



# DB2 9 for z/OS Utilities

Road show edition  
*June 2008*



**IBM Information**  
On Demand **2008**  
October 26 - 31, 2008 ~ Las Vegas  
The Premier Information Management  
Global Conference  
[www.ibm.com/events/informationondemand](http://www.ibm.com/events/informationondemand)

Information Management software

## Agenda

- But first...
- DB2 9 Utilities Enhancements
  - ▶ Support for non-utility enhancements
  - ▶ Availability enhancements
  - ▶ Performance enhancements
  - ▶ Backup and recovery enhancements
  - ▶ Other enhancements



But first...

*A word for those who are not running DB2 9 now.*



**IBM Information**  
On Demand **2008**

October 26 - 31, 2008 ~ Las Vegas  
The Premier Information Management  
Global Conference

[www.ibm.com/events/informationondemand](http://www.ibm.com/events/informationondemand)

Information Management software

## SORTNUM elimination in V8 / DB2 9

- PK45916 (V8) & PK41899 (DB2 9)
- SORTNUM no longer required
  - ▶ Correct value hard to determine, utility failure if too low or excessive sort work allocation if too high
  - ▶ DSNU3340I - UTILITY PERFORMS DYNAMIC ALLOCATION OF SORT DISK SPACE
- New zparms UTSORTAL & IGNSORTN (online changeable)
  - ▶ UTSORTAL
    - Yes, use RTS data to estimate number of rows to sort
    - No (or if RTS data not available), Use old method
    - DSNU3343I csect-name REAL-TIME STATISTICS INFORMATION MISSING FOR obj-type obj-qual. obj-name partition-information
  - ▶ IGNSORTN
    - Yes, override utility job setting of SORTNUM
    - No, utility job setting takes precedence
- Utilities:
  - ▶ CHECK INDEX, REBUILD INDEX, REORG TABLESPACE, RUNSTATS with COLGROUP

## SORTNUM elimination in V8 / DB2 9

- PK45916 (V8) & PK41899 (DB2 9)
- SORTNUM no longer required
  - ▶ Correct value hard to determine, utility failure if too low or excessive sort work allocation if too high
  - ▶ DSNU3340I - UTILITY PERFORMS DYNAMIC ALLOCATION OF SORT DISK SPACE
- New zparms UTSORTAL & IGNSORTN (online changeable)
  - ▶ UTSORTAL
    - Yes, use RTS data to estimate number of rows to sort
    - No (or if RTS data not available), Use old method
    - DSNU3343I csect-name REAL-TIME STATISTICS INFORMATION MISSING FOR obj-type obj-qual. obj-name partition-information
  - ▶ IGNSORTN
    - Yes, override utility job setting of SORTNUM
    - No, utility job setting takes precedence
- Utilities:
  - ▶ CHECK INDEX, REBUILD INDEX, REORG TABLESPACE, RUNSTATS with COLGROUP

## Other recent maintenance changes

- PK61759 (V8 & 9)
  - ▶ LOAD and REORG performance improvement
  - ▶ Sort processing performance improvement
    - 15% CPU reduction
  
- PK63324 and PK63325 (9)
  - ▶ LOAD COPYDICTIONARY
    - Allow priming of a partition with a compression dictionary
  
- PK60956 (V8 & 9)
  - ▶ SORTBLD performance improvement for indexes with small SECQTY
  - ▶ SORTBLD elapsed up to 20x improvement!!!
  - ▶ Recommendation: Set appropriate PRIQTY/SECQTY to avoid extend processing
  
- PK47093 (V8 and 9)
  - ▶ LOAD and REORG invalidation of dynamic statement cache

## Other recent maintenance changes

- PK59399 (9)
  - ▶ Allow DB2 to use DFSORT parameters from PARMLIB
  - ▶ Requires z/OS r10
  - ▶ Install options can be changed without reassembling ICEMAC module
  
- PK56334 (V8 & 9)
  - ▶ Allow REORG to reset LPL & WEPR

## DB2 9 Utilities Highlights

- More online utilities
  - ▶ Rebuild Index SHRLEVEL CHANGE
  - ▶ Reorg LOB now supports SHRLEVEL REFERENCE (space reclamation)
  - ▶ Check data, LOB and repair locate ... SHRLEVEL CHANGE
  - ▶ Check index SHRLEVEL REFERENCE supports parallel for > 1 index
  - ▶ Clones for “online LOAD REPLACE”
- Online REORG BUILD2 phase elimination
- REORG parallelism for UNLOAD, RELOAD, LOG phases
- Utility TEMPLATE switching
- UNLOAD SKIP LOCKED DATA option



## DB2 9 Utilities Highlights...

- MODIFY Recovery enhancements
  - ▶ “*Retain*” keyword added to improve management of copies
    - LAST(n), LOGLIMIT, GDGLIMIT
- Volume-based COPY/RECOVER (BACKUP SYSTEM/RESTORE SYSTEM)
  - ▶ RECOVER modified to enable object-level recovery from volume FlashCopy
  - ▶ Full integration of tape into BACKUP/RESTORE SYSTEM utilities
  - ▶ Incremental FlashCopy, APAR PK41001
- Truncate log based on timestamp
- RECOVER to any point-in-time with consistency
- RECOVER RESTOREBEFORE to use an earlier image copy
- Display progress of RECOVER during log apply
- COPY CHECKPAGE option always active
  - ▶ “Copy Pending” avoided if broken page encountered
- COPY SCOPE PENDING to copy only objects in “Copy Pending”

## DB2 9 Utilities Performance Improvements

### CPU reductions in LOAD, REORG, and REBUILD

- Reductions mostly due to improved index processing (\* with exceptions)
- 10 to 20% in Image Copy\* (even with forced CHECKPAGE YES)
- 5 to 30% in Load, Reorg, Reorg Partition, Rebuild Index
  - ▶ Except REORG TABLESPACE SHR CHG PART with NPSIs
- 20 to 40% in Load
- 20 to 60% in Check Index
- 35% in Load Partition
- 30 to 40% in Runstats Index
- 40 to 50% in Reorg Index
- Up to 70% in Load Replace Partition with dummy input



# Support for non-utility enhancements

*Being able to use everything in DB2 9*



**IBM Information**  
On Demand **2008**

October 26 - 31, 2008 ~ Las Vegas  
The Premier Information Management  
Global Conference

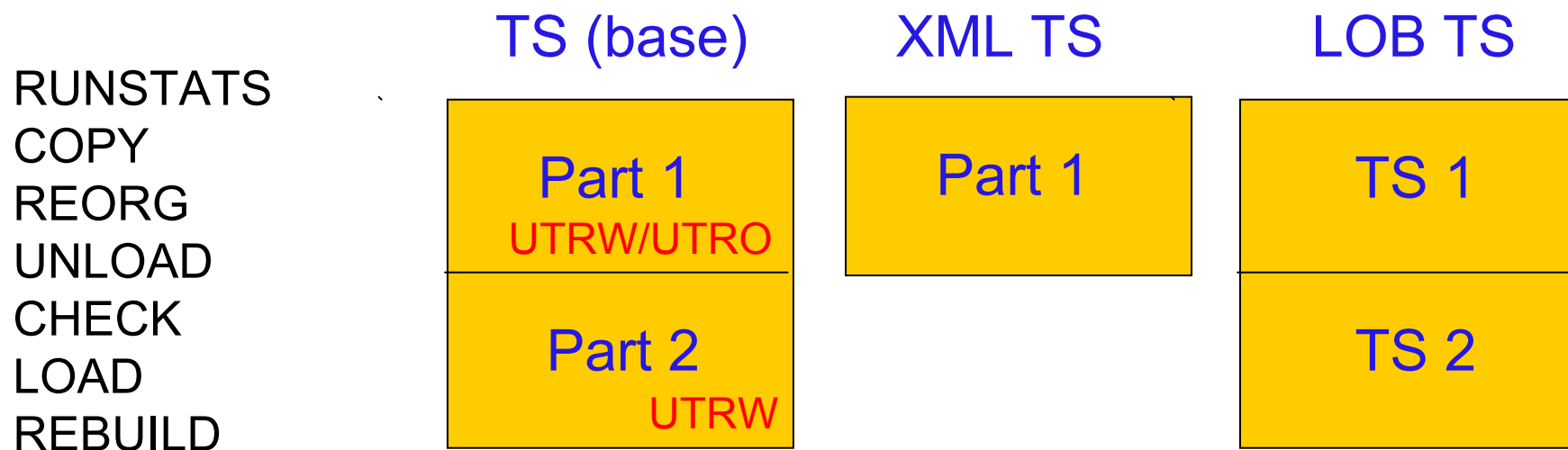
[www.ibm.com/events/informationondemand](http://www.ibm.com/events/informationondemand)

Information Management software

## DB2 9 Utilities

- Support for all new functions in DB2 9 for z/OS
  - ▶ Universal Table Spaces (UTS)
    - Partition By Growth (PBG)
    - Partition By Range (PBR)
  - ▶ XML table spaces (PBG or PBR)
  - ▶ Not logged tables/table spaces
  - ▶ Clone tables
  - ▶ Index on expression
  - ▶ New data types (BIGINT, VARBINARY, DECFLOAT XML)

