# Z27: DB2 for z/OS

# Managed System Backup and Recovery in DB2 V8

James Teng (jteng@us.ibm.com)

**IBM** DB2 Information Management
**Technical Conference**

**Sept. 20-24, 2004**

**Las Vegas, NV**



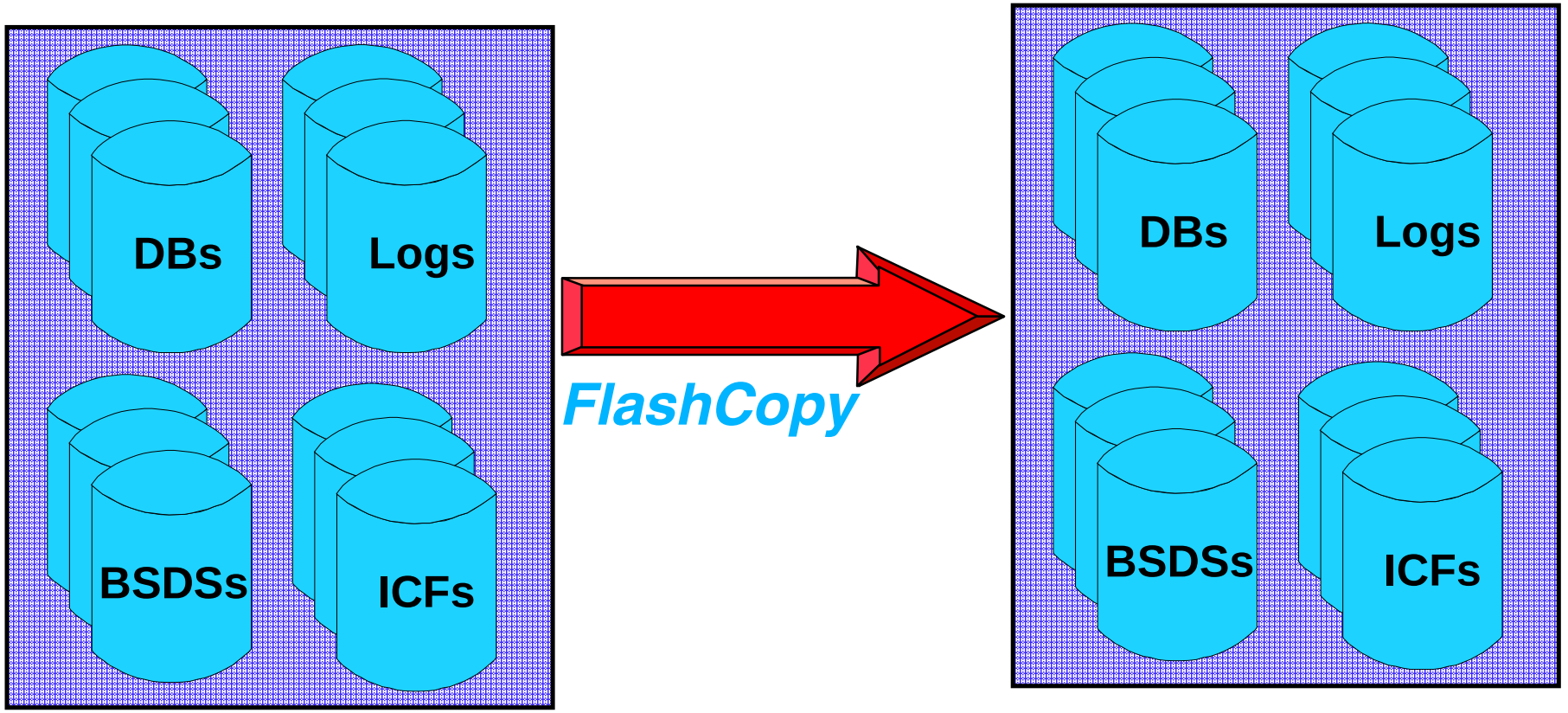
RETURN TO INDEX

# Customer Requirements

- *Need a fast non-disruptive way to backup/recovery all DB2 data*
  - *It is too slow and difficult to manage backups at table/index level*
  - *A single DB2 has more than 50,000 tables/indexes for many ERP/CRM applications*
  - *Need a way to ensure data and logs in backup are consistent for restart recovery*
  - *Consistent backup with no impact to applications*
- *Backups can be used to support*
  - *Point-In-Time Recovery on application errors*
  - *Table or index level recovery*
  - *Disaster Recovery or Cloning Systems*

