Information Management

IBM

DB2 9 for z/OS Business Value

M. Sueli Almeida, sueli@us.ibm.com
IBM DB2 for z/OS
Silicon Valley Laboratory

DB2 9 for z/OS Customer Experience

Rosana Profitti Hernanz Moreno
Casas Bahia

IBM Information On Demand 2009

DB2 z/OS 2009 Technical Conference

Important Disclaimer

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY.

WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE.

IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION.

NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, OR SHALL HAVE THE EFFECT OF:

- CREATING ANY WARRANTY OR REPRESENTATION FROM IBM (OR ITS AFFILIATES OR ITS OR THEIR SUPPLIERS AND/OR LICENSORS); OR
- ALTERING THE TERMS AND CONDITIONS OF THE APPLICABLE LICENSE AGREEMENT GOVERNING THE USE OF IBM SOFTWARE.

Introduction

- **DB2 9 for z /OS is a major release of DB2 for the IBM System Z platform**
- **Many features that provide technical and business benefit across the following areas:-**
  - Price/performance (reduced cost)
  - Improved availability
  - Reduced total cost of ownership
  - Application enablement and support for new workloads

This session is about the business value of DB2 9 for z/OS. This is a major new release of DB2 and it has been available for 2 years.

There are many functions and features in this release that provides not only technical but also business value.

During this presentation we will talk about the business value in terms of improve price/performance to reduce cost; improved availability, reduced total cost of ownership and enabling applications as well as support for new workloads.

Agenda

- **Economic downturn**
- **DB2 9 for z/OS – Business Value & Benefits**
  - Price/performance (reduced cost)
  - Improved availability
  - Reduced total cost of ownership
  - Application enablement and support for new workloads
- **Casas Bahia – Customer Experience with DB2 9 for z/OS**

For the Agenda we will go very briefly on the impact of the global economic downturn and how it changed the game;

Then we will go thru the technical and business value of Db2 9, before I hand over to Rosana Moreno, who will give us a background information on Casas Bahia and also their experiences with DB2 9 for z/OS

Succeeding in turbulent times
Top 3 Challenges for CIOs

- **Reducing Costs of Information Technology**
- **Boost Business Resilience and Reduce Risks**
- **Demand to drive more innovative solutions**

If you look at the facing challenges that most of the CIOs around the world in the global economic recession times, there are 3 recurring things:

Reducing cost of information technology;

Technically trying boost business resilience and achieve continuous availability across any kind of failure, and

Demand of driving more innovative solutions

Now, let us take a look on how Version 9 can help to address those challenges.

Improved price/performance (reduced cost)

Native SQL Procedures

- Standard SQL type language that programmers on all platforms will be able to easily understand
  - No need for COBOL skills
  - No need for external C program which must be prepared and executed
- External SQL procedures area easily converted over
- Runs entirely inside DBM1 address space and avoids SQL API trips
- Eligible for zIIP offload when invoked by DRDA over TCP/IP connection

Improved price/performance (reduced cost)

- **Larger index page size**
  - Index must be defined in 8KB, 16KB, or 32KB bufferpool
  - Reduce number of index levels and reduce index retrieval costs
  - Reduce number of painful index leaf page splits
- **Autonomic asymmetric leaf page split**
  - Performance relief for sequential key insert with better space utilization
  - Reduce number of painful index leaf page splits
- **Fast table APPEND ('insert at the end')**
  - Reduced space searching

Improved price/performance (reduced cost)

➢Data sharing logging improvement

- Now only necessary to generate unique LRSNs when log records are for same index or data page
- Reduced LRSN spin saves CPU and reduces log latch contention

➢Identify and remove unused indexes

- Difficult to determine in a dynamic SQL environment
- SYSINDEXSPACESTATS.LASTUSED (RTS) records last used date
- Improved insert and delete performance

Data sharing logging improvement

Prior to DB2 V9, successive log records produced on the same DB2 member always had different LRSNs. In order to achieve this DB2 would re-drive the pulling of the *store clock* (STCK) to produce the LRSN if it was the same as the previous one. With processor speeds continually increasing, especially with the z9 technology, this became increasingly likely. Each time the STCK was re-driven in this way, CPU cycles were effectively lost. Note that while DB2 is doing the re-driving of the pulling of the STCK, the log latch is maintained. Holding the log latch in this way aggravates the log latch contention.