## Utilities and Partition By Growth Table Spaces concurrency



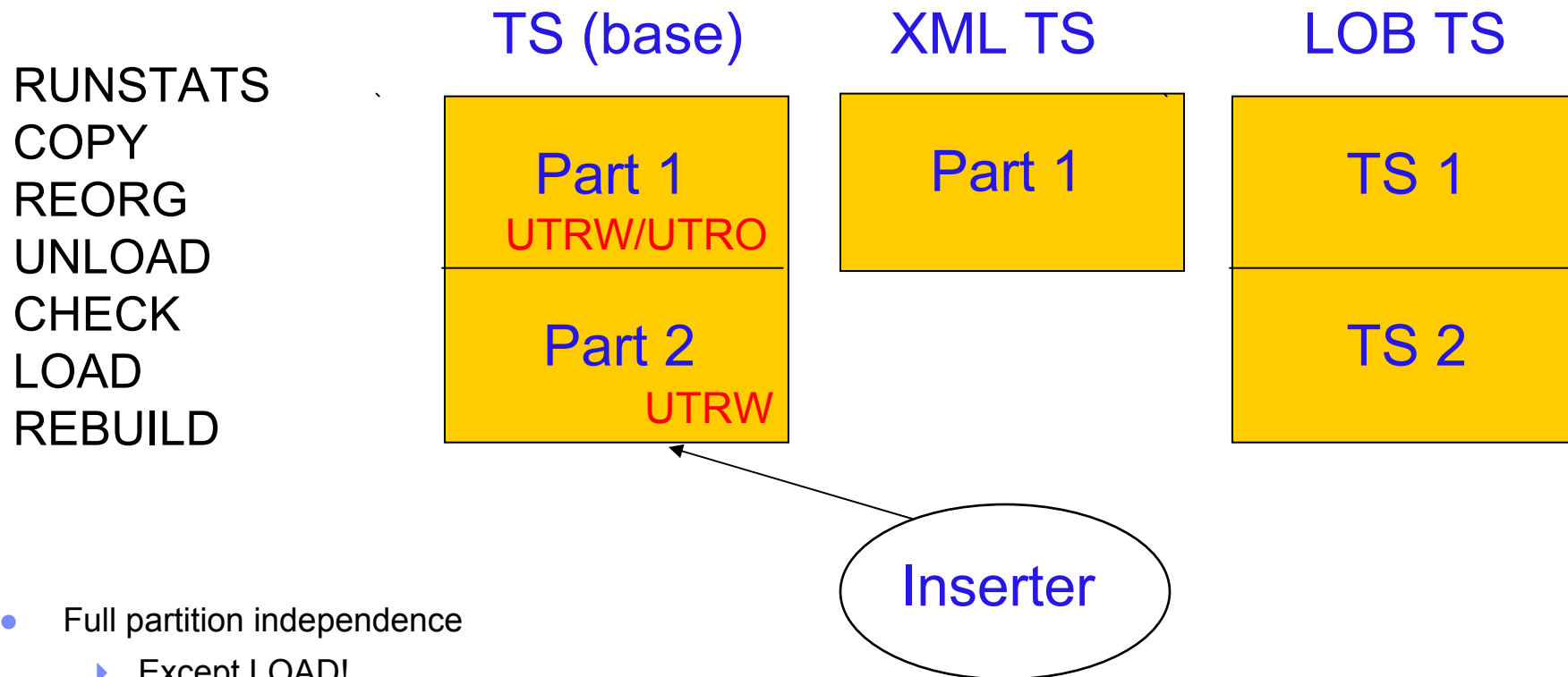
### Utilities and PBG objects

- Table space or part level supported

**RUNSTATS TABLESPACE TS SHRLEVEL CHANGE**

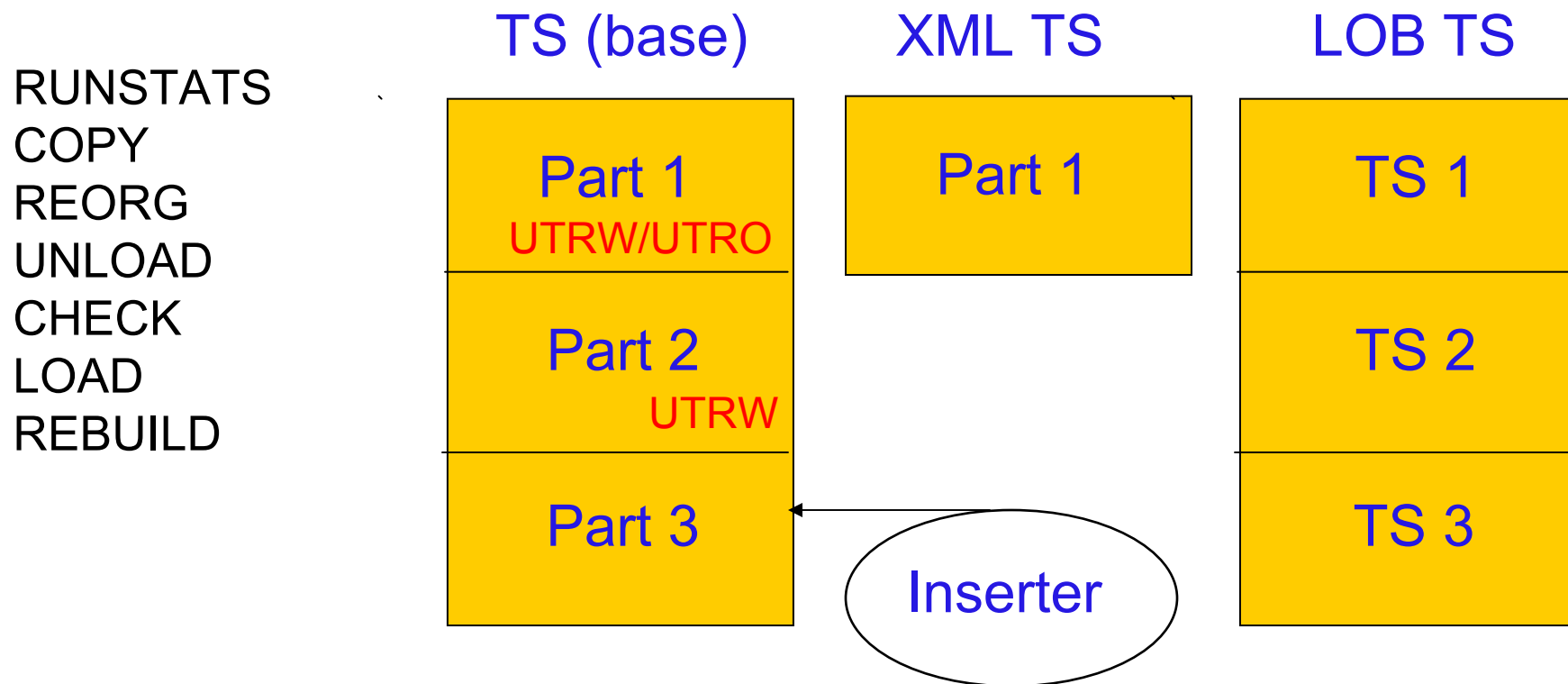
**COPY TABLESPACE TS DSNUM 1 SHRLEVEL REFERENCE**

## Utilities and Partition By Growth Table Spaces concurrency

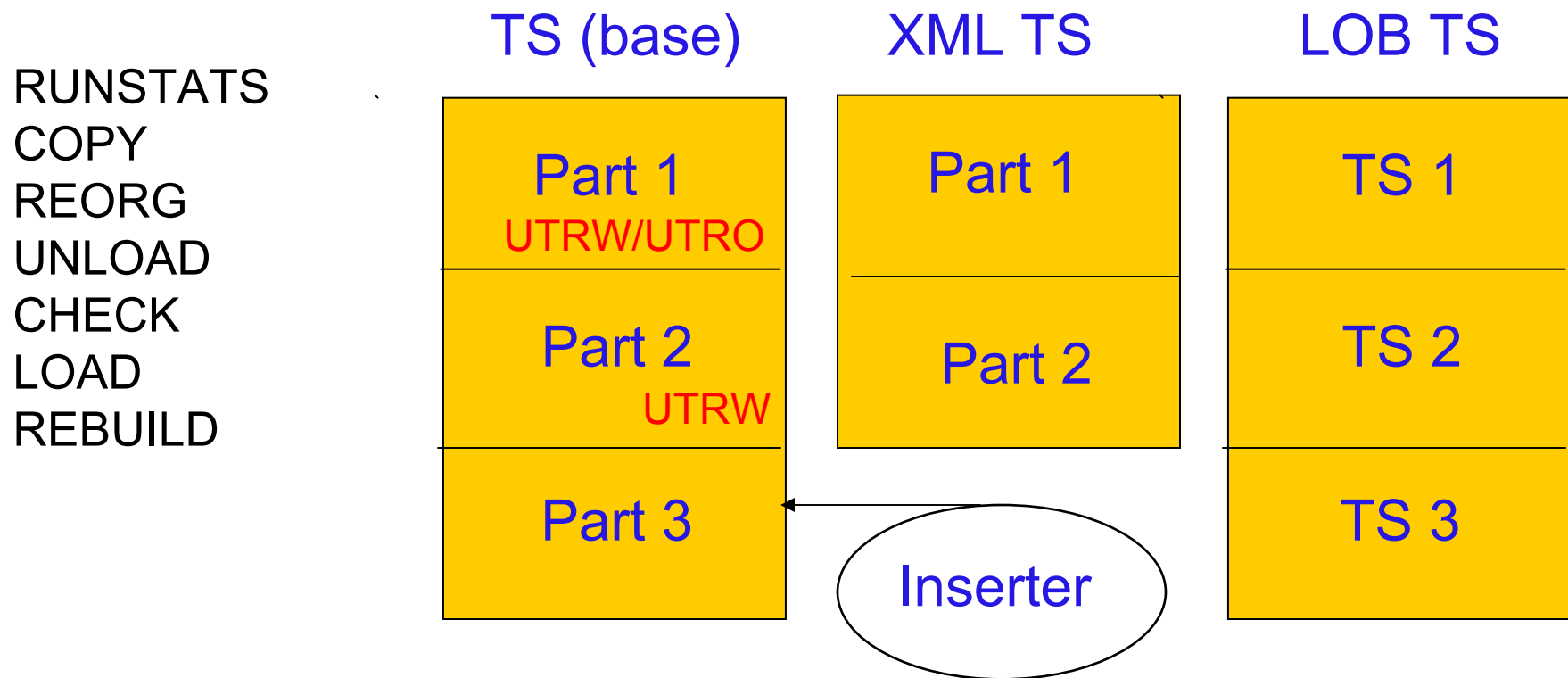


- Full partition independence
  - ▶ Except LOAD!
- Utility & SQL concurrency
  - ▶ New part may be added by DML even though cannot claim last part
  - ▶ But: Insert fails if unable to claim earlier part during space search

# Utilities and Partition By Growth Table Spaces concurrency



# Utilities and Partition By Growth Table Spaces concurrency

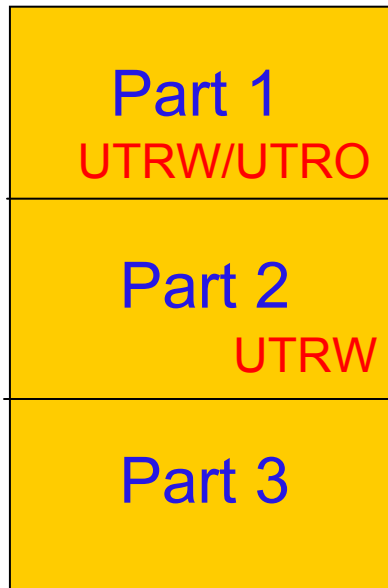


PBG base and XML partitions grow independently  
 PBG base and LOB table spaces grow together



## Utilities and PBGs

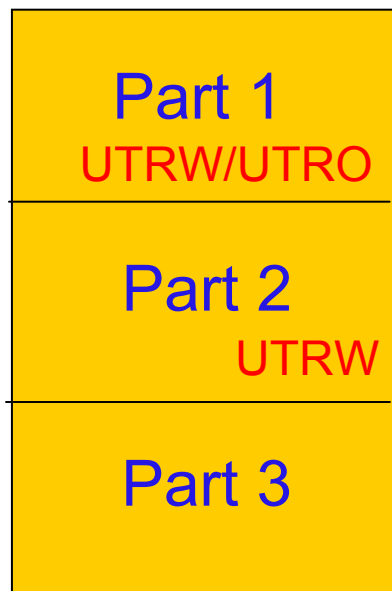
### PBG



- Drains & claims on new part are inherited from previous part
- The following DBET states are inherited by a new part from the previous part
  - ▶ Tablespace
    - RO
    - UT
    - UTUT
    - UTRO
  - ▶ NPI
    - PSRBD
    - ICOPY
  - ▶ Other states may be set by utilities directly, e.g. UTRW
- The following SHRLEVEL CHANGE utilities operate only on parts that existed at the time the utility started
  - ▶ RUNSTATS
  - ▶ UNLOAD
  - ▶ CHECK INDEX

## Utilities and PBGs

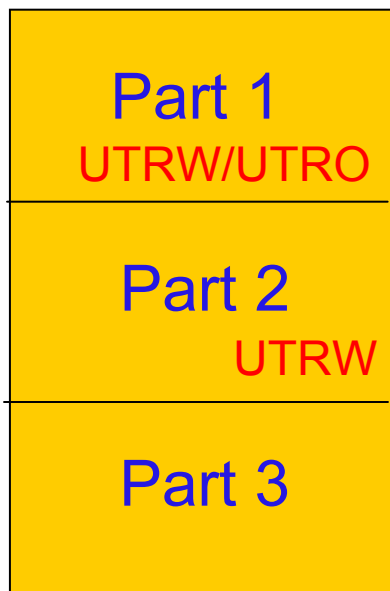
### PBG



- REORG
  - ▶ May create new parts
  - ▶ If hit MAXPARTITIONS then REORG fails
  - ▶ Dictionary copied from previous part
  - ▶ REORG of single part
    - No new part creation
    - Rows must fit back into part
  - ▶ REORG of part range
    - Data can flow from one part to another within range
    - If LOB column exists then rows will not move between parts
  - ▶ SHRLEVEL CHANGE
    - New part added to both base and shadow
  - ▶ No parallelism
  - ▶ No shrinking
  - ▶ No REBALANCE

## Utilities and PBGs

### PBG



- LOAD
  - ▶ No LOAD at part level
  - ▶ No parallelism
  - ▶ Can grow parts
  - ▶ Dictionary copied from previous part to new part
  - ▶ Excess parts remain empty
- COPY
  - ▶ Part or tablespace level
  - ▶ Will copy new parts if SHRLEVEL CHANGE at tablespace level
- REBUILD INDEX SHRLEVEL CHANGE
  - ▶ Index entries for new part included in log apply phase

