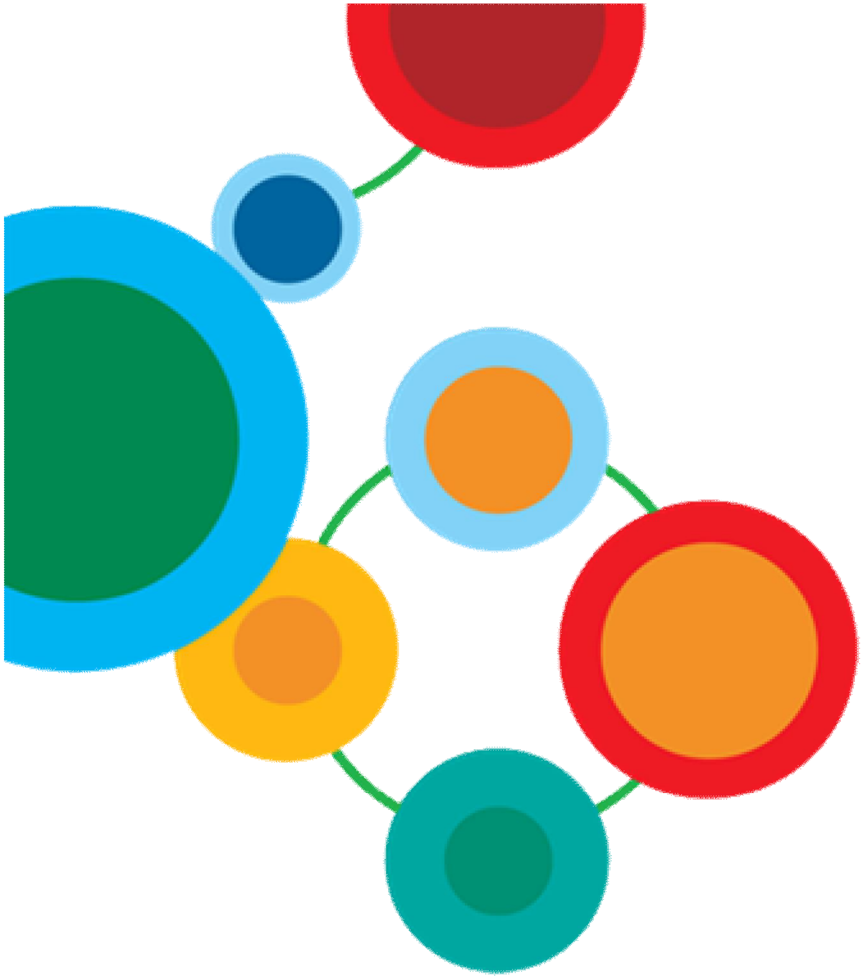


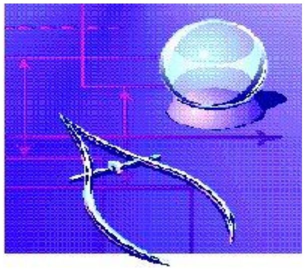
What’s New in DB2 10 for z/OS for DBA

Session Number 1260

James Teng, Ph.D.
Distinguished Engineer
IBM Silicon Valley Laboratory

IBM Software
Information On Demand 2011

Disclaimers & Trademarks*



Information in this presentation about IBM's future plans reflect current thinking and is subject to change at IBM's business discretion. You should not rely on such information to make business plans. Any discussion of OEM products is based upon information which has been publicly available and is subject to change. The opinions expressed are those of the presenter at the time, not necessarily the current opinion and certainly not that of the company.

