The DB2 for z/OS Log

Not just for System Programmers

Paul Bartak
IBM Certified Consulting IT Specialist
DB2 Information Management
St. Louis, MO

Disclaimer

The information contained in this presentation has not been submitted to any formal IBM review and is distributed on an "As Is" basis without any warranty either express or implied. The use of this information is a customer responsibility.

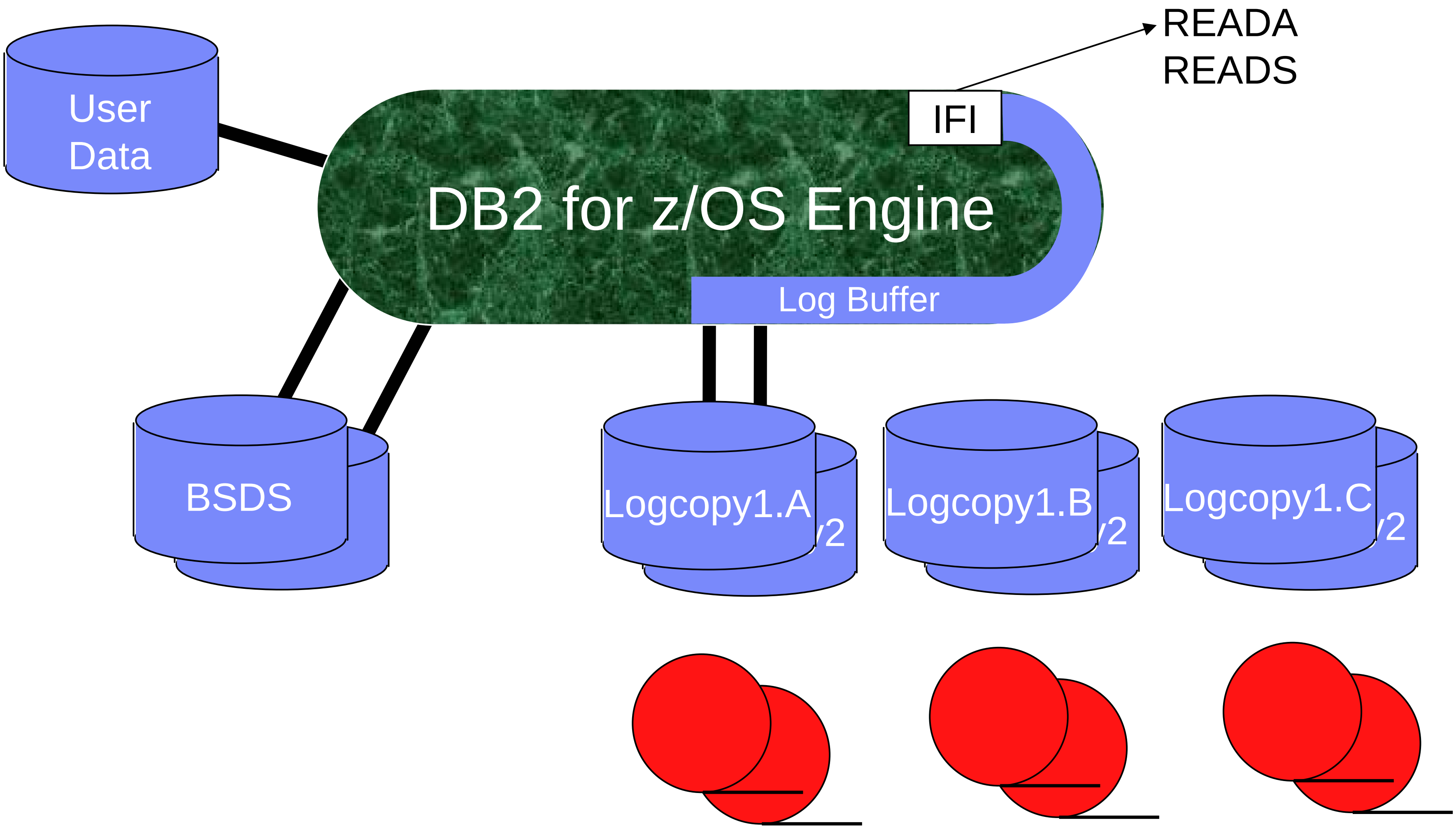
The measurement results presented here were run in a controlled laboratory environment using specific workloads. While the information here has been reviewed by IBM personnel for accuracy, there is no guarantee that the same or similar results will be obtained elsewhere. Performance results depend upon the workload and environment. Customers attempting to adapt this data to their own environments do so at their own risk.

In addition, the material in this presentation may be subject to enhancements or Programming Temporary Fixes (PTFs) subsequent to general availability of the code.

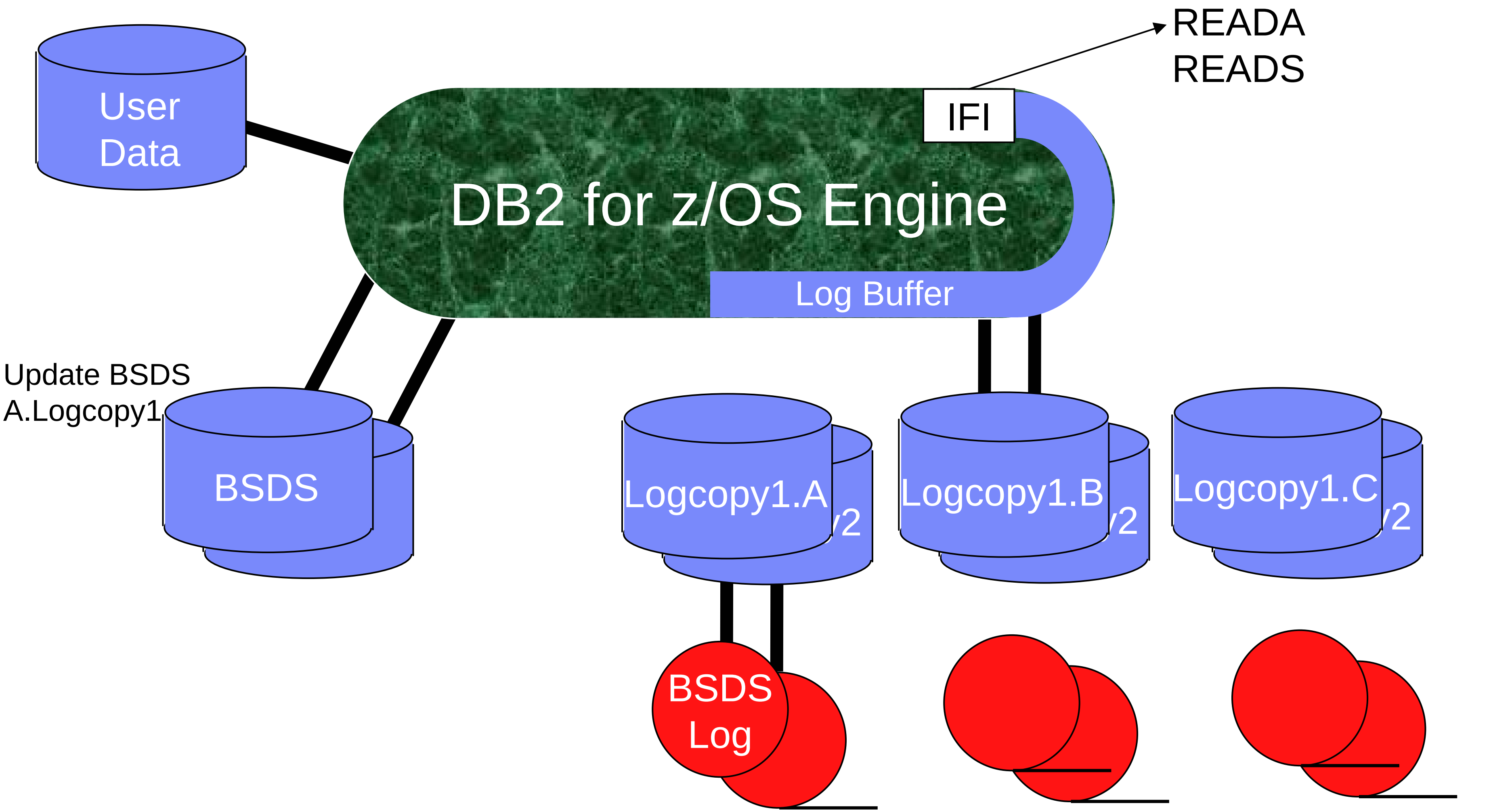
Agenda

- DB2 Log Architecture
- Supporting DB2 Operations
- Commands, Utilities
- Restart
- Recover
- Replication
- Summarized messages & ZParms
- IBM DB2 Tooling to exploit log information

