored procedures from test to production.
- We also rewrote information to make it clearer what you need to do to accomplish a particular goal.



For the V10 development cycle, the information development team focused on addressing solutions to top customer pain points. Based on customer feedback, another area we focused on improving is the stored procedure information.

Important features of information center

- Find helpful usage instructions in the “Information center home” section.
- Easily send feedback by clicking the **Feedback** link at the bottom of any topic.
- Conveniently download the PDF version of the information from the link at the bottom of any topic. (*Look for the PDF icon!*)
- Efficiently search for the information you need by using the search features.



This information center might refer to new functions that are not incorporated into the product yet. Some links in this information center will not work if the target topic is not available. Also, links from a PDF file to topics in another publication will not work.

We welcome your feedback about the DB2 Version 10 for z/OS information. Send documentation feedback by clicking the **Feedback** link at the bottom of any information center topic. Feedback is received directly by the information development team.

developerWorks Exchange Web site

- The DB2 for z/OS Exchange Web site on developerWorks is a great resource!
www.ibm.com/developerworks/exchange/dw_categoryView.jspa?categoryID=25
- Use this site to exchange sample programs, SQL statements, utility jobs, and more.
- This is a user community resource, and we encourage you to use and contribute to this site.

developerWorks > My developerWorks >

DB2 for z/OS Exchange

To narrow the number of results, you can enter a search term. Page through the results or display all the results by clicking Show all. To see more information about a code listing, click on the title.

To submit, click Submit in the Exchange options box on the right side of this page.

Search the DB2 for z/OS Exchange

Search for: within All topics

1 - 5 of 44 results [Next](#) [Show all](#) [Show summaries](#)

Title	Submitter	Date	Rating
DBRM Release Marker UDF - Version 2.1	ToddBurchDB2	Dec 31, 2009 12:50 PM	★★★★ Based on 1 rating(s)
Sample Application for changing column of CLOB datatype to XML datatype - Version 1.0	Jane_Man	Dec 30, 2009 6:51 PM	★★★★ Based on 1 rating(s)

Why Migrate to DB2 10 for z/OS?

- Business needs to save money
 - Reduce CPU time
 - Scale, reduce complexity
- Application developers need improved productivity and integration
 - pureXML for a faster, more capable interface to XML data
 - Powerful new SQL temporal enhancements & portability
- Database Administrators need
 - Improved performance
 - Availability, scalability & memory management
 - Improved security and regulatory compliance
 - More productive database administration



DB2 10 has a lot for everyone. Here are just a few of the strategic highlights.

The business needs include CPU cycle reductions that deliver for batch and transactions, and pureXML™ improvements to build a strong foundation for SOA and XML initiatives.

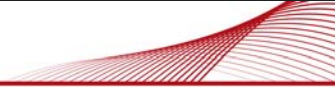
Database Administrators (DBAs) need improved database performance, scalability, and availability. Memory management is dramatically reduced, so growth is much simpler. DBAs also get more flexible security to help with regulatory compliance. A wide range of enhancements improve ERP application and data warehouse functionality and performance. Large object (LOB) performance and flexibility are improved. Many DBA tasks are simplified.

Application developers are most excited by the second release of pureXML, which improves the SQL and XML interface to access XML data stored in a native format. Application developers need powerful new SQL enhancements. Improved SQL and data definition compatibility with other DB2 platforms makes porting much easier.

DB2 10 for z/OS At a Glance

Performance, Scalability	<ul style="list-style-type: none"> • CPU reductions out-of-the-box • Hash access to data, index include columns • Ten times more threads per DB2 image
Availability Security Productivity	<ul style="list-style-type: none"> • More online schema changes • Improved concurrency: catalog, data, & utilities • Row and column access control, masking • Administrator privileges with finer granularity • Administration productivity enhancements
Application Enablement	<ul style="list-style-type: none"> • Versioned data or temporal queries • pureXML enhancements • Last committed reads • SQL improvements that simplify porting
Dynamic Warehousing	<ul style="list-style-type: none"> • Moving sum, moving average • Many query optimization improvements • Query parallelism improvements • IBM Smart Analytics Optimizer

DB2 10 for z/OS provides the best reduction in CPU for transactions and batch for 22 years, since V2R1. We expect most customers to reduce CPU times between 5% and 10%. Applications which can take advantage of additional benefits, such as hash access, can have larger CPU and memory reductions. Scalability is the second major benefit, with the ability to run five to ten times as many threads in a single subsystem by moving 80% to 90% of the virtual storage above the bar. Schema evolution or data definition on demand enhancements improves availability. SQL, pureXML, web services extend usability and application portability for this platform. Productivity improvements for application developers and for database administrators are very important as data grows in scale and complexity. DBAs can avoid running statistics, some REORGs, and benefit from memory, improved query stability, and utilities enhancements. Warehousing continues to evolve, with improvements in SQL and XML, better optimization techniques, increased parallelism and the new IBM Smart Analytics Optimizer.



Roger Miller
millerrl@us.ibm.com