With this enhancement it is now only necessary for a given DB2 member to produce unique LRSNs when the log records are for the same data page. In V9, any LRSN update spins are done without holding the log latch, so other work is not blocked. This saves both CPU cycles and reduces log latch contention.

Improved price/performance (reduced cost)

- **Improved sort avoidance and performance**
  - Use of in-memory workfile if number of rows can fit into one page
  - Use of 32KB workfile if row size > = 100 bytes to reduce IO
  - New GROUP BY sort – group collapsing during sort input phase
  - Sort avoidance for DISTINCT on non-unique index
- **SELECT, INSERT, UPDATE, DELETE for LOBs**
  - Improved performance
  - Significant reduction in locking and holding locks for shorter duration
  - No lock escalation
  - New dynamic data format (progressive streaming) for JCC T4 applications
- **Reduced CPU for LOAD and REORG for charge back**
  - Improvements related to reduced index manager costs, use of shared memory objects to avoid data movement, improved index key generation

There are several improvements in the sort avoidance an performance, on both small and large sort. The first is about of usage of in-memory. If the result rows fit in single 32K page, then DB2 will no longer allocate the DB2 internal sort and the overhead of logical workfile. Basically DB2 will use a new in-memory workfile technique to actually produce the results much faster and reduce cost of CPU; The second switch is on the converged temp space. DB2 will now make much more use of 32kb workfile even when the row size is relatively small such as 100 bytes. DB2 will preferentially use the 32KB page size and this will reduce the elapsed time and CPU cost of the sort;

There are also some optimization improvements such as a new GROUP BY sort; For DISTINCT we are now able to use a non-unique index to avoid sort in this case;

LOBs: Beginning with DB2 9, the locking technique for LOBs has changed. Locks on LOBs are taken when they are needed for an INSERT or UPDATE operations and released immediately at the completion of the operation. LOB locks are not held for SELECT and DELETE operations. A LOB lock that is acquired in the case of an application that uses uncommitted read only tests the LOB for completeness and releases immediately after it is acquired. For normal SELECT operations, DB2 avoids acquiring any LOB lock since the holding of the current lock on a base row or a data page is good enough to ensure the access of a correct copy of LOB data. But for a SELECT with ISO(UR) readers, if LOB columns are to be selected, the UR reader must acquire an S LOB lock to serialize with concurrent updater/insert operations. Instead of using LOB locks, DB2 now uses the READ LSN value to determine whether LOB storage can be reused. To summarize, in DB2 9 S-LOB lock is acquired and freed in each FETCH call as in DB2 V8. The change from DB2 V8 is that for non-UR readers and DELETE calls, no S-LOB lock is acquired. For INSERT, X-LOB lock acquired is released after allocation completes and is not held till commit. There is no change for UPDATE. Availability improves because locks are not held until commit.

Reduced CPU for LOAD and REORG → I many cases CPU for those utilities are charged back to the application user area. In V9 there are some improvements to reduce CPU cost by reducing Index Manager costs, making usage of shared memory to avoid data movement between batch and DBM1 AS and also some improvement in the index key generation.

Improve Availability

- **Online REORG**
  - Eliminates the BUILD2 phase for REORG PART operation
  - NPIs also shadowed and implicitly reorganized
  - Partition level unload/reload/log apply parallelism
  - Removes prime cause of outage
- **Online REBUILD INDEX**
  - Good for CREATE INDEX DEFER YES
- **CLONE table**
  - Fast replacement of one table with another (flip-flop)
  - Addresses requirement to replace the entire contents of a table while maintaining access to the old data until the new dataset has been loaded
    - Aka “Online LOAD REPLACE SHRLEVEL(CHANGE)”
  - Reduce or even eliminate service outage caused by batch processes

