Utilities Update – DB2 10 Preview

Haakon Roberts
IBM

Session Code: A14
10 Nov 2010 17:00-18:00 | Platform: z/OS

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.**

Agenda

- Recent DB2 9 enhancements
- Current DB2 9 work
- DB2 10 utility enhancements
- Summary

Delivery of enhancements in maintenance stream

- When prudent to do so
  - Risk vs. benefit
  - Resource
- Simplify utility processing
- Reduce resource consumption
- Improve performance
- Reduce CPU consumption

SORTNUM Elimination

- CHECK INDEX, REBUILD INDEX, REORG, RUNSTATS
- PK45916 (V8) & PK41899 (V9)
- Better performance, more robust, simpler
- SORTNUM no longer required
  - Difficult to estimate: failure if too low, excessive sort work allocation if too high
- New zparms UTSORTAL & IGNSORTN (online changeable)
  - UTSORTAL YES|NO
    - Use RTS data to estimate number of rows to sort
    - DB2 will dynamically allocate sort work datasets
      - If SORTWK DD cards not hard coded
  - IGNSORTN YES|NO
    - Override utility job setting of SORTNUM
- Recommendation
  - Turn on UTSORTAL, test it, then consider turning on IGNSORTN

DSNU3340I 168 08:13:52.66 DSNUGLSR - UTILITY PERFORMS DYNAMIC ALLOCATION OF SORT DISK SPACE

Other recent enhancements

- Permit use of ALIASes for LOAD, RUNSTATS and UNLOAD
  - PK77061 (V9)
- New DSNACCOX stored procedure to gather statistics from catalog and make utility recommendations
  - See PK44133
  - DSNACCOR still supported
- Better information for DPROPR/QRep or other IFI 306 readers
  - Write diag log record at utility termination so IFCID 306 readers can trigger refresh
  - PK78558 (V9)
- EAV dataset support
  - PK81151 (V8 & V9)
- Improved LOAD/UNLOAD processing with NUMRECS parameter
  - PK88970/PK88972/PK88974 (V9)
  - Replaces SORTKEYS at table space level with NUMRECS at table level
  - Simpler, eliminates risk of LOAD failure for load of multiple tables with skewed data distribution

Other recent enhancements

- LOAD/UNLOAD LOB file reference variable performance
  - PK75216 (V9)
  - PDS only, not HFS
  - 56% ET reduction on UNLOAD, 93% ET reduction on LOAD
- LOAD and UNLOAD to/from virtual file
  - USS named pipe support with templates
  - PK70269 (V8 & V9)
  - PK96023 (V8 & V9)
    - LBI on UNLOAD – 60% CPU reduction, 50% ET reduction
- LOAD COPYDICTIONARY
  - PK63324/PK63325 (V9)
  - REORG avoidance – prime empty partitions with compression dictionary

Performance – utility CPU consumption

- Focus on real CPU reduction & zIIP exploitation
- DB2 utilities have been zIIP-enabled since 2006
- Real CPU cost reduction in V9
  - 10-20% for COPY & RECOVER
  - 5-30% for LOAD, REORG, REBUILD INDEX
  - 20-60% for CHECK INDEX
  - 35% for LOAD partition
  - 30-40% for RUNSTATS INDEX
  - 40-50% for REORG INDEX
  - 70% for LOAD REPLACE partition with dummy input
- Flashcopy exploitation in DB2 10 dramatically reduces CPU consumption for COPY & reduces CPU for RECOVER & inline copies
- More zIIP offload in DB2 10 with RUNSTATS

Performance – zIIP exploitation for sort processing

- In spite of CPU reduction in V9, there is continued focus on CPU consumption for utilities
- Sort can consume ~60% of total utility CPU time
- DB2 in concert with DFSORT provides zIIP offload of DB2 utility memory-object fixed-length record sort processing
- Requirements:
  - DB2 APAR PK85889 (V8 or V9)
  - DFSORT APAR PK85856
  - z/OS 1.10
- PTFs can be applied independently of each other
- Exploitation is automatic

New solutions for DB2 9

- LOAD/UNLOAD FORMAT INTERNAL
  - Unload and load data in true internal format
  - Avoid field processing
  - 50% CPU reduction, 30% ET reduction measured for LOAD
- LOAD PRESORTED
  - Avoid sort overhead
  - Up to 25% CPU reduction, 33% ET reduction depending on no of indexes
  - Can combine with Utility Enhancement Tool PRESORT option
- Avoid FRVs for LOAD/UNLOAD of zero length LOBs
  - PM12286 (V9)
- Support REORG of multiple part ranges
  - PK87762 (V9)
  - E.g. REORG PART 1,45:71,500:503,4010
  - More efficient, improved availability, exploit parallelism