The following terms are trademarks or registered trademarks of the IBM Corporation in the United States and/or other countries: AIX, AS/400, DATABASE 2, DB2*, Enterprise Storage Server, ESCON*, IBM, iSeries, Lotus, NOTES, OS/400, pSeries, RISC, WebSphere, xSeries, z/Architecture, z/OS, zSeries, System p, System I, System z

The following terms are trademarks or registered trademarks of the Microsoft Corporation in the United States and/or other countries: MICROSOFT, WINDOWS, ODBC

For more copyright & trademark information see ibm.com/legal/copytrade.phtml

Agenda

- Online Schema Enhancements
- Index Enhancements
- Buffer Pool Enhancements
- Universal Table Space Enhancements
- Auto-Compress
- Concurrency Enhancements
- RTS Enhancements
- Hash Table Access
- Logging Enhancements
- Utility Enhancements

Without Online Schema

- **Unload data**
- **Drop base tables**
- **Create base tables**
- **Create indexes**
- **Create views**
- **Define authorizations**
- **Load data (Copy, Stats, Check)**
- **Rebind**

Data available for read

Data unavailable

(minutes, hours, days)

Data available

Online Schema Evolution – before V10

- Table or column Changes
  - Increase column size within datatype integer, float
  - Change to expand datatype: char, numeric
  - Change varchar to / from char
  - Rename Table/Column
- Index Changes
  - Add new column to index
  - Change index column type
  - Rename Index
- Dynamic Partitions
  - Add partition
  - Rotate partition

Online Schema Enhancements – V10

- Table Space Changes - deferred
  - Page Size – UTS only
  - DSSIZE – UTS only
  - SEGSIZE – UTS only
  - MAXPARTITIONS
  - Convert single table simple/segmented into PBG
  - Convert classic partitioned table into PBR
  - Convert PBR to PBG
  - Convert Classic Partitioned/PBR/PBG to Member Cluster
  - Table Space is put into Advisory REORG pending
- Index Changes - deferred
  - Page Size
  - Index is put into Advisory REORG pending

Online Schema Enhancements – V10…

- Deferred Alters are materialized at the next table/index space level REORG
  - ➢ Before REORG, pending changes can be dropped via ALTER … DROP …
  - ➢ Quiesce applications using the DBD, plan/package locks during the SWITCH phase
  - ➢ Invlaidate plans/packages/dynamic statement cache when altering table space types
- xAlter Buffer Pool with the same page size – immediate
  - ➢ No longer need to STOP TS/IS in data sharing
  - ➢ Use DRAIN(ALL) to quiesce applications
  - ➢ New BP will be used after commit

Execute ALTER Statement

- Statement is validated.

DB2 semantically validates stmt and checks auth by performing pending DDL (if any), then the requested ALTER (then undoes the catalog changes).

- Assuming all checks out ok:
  - statement is put on to-do-list.
  - Table space is placed in Advisory REORG Pending (non-restrictive).
  - Statement completes with +610 to advertise the Advisory state.

The to-do list is new catalog table, SYSPENDINGDDL

SYSIBM.SYSPENDINGDDL:

DBNAME	TSNAME	DBID	PSID	…	OPTION_ KEYWORD	OPTION_ VALUE	…	STATEMENT_ TEXT

REORG – materialize deferred alters

- Pending DDL is materialized during the SWITCH phase.
- SYSPENDINGDDL entries are removed.
- Stats are collected.
  - Default is TABLE ALL INDEX ALL UPDATE ALL HISTORY ALL unless overridden.
  - Warning message DSNU1166I issued to indicate that some partition statistics may no longer be accurate (COLGROUP, KEYCARD, HISTOGRAM …).
- SYSCOPY entries created to record manifestation.
- Advisory REORG Pending state is reset.

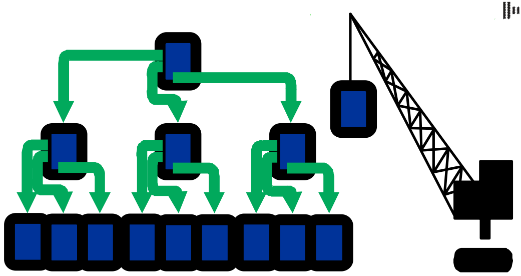
Deferred Alter Restrictions

- May not mix immediate and deferred options in an ALTER statement (-20385).
- Many immediate DDL statements are not allowed while there is pending DDL awaiting manifestation (-20385).
  - CREATE/DROP/ALTER TABLE with pending DDL.
- RECOVER before REORG is not allowed