Basic DB2 Architecture

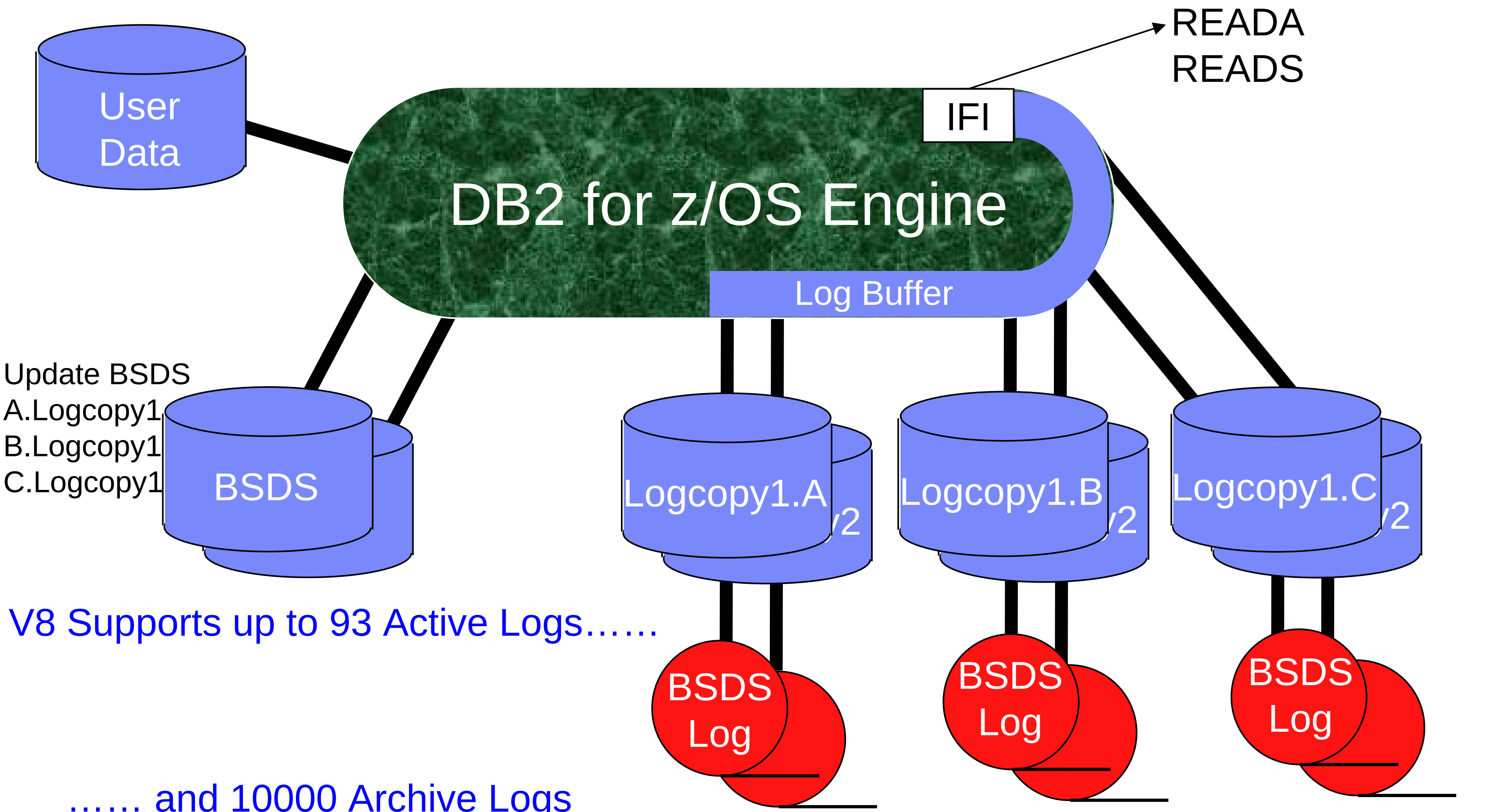


Basic DB2 Architecture



Attempt to fit one active log on to an archive tape

Basic DB2 Architecture

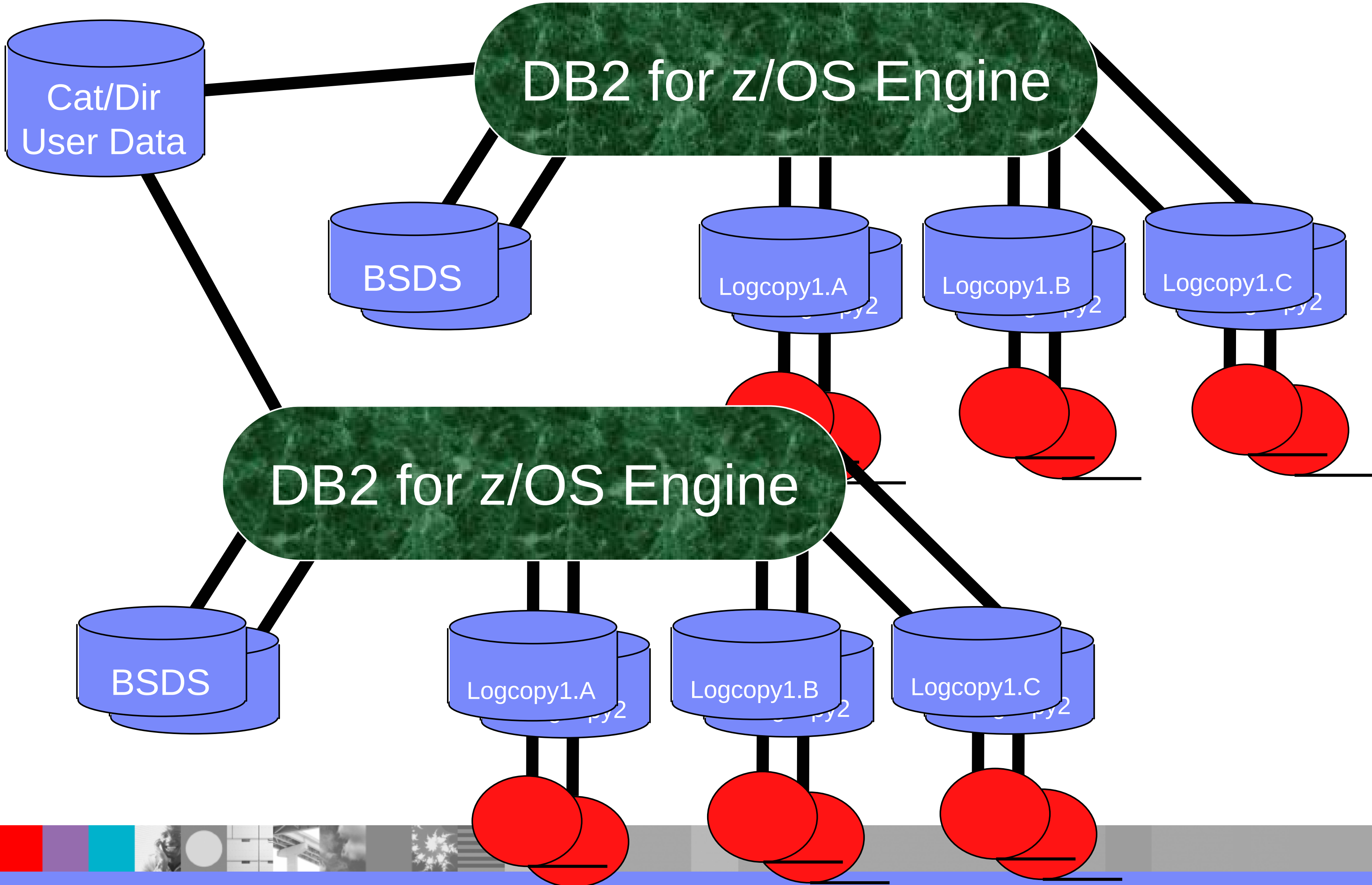


V8 Supports up to 93 Active Logs……

