

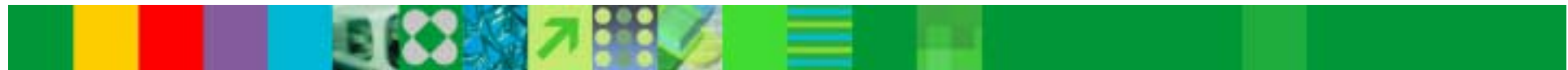


| IBM Software Group

# DB2 for z/OS Utilities Update

*Haakon Roberts,  
IBM Silicon Valley Lab  
haakon@us.ibm.com*

*August 2009*



# Agenda

---

- Availability
- Performance
- Features & function
- Summary

## Availability

---

- Ensure utilities are non-disruptive
- Introduction of shadow page set technology
- Introduction of claim & drain processing
- Exploitation of ISO(UR) processing
- SHRLEVEL NONE
- SHRLEVEL REFERENCE
- SHRLEVEL CHANGE

## Availability – what has changed recently?

---

- Online create or rebuild of non-unique indexes
  - ▶ REBUILD INDEX SHRLEVEL CHANGE
- Eliminate outage for partition-level REORGs
  - ▶ Eliminate BUILD2 phase
  - ▶ New restriction on concurrent part level REORGs
- REORG avoidance for data compression
  - ▶ LOAD COPYDICTIONARY
  - ▶ PK63324 & PK63325 (V9)
- Online data consistency checking and repair
  - ▶ CHECK DATA SHRLEVEL CHANGE
  - ▶ CHECK LOB SHRLEVEL CHANGE
  - ▶ REPAIR LOCATE... SHRLEVEL CHANGE

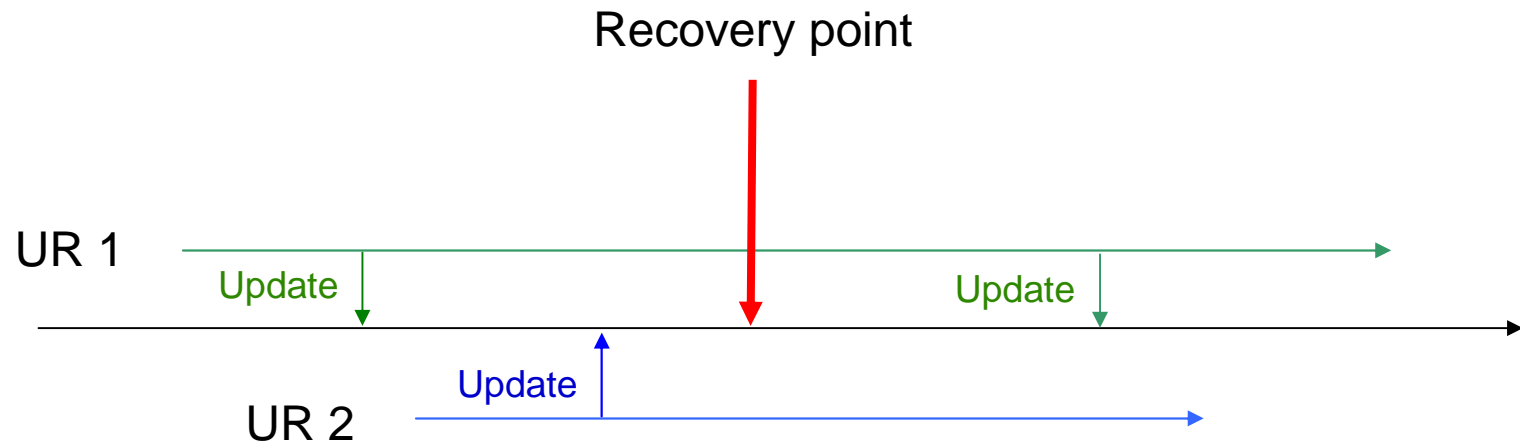
## Availability – what has changed recently?

---

- Run data consistency checks without impacting BACKUP SYSTEM or disk mirroring
  - ▶ PK41711 (V9)
- Avoid CHKP after PIT recovery of RI set in V9
  - ▶ PK80304 (V9)

## Availability – what has changed recently?

- Replace data with virtually no outage
  - ▶ CLONEs effectively provide LOAD REPLACE SHRLEVEL CHANGE
  - ▶ UTS only
- Read LOB data during REORG
  - ▶ REORG SHRLEVEL REFERENCE for LOBs
- RECOVER to point in time with consistency
  - ▶ Avoid need for QUIESCES



## Performance

---

- What is important to you?
- Elapsed time
  - ▶ DB2 enhancements
  - ▶ z/OS & architecture improvements
  - ▶ Parallelism
- CPU cost
  - ▶ DB2 enhancements
  - ▶ z/OS & architecture improvements
  - ▶ zIIP