Online Schema – Immediate

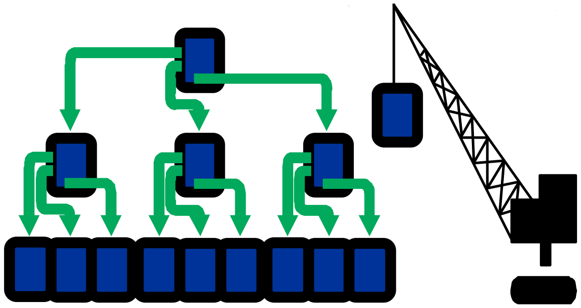
- Alter Index to add included columns - immediate
  - Add non-key columns
  - Index is marked in Rebuild Pending unless new columns are added within the same COMMIT
  - Need to REBUILD index and rebind to benefit for index-only access
- Alter PBR/PBG into Hash - immediate
  - Table is not accessable through hash access due to the overflow index is marked in Rebuild Pending
  - Unable to insert new rows until REORG or REBUILD the overflow index
- Alter LOB to become inline - immediate
  - Use REORG to convert inline for all existing LOBs

Index Improvements – V10



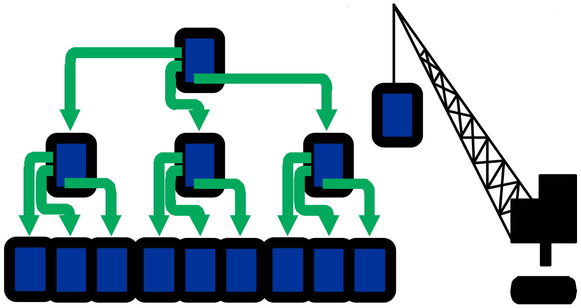
- Parallel index update at insert
  - For tables with more than 2 indexes
  - 2 or more indexes for Member Cluster or APPEND tables
