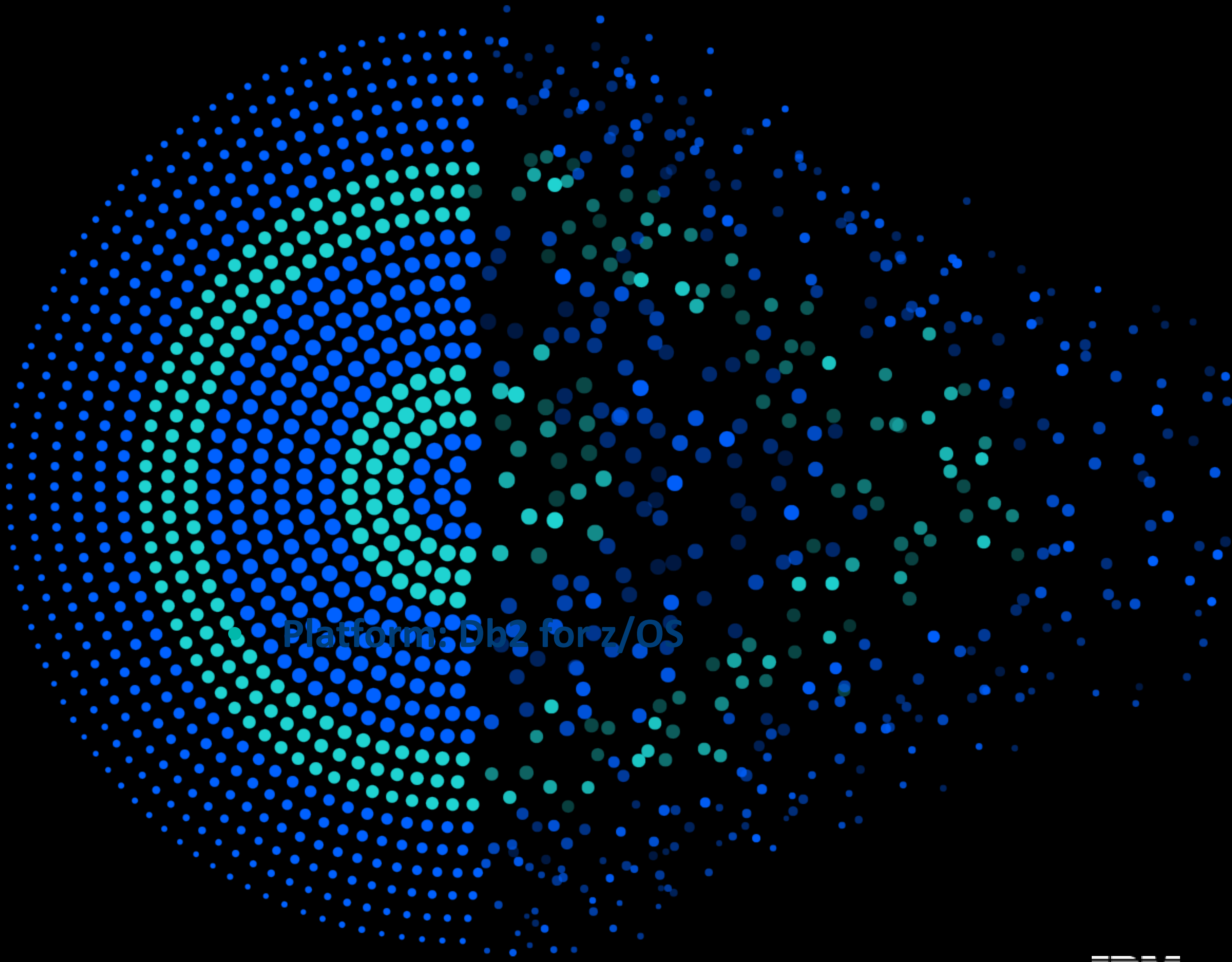
# Starter's Guide to Db2 for z/OS Data Sharing Monitoring and Tuning