## Performance – what has changed recently?

---

- Faster REORGs
  - ▶ Parallel unload of partitions
  - ▶ Parallel reload of partitions
  - ▶ Parallel log apply
    - ▶ Greater likelihood of REORG keeping up with logging rates
- Faster CHECK INDEX SHRLEVEL REFERENCE
  - ▶ Parallel index processing
- Up to 40% faster COPY & RECOVER RESTORE phase to/from tape
  - ▶ Support Large Block Interface for image copies to tape
- Reduced impact on applications when running COPY
  - ▶ COPY uses MRU for buffers to improve BP hit ratio for online applications
- Reduced impact on applications when running LOAD & REORG
  - ▶ Auto-invalidate of cached dynamic statements on completion of LOAD & REORG
  - ▶ PK47083 (V8 & V9)



## Performance – what has changed recently?

---

- Greater utility parallelism with SORTNUM elimination
  - ▶ PK45916 (V8), PK41899 (V9)
  - ▶ Major improvement in utility sort processing
  - ▶ Simpler, more efficient, more reliant on RTS
- SORTBLD performance improvement
  - ▶ PK60956 (V8 & V9)
  - ▶ Up to 20X performance improvement in SORTBLD for indexes with small SECQTY
- LOAD & REORG performance improvement
  - ▶ PK61759 (V8 & V9)
  - ▶ 10% CPU & elapsed time improvement in RELOAD phase
  - ▶ 10% CPU reduction in SORT phase
- COPY performance improvement
  - ▶ PK74993 (V9)
  - ▶ 20% elapsed time improvement for copy of multiple small datasets to tape
- COPY performance with large LISTDEF lists
  - ▶ PK78865 (V8 & V9)
  - ▶ Reduce writes to SYSUTILX

## Performance – what has changed recently?

---

- Crossloader performance improvement for CCSID data conversion
  - ▶ PK76860 (V8 & V9)
- LOAD/UNLOAD LOB file reference variable performance
  - ▶ PK75216 (V9)
  - ▶ PDS only, not HFS
  - ▶ 56% ET reduction on UNLOAD, 93% ET reduction on LOAD
- UNLOAD performance for multi-table table spaces
  - ▶ UTILINIT phase – use DBD rather than catalog lookup
  - ▶ PK77313 (V8 & V9)
  - ▶ In one case 1.5 hours -> 11 secs

## Performance – what has changed recently?

---

- REORG PART of empty partition performance
  - ▶ Avoid NPI scan for non-clustering indexes
  - ▶ PK67154 (V8 & V9)
  - ▶ Sample SORTBLD phase: 98% CPU reduction, 70% ET reduction
- COPY of partitioned tablespace with many parts
  - ▶ PK81232 (V9)
  - ▶ Correct CPU regression in V9 - up to 80% CPU reduction
- COPY SHRLEVEL CHANGE performance improvement for LOBs
  - ▶ PK83096 (V9)

## Performance – what has changed recently?

- **LOAD and UNLOAD to/from virtual file**
  - ▶ USS named pipe support with templates
  - ▶ PK70269 (V8 & V9)
- **DSN1COPY performance**
  - ▶ Improved VSAM buffer allocation for page sets with cylinder allocation
  - ▶ Up to 20% ET improvement
  - ▶ PK78516 (V8 & V9)
- **RUNSTATS histogram statistics**
  - ▶ Improved query optimization for non-uniform distribution
  - ▶ Example - 1, 3, 3, 4, 4, 6, 7, 8, 9, 10, 12, 15 (sequenced), cut into 3 quantiles

Seq No	Low Value	High Value	Cardinality	Frequency
1	1	4	3	5/12
2	6	9	4	4/12
3	10	15	3	3/12

## Performance – what has changed recently?

---

- 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
- zIIP enablement for utility index processing in V8

## Performance – what has changed recently?

---

- 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 will provide 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

## Performance – UNLOAD

---

- Two UNLOAD products from IBM
  - ▶ DB2 UNLOAD Utility (in the IBM DB2 Utilities Suite)
  - ▶ DB2 High Performance Unload (HPU) Utility
  - ▶ (DSNTIAUL is only a sample!)
- HPU was delivered before the UNLOAD utility – had this not been the case, we would never have used the words “High Performance”
- In elapsed time, they are comparable (sometimes UNLOAD is faster, sometimes HPU is faster)
- In CPU time, HPU consumes approximately half the CPU in many situations (but not always)
- UNLOAD is geared towards user of DB2 Utilities (Utilities interface)
- HPU is geared towards application developers (SQL interface)

## Features & function

---

- More powerful utilities for greater flexibility...
- ... yet simpler utilities for reduced complexity
- New utilities & more options
  - ▶ COPYTOCOPY
  - ▶ BACKUP SYSTEM & RESTORE SYSTEM
  - ▶ LISTDEF
  - ▶ TEMPLATE
  - ▶ File Reference Variables
  - ▶ ...
- Intelligent defaults
- Autonomics
- Synergy with Information Management Tools