## Utilities changes for XML

- Enhanced to handle new XML type, XML table spaces, and XML indexes
- CHECK DATA
- CHECK INDEX
- COPY INDEX
- COPY TABLESPACE
- COPYTOCOPY
- LISTDEF
- LOAD
- MERGECOPY
- QUIESCE
- TABLESPACESET
- REAL TIME STATISTICS
- REBUILD INDEX
- RECOVER INDEX
- RECOVER TABLESPACE
- REORG INDEX
- REORG TABLESPACE
- REPORT TABLESPACESET
- UNLOAD
- RUNSTATS

## DB2 9 Conversion Mode versus New Function Mode

Utility enhancements available in both CM and NFM are:

- CPU reductions
- Parallel unload, reload and log apply processing by the REORG utility
- Removal of the BUILD2 phase for REORG at the partition level
- Log Phase addition for REORG SHRLEVEL REFERENCE PART
- REORG SHRLEVEL REFERENCE for LOB table spaces
- CHECK INDEX SHRLEVEL REFERENCE with Parallelism
- Histogram Statistics for RUNSTATS
- CHECK LOB SHRLEVEL CHANGE
- CHECK DATA SHRLEVEL CHANGE
- REPAIR LOCATE SHRLEVEL CHANGE
- REBUILD INDEX SHRLEVEL CHANGE
- MODIFY RECOVERY support for the RETAIN option
- Object level recovery from a system-level backup by the RECOVER Utility
- BACKUP SYSTEM & RESTORE SYSTEM system-level backups to tape
- Skipping locked rows for transaction updates by the UNLOAD Utility
- TEMPLATE switching support based on size for image copy datasets

Other Utility enhancements are only available for DB2 9 NFM, such as...

- LBI (large Block Interface) support for sequential datasets on tape
- Recover to Point in Time (PIT) with consistency (defaults as new behavior)
- Clones effectively provide LOAD REPLACE SHRLEVEL CHANGE
- Online Load (LOAD RESUME YES SHRLEVEL CHANGE) without logging using NOT LOGGED



# Online Utility Enhancements

*Improved concurrency*



**IBM Information**  
On Demand **2008**

October 26 - 31, 2008 ~ Las Vegas  
The Premier Information Management  
Global Conference

[www.ibm.com/events/informationondemand](http://www.ibm.com/events/informationondemand)

Information Management software

## More online utilities

### REBUILD INDEX SHRLEVEL CHANGE

- ▶ Great for building new non-unique indexes or when index is in RBDP
- ▶ Index is built in place with no shadow
  - To move indexes to different volumes with availability still use REORG INDEX SHRLEVEL CHANGE)
- ▶ Table space must be in LOGGED state (new log phase)

## More online utilities (cont...)

### REORG enhancements

- Partition parallelism (UNLOAD/RELOAD) in a single utility statement (performance)
- Parallel log apply (performance)
- Elimination of BUILD2 phase (availability)
  - NPIs also shadowed
  - NPIs implicitly reorged
  - REORG SHRLEVEL REFERENCE PART now has a log phase
  - Improved availability
    - ◇ But concurrent REORGs at part level for same tablespace not permitted



## CHECK SHRLEVEL CHANGE design (generic)

- SHRLEVEL REFERENCE CHECK INDEX causes data and indexes to be unavailable for update for the duration of the utility
- Claim as reader for target data and indexes
- Create shadow datasets
  - ▶ Same dataset naming convention as REORG SHRLEVEL CHANGE
- Drain writers for target data and indexes
- Copy datasets with DFSMSdss ADRDSSU with FCNOCOPY to shadows
  - ▶ Uses dataset-level FlashCopy2 if available
  - ▶ Else, traditional media copy – still smaller r/o outage than SHR REF
- After logical complete for datasets,
  - ▶ Dedrain target data and indexes
  - ▶ Run CHECK on shadow data and indexes
- At utilterm delete shadow datasets when DB2 managed

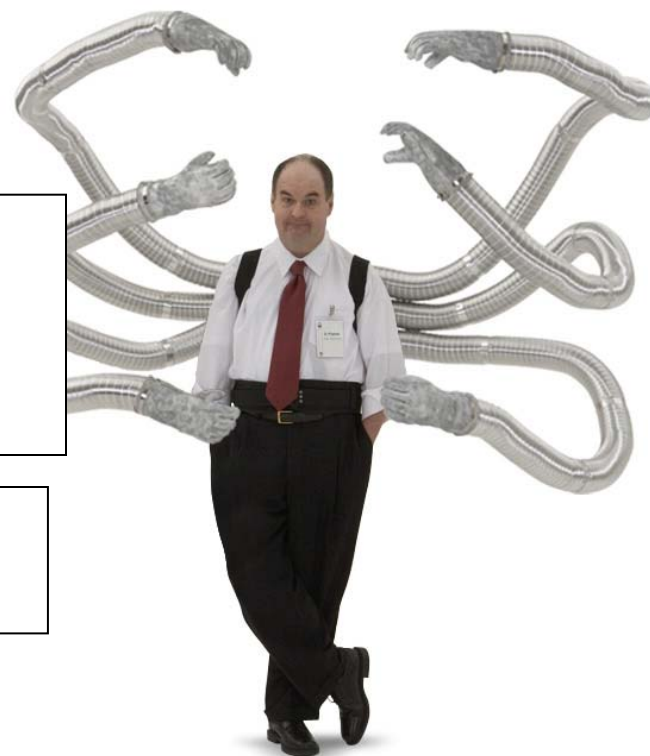
## More online utilities (cont...)

- CHECK DATA SHRLEVEL CHANGE
- CHECK LOB SHRLEVEL CHANGE
- REPAIR LOCATE ... SHRLEVEL CHANGE
- REORG LOB now supports SHRLEVEL REFERENCE
- Clones effectively provide LOAD REPLACE SHRLEVEL CHANGE
  - ▶ UTS only
- UNLOAD with ISO(CS) supports skipping rows that are locked for transaction updates



# Performance Enhancements

Reducing elapsed and CPU times



**IBM Information**  
On Demand **2008**

October 26 - 31, 2008 ~ Las Vegas  
The Premier Information Management  
Global Conference

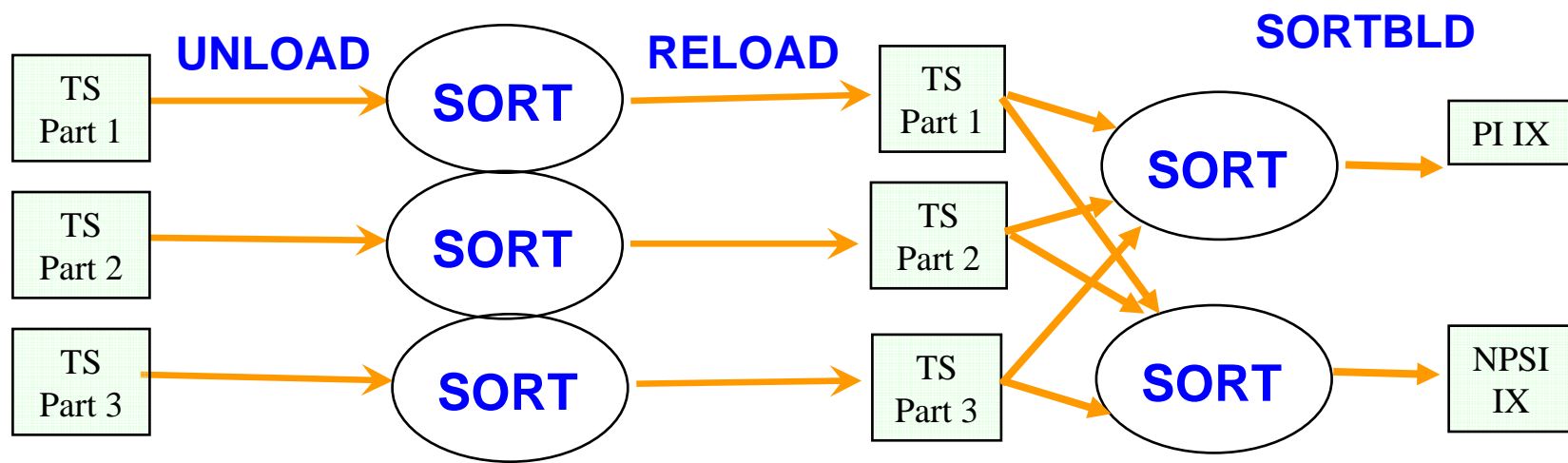
[www.ibm.com/events/informationondemand](http://www.ibm.com/events/informationondemand)

Information Management software

# REORG Partition Parallelism

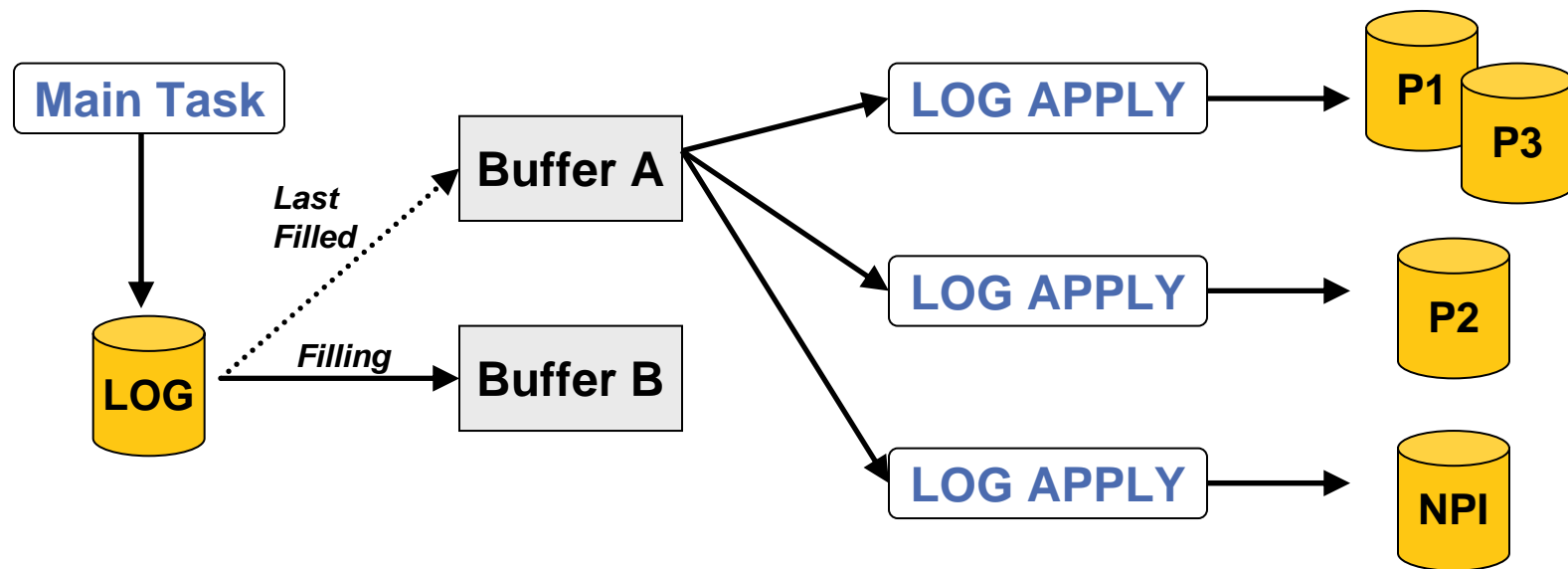
## REORG TABLESPACE TS

Three partitions, one PI, one NPSI



- Multiple concurrent jobs no longer needed
- No longer allowed with PART specification and NPSIs