Improve Availability …

- REORG of LOB table space
  - Complete REORG of LOB data to reclaim space
- Modify EARLY code with no IPL needed
  - New Command to refresh early code and then recycle DB2
- Consistent RECOVER
  - Automatically detects uncommitted transactions that are running at the PIT recovery time
  - Rolls back changes on the object to be recovered to ensure data consistency after the PIT recovery
  - URs that are INFLIGHT, INABORT, POTPONED ABORT are rolled back
  - Leaves the recovered objects in a consistent state from at transaction point of view
  - Reduces even eliminates the need for taking successful QUIESCE points

Improve Availability …

- Data sharing restart availability enhancements
  - Initiating automatic GRECP recovery at the end of restart
  - Deferring the updates of SYSLGRNX beyond end of restart
  - Opening data sets earlier in restart processing
  - Removing need for conversion locks during special open
  - Allowing table-level retained locks to support postponed abort unit of recovery
- Cancel in progress database commands
- Online schema change
  - RENAME COLUMN and RENAME INDEX
  - Eliminate destructive changes

Deferring the updates of SYSLGRNX till after end of restart

The closing off of SYSLGRNX records is important for the performance of recoveries. In DB2 V8 non-GBP-dependant objects that were opened but not involved in the restart log apply phases (forward log recovery and backward log recovery) have SYSLNGRNX entries closed off at the end of restart. These updates at the end of restart could contend with other member updates, resulting in extended restart times.

In DB2 V9 the updating of non GBP dependent objects SYSLGRNX entries is deferred beyond restart, therefore allowing restart to complete quicker. The updating of the SYSLGRNX entries are now triggered by the first system checkpoint following restart. Note that this benefits DB2 restart in a non data sharing environment as well.

Cancel in progress DB Commands:

A -STOP DB (DSN0170) SPACE(TESTIMP*) command while some other transactions were still operating on objects in this table space. As a result, the page set status is changed from RW to STOPP.
The -DISPLAY THREAD(*) command has now the extension: TYPE(SYSTEM) keyword. So, now the STOP command can easily be identified. For example, if the system agent command has token 93 associated to it, which makes it now possible for you to cancel this system thread using the usual -CANCEL THREAD(93) DB2 command.

Reduced Total Cost of Ownership

- Plan stability
  - Ability to backup your static SQL packages
  - Save old copies of packages in Catalog/Directory
  - Can switch back to previous or original version when bad access path change
  - Removes the fear of REBIND
- Histogram statistics
  - Represents pockets of data
  - Improved filter factor estimation when gaps in the range
- Universal table space partition-by-growth (PBG)
  - New partition added automatically when more space needed
  - Max size controlled by MAXPARTITIONS, DSSIZE, and page size
  - Help deal with potentially unpredictable data volumes

Maximum size of SPT01 = 64GB

APAR for SPT01 contraction:

PK80375 open, there is a usermod though...

Reduced Total Cost of Ownership …

- Trusted network context and SQL ROLE
  - Addresses security/audit in 3-tier architectures where a ‘surrogate’ user id (or function id) is used to access DB2
  - Provides better control access to applications
  - Provides better audit ability both when making database changes and when user executes “transactions”

- Selective tracing
  - New trace filters available to help minimize trace overhead
  - Filters include the ability to include or exclude data with wild card capability
  - Use filters to target detailed trace classes selectively and reduce CPU overhead

Reduced Total Cost of Ownership …

- **Incremental DBM1 31-bit storage VSCR (5-10%)**
  - Reduced EDM Pool requirement for static SQL
  - Reduced Local Dynamic Statement Cache when using KEEPDYNAMIC(YES)
- **Index compression for informational systems**
  - Save DASD space
  - Requires large index page size
  - Target large indexes e.g., NPIs
- **Utility TEMPLATE switching**
  - Extends the capacity of the template command to allow different output locations to be specified based on the size of the dataset
  - Reduces the ongoing effort required to monitor and maintain backup jobs by automatically selecting the correct output location as DB2 tables grow over time
- **MODIFY RECOVERY**
  - Simplification and safety