John Campbell
Distinguished Engineer
Db2 for z/OS Development



IBM Data and AI

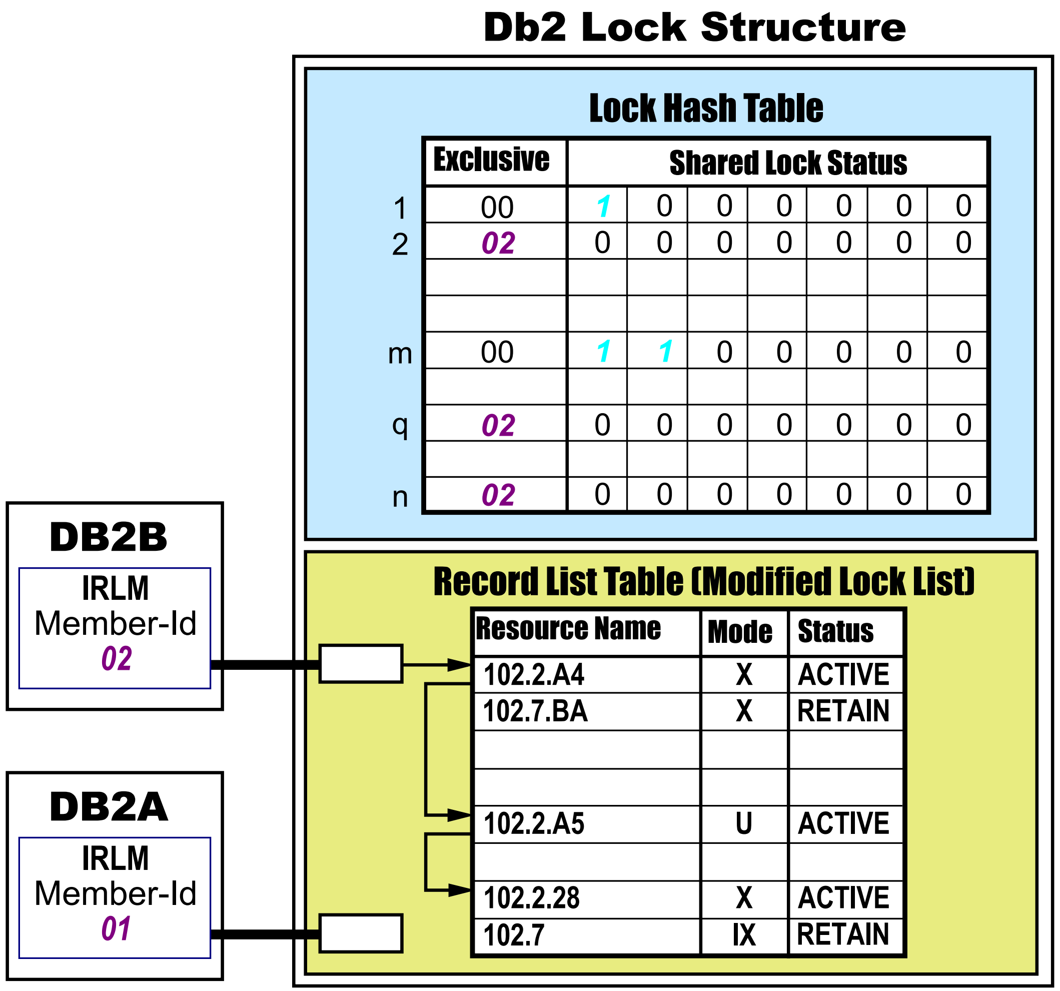
## Agenda

- **Global locking**
  - Db2 LOCK1 structure
  - Global contention monitoring and tuning

- **Group buffer pools**
  - GBP reads
  - GBP writes
  - GBP castout
  - GBP monitoring and tuning

## Db2 LOCK1 structure

- **What is it used for?**
  - Fast inter-system global lock conflict detection
  - Optimisation for fast grant of global lock where no contention
  - Fast inter-system page latch conflict detection for cases where sub-page concurrency is allowed
    - Row level locking, space map pages, index leaf pages
  - Inter-system read/write interest tracking for Db2 objects
  - Retained locks in case of Db2 member failure