…… and 10000 Archive Logs

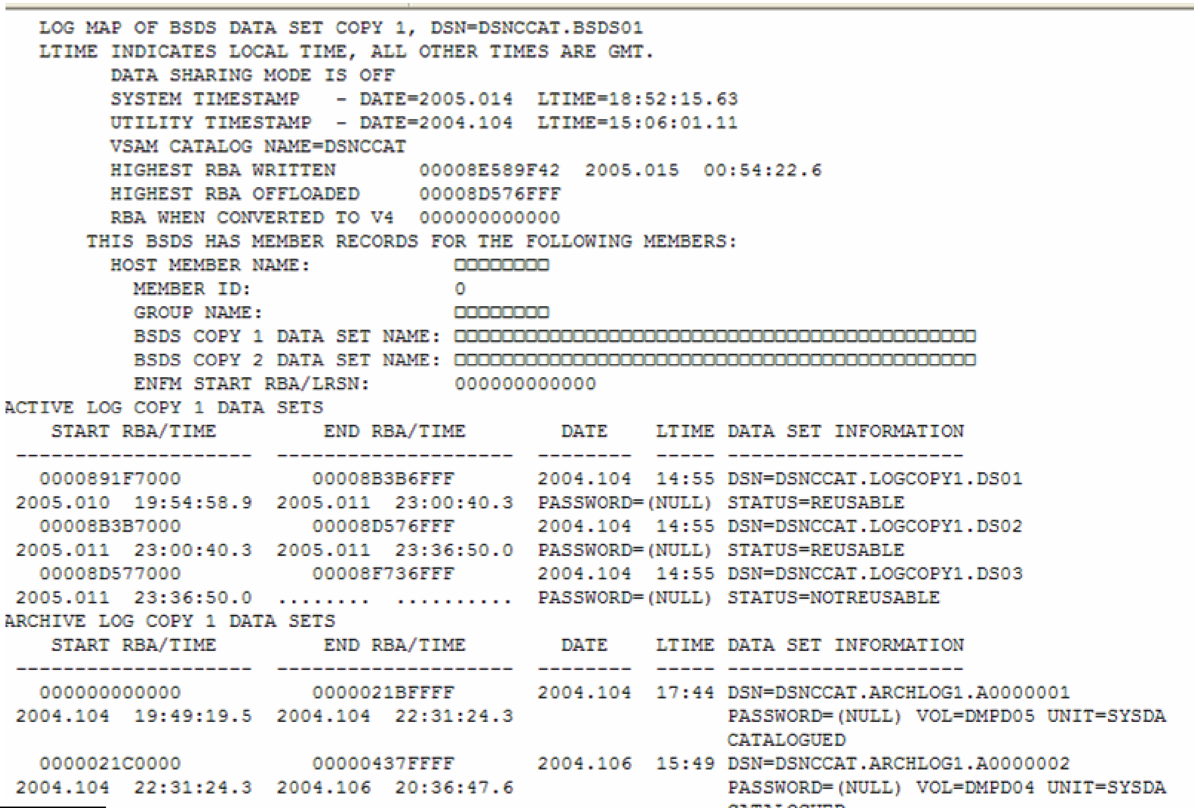
Attempt to fit one active log on to an archive tape

Basic DB2 Architecture – Data Sharing

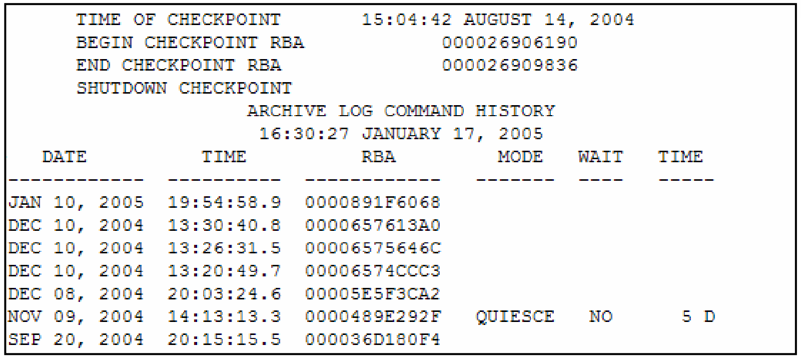


Bootstrap Dataset (BSDS)