# Use FlashCopy to take Backups - Today's Solution

**SET LOG SUSPEND** - Suspend Writes




**SET LOG RESUME** - Resume Writes

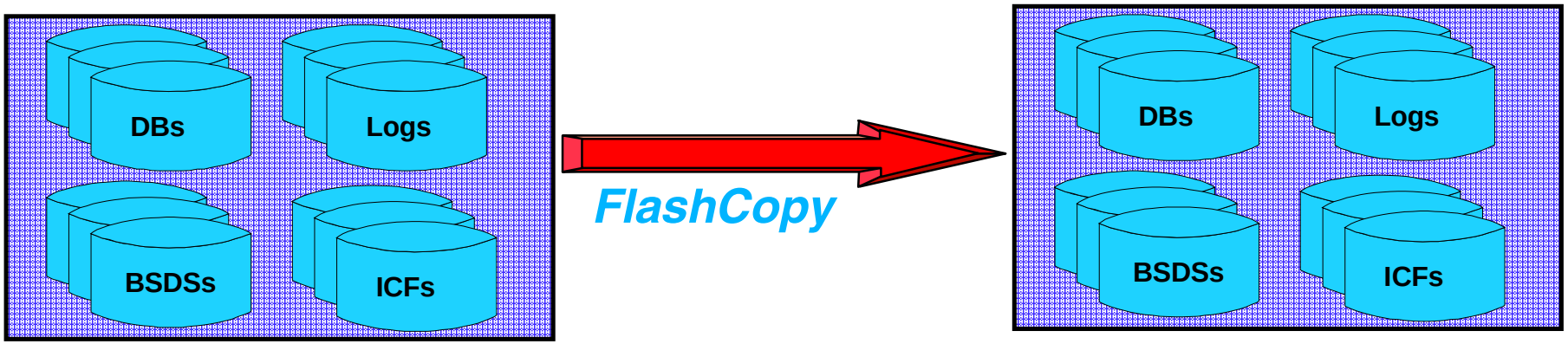
***Save to Tape/DASD***

# Notes: Use FlashCopy to take Backups

- *Issue SET LOG SUSPEND command to*
  - *Temporarily "freeze" all DB2 update activity*
  - *Reads are still allowed*
- *Allow fast volume level copy entire DB2 system*
  - *Use ESS FlashCopy*
  - *Parallel copy all data and log volumes*
  - *Copies can be used for point-in-time recovery or disaster recovery*
- *Issue SET LOG RESUME command to*
  - *Resume update operations*
- *Option to physical copy target volumes to tapes or to remote storage system via PPRC or XRC*

# DB2 Managed ESS FlashCopy Solution in V8

- Provide an easier and less disruptive way for fast volume-level backup and recovery
  - Use ESS FlashCopy to backup DB2 data and logs
  - No longer need to suspend logs
  - Backups are managed by DB2 and DFSMShsm to support system level PIT recovery



# DB2 Managed ESS FlashCopy Solution in V8 ...

- Two new utilities in DB2 for z/OS V8:
  - BACKUP SYSTEM
  - RESTORE SYSTEM
- Backup copies can also be used for:
  - Disaster recovery
  - System cloning
- Enhancement in the SET LOG SUSPEND command
  - As an alternative to BACKUP SYSTEM

# BACKUP SYSTEM utility

- Invokes DFSMShsm to take fast volume copies of the DB2 data and / or logs
- DB2 data and logs must be SMS-managed
- No DB2 quiesce point is required, nothing stops as in SET LOG SUSPEND
- Manages up to 50 backup versions on DASD
  - BSDS limit
- Backup information are kept in BSDS and in DFSMShsm
- Two flavors: FULL / DATA ONLY

# BACKUP SYSTEM utility ...

- Full backup
  - BACKUP SYSTEM - FULL
  - Allow recovery of the entire system in later stage
  - Have to define the "database" and "log" COPYPOOLs
  - Backup both database and then log (active logs and BSDS)