Lock Hash Table

| | Exclusive | Shared Lock Status | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | 00 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | | |
| | | | | | | | | |
| m | 00 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | | |
| q | 02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | | | | | | | |
| n | 02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Record List Table (Modified Lock List)

| Resource Name | Mode | Status |
|---|---|---|
| 102.2.A4 | X | ACTIVE |
| 102.7.BA | X | RETAIN |
| | | |
| | | |
| 102.2.A5 | U | ACTIVE |
| | | |
| 102.2.28 | X | ACTIVE |
| 102.7 | IX | RETAIN |

# Db2 LOCK1 structure …

- **LOCK1 structure size is defined in the CFRM policy**

```
STRUCTURE NAME(DSNDB2_LOCK1)
          SIZE(1024M)
          INITSIZE(512M)
          ALLOWAUTOALT(NO)
          PREFLIST(COUPLE01,COUPLE02)
```

By default, Db2 tries to do a 50-50 split between Lock Hash Table and Record List Table

Lock Hash Table (LTEs) = 256MB
Record List Table (RLEs) = 256MB

*LTE size depends on number of data sharing members*
*2 bytes for 1-7 members*
*4 bytes for 8-23 members*
*8 bytes for 24-32 members*

```
STRUCTURE NAME(DSNDB2_LOCK1)
          SIZE(1024M)
          INITSIZE(768M)
          ALLOWAUTOALT(NO)
          PREFLIST(COUPLE01,COUPLE02)
```

The Lock Hash Table has to be a power of 2

Lock Hash Table (LTEs) = 256MB
Record List Table (RLEs) = 512MB

⚠ The record table is susceptible to storage shortages if the structure is too small or if the allocation of the lock table leaves too little storage for the record table

## Db2 LOCK1 structure …

- **Shortage of RLEs**

```
DXR170I    irlmx THE LOCK STRUCTURE wwwwwwww IS zz% IN USE      ← Alert at 50%, 60%, 70%
DXR142E    irlmx THE LOCK STRUCTURE wwwwwwww IS zzz% IN USE     ← Alert at 80%, 90%, 100%
```

  - At 80% full, data sharing continues with no restrictions, but storage is approaching a critical threshold
  - At 90% full, only 'must-complete' type of requests that require lock structure storage are processed
  - At 100% full, any request that requires lock structure storage is denied with an 'out of lock structure storage’ error

## Global contention

```
DATA SHARING LOCKING            QUANTITY  /SECOND  /THREAD  /COMMIT
---------------------------     --------  -------  -------  -------
GLOBAL CONTENTION RATE (%)          0.64
FALSE CONTENTION RATE (%)           0.11
...
SYNCH.XES - LOCK REQUESTS         227.5M    10.6K  1368.75   458.86
SYNCH.XES - CHANGE REQUESTS      1340.7K    62.24     8.07     2.70
SYNCH.XES - UNLOCK REQUESTS       225.8M    10.5K  1358.14   455.30
ASYNCH.XES -CONVERTED LOCKS      3485.41     0.48     3.87     0.00
...
SUSPENDS - IRLM GLOBAL CONT     34192.00     1.59     0.21     0.07
SUSPENDS - XES GLOBAL CONT.      6076.00     0.28     0.04     0.01
SUSPENDS - FALSE CONT. MBR      21575.00     1.00     0.13     0.04
```

**ROT**

*Global Contention should be less than 3-5% of XES IRLM Requests*

Global Contention = SUSPENDS – IRLM + XES + FALSE (A)

XES IRLM Requests = (SYNCH. XES – LOCK + CHANGE + UNLOCK)
+ ASYNCH.XES – CONVERTED LOCKS + (SUSPENDS – IRLM + XES + FALSE) (B)

Global Contention Rate (%) = (A)/(B)*100

**ROT**

*False Contention should be less than 1-3% of XES IRLM Requests*

False Contention = SUSPENDS – FALSE (C)

False Contention Rate (%) = (C)/(B)*100

False Cont = false contention on lock table hash anchor point
Could be minimised by increasing the number of LTEs in the Lock Hash Table

## Resizing the Db2 LOCK1 structure

- **Increase size of LOCK1 structure if high False Contention rate and/or shortage of RLEs**
- **If shortage of RLEs but False Contention rate is OK, consider using the IRLM option LTE= to control the size of the Lock Hash Table**
  - Specify a value for the LTE= parameter in the IRLMPROC or issue the MODIFY irlmproc SET,LTE= command
  - Requires a REBUILD of LOCK1 structure