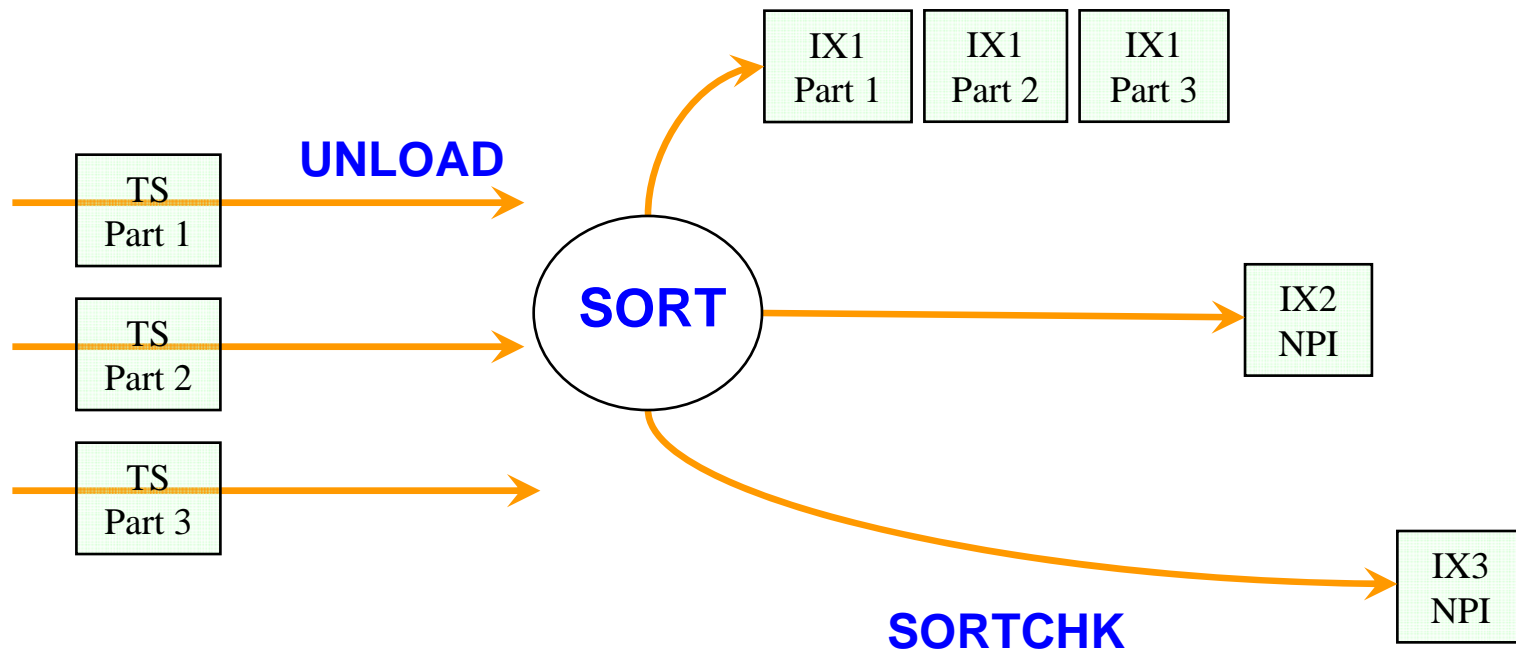
# Reorg Log Apply Parallelism



- Double Buffering
- Parallel log apply for SHRLEVEL CHANGE and SHRLEVEL REFERENCE PART if multiple NPIs

# Check Index Shrlevel Reference Parallelism

CHECK INDEX(ALL) TABLESPACE TS SHRLEVEL REFERENCE  
 three partitions, three indexes



Improved elapsed time performance (like REBUILD INDEX)

## Performance: Runstats Histogram Statistics

- Summarizes data distribution on an interval scale
  - ▶ Extends non-uniform distribution stats
- DB2 uses equal-depth histograms
  - ▶ Each quantile has about the same number of rows
  - ▶ 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



# Backup and Recovery Enhancements



**IBM Information**  
On Demand

**2008**

October 26 - 31, 2008 ~ Las Vegas  
The Premier Information Management  
Global Conference

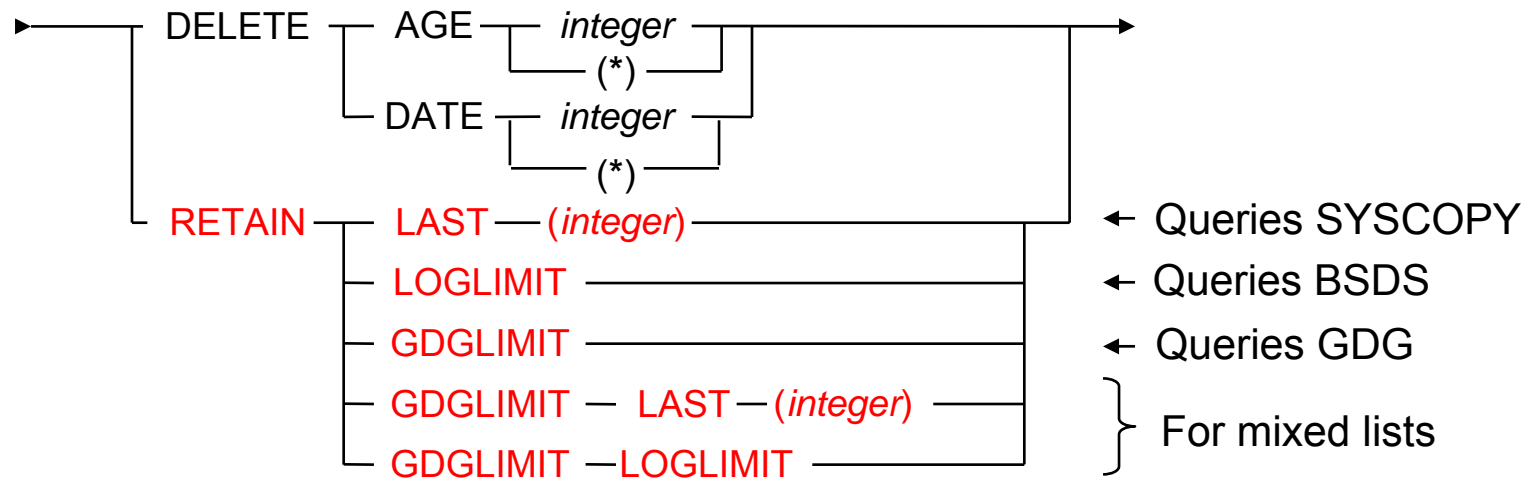
[www.ibm.com/events/informationondemand](http://www.ibm.com/events/informationondemand)

Information Management software



# Backup and Recovery Enhancements

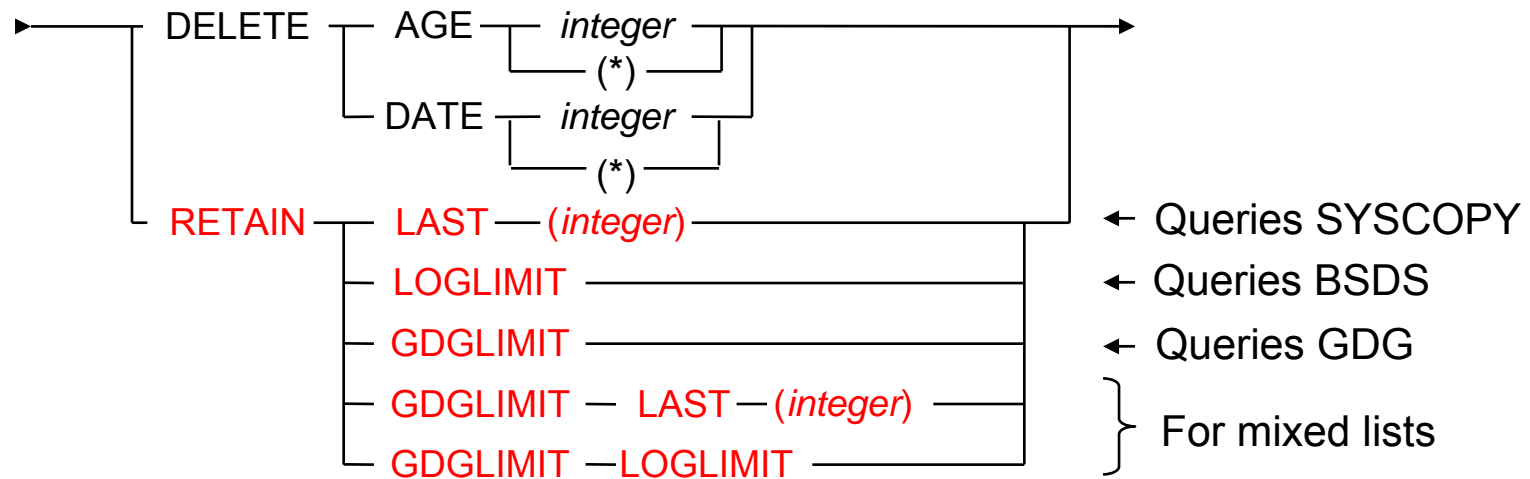
- Modify Recovery simplification and safety



- ▶ SYSLGRNX records deleted even if no SYSCOPY records deleted
- ▶ Deletion works on dates not on timestamps, so more entries than requested might be kept

# Backup and Recovery Enhancements

- Modify Recovery simplification and safety



- ▶ SYSLGRNX records deleted even if no SYSCOPY records deleted
- ▶ Deletion works on dates not on timestamps, so more entries than requested might be kept

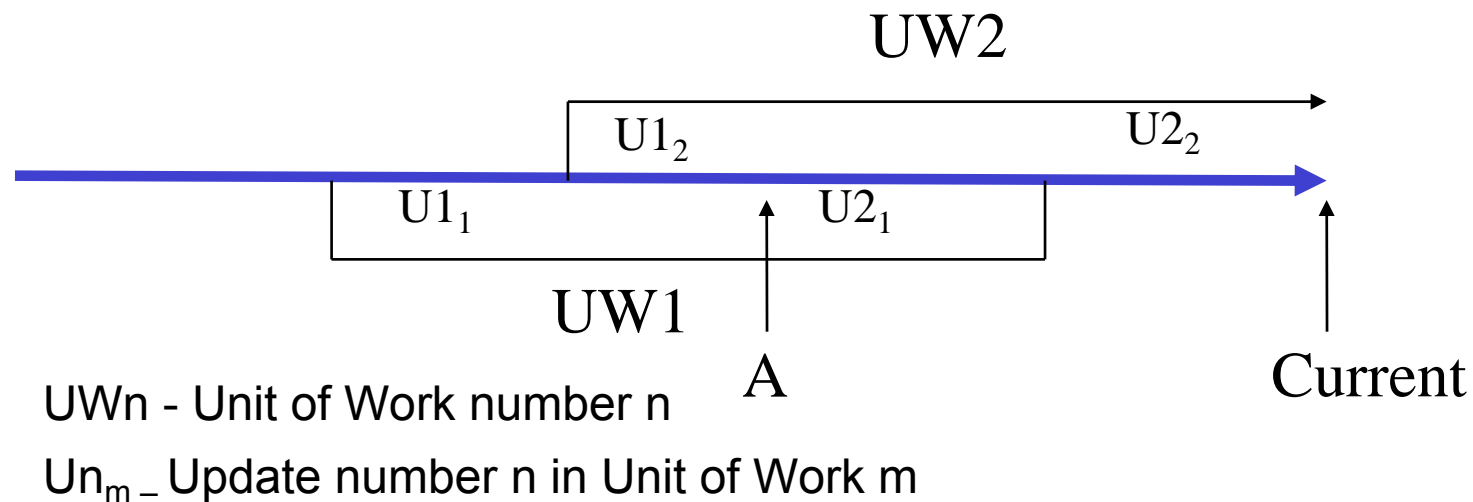
## Copy Utility Changes

- Always perform CHECKPAGE on the COPY utility
- The COPY utility includes SCOPE PENDING support to improve usability
- COPY utility bufferpool usage uses MRU management of those pages read by the COPY utility
- Template switching (e.g., copy to tape if large; to disk if small)

```
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)
```

