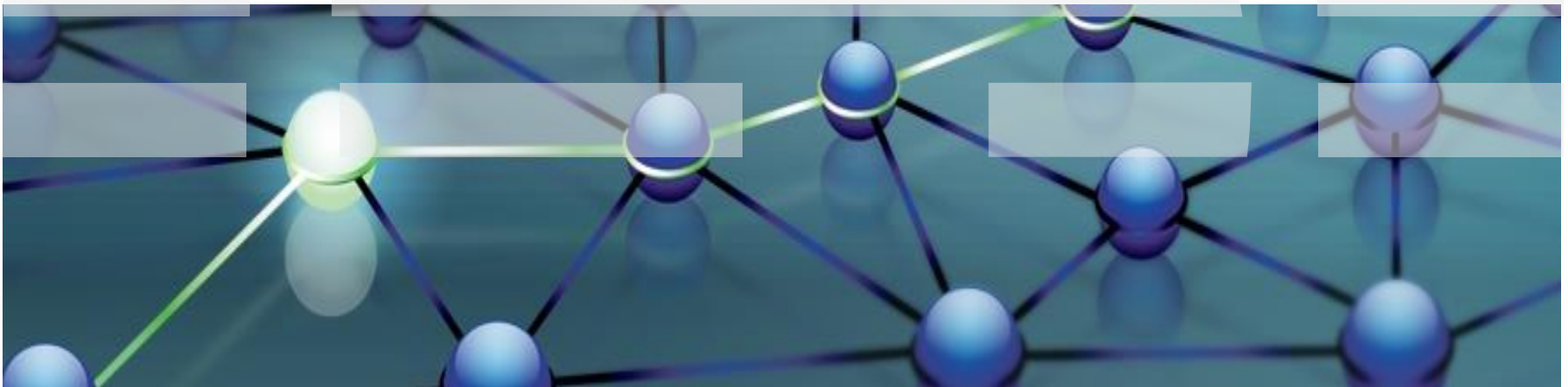


# Db2 for z/OS Utilities – Hints and Tips

Feb 2020

Haakon Roberts  
haakon@us.ibm.com



## Disclaimer

- Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

---

## Agenda

- General recommendations
- COPY
- RECOVER
- LOAD/UNLOAD
- REORG
- RUNSTATS
- CHECK
- DSN1COPY
- Summary

## General Recommendations

- New function, performance and availability enhancements are continually being released via APARs and PTFs
- Visit the Db2 for z/OS News from the Lab blog
  - <http://ibm.biz/db2znews>
- Keep an eye on recent enhancements delivered in Db2
  - [https://www.ibm.com/support/knowledgecenter/en/SSEPEK\\_12.0.0/wnew/src/tpc/db2z\\_minorenancementsinapars.html](https://www.ibm.com/support/knowledgecenter/en/SSEPEK_12.0.0/wnew/src/tpc/db2z_minorenancementsinapars.html)
  - [https://www.ibm.com/support/knowledgecenter/en/SSEPEK\\_12.0.0/wnew/src/tpc/db2z\\_db2functionlevels.html](https://www.ibm.com/support/knowledgecenter/en/SSEPEK_12.0.0/wnew/src/tpc/db2z_db2functionlevels.html)

## General Recommendations

- Defaults are now generally recommended
- Basic recommendations are assumed and not covered in this session – e.g.
  - SHRLEVEL CHANGE
  - Parallelism for improved performance
  - Inline copy to avoid copy-pending
  - Work datasets allocated to disk, not tape
  - SORTNUM elimination
    - SORTNUM or hard-coded sortwork datasets should be a rare exception

---

## General Recommendations

- Maximise prefetch for utilities performance
  - Up to 128 pages
  - $VPSEQT * VPSIZE \geq 80,000$

## PBGs

- No REORG partition parallelism
- No LOAD or UNLOAD partition parallelism
- No support for loading or unloading subsets of partitions
- V12 supports growing a new partition even if reorging a subset of parts
  - Prior to V12, needed to include last partition or reorg entire tablespace to avoid failures
  - REORG\_IGNORE\_FREESPACE zparm removed
- New DROP\_PART REORG parameter in V12
  - Complements zparm delivered in V11
  - No support for PIT recovery prior to REORG that removed partitions
- REORG of multiple parts will default to AUX YES and pull in LOB table spaces
  - Expensive, but necessary to allow move of rows between parts
  - Skipped if MAXPARTITIONS is 1 – not if # parts is 1

## COPY

- COPY FLASHCOPY CONSISTENT does not need FlashCopy support
  - Create transaction-consistent image copies without any application impact
- RECOVER from a transaction-consistent copy can take longer
- If you do not take incremental copies, then alter pagesets to TRACKMOD NO
  - Reduced application overhead by avoiding maintaining modified page flags in spacemap pages
- FlashCopy to an XRC primary volume is now supported
- Use LBI for better COPY performance when writing to tape
  - Ref. TAPEBLKSZLIM in DEVSUPxx
- Consider copying large indexes
  - RECOVER is likely to be a lot faster than REBUILD INDEX
- Leave BUFNO alone – let the utility choose a default value
- Maximize prefetch quantity for faster performance