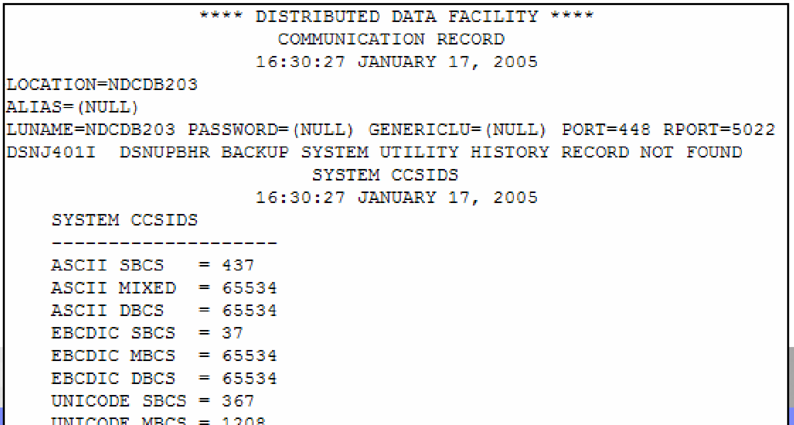
- Two small, identical, VSAM, KSDS datasets
- Used to track key information for DB2 to start / recover
- Tracks
  - Active Logs, the archive status, and RBA range
  - Archive Logs
  - Recent log point
  - Checkpoints
  - Data Sharing definitions
  - DDF Information
  - HLQ of the ICF Catalog Name
- V8 BSDS Reformat
  - DSNJNCVB is run after migration / installation is complete



```
   LOG MAP OF BSDS DATA SET COPY 1, DSN=DSNCCAT.BSDS01
   LTIME INDICATES LOCAL TIME, ALL OTHER TIMES ARE GMT.
         DATA SHARING MODE IS OFF
         SYSTEM TIMESTAMP   - DATE=2005.014  LTIME=18:52:15.63
         UTILITY TIMESTAMP  - DATE=2004.104  LTIME=15:06:01.11
         VSAM CATALOG NAME=DSNCCAT
         HIGHEST RBA WRITTEN        00008E589F42  2005.015  00:54:22.6
         HIGHEST RBA OFFLOADED      00008D576FFF
         RBA WHEN CONVERTED TO V4   000000000000
       THIS BSDS HAS MEMBER RECORDS FOR THE FOLLOWING MEMBERS:
         HOST MEMBER NAME:             □□□□□□□□
           MEMBER ID:                  0
           GROUP NAME:                 □□□□□□□□
           BSDS COPY 1 DATA SET NAME:  □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
           BSDS COPY 2 DATA SET NAME:  □□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
           ENFM START RBA/LRSN:        000000000000
ACTIVE LOG COPY 1 DATA SETS
    START RBA/TIME          END RBA/TIME        DATE     LTIME DATA SET INFORMATION
 --------------------  --------------------  --------  ----- --------------------
   0000891F7000            00008B3B6FFF      2004.104  14:55 DSN=DSNCCAT.LOGCOPY1.DS01
 2005.010  19:54:58.9  2005.011  23:00:40.3  PASSWORD=(NULL) STATUS=REUSABLE
   00008B3B7000            00008D576FFF      2004.104  14:55 DSN=DSNCCAT.LOGCOPY1.DS02
 2005.011  23:00:40.3  2005.011  23:36:50.0  PASSWORD=(NULL) STATUS=REUSABLE
   00008D577000            00008F736FFF      2004.104  14:55 DSN=DSNCCAT.LOGCOPY1.DS03
 2005.011  23:36:50.0  ........  ..........  PASSWORD=(NULL) STATUS=NOTREUSABLE
ARCHIVE LOG COPY 1 DATA SETS
    START RBA/TIME          END RBA/TIME        DATE     LTIME DATA SET INFORMATION
 --------------------  --------------------  --------  ----- --------------------
   000000000000            0000021BFFFF      2004.104  17:44 DSN=DSNCCAT.ARCHLOG1.A0000001
 2004.104  19:49:19.5  2004.104  22:31:24.3                  PASSWORD=(NULL) VOL=DMPD05 UNIT=SYSDA
                                                             CATALOGUED
   0000021C0000            00000437FFFF      2004.106  15:49 DSN=DSNCCAT.ARCHLOG1.A0000002
 2004.104  22:31:24.3  2004.106  20:36:47.6                  PASSWORD=(NULL) VOL=DMPD04 UNIT=SYSDA
                                                             CATALOGUED
```



```
      TIME OF CHECKPOINT       15:04:42 AUGUST 14, 2004
      BEGIN CHECKPOINT RBA              000026906190
      END CHECKPOINT RBA                000026909836
      SHUTDOWN CHECKPOINT
                     ARCHIVE LOG COMMAND HISTORY
                      16:30:27 JANUARY 17, 2005
   DATE          TIME          RBA           MODE    WAIT    TIME
------------  ----------  ------------   -------  ----   -----
JAN 10, 2005  19:54:58.9  0000891F6068
DEC 10, 2004  13:30:40.8  0000657613A0
DEC 10, 2004  13:26:31.5  00006575646C
DEC 10, 2004  13:20:49.7  00006574CCC3
DEC 08, 2004  20:03:24.6  00005E5F3CA2
NOV 09, 2004  14:13:13.3  0000489E292F   QUIESCE   NO       5 D
SEP 20, 2004  20:15:15.5  000036D180F4
```



```
                 **** DISTRIBUTED DATA FACILITY ****
                        COMMUNICATION RECORD
                      16:30:27 JANUARY 17, 2005
LOCATION=NDCDB203
ALIAS=(NULL)
LUNAME=NDCDB203 PASSWORD=(NULL) GENERICLU=(NULL) PORT=448 RPORT=5022
DSNJ401I  DSNUPBHR BACKUP SYSTEM UTILITY HISTORY RECORD NOT FOUND
                           SYSTEM CCSIDS
                      16:30:27 JANUARY 17, 2005
    SYSTEM CCSIDS
    --------------------
    ASCII SBCS   = 437
    ASCII MIXED  = 65534
    ASCII DBCS   = 65534
    EBCDIC SBCS  = 37
    EBCDIC MBCS  = 65534
    EBCDIC DBCS  = 65534
    UNICODE SBCS = 367
    UNICODE MBCS = 1208
```

RBAs & LRSNs

- Relative Byte Address OR Log Record Sequence Number
- RBA
  - Ever increasing hexadecimal number
- LRSN
  - Based on timestamps from the Sysplex Timer
- Starts with x’0000000000000000’ when a new (non-data sharing) DB2 subsystem is started.
- Each log record is assigned a unique RBA
- Rate of log record creation (and therefore RBA progression) is based on DB2 activity volume, not time
- Tracked in the BSDS

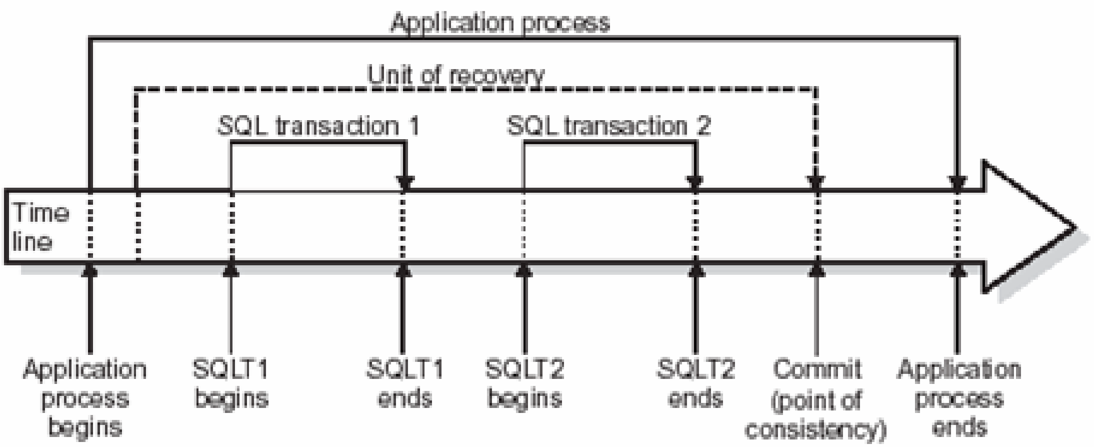
X’0000000000000000’

Types of Log Records - Summary

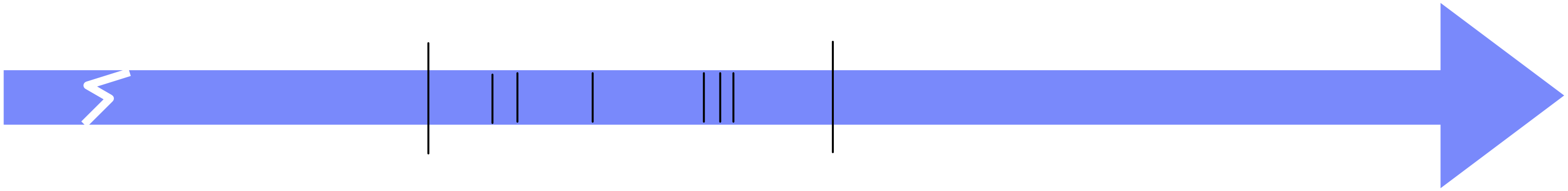
- Unit of Recovery (UR)
  - Begin UR
    - First database change
    - IDs remaining Undo / Redo records in the LRH
  - Undo / Redo (DATA CAPTURE CHANGES logs entire row)
  - End Phase 2
    - Commit or Rollback
  - There are others based on connection type
- Compensation (Redo / Undo)
  - Includes the tracking of dataset creations
- DBET
  - Database Exception Table
    - Logical Page List (LPL)
    - Write Error Page Range (WEPR)
  - DISPLAY DATABASE…. RESTRICT
  - Tracks image copies of SYSUTILX, DBD01, and SYSCOPY
- Checkpoints
- Dataset page set control records
  - Similar to SYSLGRNX

Units of Recovery (UR) or Logical Units of Work (LUW)