# Recover Utility Changes

- Recovery to a point in time with consistency (NFM mode)
  - ▶ Uncommitted changes are backed out
  - ▶ Significantly reduces (eliminates?) the need to run QUIESCE
  - ▶ Does not apply to RECOVER TOCOPY, TOLASTCOPY and TOLASTFULLCOPY using SHRLEVEL CHANGE copy (consistency is not ensured – use RBA/LRSN after COPY point)
  - ▶ Include all relevant objects in same RECOVER to ensure data consistency from the application point of view



## Recover to a Point In Time (cont...)

- Recovery to a point in time with consistency (contd)
  - ▶ Two new phases in RECOVER: LOGCSR and LOGUNDO (will each be executed by member)
  - ▶ RECOVER can be restarted in any LOG phase, LOGCSR will restart at beginning, others at last commit point
  - ▶ Fast Log Apply not used for LOGUNDO phase
  - ▶ Progress of LOGUNDO phase can be monitored by regular status message DSNU1555I showing current and target log RBA (e.g. to detect backout pf long running URs)
  - ▶ This function can greatly reduce your need to run QUIESCE
    - If you want to capture a log point non-disruptively
      - -DISPLAY LOG, or
      - QUIESCE against SYSEBCDC (or some other dummy table space), or
      - SELECT CURRENT\_TIMESTAMP from SYSDUMMY1 and then convert it to an LRSN

## More Backup and Recover Enhancements...

- RECOVER RESTOREBEFORE x'rba/lrsn' directs RECOVER to use a recovery base prior to the rba/lrsn
- Progression of RECOVER log apply shown via -DIS UTIL
  - ▶ To help see progress and estimate elapsed time:

```
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
```

## More Backup and Recover Enhancements

- Support large block interface (allows blocksize > 32,760 bytes) for tapes
  - ▶ Up to 40% reduction elapsed time for COPY and RECOVER RESTORE phase
- Support large format datasets (allows > 65,535 tracks per DASD volume)
  - ▶ Handy for COPYing those large tablespaces with TEMPLATES
  - ▶ Requires NFM for creating large format datasets

## More Backup and Recover Enhancements

- Support large block interface (allows blocksize > 32,760 bytes) for tapes
  - ▶ Up to 40% reduction elapsed time for COPY and RECOVER RESTORE phase
- Support large format datasets (allows > 65,535 tracks per DASD volume)
  - ▶ Handy for COPYing those large tablespaces with TEMPLATES
  - ▶ Requires NFM for creating large format datasets



## BACKUP/RESTORE SYSTEM

- What's new in DB2 9 for BACKUP and RESTORE SYSTEM?
  - ▶ The ability to recover at the object level using system-level backups
    - RECOVER to point in time or to current
    - COPY YES indexes can be included in this recovery
    - Will restore from the previous system level backup or image copy
    - Requires z/OS V1R8
  - ▶ Tape Support
  - ▶ Support for Incremental FlashCopy
    - PK41001 & PK42014
  - ▶ RTS COPY columns also updated for BACKUP SYSTEM

## RECOVER utility changes

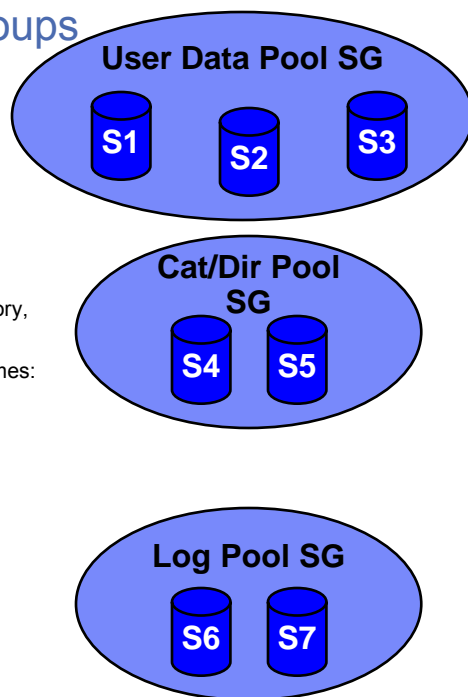
- Object level recovery from a system level backup
- New zparms
  - ▶ SYSTEM-LEVEL BACKUPS
    - YES – Permit RECOVER to use system level backups as a base
  - ▶ RESTORE/RECOVER
    - YES – RESTORE & RECOVER to use tape backup, not disk
    - RECOVER statement override: FROMDUMP
  - ▶ DUMP CLASS NAME
    - Name of DFSMSHsm dump class if using backup on tape
    - RECOVER statement override: DUMPCLASS
- No REPORT RECOVERY support at this time

# DB2 9 Utilities

- **BACKUP/RESTORE SYSTEM**

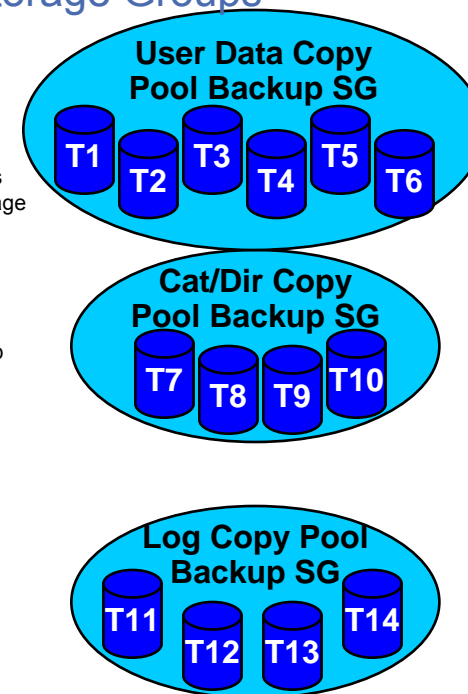
- ▶ But first, a quick primer... DFSMS terms explained

## POOL Storage Groups



- Similar in concept to DB2 STOGROUPs
- A collection of (source) volumes
- Stores user data, catalog/directory, log data sets
- This example uses source volumes: S1 – S7

## COPY POOL BACKUP Storage Groups



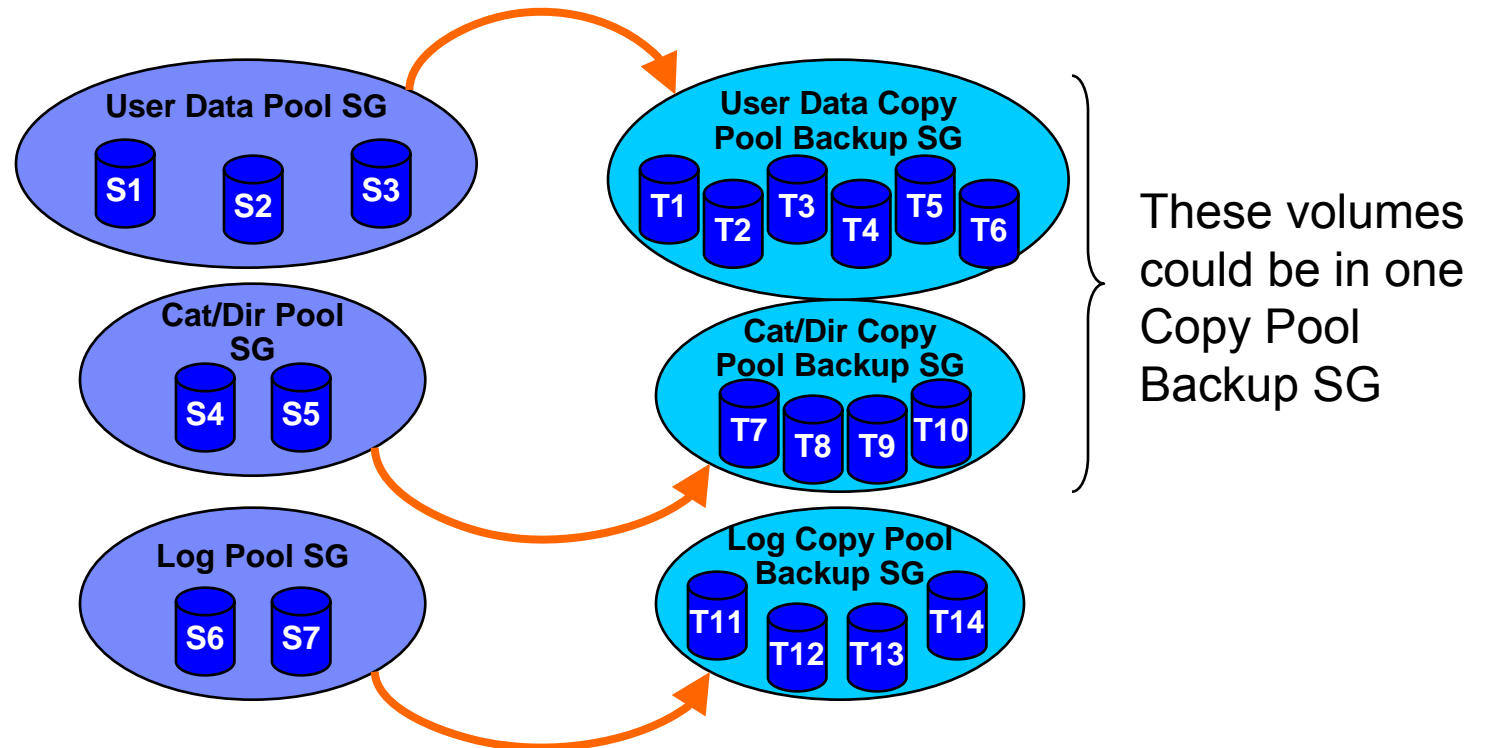
- A collection of volumes dedicated as FlashCopy targets for a source storage group
- Stores backups of user data, catalog/directory, log data sets
- Should contain a fixed multiple of volumes of the source storage group
- This example uses target volumes: T1 – T14

# DB2 9 Utilities

- BACKUP/RESTORE SYSTEM

- ▶ But first, a quick primer... Defining the FlashCopy target SG

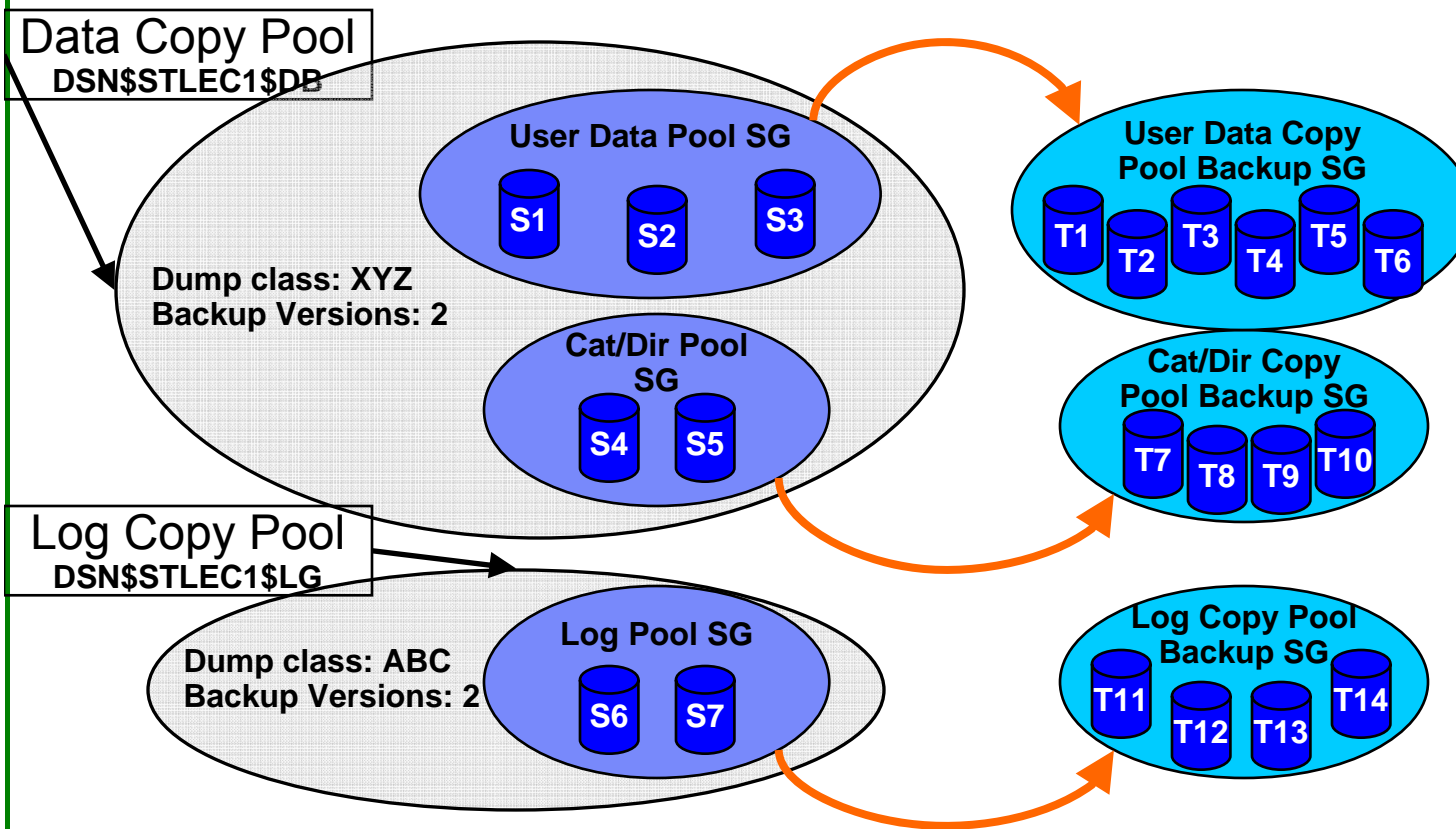
The Copy Pool Backup SG is an attribute of the Pool SG