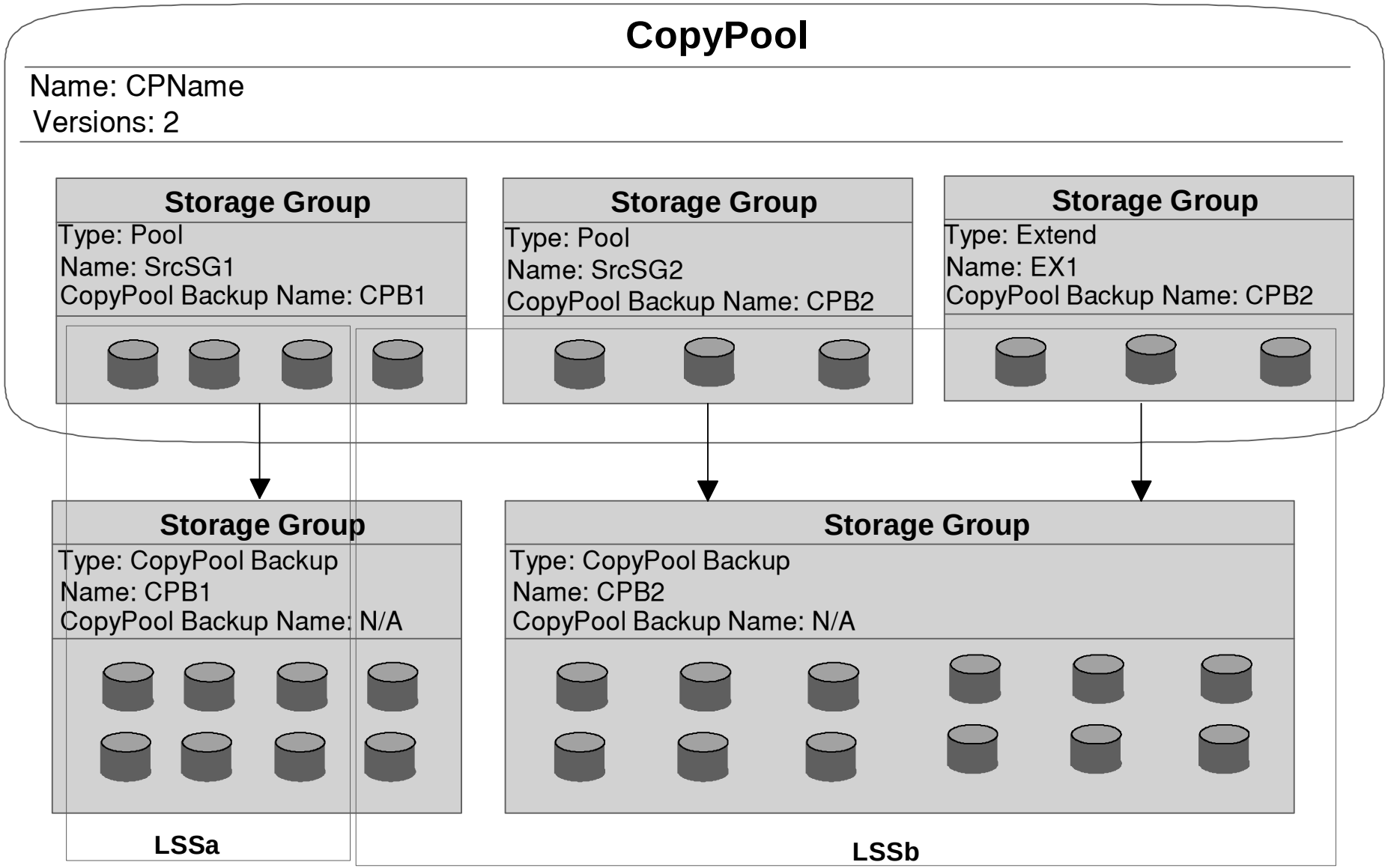
- Data only system backup
  - BACKUP SYSTEM - DATA ONLY
  - Only "database" COPYPOOL has to be defined for database backup

# COPYPOOL

- New SMS construct
- Set of SMS storage groups - maximum 256
- Has a VERSIONS attribute - maximum 85
- Each DB2 system has two SMS COPYPOOLs
  - DATA COPYPOOL (DSN$location_name$DB)
  - LOG COPYPOOL (DSN$location_name$LG)

- Copy Pool  Backup
  - New storage group type
  - Used to hold volume copies of DASD defined in the COPYPOOL