## Backup solutions

- **BACKUP SYSTEM**
  - Volume-level FlashCopy
  - Significant DASD investment required
  - Can be complex to set up & administer, but invocation simple
- **Sequential image copies**
  - Tried and trusted solution since V1.1
- **Other external backups, such as volume-level backups, DSN1COPY**
  - Outside of Db2's control
  - Requires careful management and co-ordination
- **FlashCopy**
  - DASD investment required
  - Massive CPU, ET, resource reduction on z
- **Choice is dependent on environment and requirements, all options will continue to be supported**

## RECOVER

- Duration is (restore + log apply)
- Aim is to minimize one or the other or both
- Consider using FlashCopy
  - Careful: Migrated FCICs are not automatically recalled by hsm and will be skipped by RECOVER
- Maximise exploitation of parallel restore and Fast Log Apply
  - Recover multiple objects in a list in parallel but ideally <100
  - Avoid running more than 51 RECOVER jobs per subsystem
- Split off page sets that are not updated and recover separately
- For recovery of many objects, ruthlessly prioritise based on application criticality

# RECOVER

## ■ PIT recovery:

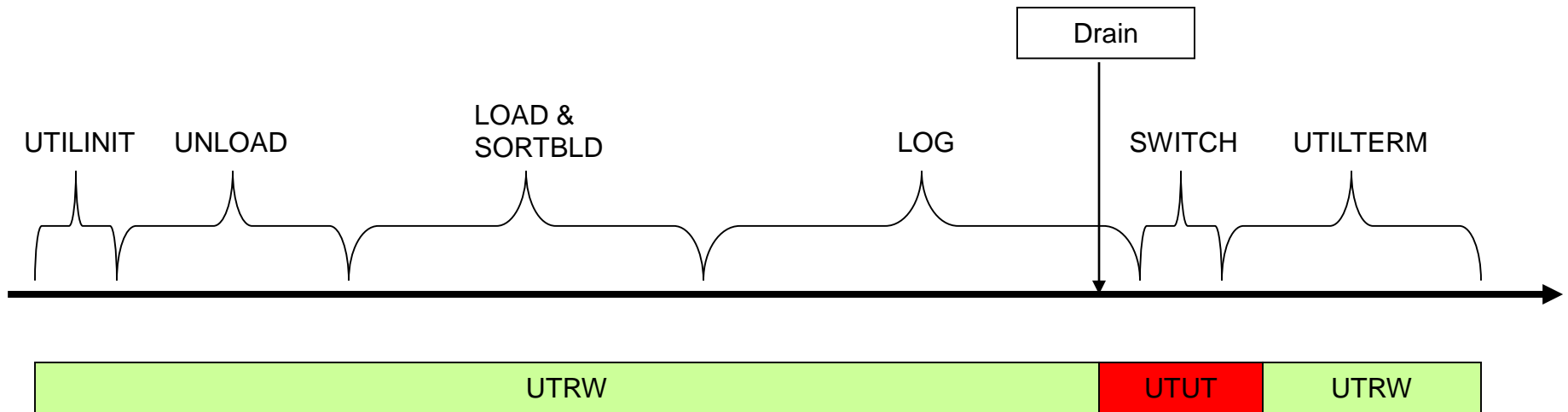
- Copy indexes and include in recovery list
- Consider SCOPE UPDATED and BACKOUT YES
  - SCOPE UPDATED is the default in V12
- Include whole RI set in same RECOVER statement
- Include base and aux objects in same RECOVER statement
- Consider using ENFORCE NO VERIFYSET NO to reduce outage duration
  - ENFORCE NO means CHKP/ACHKP will not be set if entire set is not included
  - VERIFYSET NO allows RECOVER to run even though related objects are not included

## LOAD/UNLOAD

- For LOAD REPLACE, use SHRLEVEL REFERENCE
  - Avoid leaving object in RECP if utility fails
- Use SORTDEVT to drive parallel index build
- If input data can be split, then partition parallelism is likely to be quicker than a single input dataset with the PARALLEL option
  - Exception is if there is a lot of data skew
- For LOAD SHRLEVEL CHANGE use the PARALLEL keyword
- Specify NUMRECS if input is on tape or variable length, not SORTKEYS
- For LOAD RESUME, take inline image copies to avoid copy-pending or need for a separate COPY step
- Use SPANNED YES for tables with LOB or XML columns
- Consider REGISTER NO option for UNLOAD in V12 to cut data sharing overhead
- If table is partitioned by date/timestamp, use LOAD COPYDICTIONARY to pre-allocate compression dictionaries to empty partitions