  - I/O parallelism
- Index with included columns
  - Add non-key columns
  - Unique index only
  - Also applies to primary index and indexes used to enforce unique constraints
  - Index will be marked in Rebuild Pending unless columns are added to the table within the same commit
  - Rebind to benefit for index-only access

Index Improvements – V10 …



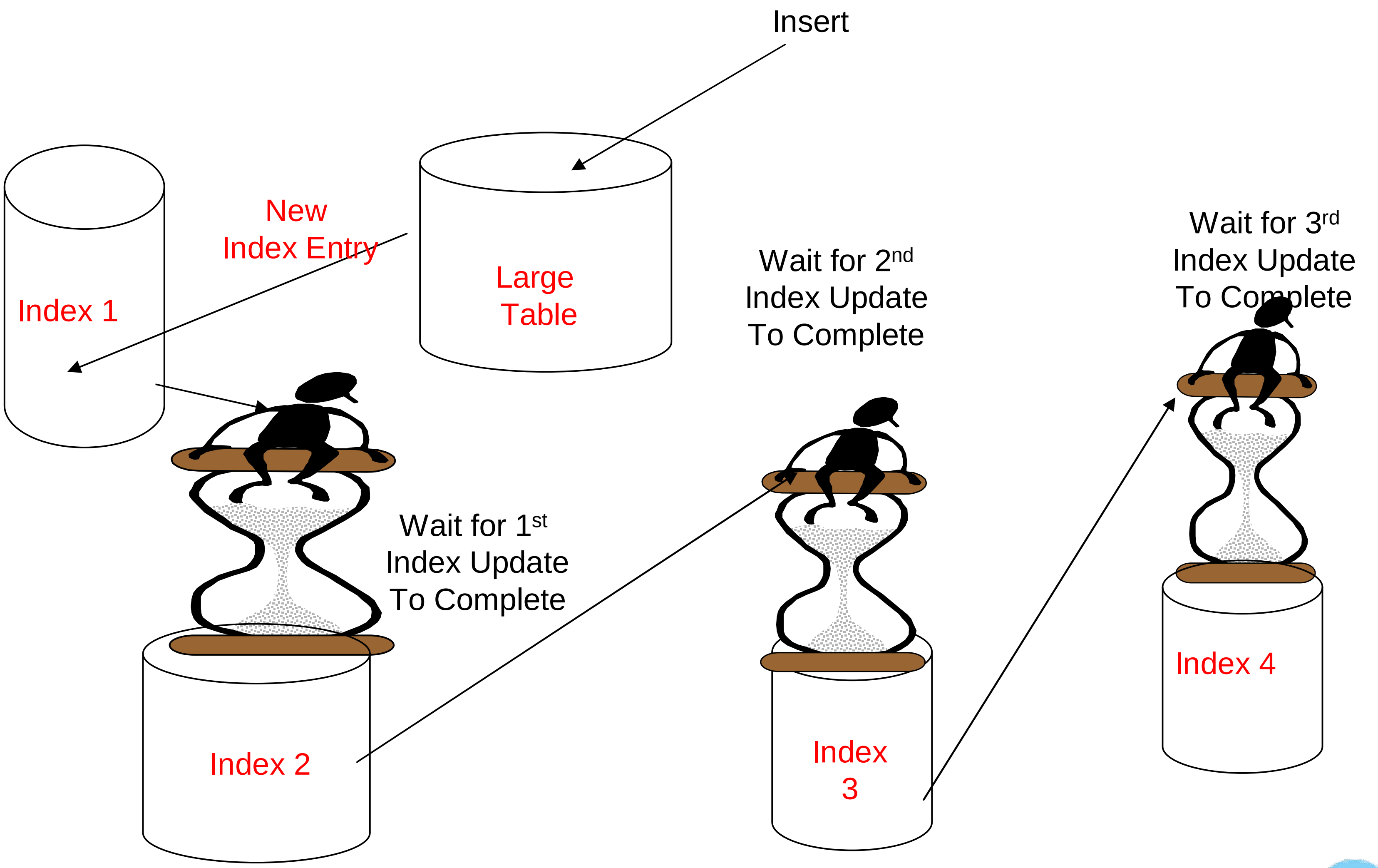
- Impove index look-aside performance
  - ➢Cache the last accessed leaf page
- Improve index split performance
  - ➢Avoid searching a free page from the first space map
- Cache the index root page in buffer pool during open
  - ➢Eliminate one GETP/RELP on each index access
- List prefetch for disorganized indexes – forward scan only
  - ➢Less need for REORG