# CopyPool



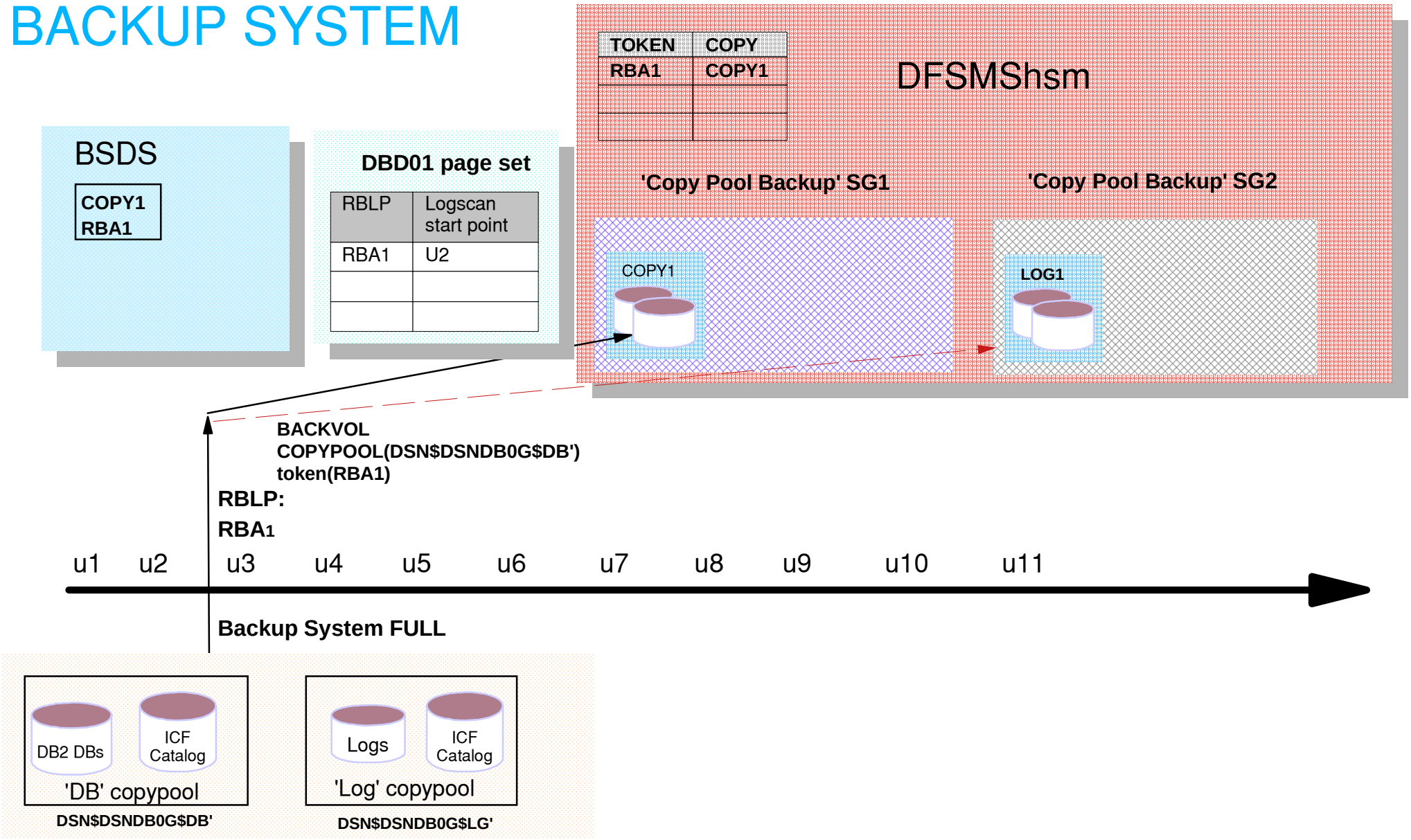
# BACKUP SYSTEM process

- When BACKUP SYSTEM utility is issued, DB2
  - Suspends  32K page writes for objects created before V8 NFM
    - Use REORG to convert them to avoid suspend writes
  - Suspends data set creation, deletion, rename and extension operations
  - Prevents data sets from pseudo closed
  - System checkpoints will not advance restart REDO log point
  - *Records* the Recover Based Log Point (RBLP) in DBD01
    - Recommend to control checkpoints based on time interval for data sharing systems