# DB2 9 Utilities

- BACKUP/RESTORE SYSTEM

- ▶ But first, a quick primer... Copy Pools are a collection of source SGs

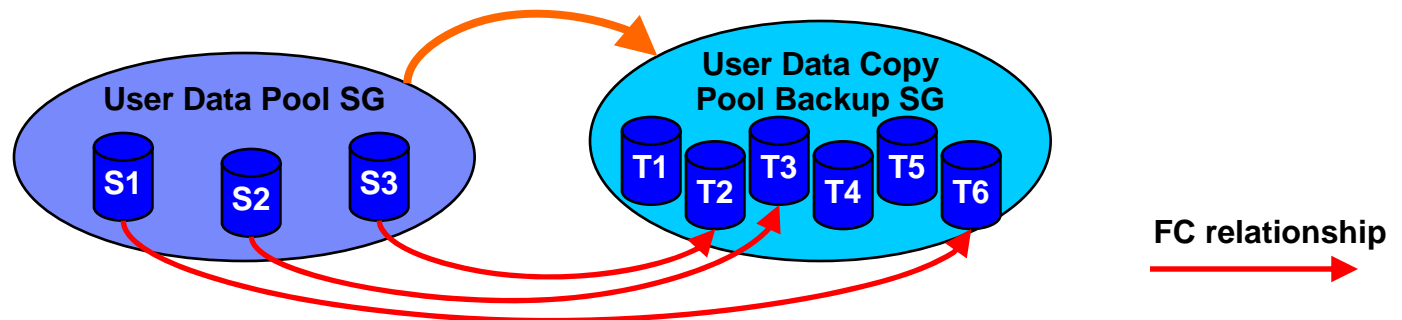


- A Copy Pool contains up to 256 (source) storage groups
- BACKUP SYSTEM/ RESTORE SYSTEM operate on Copy Pool Level

## DB2 9 Utilities

- BACKUP/RESTORE SYSTEM

- ▶ But first, a quick primer... FlashCopy (via the BACKUP SYSTEM DATA ONLY utility statement)
  - A relationship is established between source volume and target volume
  - The copy is considered successful when the relationship is established.

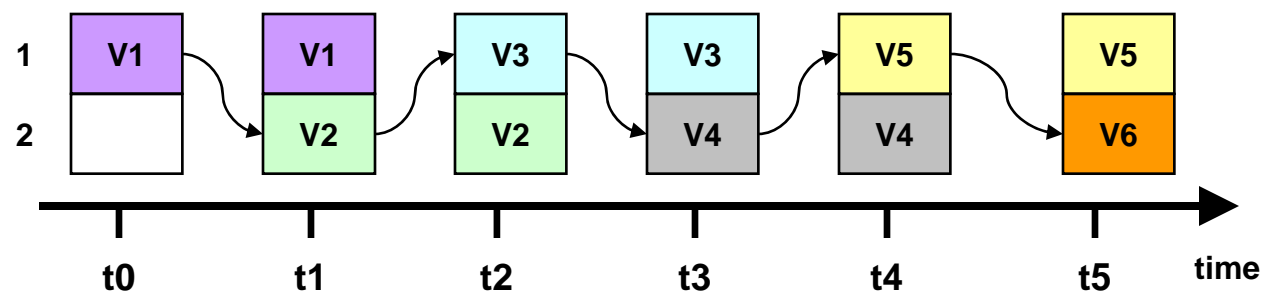


- A background copy is then started which will result in a target volume that looks like the source volume when the relationship was established.
  - *This can result in a performance degradation because of the I/O*
  - *If a track on the source volume changes before the background copy is complete then that track will be copied to the target volume before the new contents of the track is written to the source volume.*
- The relationship goes away when the background copy completes.

## DB2 9 Utilities

- What are Fast Replication versions and generations?
  - ▶ A Copy Pool is defined with a number of Versions that can be maintained in parallel
  - ▶ Version numbers are increased with every FRBACKUP invocation (1..999)
  - ▶ If more versions are created than the Copy Pool can support then the oldest versions will be rolled off:

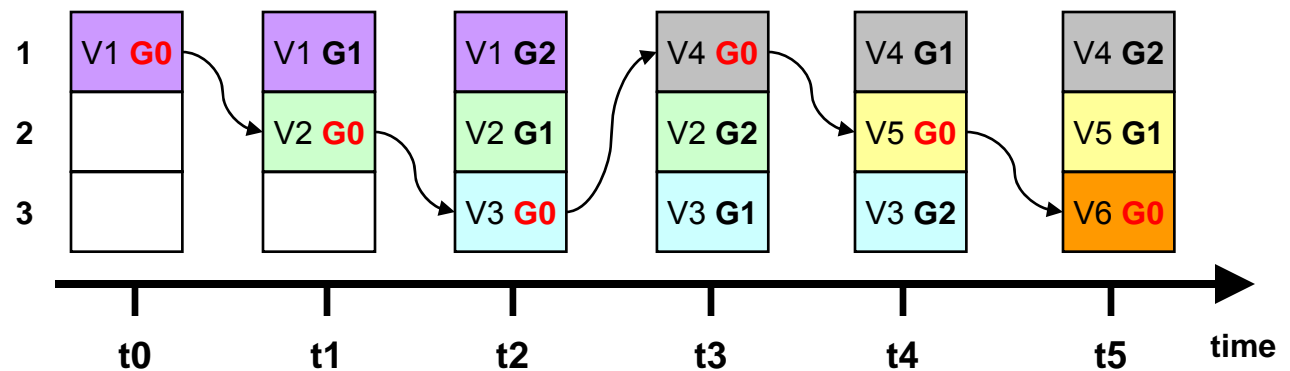
Copy Pool with  
2 versions:



# DB2 9 Utilities

- What are Fast Replication versions and generations?
  - ▶ Generations are a relative way to address versions in the Copy Pool
  - ▶ Generation 0 always refers to the most current version, generation 1 refers to the second most current version, etc.
  - ▶ Same concept as for Generation Data Groups (GDG)

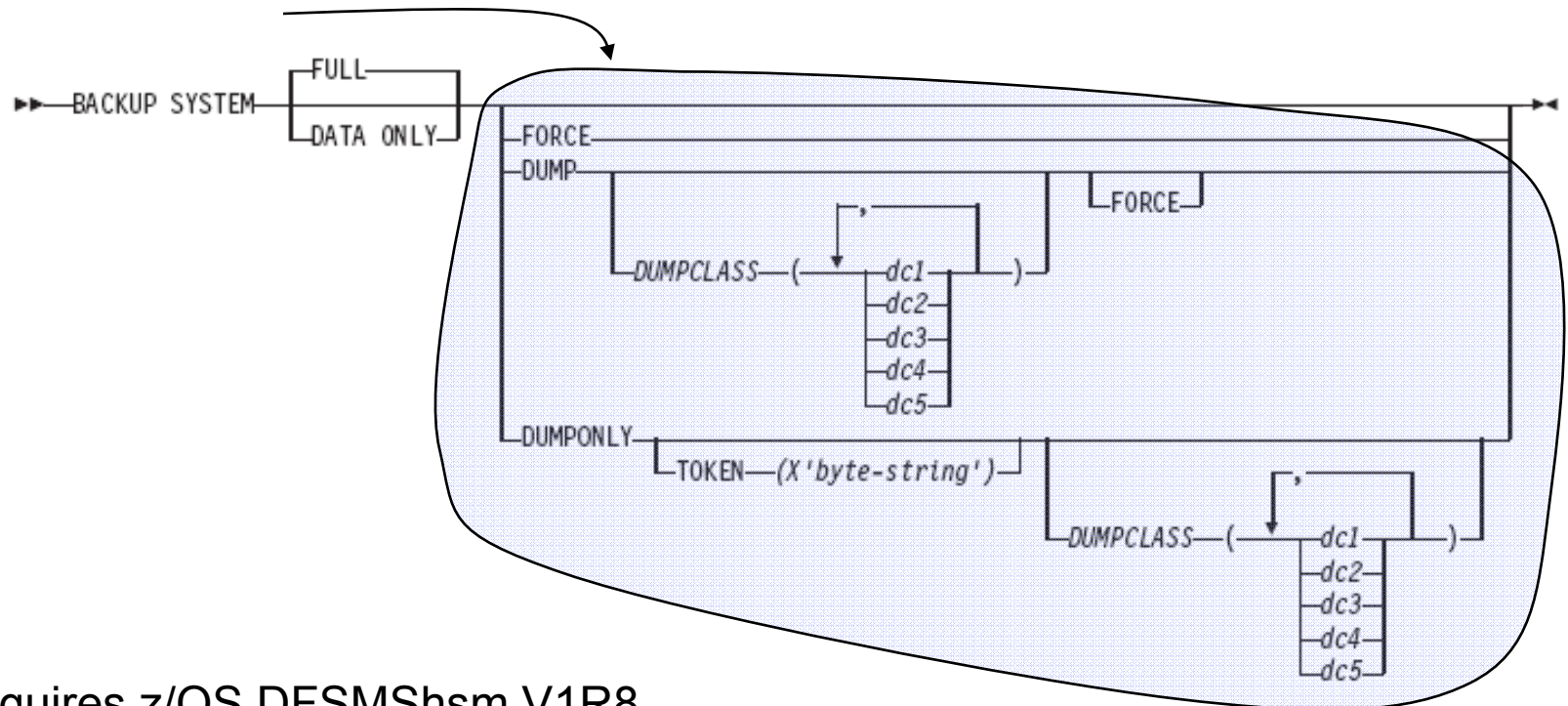
Copy Pool with 3 versions:





# DB2 9 Utilities

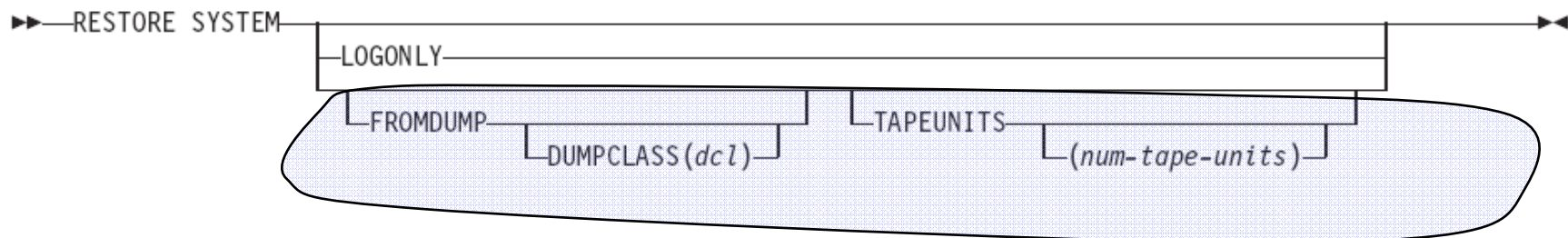
- BACKUP/RESTORE SYSTEM (what's new in DB2 9)
  - ▶ Tape control



- Requires z/OS DFSMSHsm V1R8
- FORCE indicates that oldest copy can be overwritten even if dump to tape is in progress
- Use LIST COPYPOOL with DUMPVOLS option to verify dump status

## DB2 9 Utilities

- BACKUP/RESTORE SYSTEM (what's new in DB2 9)



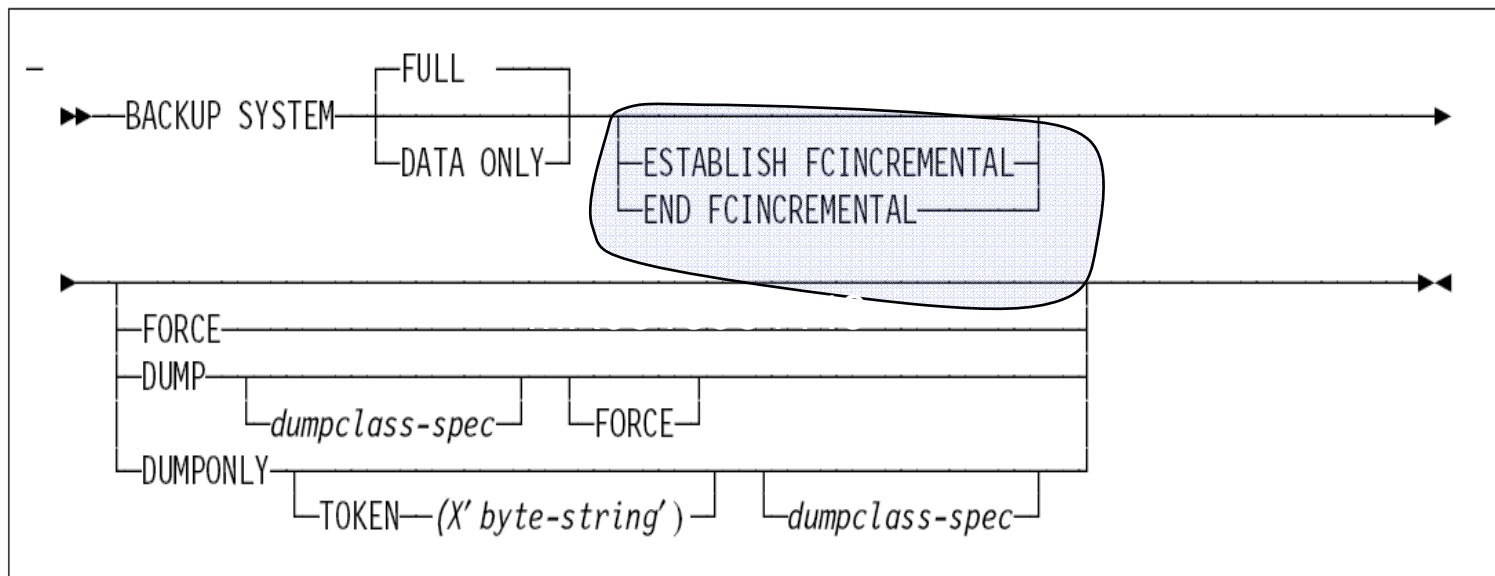
- ▶ Restore from dumps will use parallelism; limited by number of distinct tape volumes that the dump resides on; capped by TAPEUNITS
- ▶ ZPARMs to override tape options (DSNTIP6):
  - ▶ FROMDUMP           ↔ RESTORE/RECOVER FROM DUMP
  - ▶ DUMPCLASS           ↔ UTILS DUMP CLASS NAME
  - ▶ TAPEUNITS           ↔ RESTORE TAPEUNITS

## DB2 9 Utilities

- BACKUP/RESTORE SYSTEM (what's new in DB2 9)
  - ▶ Another primer, this time on Incremental FlashCopy:
    - A persistent relationship is established between two DASD devices
    - One is a source volume and the other is a target volume
    - DFSMSHsm chooses the volumes
    - All tracks on the source volume are considered to be changed when the relationship is established so all tracks are copied.
    - Subsequent incremental copies will copy only the tracks that have changed on the source volume since the last copy was taken
    - A DASD volume can have only one incremental relationship
    - If a Copy Pool has more than 1 version then the remaining versions will be full backups

## DB2 9 Utilities

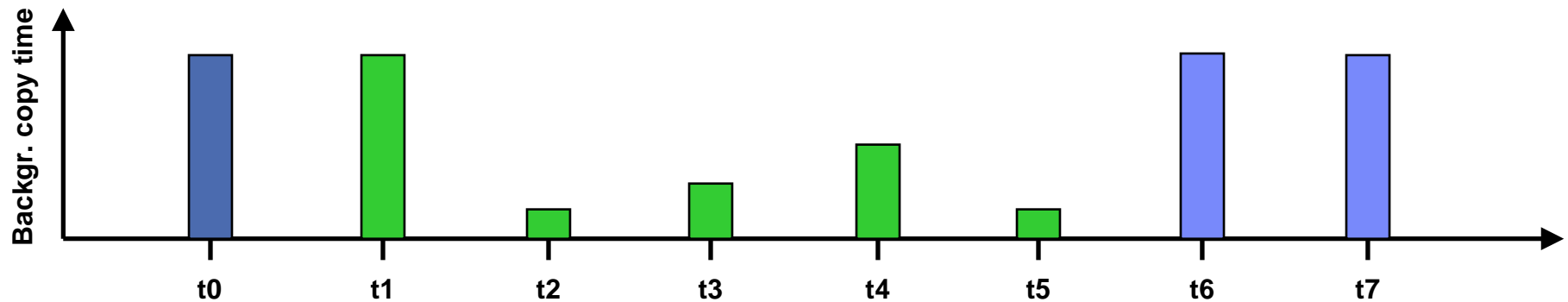
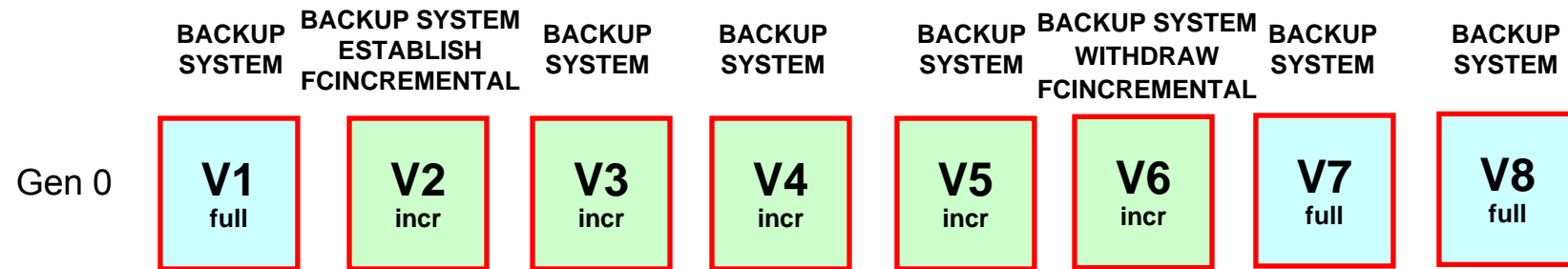
- BACKUP/RESTORE SYSTEM (what's new in DB2 9)
  - ▶ Support for Incremental FlashCopy (Post GA via APAR PK41001)



- ▶ Incremental FlashCopy can also be activated using HSM commands; `BACKUP SYSTEM` will then use it

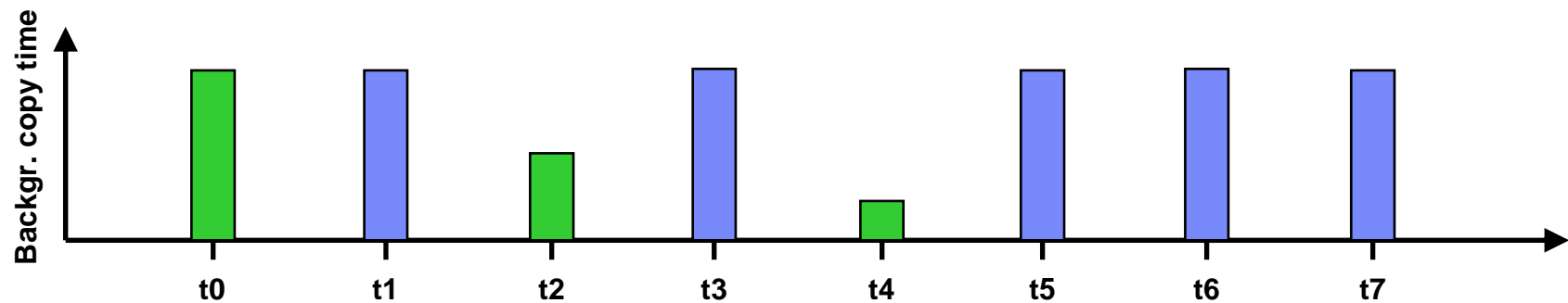
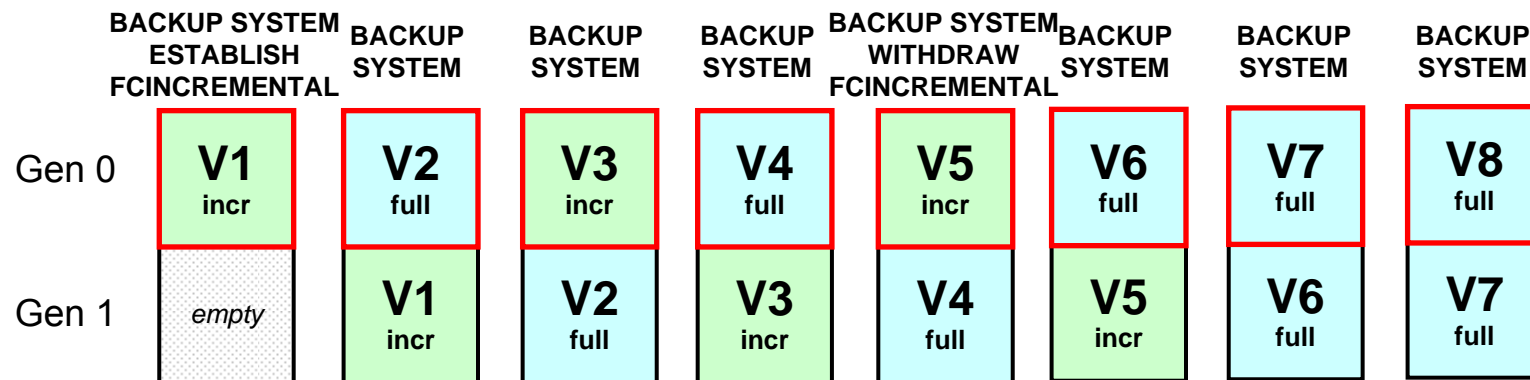
# DB2 9 Utilities

- BACKUP/RESTORE SYSTEM (what's new in DB2 9)
  - ▶ Incremental Flashcopy with a single version:



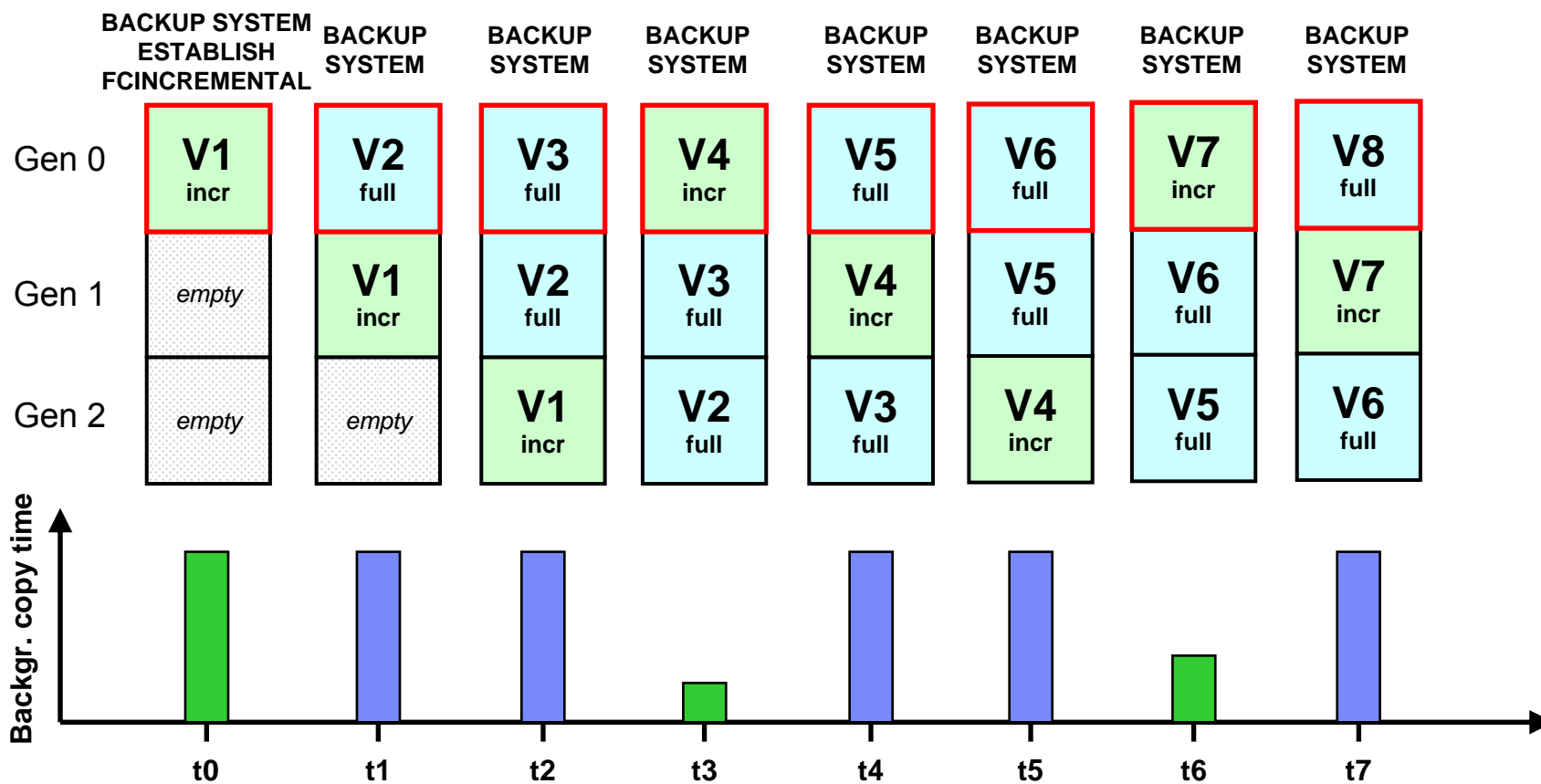
# DB2 9 Utilities

- BACKUP/RESTORE SYSTEM (what's new in DB2 9)
  - ▶ Incremental Flashcopy with two FC versions:



# DB2 9 Utilities

- BACKUP/RESTORE SYSTEM (what's new in DB2 9)
  - ▶ Incremental Flashcopy with three FC versions:



## Miscellaneous Enhancements

- All utility messages have timestamp and julian date (day)

DSNU000I 158 22:52:30.07 DSNUGUTC...

- Limit of 254 parts per REORG on a compressed table space lifted; storage consumption reduced
- New DSNACCOX stored procedure to gather stats from the catalog (including new Real Time Stats) & recommend REORG, RUNSTATS, and COPY actions for objects ([see APAR PK44133](#))



# Future enhancements

- Simplification, automation, availability, performance, flexibility
- Web-based Data Studio Admin Console
- DB2 Utilities Enhancement Tool (UET)
- High priority requirements
  - ▶ Autonomic stats
  - ▶ Autonomic compression
  - ▶ Recover to a different table space in a different DB2 with consistency
  - ▶ Reduce RUNSTATS resource consumption
  - ▶ Data set-level FlashCopy support
  - ▶ Online consistent image copies
  - ▶ Policy-based recovery
  - ▶ UTSERIAL elimination
  - ▶ REORG enhancements (LOBs, etc.)
  - ▶ ...

Name	Monitoring Status	Data Server Status	Critical	Warning	CPU Usage	Disk Space	Memory Usage	Locking	SQL Performance	Connections	Transactions	Logging	Maintenance
Production	3	8											
Web	1	1											
eCommerce	0	0											
Support	1	1											
Retail	0	0											
New York	0	0											
Los Angeles	0	0											
Accounts	2	3											
Marketing	0	4											
Test	0	0											
Development	0	0											

~~Manual invocation of  
 •RUNSTATS  
 •COPY/BACKUP SYSTEM  
 •QUIESCE  
 •MODIFY RECOVERY  
 •REORG~~

# Future enhancements

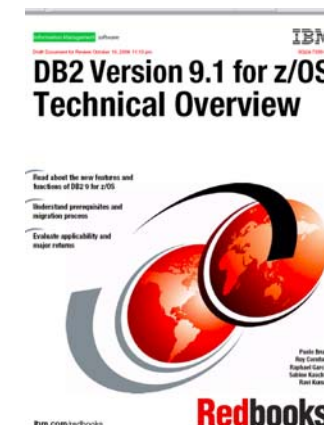
- Simplification, automation, availability, performance, flexibility
- Web-based Data Studio Admin Console
- DB2 Utilities Enhancement Tool (UET)
- High priority requirements
  - ▶ Autonomic stats
  - ▶ Autonomic compression
  - ▶ Recover to a different table space in a different DB2 with consistency
  - ▶ Reduce RUNSTATS resource consumption
  - ▶ Data set-level FlashCopy support
  - ▶ Online consistent image copies
  - ▶ Policy-based recovery
  - ▶ UTSERIAL elimination
  - ▶ REORG enhancements (LOBs, etc.)
  - ▶ ...

Name	Monitoring Status	Data Server Status	Critical	Warning	CPU Usage	Disk Space	Memory Usage	Locking	SQL Performance	Connections	Transactions	Logging	Maintenance
Production	3	8											
Web	1	1											
eCommerce	0	0											
Support	1	1											
Retail	0	0											
New York	0	0											
Los Angeles	0	0											
Accounts	2	3											
Marketing	0	4											
Test	0	0											
Development	0	0											

~~Manual invocation of  
 •RUNSTATS  
 •COPY/BACKUP SYSTEM  
 •QUIESCE  
 •MODIFY RECOVERY  
 •REORG~~

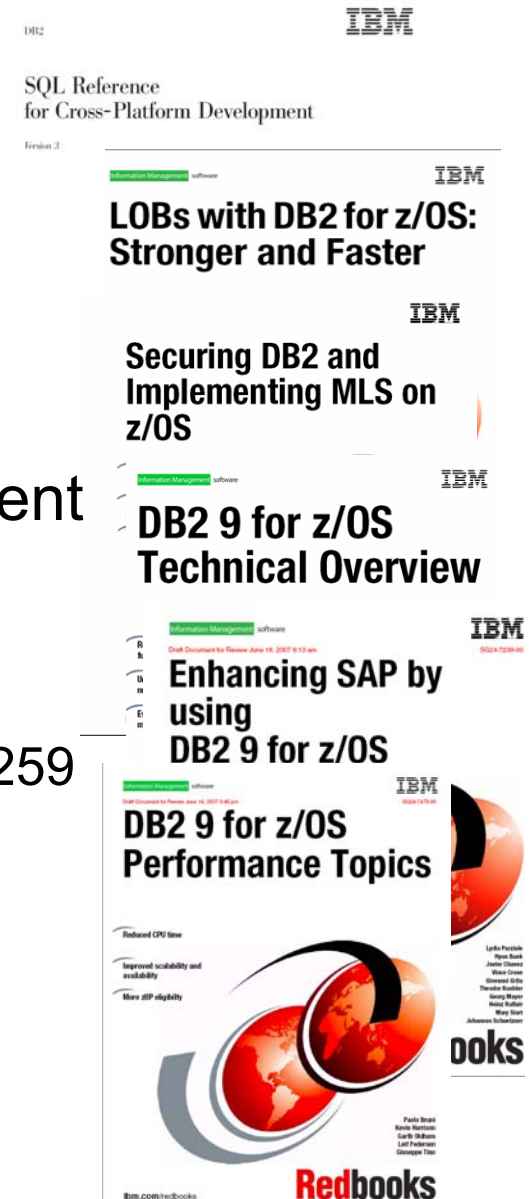
Get updated books February & March 2008  
<http://publib.boulder.ibm.com/infocenter/imzic/>

- ✓ Utility Guide and Reference
- ✓ Administration Guide
- ✓ Data Sharing: Planning and Administration
- ✓ Performance Guide
- ✓ Messages
- ✓ Codes
- Program Directories
- Installation Guide
- Redbooks
- Application Programming Guide & Reference for JAVA™
- RACF Access Control Module Guide
- ODBC, Spatial, ...
- Internationalization Guide (Unicode)



# DB2 9 in **IBM Redbooks**

- DB2 9 Technical Overview SG24-7330
- DB2 9 Performance Topics SG24-7473
- DB2 9 Stored Procedures SG24-7604
- Index Compression with DB2 9 for z/OS paper
- SQL Reference for Cross-Platform Development
- DB2 9 Optimization Service Center SG24-7421
- LOBs with DB2 for z/OS SG24-7270
- Powering SOA with IBM Data Servers SG24-7259
- Enhancing SAP - DB2 9 SG24-7239
- Best practices SAP BI - DB2 9 SG24-6489-01
- Data Sharing in a Nutshell, SG24-7322
- Securing DB2 & MLS z/OS SG24-6480-01



## References Utilities

- DB2 for z/OS home page  
<http://www.ibm.com/software/data/db2/zos/>
- DB2 for z/OS Version 8 Performance Topics, SG24-6465
- DB2 for z/OS Version 7 Using the Utilities Suite, SG24-6289
- Recommendations for Tuning Large DFSORT Tasks  
<http://www.ibm.com/servers/storage/support/software/sort/mvs/tuning/index.html>
- DFSMSrmm SMS ACS Support  
<http://www.redbooks.ibm.com/abstracts/TIPS0530.html?Open>

## More information on the web

COBOL and DB2 V8 multirow fetch, insert

<http://www.ibm.com/support/docview.wss?&uid=swg27012149>

<http://www.ibm.com/support/docview.wss?&uid=swg27012160>

DB2 9 education from BetaWorks

<http://www.ibm.com/support/docview.wss?&uid=swg27012156>

<http://www.ibm.com/support/docview.wss?&uid=swg27012181>

<http://www.ibm.com/support/docview.wss?&uid=swg27012150>

<http://www.ibm.com/support/docview.wss?&uid=swg27012151>

<http://www.ibm.com/support/docview.wss?&uid=swg27012152>

<http://www.ibm.com/support/docview.wss?&uid=swg27012153>

<http://www.ibm.com/support/docview.wss?&uid=swg27012155>

<http://www.ibm.com/support/docview.wss?&uid=swg27012158>

<http://www.ibm.com/support/docview.wss?&uid=swg27012154>