INITSIZE = 1024MB
Lock Hash Table (LTEs) = 512MB
Record List Table (RLEs) = 512MB

INITSIZE = 1280MB
LTE = 256 (based on 2-byte entries)
Lock Hash Table (LTEs) = 512MB
Record List Table (RLEs) = 768MB

INITSIZE = 1024MB
LTE = 128 (based on 2-byte entries)
Lock Hash Table (LTEs) = 256MB
Record List Table (RLEs) = 768MB

| For LTE= | 2-byte entries | 4-byte entries |
|---|---|---|
| 8 | 16 MB | 32 MB |
| 16 | 32 MB | 64 MB |
| 32 | 64 MB | 128 MB |
| 64 | 128 MB | 256 MB |
| 128 | 256 MB | 512 MB |
| 256 | 512 MB | 1024 MB |
| 512 | 1024 MB | 2048 MB |
| 1024 | 2048 MB | 4096 MB |
| 2048 | 4096 MB | |

## Global contention …

- **Use a light-weight locking protocol (isolation level) and exploit lock avoidance**
  - Benefits of lock avoidance
    - Increased concurrency by reducing lock contention
    - Decreased lock and unlock activity and associated CPU resource consumption
    - In data sharing, decreased number of CF requests and associated CPU overhead
    - Minimise impact of retained locks
  - Use ISOLATION(CS) <u>CURRENTDATA(NO)</u> or use ISOLATION(UR)
- **Commit frequently**
  - Reduce lock contention
  - Improve effectiveness of global lock avoidance
- **Avoid serialisation points e.g. single row table used as a ‘counter’**
  - Use IDENTITY column or pull value from SEQUENCE object <u>with CACHE</u>
- **Exploit table and index partitioning**

## Global contention …

ROT

*P-lock Negotiation should be less than 3-5% of XES IRLM requests*

```
DATA SHARING LOCKING           QUANTITY  /SECOND  /THREAD  /COMMIT
---------------------------    --------  -------  -------  -------
...
SYNCH.XES - LOCK REQUESTS        227.5M    10.6K  1368.75   458.86
SYNCH.XES - CHANGE REQUESTS     1340.7K    62.24     8.07     2.70
SYNCH.XES - UNLOCK REQUESTS      225.8M    10.5K  1358.14   455.30
ASYNCH.XES -CONVERTED LOCKS     1315.00     0.06     0.01     0.00
...
PSET/PART P-LCK NEGOTIATION    18037.00     0.84     0.11     0.04
PAGE P-LOCK NEGOTIATION         2863.00     0.13     0.02     0.01
OTHER P-LOCK NEGOTIATION        9067.00     0.42     0.05     0.02
P-LOCK CHANGE DURING NEG.      15991.00     0.74     0.10     0.03
```

P-lock Negotiation = PSET/PART P-LCK NEGOTIATION + PSET/PART P-LCK NEGOTIATION + OTHER P-LCK NEGOTIATION (A)

XES IRLM Requests = (SYNCH. XES – LOCK + CHANGE + UNLOCK) + ASYNC.XES – CONVERTED LOCKS + (SUSPENDS – IRLM + XES + FALSE) (B)

P-lock Negotiation Rate (%) = (A)/(B)*100

- **P-lock contention and negotiation can cause IRLM latch contention, page latch contention, asynchronous GBP write, active log write, GBP read**
  - Page P-lock contention by one thread causes Page Latch contention for all other threads in the same member trying to get to the same page