V10 REORG – improved availability & removed restrictions

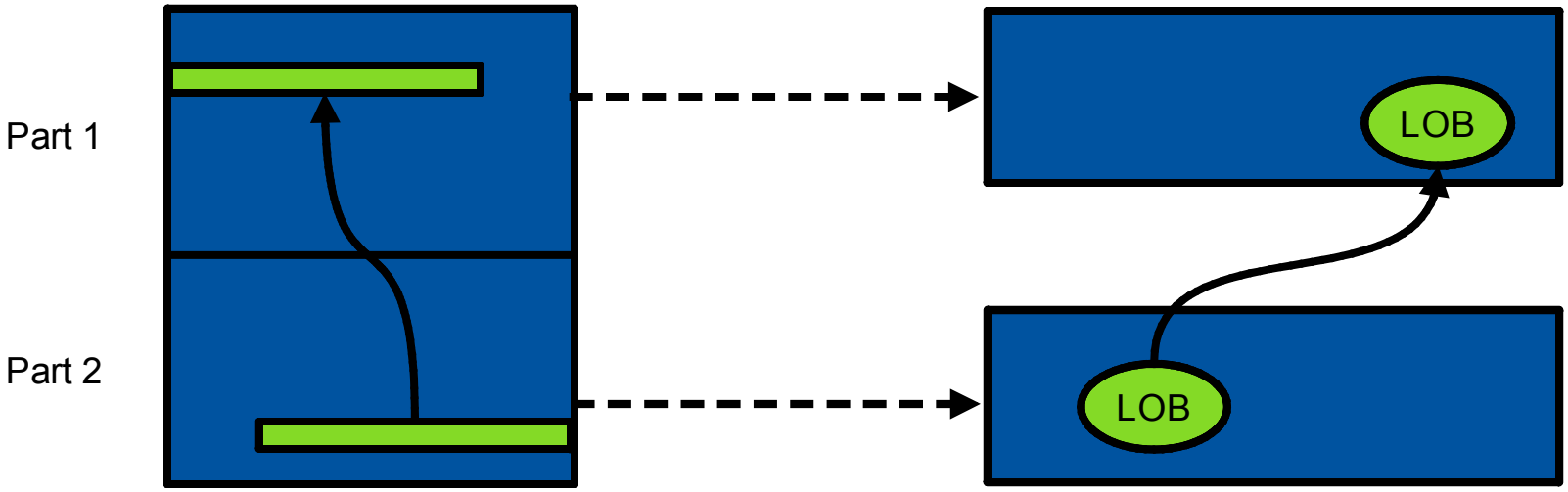
- Reduced need for REORG INDEX
  - List prefetch of index leaf pages based on non-leaf information for range scans
- Improved performance for part-level REORG with NPIs & REORG INDEX
  - Index list prefetch results in up to 60% elapsed time reduction
- Reduced need for REORG with compress on insert
- New REORGCLUSTERSENS RTS column
  - If no clustering-sensitive queries then avoid REORG to restore clustering
  - DSNACCOX enhanced
- REORG SHRLEVEL CHANGE for all cat/dir page sets
- REORG SHRLEVEL REFERENCE|CHANGE to remove REORP

V10 REORG – improved availability & removed restrictions

- REORG SHRLEVEL CHANGE for LOBs
  - Independent of whether LOBs are LOG NO or LOG YES
  - No mapping table required
  - Base table space must be LOGGED
- REORG FORCE option to cancel blocking threads
  - FORCE ALL or just READERS
  - Same process as –CANCEL THREAD so requires thread to be active in DB2 for it to be cancelled
  - Threads cancelled on final drain
- Reduced application outage for REORG with inline stats
  - Update catalog after dedrain
- REORG support for multiple part ranges
  - REORG support retrofitted to V9 in PK87762
    - LISTDEF support is not retrofitted
- REORG support for hashed table spaces
  - New AUTOESTSPACE parameter to determine hash space