- A grouping of database modifications that are considered atomic
- UR begins when an connection first modifies the database
- UR ends when the connection commits or does a rollback
  - Implicit or Explicit
- Used by:
  - Crash Recovery
    - Forward Log Apply
    - Backwards Log Apply
  - Utilities
    - Roll Forward
    - Log Apply



$$$ from Savings…..to Checking

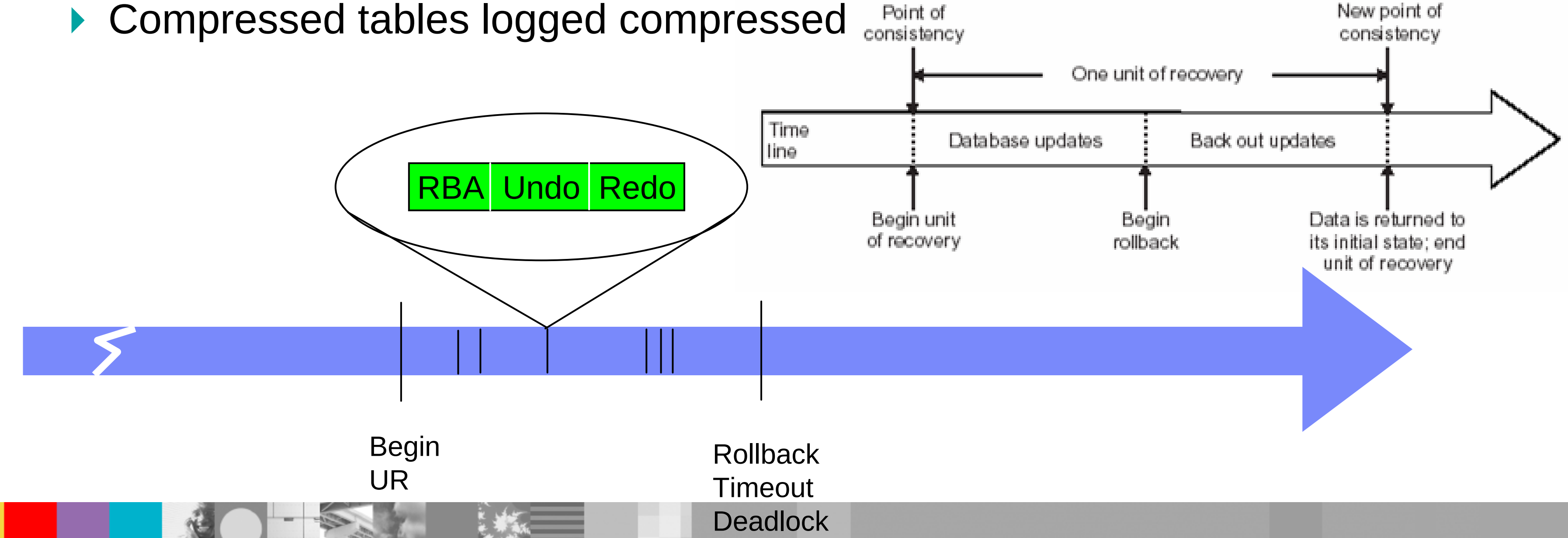


Begin UR

Commit or Syncpoint

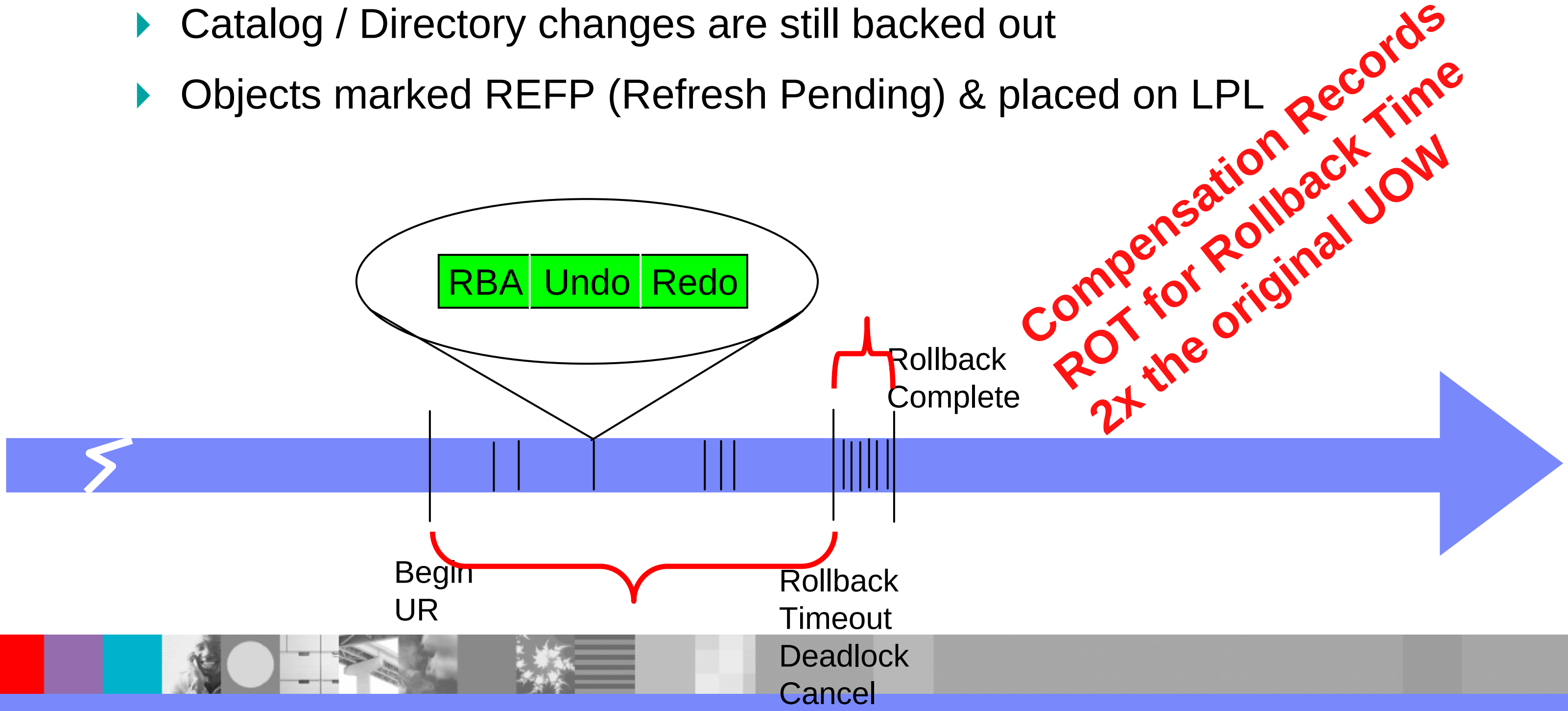
Units of Recovery (UR) or Logical Units of Work (LUW)

- Undo / Redo records
  - Insert
    - Redo records contain the Inserted values
  - Delete
    - Undo records contain the row values prior to the Delete
  - Update
    - Undo records contain the row values prior to the Update
    - Redo records contain the row values after the Update
  - Compressed tables logged compressed