# REORG - Introduction

- SHRLEVEL CHANGE & DRAIN ALL
  - Not to scale



# REORG

- Two key parameters governing outage duration:
  - MAXRO
    - Number of seconds of processing remaining on the log before we attempt to drain
  - DRAIN\_WAIT
    - Number of seconds that the drain attempt should wait for before giving up
- SHRLEVEL CHANGE main recommendations
  - Use DRAIN ALL to minimize application impact
  - Use TIMEOUT TERM to free up objects on timeouts
  - $(DRAIN\_WAIT + MAXRO) < (IRLMRWT - 5 \text{ or } 10 \text{ seconds})$  to avoid application impact
  - RETRY=6
  - $RETRY\_WAIT = DRAIN\_WAIT * RETRY$
  - Consider MAXRO DEFER & -ALTER UTILITY or SWITCHTIME if REORG needs to complete in short window
- FlashCopy inline copy reduces outage duration
  - Avoid sequential inline copy processing after drain

## REORG

- Careful using part-level inline copies to tape, at least until PI75518 is available
  - Each inline copy must be open for write simultaneously
  - Part-level inline copies are required for PBR RPN
- If interested in the FORCE option:
  - Look for PH12240
  - Careful cancelling long-running URs that must roll back

## REORG INDEX vs. REBUILD INDEX

- REORG INDEX can materialize schema changes
- REBUILD INDEX does not materialize schema changes
- If index is disorganized, then REBUILD INDEX can be faster than REORG INDEX
- REBUILD INDEX SHRLEVEL CHANGE does have an availability impact
  - The data is UTRW, but the index is UTUT, RBDP
- Use CREATE INDEX DEFER YES followed by REBUILD INDEX SHRLEVEL CHANGE to create new non-unique indexes



## RUNSTATS

- Do not use RUNSTATS to gather space statistics – rely on RTS
- Use TABLESAMPLE SYSTEM AUTO
  - Db2-managed page sampling
  - Db2 12 FL505 will by default override row sampling to this
- Don't bother running RUNSTATS on LOB table spaces
  - RTS contains all the information you need
- Use stats profiles in V12
- Use new REGISTER NO option in V12 to avoid data sharing overhead

# CHECK

- If you have FlashCopy, then run CHECK SHRLEVEL CHANGE
  - If you don't, then don't!
- CHECK SHRLEVEL CHANGE does not reset CHKP/ACHKP
  - Plan to change this in an APAR in future
- If you know your data well then consider avoiding setting CHKP/ACHKP in the first place
  - Use NOCHECKPEND in LOAD or REORG DISCARD
  - Use ENFORCE NO in PIT recovery

## DSN1COPY

- DSN1COPY runs standalone and cannot ensure that data matches definition at target
- All target datasets must be pre-allocated for multi-piece table spaces
- XML
  - Data-dependent information is kept in XMLSTRINGS so DSN1COPY of XML data from one Db2 system to another will not work
- Db2 12 has enhanced validation on pageset open to check for schema mismatch
  - Both table space and table definitions are checked
- Run REPAIR CATALOG to fix versioning information and validate schema matches after DSN1COPY

# DSN1COPY

- REPAIR CATALOG – major improvement over REPAIR VERSIONS

DSNU650I <DBU2 346 21:32:35.19 DSNUCBVR - CATALOG TABLESPACE GK200541.TSGK2100  
DSNU671I <DBU2 346 21:32:35.30 DSNUCBVR - DBID=X'03FE' PSID=X'0049' OBID=X'0060'  
TABLE SCHEMA IN THE CATALOG DOES NOT MATCH THE PAGE SET.  
DSNU667I <DBU2 346 21:32:35.30 DSNUCBVR - NUMBER OF COLUMNS IS 5 IN THE CATALOG, BUT 9 IN THE PAGE SET.  
DSNU667I <DBU2 346 21:32:35.30 DSNUCBVR - COLUMN 1 DIFFERENCE:  
DSNU667I <DBU2 346 21:32:35.30 DSNUCBVR - TYPE IS CHARACTER IN THE CATALOG, BUT TIMESTAMP/TSTZ IN THE  
PAGE SET.  
DSNU667I <DBU2 346 21:32:35.30 DSNUCBVR - REPAIR CANNOT FIX THE ERROR.  
DSNU667I <DBU2 346 21:32:35.30 DSNUCBVR - TABLE DEFINITION ERROR CANNOT BE FIXED.  
DSNU690I <DBU2 346 21:32:35.30 DSNUCBVR - CAN NOT UPDATE INFORMATION IN THE CATALOG  
THAT DOES NOT MATCH THE PAGE SET  
DSNU012I 346 21:32:35.35 DSNUGBAC - UTILITY EXECUTION TERMINATED, HIGHEST RETURN CODE=8

## DSN1COPY

### ■ Table versioning improvements

- New version 0 system page support makes pagesets self-describing to avoid DSN1COPY failure when pageset only contains version 0 records but table has been altered to a higher version
- REORG, LOAD REPLACE, INSERT or UPDATE will now insert a version 0 system page
- REPAIR enhanced with two new options
  - INSERTVERSIONPAGES inserts missing system pages for tables that are version 0 to avoid need for REORG
  - SETCURRENTVERSION updates catalog current version for all tables to the same number as CURRENT\_VERSION in SYSTABLESPACE. REORG will then bring all tables in table space up to the same version number
- Ref. PI86880, PI88940

---

## Summary

- Stay reasonably current on maintenance
- Keep up to date with utility enhancements
- Revisit existing utility jobs to benefit from new options