Index Improvements – V10 …

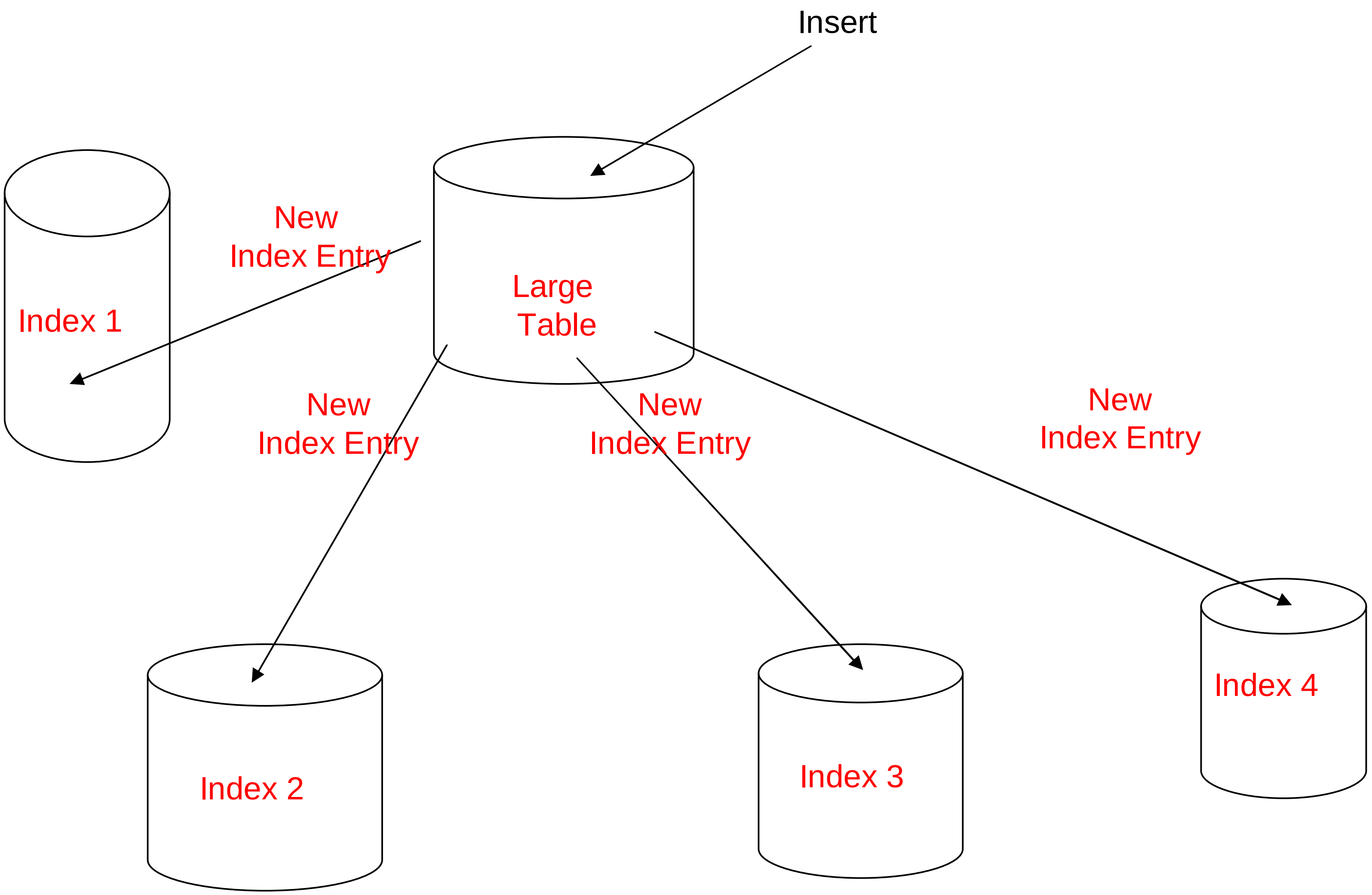


- Improve performance for Referential Integrity and nested loop join
  - ➢Enable index look-aside and sequential detect
- CPU reduction for index access especially with NOT PADDED indexes on VARCHAR
- Remove empty pages from the index tree is no longer limited to the covering space map on delete
- New IFCID 359 to monitor index split

Index Update without I/O parallelism



I/O Parallelism for Index Updates



Buffer Pool Enhancements

- Use 1MB real storage frame, instead of 4K, for PGFIX = YES Buffer Pools
  - ➢Reduce cpu overhead when accessing pages in BP
  - ➢Available only for z10 and zEnterprise
- Prefetch/Deferred Writes are running under zIIP
- Avoid BP scans in data sharing
  - ➢Switch from non-Shared to Shared and vice versa
  - ➢Data set close
  - ➢During STOP DB2
  - ➢Significantly improve performance for large BPs

Buffer Pool Enhancements …

- In-memory tables/Indexes
  - ➢A new PGSTEAL = NONE on ALTER BUFFERPOOL
  - ➢Load data into the buffer pool at data set open
    - Done by DB2 Prefetch Engines
  - ➢Ensure BP size is large enough to cache all data/index pages
  - ➢BP is managed in FIFO (First In First Out) order
    - Not managed in LRU
    - Disable all prefetch requests
  - ➢Query optimizer uses zero I/O cost for tables and indexes using this Buffer Pool
  - ➢Recommend to use AUTOSIZE(NO) which is the default
  - ➢Recommend to set CLOSE = NO for tables/indexes

Currently Committed Locking Option

- A new Bind/Prepare option – Currently Committed versus Wait for Outcome
  - ➢Applicable only for UTS table space
  - ➢Support both Row and Page level locking
- Applies to iso(cs) cd(no) to avoid lock wait for uncommitted insert/delete
- Applies to iso(rs) to avoid lock wait for uncommitted insert
- Still need to wait for uncommitted updated rows
- Pseudo delete data rows will always be done for UTS
  - – Space can be reused after commit