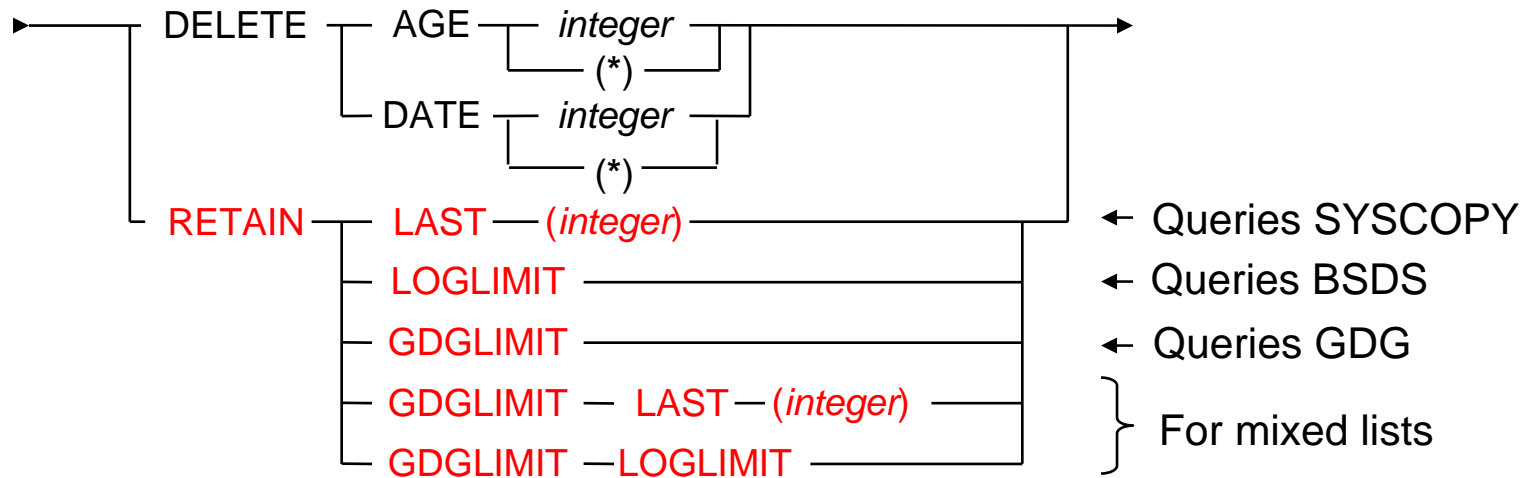


## Features & function – what has changed recently?

- BACKUP SYSTEM & RESTORE SYSTEM enhancements
  - ▶ Support for tape
  - ▶ Support for incremental FlashCopy
- Object-level recovery from system-level backup
- RECOVER to any point in time with consistency
- SORTNUM elimination
  - ▶ Simplified utility invocation
- Remove restriction on REORG of >254 compressed parts
  - ▶ ZPARM restricts LOAD in V9 – restriction removed in X
- 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)

# Features & function – what has changed recently?

- **MODIFY RECOVERY simplification & safety**



- **Template switching for COPY utility**

- ▶ E.g. copy to disk if small, to tape if large

```

TEMPLATE LRG DSN &DB..&TS..D&DA..T&TI. UNIT=TAPE
TEMPLATE SML DSN &DB..&TS..D&DA..T&TI. UNIT=SYSALLDA LIMIT(20 CYL, LRG)
COPY TABLESPACE SMALL.TS COPYDDN(SML)
COPY TABLESPACE LARGE.TS COPYDDN(SML)
    
```

## Features & function – what has changed recently?

- 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
- More information
  - ▶ All utility messages in job output have julian date & timestamp
  - ▶ -DISPLAY UTILITY enhanced to show progress of logapply

**DSNU116I csect-name RECOVER LOGAPPLY PHASE DETAILS:**

**STARTING TIME = timestamp**

**START RBA = ss START LRSN = rr**

**END RBA = ee END LRSN = nn**

**LAST COMMITTED RBA = cc LAST COMMITTED LRSN = ll**

**ELAPSED TIME = hh:mm:ss**

## What's coming?

---

- Remove usability restrictions for REORG
  - ▶ LOBs
  - ▶ PBG
  - ▶ Catalog/directory SHRLEVEL CHANGE REORG
  - ▶ Rebalance of partitioned page sets with LOB columns
  - ▶ Disparate parts
- REORG avoidance
- Remove UTSERIAL lock for greater utility concurrency
- RTS enhancements & greater reliance upon RTS
- Intelligent & autonomic statistics gathering
- BACKUP SYSTEM / RESTORE SYSTEM enhancements

## What's coming?

---

- FlashCopy exploitation
- Faster & better COPY processing
  - ▶ Incremental, CHANGELIMIT, FlashCopy
- LOAD & UNLOAD enhancements
  - ▶ Improved LOB/XML processing
  - ▶ Improved UTF-16 support
  - ▶ Performance options
- CHECK utility enhancements
  - ▶ XML, availability, data correction,...
- Faster point in time recovery
- ... and more

## Summary

---

- Continuing commitment to, & investment in, utilities
- Toleration, support & exploitation of new features from day 1
- Ensure utilities are non-disruptive
  - ▶ Eliminate outages
  - ▶ Improve performance
  - ▶ Reduce resource cost
- Provide function that adds real value
- Reduce complexity & improve automation