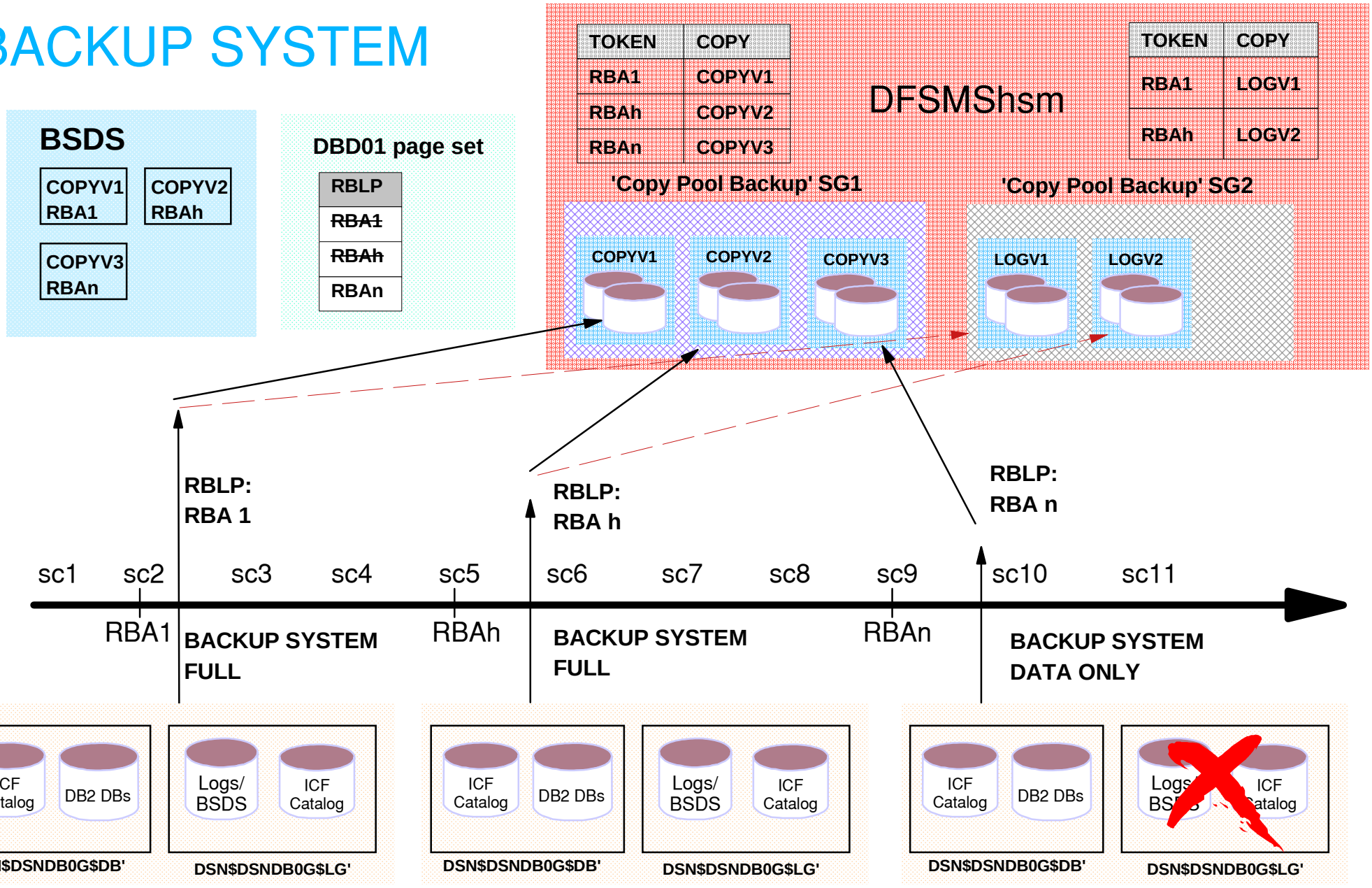
# BACKUP SYSTEM process ...

- During the backup . . .
  - Invokes DFSMShsm to take FlashCopy of 'DB' COPYPOOL
    - HSM uses the DSS COPY to copy the volumes in the COPYPOOL
  - Invokes DFSMShsm to take FlashCopy of the 'LG' COPYPOOL, if for Backup System Full
  - Each member updates BSDS with the system backup information
    - In data sharing only the submitting member logs BSDS information
  - Resume the quiesced activities

# BACKUP SYSTEM



# BACKUP SYSTEM



# RESTORE SYSTEM utility

- RESTORE SYSTEM utility is only needed to recover system to an arbitrary PIT
- To recover system only to the PIT at which the backup copy was taken
  - Use copies from BACKUP SYSTEM FULL