Auto-Compress

- Extend insert to build compression dictionary
  - ➢Applies to COMPRESS YES tables without a dictionary
  - ➢Start to compress rows without a REORG or LOAD
  - ➢Applicable for inserts due to online LOAD and MERGE SQL statement
- Dictionary is built by an asynchronous DB2 task
  - ➢Trigger based on DATASIZE in RTS catalog table
  - ➢Use iso(ur) to perform table scan existing data rows
- Use the default SYSTEMPAGES YES option to taken image copies
  - ➢ UNLOAD utility needs to have the dictionary before processing any data rows

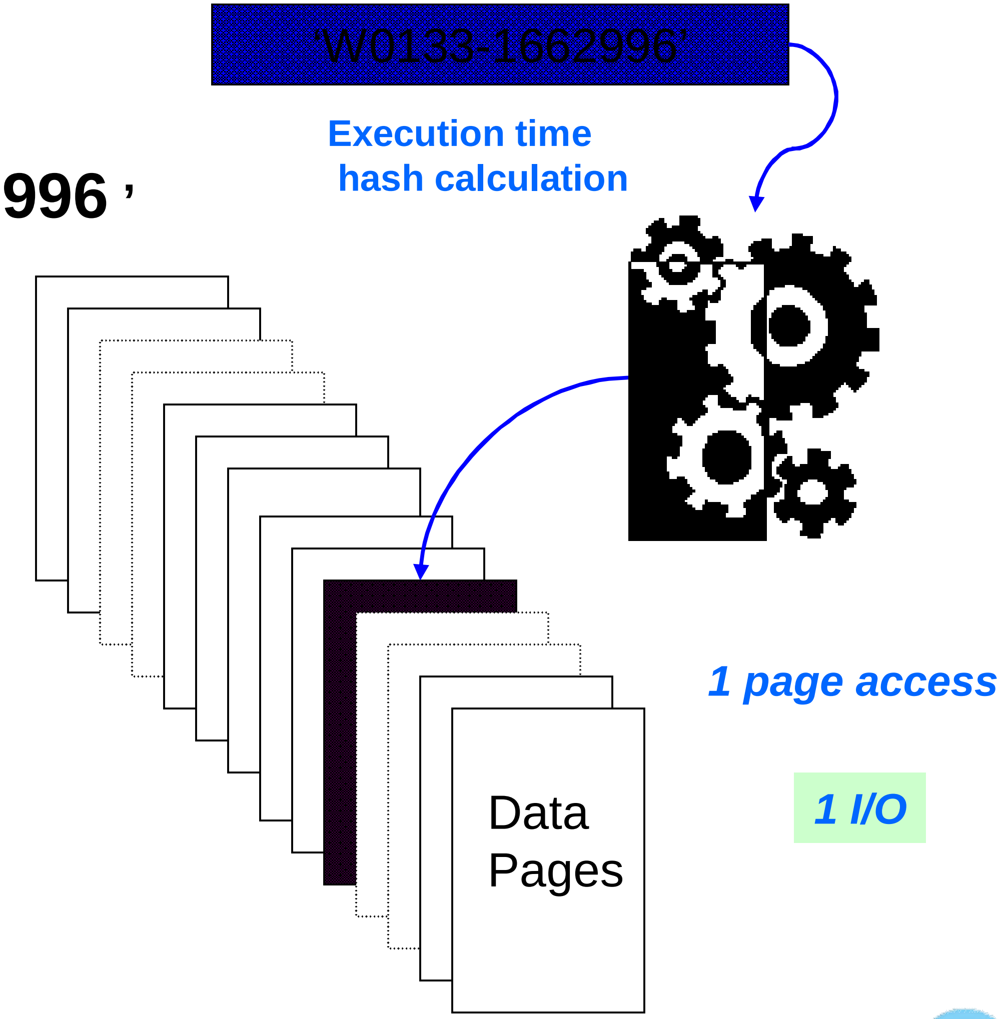
Hashed Table

SELECT * …WHERE
ITEMNO = 'W0133-1662996 '

Locate the data row
by hashing the key value.