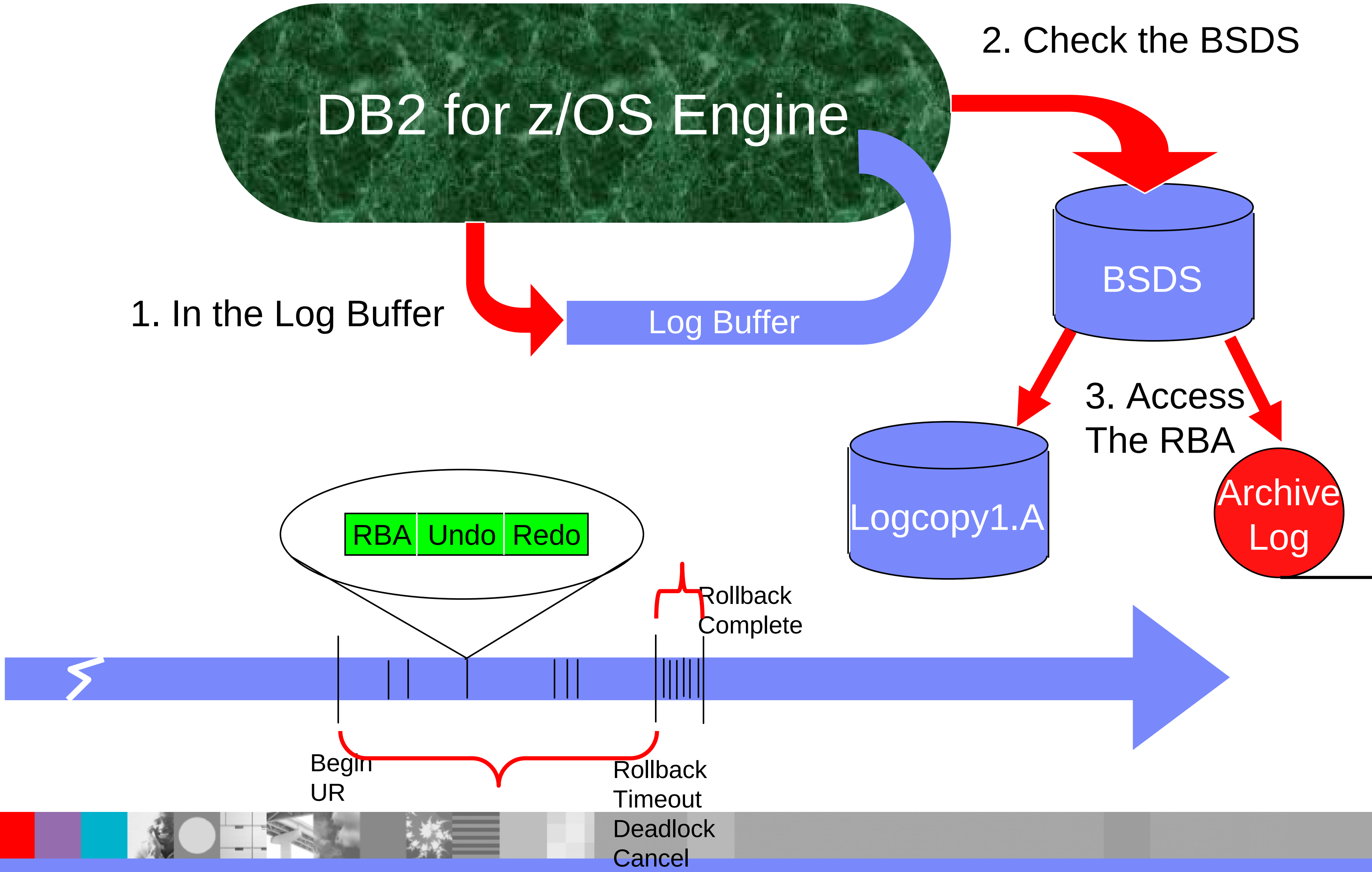
Units of Recovery (UR) or Logical Units of Work (LUW)

- LOBs
  - LOB changes are backed out with the non-LOB data during rollbacks, even if the auxiliary Tablespaces has the LOG NO attribute
- CANCEL THREAD(x) NOBACKOUT
  - Option to cancel long running threads
    - Does not read log records
    - Does not write or apply compensation records
  - Catalog / Directory changes are still backed out
  - Objects marked REFP (Refresh Pending) & placed on LPL



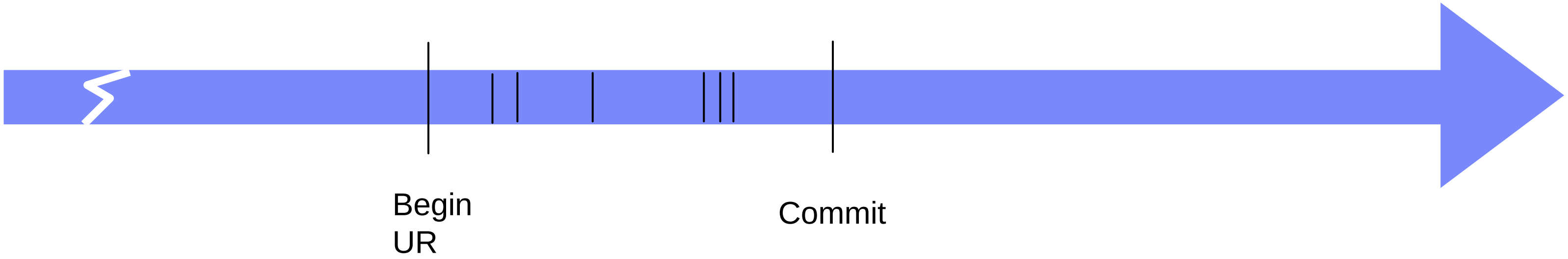
Compensation Records
ROT for Rollback Time
2x the original UOW

Where DB2 goes looking for Log Record(s)



Support of Deferred Write

- During processing, User Data is read / modified in Bufferpools
- This data is written to disk based on several thresholds
  - Bufferpool deferred write thresholds
  - Checkpoints
  - But not specifically on Commit
- However, Log Buffers must be force written to the Log at commit
  - This provide transactional integrity
  - For data sharing this can also involve forced writes to the GBP
  - The Log can then be used to reconstruct the data on disk
- Cancel without roll back places objects in a Refresh Pending state
  - May fail if Cat / Dir cannot be rolled back OR if part of global transaction



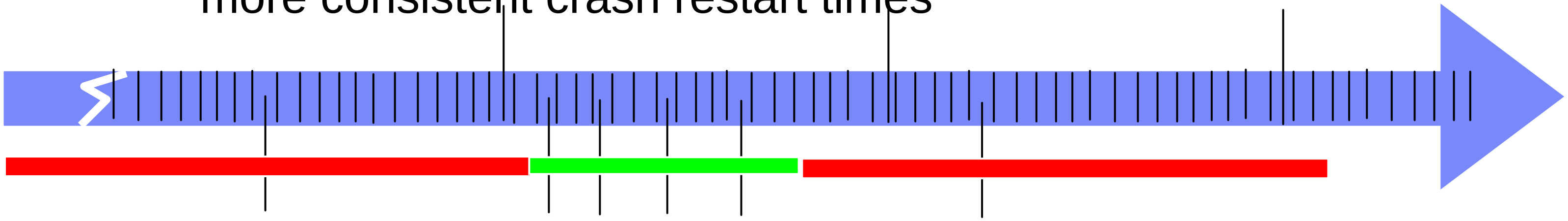
Checkpoints

- Based on time or number of log records
- Enables DB2 to reconstruct a “current state” of the subsystem during start up
  - Did it end normally?
  - Was it an abnormal completion?
  - If so, what was active at the time of the abend?

Log Record based:

consistent spacing of checkpoints

more consistent crash restart times



Time based:

consistent timing of checkpoints

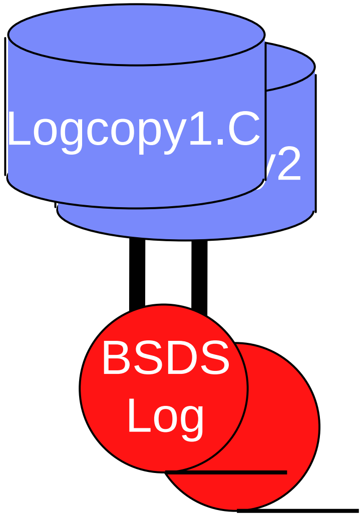
changing transaction volumes result in skewed frequency

Checkpoints

- At start up
  - DB2 first locates it’s BSDS(s)
  - DB2 finds the last checkpoint in the Log
  - The BSDS points to where DB2 can find the Checkpoint
  - CURRENT STATUS REBUILD is the restart phase where DB2 reconstructs what was happening at Checkpoint time
    - In flights, In Aborts, In Commit, In Doubt
  - FORWARD LOG APPLY is when DB2 reads forward from the Checkpoint and applies log records
    - Data / Indexes are retrieved into the Bufferpool and modified
  - BACKWARD LOG APPLY is the last phase where any remaining URs (not dependent on another 2PC resource) are reversed out of the system

Offloading

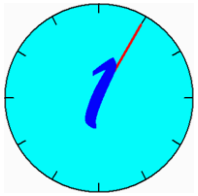
- The process of copying Active Log records to archive logs (QSAM)
  - Each archive log record is a VSAM CI from the active log
- Triggered by:
  - An Active Log is full
  - At DB2 start when an Active Log is full
  - **ARCHIVE LOG** command
    - Current Active Log(s) is truncated
  - A write error for a log record
    - Current Active Log(s) is truncated
- If all Active logs fill without a completed offload DB2 stops processing until an offload completes
- Archive log record is a VSAM CI
- Can choose dual archiving
- Can offload to disk