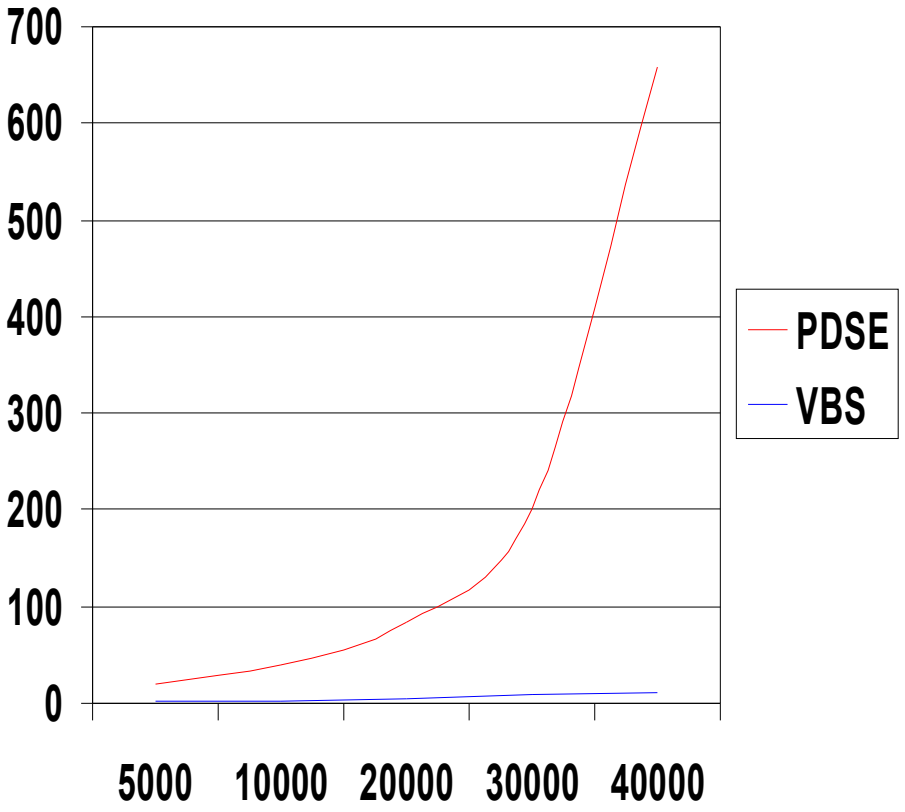
V10 REORG – improved availability & removed restrictions

- New AUX keyword on REORG of partitioned base for improved LOB handling
  - Permit rows to flow between partitions
  - Allows REORG REBALANCE with LOB columns
  - Allows ALTER of LIMITKEY with LOB columns
  - Permits move of rows between parts on PBG REORG
  - Permits deletion of corresponding LOBs on REORG DISCARD
  - Default is AUX NO unless LOB objects required to complete REORG
  - No XML column support for classic partitioned or PBR
  - No mapping table change