    **HSM FRRECOV COPYPOOL(cpname)**

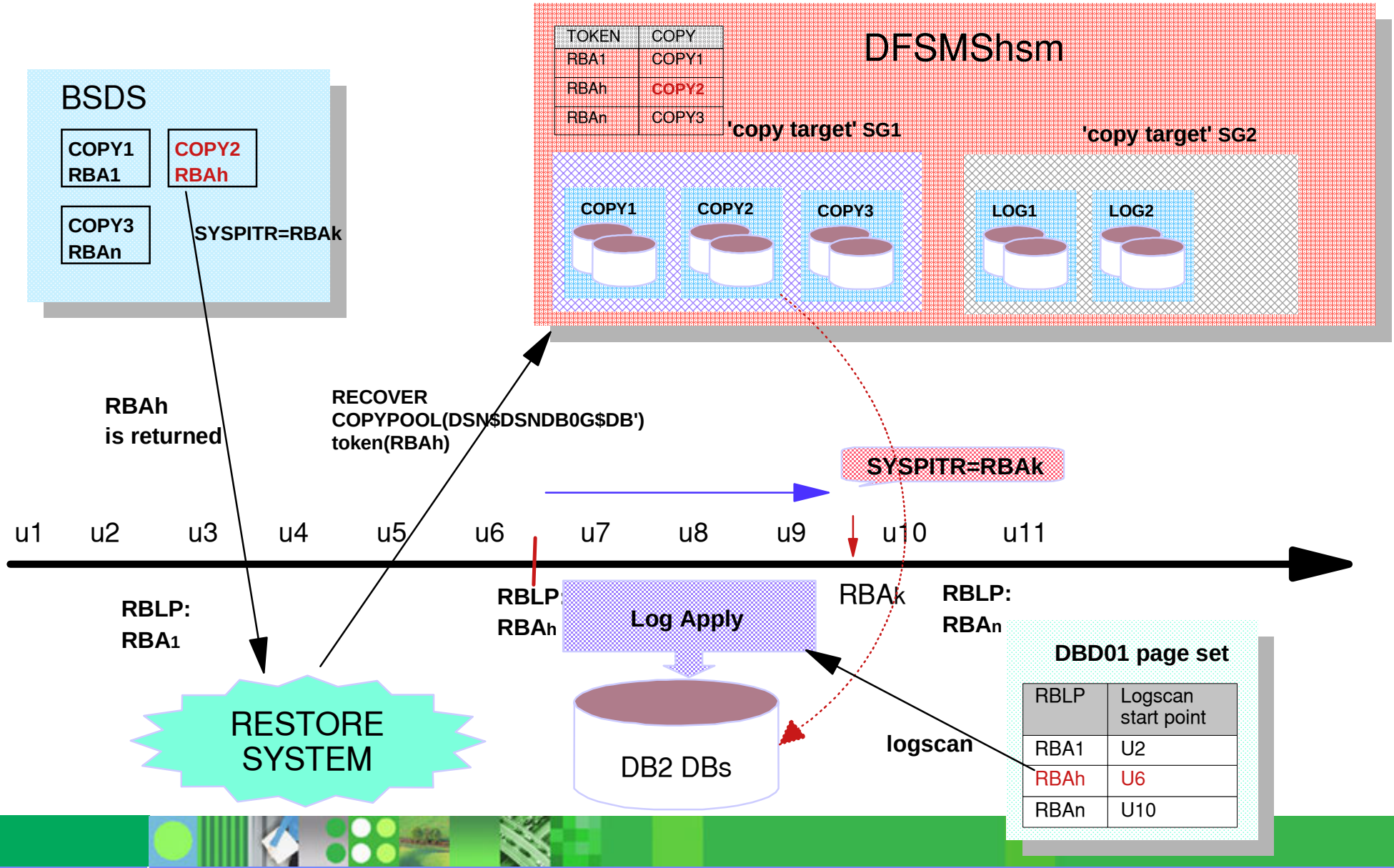
    **GEN(gen)**

    to restore the database and log copypool
  - Start DB2 and inflight URs are backed out

# RECOVER SYSTEM to an arbitray PIT

- RESTORE SYSTEM utility is needed
  - Use copies from BACKUP SYSTEM FULL or DATAONLY
    - RESTORE SYSTEM does not restore LOG backup copies, therefore copies from DATAONLY is enough
  - Two phases
    - RESTORE phase: recover the database volumes from the latest BACKUP version prior to the arbitrary PIT
    - LOG APPLY phase: apply log records to recover database object to that arbitrary PIT