-The hash Key must be unique
-Reduced page visits
-Reduced CPU & elapsed time
-Possibly eliminate an index

-Tradeoff: extra space used



'W0133-1662996'

Execution time
hash calculation

Data
Pages

1 page access

1 I/O

Hashed Table Access

- **Tables**
  - Hash access is good for tables:
    - With **a unique key**
    - Queried by applications (such as OLTP) needing **single row access via the unique key or IN predicates**
    - With **known approximate size**

- **Applications**
  - Hash access path will be chosen when:
    - The SELECT statement includes an **equal predicate on all hash key columns or IN-List**
    - DB2 APS determines via hash availability and the specific q **if hash access is suitable**

- **Syntax**
  - CREATE TABLE … ORGANIZE BY HASH UNIQUE (itemno) HASH SPACE 2G
  - **ALTER PBR/PBG into Hash**
    - **Table is not accessible through hash or Insert until REORG**

Universal Table Space Enhancements

- Member Cluster support for UTS
  - ALTER support – deferred alter with REORG
- ALTER to add new partitions to PBG
- Ability to create multiple partitions on create of PBG
  - Allow to use DSN1COPY to copy data
- PBR is the default for new ranged partitioned tables
  - Need to specify SEGSIZE 0 to create classic ranged partitioned tables
  - A ZPARM option which allows to change the default

Workfile Enhancements

- The workfile record size can be up to 64K bytes
  - Spanned workfile record to support JOINs and large SORTs
  - The sort key length can be up to 32K bytes
- PBG can be used for workfile table space
  - The maximum size can go beyond 64GB
  - MAXPARTITIONS can be used to limit the size
  - ALTER can't be done for
- Extend the usage of In-memory workfile
  - Allow simple search predicates

New RTS data in DB2 10

- REORGSCANACCESS
  - Number of queries use table scans
- REORGHASHACCESS
  - Number of queries use hash access
- REORGINDEXACCESS
  - Number of queries use this index to access data
- NPAGES – number of index pages that contains only pseudo deleted keys
  - Unused SYSINDEXSPACESTATS column in V9
- DRIVETYPE – HDD or SSD (Solid State Drive)

Logging Enhancements

- Reduce Log Write Latch (LC19) contention
  - ➢ Eliminate CPU spin loop to generate an unique LRSN within a data sharing member – V9
  - ➢ LRSN will only need to be unique for updates against the same data/index page – V9
  - ➢ Allow multiple rows inserted into the same data page with the same LRSN – V10
- Parallel log force for dual active logs
- Option to take system checkpoint based on time or logs, whichever occurs first
- Online to add new active log data sets via
  - – SET LOG NEWLOG(dsn) COPY(1 or 2)
  - – Recommend to format the data set with DSNJLOGF

Online REORG Enhancements

- Allow REORG SHRLEVEL REFERENCE after ALTER of limitkey
  - Involved partitions will still be marked in REORG pending
  - Helps to avoid data recovery when REORG failed
- SHRLEVEL(CHANGE) for LOBs
- Allow disjoint partition ranges – available in V9
- New AUX keyword for REORG to include LOB
  - Permit movement of rows between partitions
  - Allows DISCARD to delete LOB
- Option to cancel blocking threads
- Messages to estimate length of REORG phases and time to completion

Copy/Recover Enhancements

- Consistent COPY with Shrlevel Change
  - Will not quiesce applications
  - Need data set level FlashCopy capability
- Extend COPY, LOAD, and REORG to take backups using data set level FlashCopy
- Data set level FlashCopy backups as input to
  - RECOVER, COPY, DSN1COPY, DSN1PRNT
- Extend RECOVER to support a point-in-time recovery via rollback using logs

Other Miscellaneous Enhancements

- Major changes in catalog and directory
  - Use Row level Locking
    - Improve concurrency: BIND, DDL, utilities
  - Remove links from catalog tables
    - Many more table spaces
  - LOB columns for long strings
  - DB2/SMS-managed (not user-defined)
- Allow REPAIR utility to set an index part in rebuild pending
  - REBUILD INDEX PART after rows are inserted
- Eliminate the UTSERIAL lock for utilities

Summary

- Universal Table Space
  - Online to alter page size, DSSIZE, …
  - A new option to allow queries to access committed data without waiting for locks
- Index Enhancements
  - Online to alter page size,
  - Parallel index updates to improve insert performance
- Online to alter table space types
- Extend LOAD/REORG/COPY/RECOVER to support data set level FlashCopy
- More online REORG enhancements

Thank
You