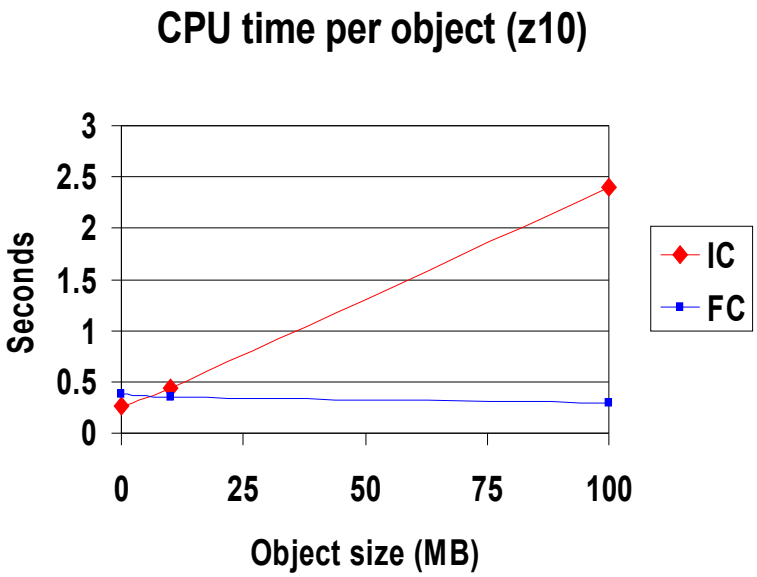
V10: LOAD/UNLOAD

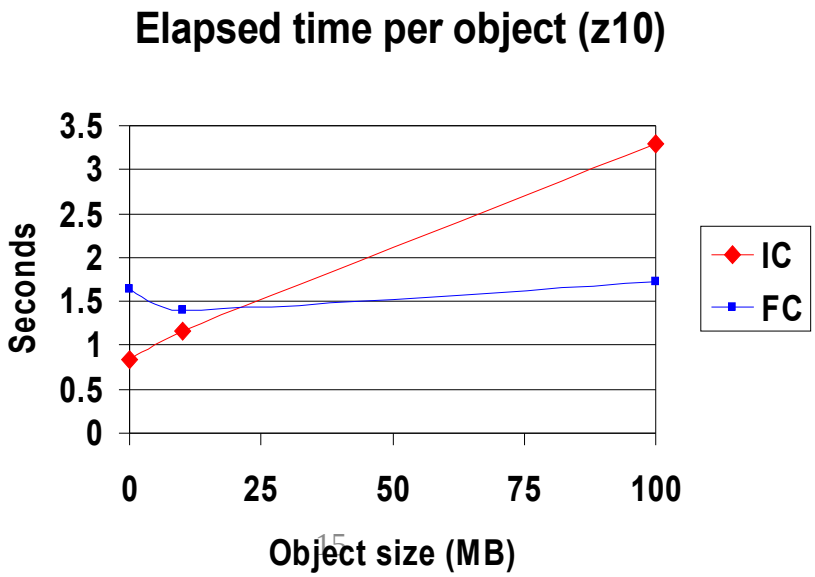
- Remove MAX_UTIL_PARTS zparm
  - Restriction removed for REORG in V9
- Improved performance for LOAD REPLACE with LOB data
  - Up to 50% elapsed time reduction
- Spanned record support for LOB/XML data
  - Option in addition to FRVs
  - Performance & portability



V10: COPY

- Dataset-level Flashcopy support
  - COPY, RECOVER, REORG, LOAD, REBUILD INDEX, REORG INDEX
  - New zparms & utility parms to govern
  - Significant CPU & elapsed time reduction
  - Create transaction-consistent imagecopies from SHRLEVEL CHANGE





V10: COPY

- Improved dataset management & performance
  - CHANGELIMIT will not allocate copy dataset unless copy taken
    - &ICTYPE now reflects actual image copy type
  - CHANGELIMIT to use RTS to determine full vs. incremental copy
  - Incremental copy will not allocate copy dataset unless pages changed
  - Insert dummy SYSCOPY record for incremental copy even though no pages changed
  - Delivery in maintenance stream post-GA

V10: RECOVER

- New BACKOUT YES option for point in time recovery
  - True rollback, not run of generated SQL undo statements
  - Requires COPY YES for indexes
- VERIFYSET option to fail PIT recovery if entire set not included
  - Base, LOB, XML, history – not RI
- ENFORCE option to avoid CHKP/ACHKP when subset of set recovered
  - Improved performance due to avoidance of set checking (RI, aux)

V10: Stats

- RUNSTATS PROFILE support for simplification
  - Includes ability to set profiles based on existing statistics
- RUNSTATS on views
- Autonomic features through new stored procedures & catalog tables
- All catalog statistics columns made updatable
- RUNSTATS SHRLEVEL REFERENCE updates RTS
  - TOTALROWS & TOTALENTRIES columns
- zIIP-enablement for RUNSTATS
- Auto sampling rates & page sampling instead of row sampling
  - Significant CPU & ET savings
  - TABLESAMPLE SYSTEM AUTO

V10: CHECK

- CHECK utilities will no longer set CHKP/ACHKP
- CHECK SHRLEVEL CHANGE default changed to fail if Flashcopy not available
- CHECK DATA enhanced for XML support
  - Document validation
  - Schema validation
- Automated exception table processing for XML documents

V10: Other

- Removed UTSERIAL lock for greater utility concurrency
- SQL SELECT on SYSLGRNX
  - Expect delivery post-GA
- LISTDEF & TEMPLATE enhancements
  - LISTDEF support for CHECK DATA
  - LISTDEF support for multiple part ranges on REORG
  - LISTDEF support for DEFINED YES|NO|ALL
    - Improved utility performance since unnecessary to build & then discard structures for undefined objects
    - Default changed to DEFINED YES & empty lists result in RC4
- REPORT RECOVERY support for SLBs
- DSNACCOX enhancements
  - Support hashed pagesets
  - New RTS columns for SSD, other

Summary

- This presentation does not cover utility support of core DB2 10 function that is available from day 1 of GA
  - Hashed tables
  - Materialisation of deferred alters
  - DEFINE NO for LOBs/XML
  - Etc.
- Continued delivery of performance improvements & features of real value
- Toleration, support & exploitation of new features from day 1
- Ensure utilities are non-disruptive
  - **Eliminate outages**
  - **Improve performance**
  - **Reduce resource cost**
- Reduce complexity & improve automation

Haakon Roberts
IBM
haakon@us.ibm.com

Session A14
DB2 10: Utilities Update