# RESTORE SYSTEM



# System level restore to an arbitrary PIT -- single system

- Establish the 'PITR' conditional restart record
  - CRESTART CREATE SYSPITR=log-point (*truncation RBA value*)
  - MUST be in New Function Mode
- Start DB2 with a PITR CRCR
  - DB2 system enters into System Recover Pending mode
  - Implictly apply DEFER ALL, FORWARD = NO (except for in-doubt URs), and Access(Maint)
  - Write logs to rollback uncommitted changes
  - Reset database restrict status and utility job status

# System level restore to an arbitrary PIT -- single system

- RESTORE SYSTEM
  - Restore the "database" COPYPOOL version that was taken by BACKUP SYSTEM prior to the specified PIT recovery point
  - Perform log apply function
- RESTORE SYSTEM with LOGONLY specified
  - Performs log apply function only
  - Note: this option can run in z/OS 1.3 without BACKUP SYSTEM utility
    - Using Log Suspend/Resume and backup volumes manually
- Recover all objects that are marked in recover or rebuild pending state

# System level restore to an arbitrary PIT -- data sharing

- Establish the LRSN truncation point on all active members
  - CRESTART CREATE SYSPITR= end-lrsn
- Delete all CF structures
- Group restart each active member with the SYSPITR CRCR
  - All members MUST be restarted
- Restore system
  - Similar to the steps as in the non-data sharing environment

# System level restore -- notes

- When DB2 is in System Recover Pending state
  - Only RESTORE SYSTEM utility is allowed
  - START DATABASE command is not allowed
  - TERM UTIL command is not allowed
  - DISPLAY UTIL command will display only the status of RESTORE SYSTEM utility
  - SQL operation is not allowed
    - Claim request on any DB2 objects will be rejected with a -904 SQL code (reason code of 00C20269)
- Restore of the database volumes is done in parallel

# System level restore -- log recovery

- Read the DBD01 header page to retrieve RBLP - the log scan starting point
- Handles table space and index space
  - CREATEs
  - DROPs
  - EXTENDs
  - LOG NO events
    - Objects are marked in RECP or RBDP state
- Log apply phase will use fast log apply (FLA) function to recover objects in parallel

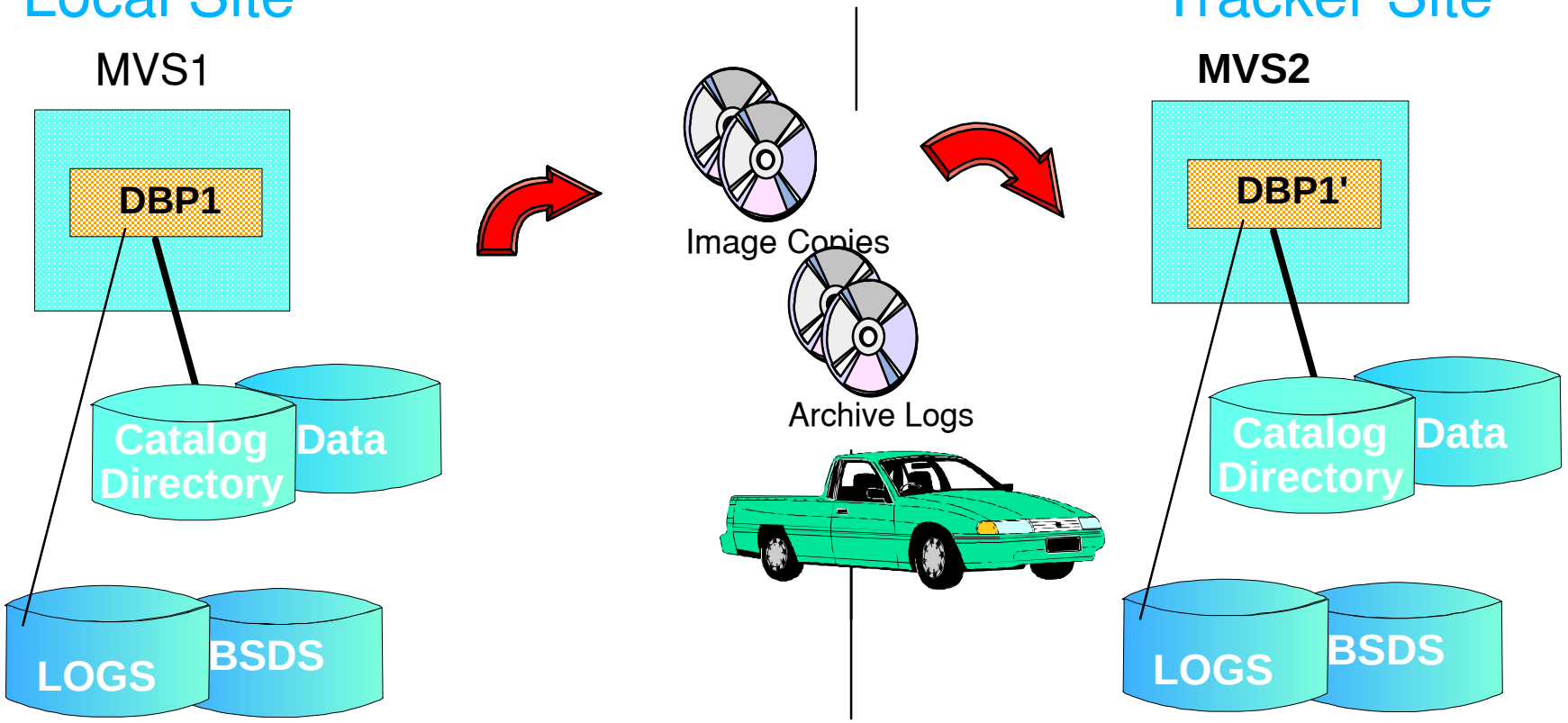
# System level restore -- log recovery (cont.)