Reduced Total Cost of Ownership …

- **Exploitation of volume level backups**
  - Tape support/control for BACKUP and RESTORE SYSTEM utilities
  - Recovery of individual tables places and indexes from volume-level backups

- **Automatic object creation**
  - Implicit creation of
    - Database
    - Primary key index
    - Unique key index
    - ROWID index
    - LOB table space, table & auxiliary index

Application enablement and support for new workloads

- **Integrated XML support**
  - Declarative language, reduce complexity, dramatically improve application development
  - Directly store and query XML in inherent hierarchical format
    - No decomposition/composition
    - No normalize/de-normalize
  - Native processing with good XML index design = high performance
  - Ideally suited
    - Versatile schemas that are diverse and evolve, and end-user customizable applications
    - Sparsely populated attribute values (null vs. absence)
  - Manage XML data with ACID properties, auditing and regulatory compliance, together with relational data

Application enablement and support for new workloads

- INSTEAD OF triggers
  - Usability feature provides an extension to the updatability of views
  - Trigger logic performs the operation against the table on behalf of the view
  - Transparent to the application

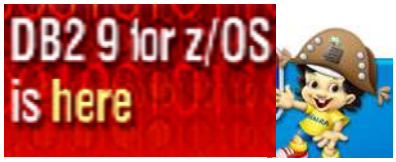
- Index on expression
  - General application for multi key column browsing
  - Eliminate non-matching index scans
  - Remove column concatenation
  - Reduce number of destructive index changes

Application enablement and support for new workloads

- **FETCH FIRST and ORDER BY in subselect**
  - Can perform mass insert/update/delete in increments

- **Optimistic locking control**
  - Positioned updates and deletes performed with optimistic concurrency control method
  - Uses RID and a row change token to test whether data has been changed by another application since the last read operation
  - Ensures data integrity while limiting the time that locks are held
  - Faster and more scalable than database locking for concurrent data access

Casas Bahia: Experiências com DB2 9 for z/OS



Rosana Profitti Hernanz Moreno
Analista de Suporte Técnico

Casas Bahia

- Líder do mercado Brasileiro na área de varejo
- 30 milhões de clientes
- Cerca de 500 lojas
- Casas Bahia loja virtual desde Fevereiro/2009
- Centro Tecnológico em São Caetano do Sul

Casas Bahia – Ambiente Tecnológico

- **2 z10 com z/OS V1.8 (Sysplex)**
- **10 CICS V3.1 (CICSPLEX)**
- **2-way Data Sharing Group para Produção**
- **2-way Data Sharing Group para desenvolvimento**
- **2-way Data Sharing Group para pre-produção**
- **2 iSeries nos depósitos**
- **~500 Linux nas lojas**
- **~600 transações por segundo**
- **~3TB de disco para dados em DB2**
  - 3000 tabelas
  - Maior tabela com 200.000.000 de linhas

Casas Bahia: Preparação para DB2 9 for z/OS

- **Conversão de CCSID**
  - Recriação das Views
  - Rebind de todos os planos e packages
  - Revisão de todos os planos de acesso
  - CCSID na compilação
    - Recompilação de alguns programas

- **Migração do DB2PM para OMEGAMON**

Casas Bahia: Migração para DB2 9 for z/OS

- **Migração para CM – Dezembro/2008**
  - ☹ Surpresas
    - Área de sort de 32K
    - Planos de Acesso – Regressão de performance
      - PTFs para corrigir
- **Migração para NFM – Abril/2009**
  - 😐 Surpresas
    - Campos VARCHAR
    - Simple Tablespace

Casas Bahia explorando novas funções do DB2 9

- ☺ **Online schema change**
  - ☺ RENAME COLUMN
  - ☺ RENAME INDEX
- ☺ **CATMAINT UPDATE SCHEMA SWITCH**
- **CLONE TABLE**
- **PBG – Partition by Growth**
- **Native SQL Procedure**

Casas Bahia e DB2 9 for z/OS

Sumário

- Migração muito mais fácil do que na Versão 8
- Processo com 3 fases é mais seguro
- …