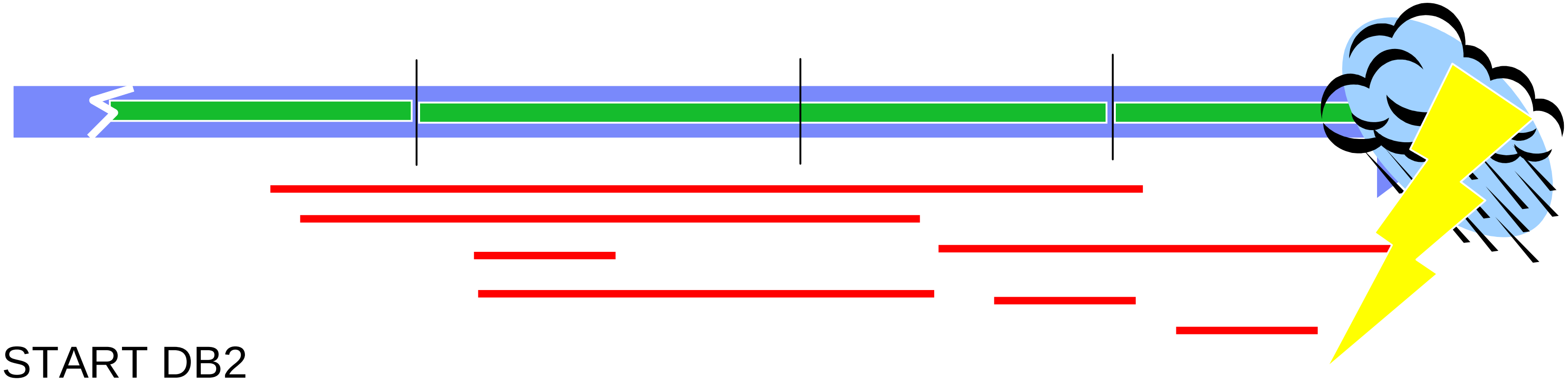
Commands, Utilities, & Controls

- ARCHIVE LOG
  - Requires the SYSADM authority or ARCHIVE privilege
  - Truncates the currently active log and triggers an offload
  - Extensions for Scope (Data Sharing) Quiesce & Offload task management
  - QUIESCE option
    - Quiesce point is recorded in the BSDS, not SYSCOPY
  - Could then use service aids like DSN1LOGP for analysis / diagnosis
- Numerous ZParms
  - SET LOG command for dynamic changes
    - SET LOG LOGLOAD(0) or SET LOG CHKTIME (0) to force a checkpoint
- DISPLAY LOG to see current setting & recent Checkpoint activity
- Print Log Map utility (DSNJU004) for checkpoint history

TIME(60)



DB2 Restart



-START DB2

Read the BSDS to find the latest Log and Checkpoint(s)

LOG INITIATION

CURRENT STATUS REBUILD

Indicates: In Flights, In Commits, In Aborts, In Doubt

FORWARD LOG APPLY

Assisted by Fast Log Apply

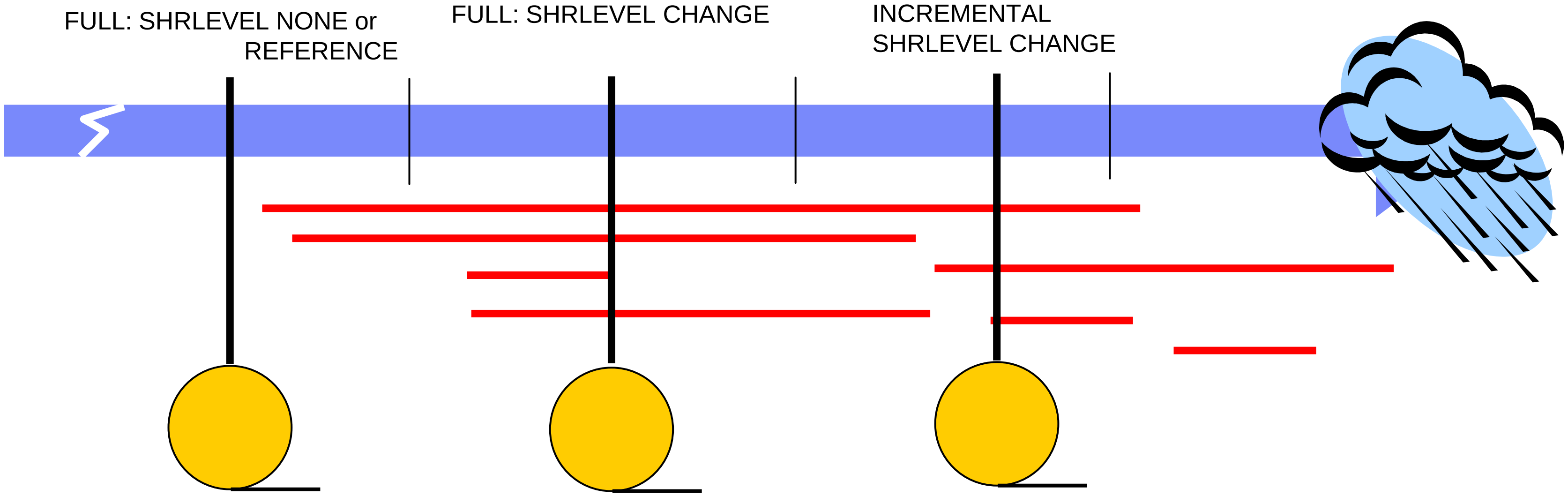
BACKWARD LOG APPLY

Can be limited & objects left deferred

A Checkpoint is taken

Open for business

DB2 Object Recovery



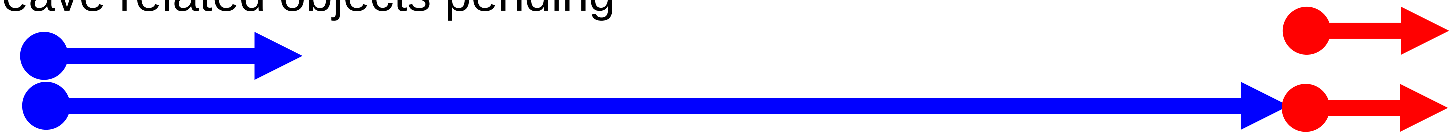
RECOVER x:

TOCOPY ●

May leave related objects pending

TO PIT

To current



Consider CLOSE / Soft Close parameters

SYSLGRNX

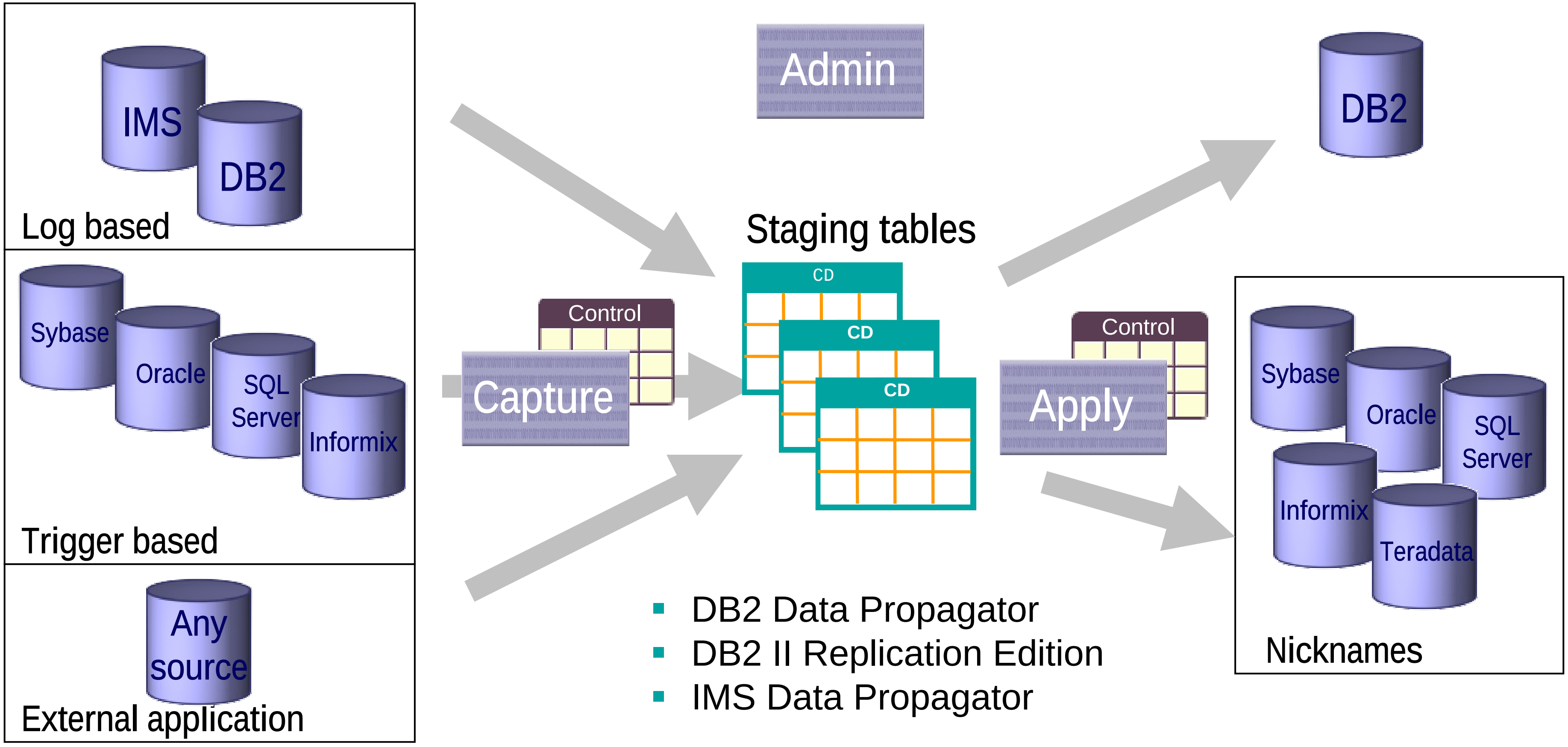
Additional Points

- LIMIT BACKOUT & BACKOUT DELAY options to mitigate long running impact on DB2 restart
- A Conditional Restart record can be added to the BSDSs via the Change Log Inventory utility.
  - This modifies RBAs and Restart behavior for specialized conditions

DB2 Tracker Site

- A separate DB2 subsystem or data sharing group
- Exists solely for the purpose of keeping shadow copies of the primary site’s data
- Tracker site is not available for independent work
- ZParm TRKSITE=YES
- Logs are shipped to the Tracker site
- RECOVERY LOGONLY run to update the data / indexes

Log Support of Replication – SQL Replication



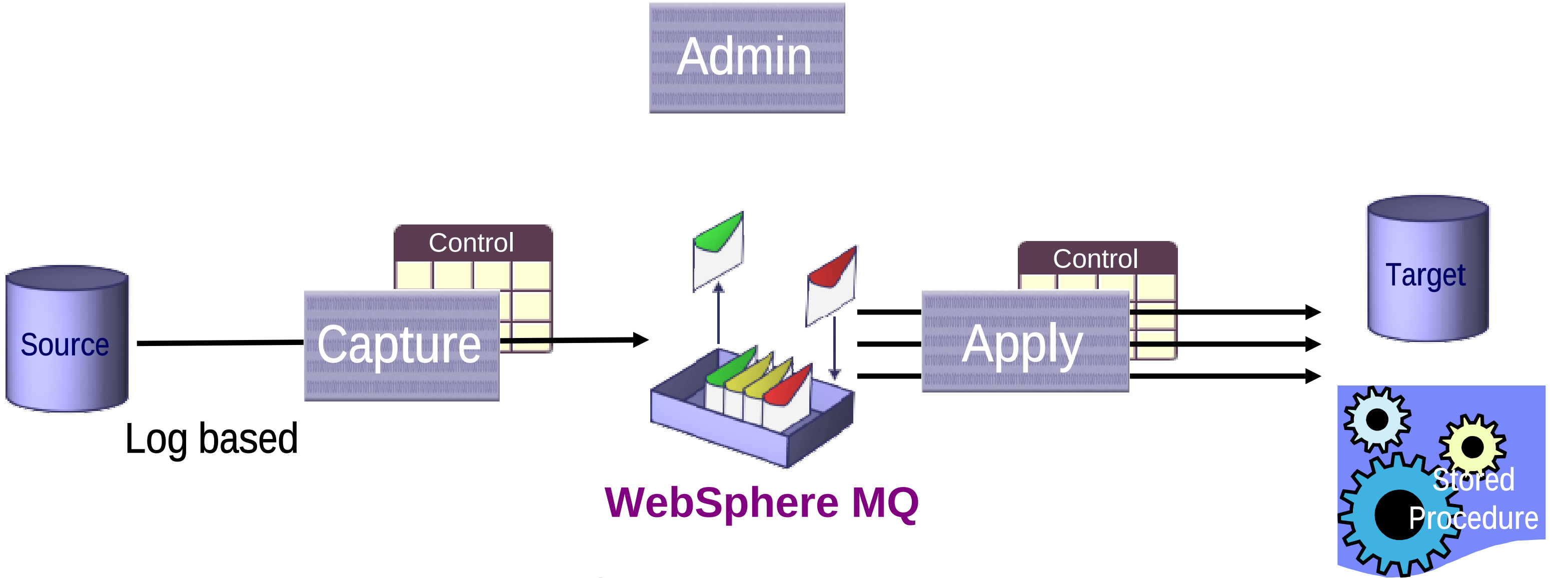
- DB2 Data Propagator
- DB2 II Replication Edition
- IMS Data Propagator

Registration → Subscription

Subscription

Subscription

Log Support of Replication – Q Replication



- Each message represents a transaction
  - A Queue represents a database log file or set of related tables from a database log file
- Highly parallel apply process
  - Non dependent transactions re-parallelized at the target
- Capture and Apply work asynchronously and disconnected
- Continuous replication, not cycle based like SQL Replication
- Differentiated conflict detection and resolution
- Integrated infrastructure for replication and publishing
- DB2 to DB2 today
  - Staged availability of heterogeneous support
- Data Integrity
  - Persistent messaging with WebsphereMQ
  - Detects missing messages

Messages Sampling

- If switching active logs & DB2 determines that a previous offload is still active:
  - DSNJ017E: WARNING - OFFLOAD TASK HAS BEEN ACTIVE SINCE *date-time* AND MAY HAVE STALLED
  - **ARCHIVE LOG CANCEL OFFLOAD** to cancel and restart
  - **DISPLAY LOG** to see offload task status
- During the last log messages will appear in the Master log indicating percentage of capacity
  - DSNJ110E capacity of log remaining message
  - IFCID trace record 0330 if Statistics Class 3 is on
- A message is indicated when all Active Logs are full
  - DSNJ111E
  - Halts processing that requires log writes
- Long running URs
  - DSNJ031I – log records per UR
  - DSNR035I – checkpoints per UR
  - IFCID 0313 written if statistics class 3 is on

ZParms Sampling

Output Buffer	OUTBUFF	40k – 400000k (4000k)
Log Apply Storage	LOGAPSTG	0M – 100M (0M)
Write to Oper	ARCWTOR	YES, NO (YES)
Archive Retn Period	ARCRETN	0 – 9999 Days (9999)
Read Copy 2 Archive	ARC2FRST	YES, NO (NO)
Checkpoint Frequency	CHKFREQ	0 – 60, 200 - 16000000
UR Log Write Freq	URLGWTH	0 – 1000K (0)
UR Check Frequency	URCHKFREQ	0 – 255 (0)
Limit Backout	LBACKOUT	AUTO,YES,NO (AUTO)
Backout Duration	BACKODUR	0 – 255 (5)
RO Switch Checkpoints	PCLOSEN	1 – 32767 (5)
RO Switch Time	PCLOSET	1 – 32767 (10)