- Log apply phase will take periodic checkpoints
  - Force modified data pages to DASD and trigger system checkpoint
  - Update the DBD01 header page with the new RBLP value
- At the end of log apply phase
  - An informal message is issued, if any object is marked RECP, RBDP or LPL during the log apply phase
  - Reset the PITR state of each member
- RESTORE SYSTEM utility is restartable

# SET LOG SUSPEND / RESUME command

- Can be used as an alternative to the BACKUP SYSTEM if it is not available
- More disruptive than BACKUP SYSTEM utility
- Issue this command from each *member* if in data sharing environment
- *Updates recovery based log point* (RBLP) in DBD01
- Backups taken either of two ways:
  - Use existing volume copy solution or
  - By HSM COPYPOOLs if running on z/OS R5

# Overview of Local and Tracker Site




***"Tracker " is a active DB2 at the recovery site which can keep shadow copies of local site data close to current***

***-can take over in case of disaster***

# "Tracker Site" Recovery

- Use BACKUP SYSTEM or Set Log Suspend/Resume to establish a tracker site
  - Dump FlashCopy target volumes to tapes
  - Send tapes to remote site
  - Restore data and logs from tapes
  - Don't start tracker DB2 until additional logs are received from the primary site
- Periodically send active, BSDS and archive logs to tracker site
  - PPRC, XRC, FTP, or Tapes
- Send image copies after load/reorg log(no)

# "Tracker Site" Recovery ...

- Each tracker recovery cycle
  - Run RESTORE SYSTEM LOGONLY to roll database forward using logs
  - Use image copies to recover objects that are in recover pending state
  - Rebuild indexes that are in rebuild pending state
- When disaster happened
  - Process the last tracker cycle if there are additional logs
  - Start the remote DB2 as a non-tracker
    - Perform normal restart work

# Prerequisites

- z/OS V1R5 and DFSMShsm
- DASD control units which support ESS Flashcopy APIs
- DB2 datasets must reside on SMS-managed volumes
- Must be in New Function Mode
- RESTORE SYSTEM LOGONLY can be executed under z/OS 1.3
  - Assumes you have used -Set Log Suspend
  - Manually dumped volumes (as today)
  - Manually restored volumes (as today)
  - Eliminates complex recovery procedures for Disaster Recovery

# Future directions

- *The FlashCopy source and target volumes can reside on different Sharks*
- *DFSMShsm will automatically manage FlashCopy target volumes to tapes*
- *Use volume level backups as the source for DB2 object level recovery*
- *Manage data set level FlashCopy*
- *Support object level point-in-time recovery*
  - *Rollback uncommitted changes*

# Summary

- *A fast and non-disruptive backup solution using*
  - *ESS FlashCopy or RVA SnapShot and*
  - *DB2 Backup System Utility*
- *No longer need to suspend logs*
- *Backups are managed by DB2 and DFSMShsm to support system level PIT recovery*
- *Restore System Utility can recover DB2 system to an arbitrary PIT*
- *Restore System Logonly can support DB2 Tracker for Disaster Recovery*