## Global contention …

- **Breakdown by page P-lock type in GBP statistics**

```
GROUP TOTAL  CONTINUED      QUANTITY  /SECOND  /THREAD  /COMMIT
-----------------------     --------  -------  -------  -------
                                                            ...
PAGE P-LOCK LOCK REQ          877.4K   122.88    14.91     3.64
 SPACE MAP PAGES            83552.00    11.70     1.42     0.35
 DATA PAGES                 10663.00     1.49     0.18     0.04
 INDEX LEAF PAGES             783.2K   109.69    13.31     3.25

PAGE P-LOCK UNLOCK REQ        926.8K   129.80    15.75     3.84

PAGE P-LOCK LOCK SUSP        8967.00     1.26     0.15     0.04
 SPACE MAP PAGES              593.00     0.08     0.01     0.00
 DATA PAGES                   143.00     0.02     0.00     0.00
 INDEX LEAF PAGES            8231.00     1.15     0.14     0.03

PAGE P-LOCK LOCK NEG        10285.00     1.44     0.17     0.04
 SPACE MAP PAGES                8.00     0.00     0.00     0.00
 DATA PAGES                    10.00     0.00     0.00     0.00
 INDEX LEAF PAGES           10267.00     1.44     0.17     0.04
```

For insert-intensive workloads with high page P-lock contention on space map pages, consider MEMBER CLUSTER (optionally combined with APPEND + LOCKSIZE ROW)

Do not use APPEND or LOCKSIZE ROW without MEMBER CLUSTER for an INSERT-at-the-end intensive workload → may result in excessive page p-lock contention on data pages and space map pages

For heavily concurrent insert activity (many concurrent threads) with high page P-lock contention on data pages caused by space search and false leads, consider INSERT ALGORITHM 2 (aka Fast Un-clustered INSERT)

If data page P-lock contention on small tables with LOCKSIZE ROW, consider MAXROWS=1 and LOCKSIZE PAGE to ‘simulate’ row locking and reduce spacemap free space update

If index tree P-lock (high index splits), consider

- Freespace tuning (PCTFREE, FREEPAGE)
- Exploit data and index partitioning to ‘dilute’ hot spot
- Increase index page size – warning: could also aggravate contention!

## Global contention …

- **Db2 accounting for more granular information**

```
CLASS 3 SUSPENSIONS     AVERAGE TIME  AV.EVENT  TIME/EVENT
--------------------    ------------  --------  ----------
LOCK/LATCH(DB2+IRLM)        0.000097      0.58    0.000167
 IRLM LOCK+LATCH            0.000004      0.21    0.000021
 DB2 LATCH                  0.000092      0.37    0.000251
..
PAGE LATCH                  0.000595      1.89    0.000314
NOTIFY MSGS                 0.000000      0.00         N/C
GLOBAL CONTENTION           0.004844      9.26    0.000523
```

```
GLOBAL       CONTENTION       L-LOCKS  AVERAGE TIME  AV.EVENT    GLOBAL       CONTENTION       P-LOCKS  AVERAGE TIME  AV.EVENT
-------------------------------------  ------------  --------    -------------------------------------  ------------  --------
L-LOCKS                                    0.000011      0.05    P-LOCKS                                    0.004833      9.21
 PARENT (DB,TS,TAB,PART)                   0.000000      0.00     PAGESET/PARTITION                         0.000000      0.00
 CHILD  (PAGE,ROW)                         0.000000      0.00     PAGE                                      0.004790      9.16
 OTHER                                     0.000011      0.05     OTHER                                     0.000043      0.05
```

```
LOCKING                AVERAGE     TOTAL
--------------------- --------  --------
...
LOCK REQUEST            139.05      2642
UNLOCK REQUEST           34.63       658
QUERY REQUEST            56.26      1069
CHANGE REQUEST           13.32       253
OTHER REQUEST             0.00         0
TOTAL SUSPENSIONS         0.32         6
 LOCK SUSPENSIONS         0.00         0
 IRLM LATCH SUSPENS.      0.32         6
 OTHER SUSPENS.           0.00         0
```

```
DATA SHARING          AVERAGE     TOTAL
-------------------  --------  --------
GLOBAL CONT RATE(%)      3.88       N/A
FALSE CONT RATE(%)       0.14       N/A
...
LOCK REQ    - XES      128.00      2432
UNLOCK REQ - XES        78.21      1486
CHANGE REQ - XES         5.63       107
SUSPENDS    - IRLM       8.26       157
SUSPENDS    - XES        0.00         0
CONVERSIONS- XES         0.68        13
FALSE CONTENTIONS        0.32         6
```

```
GROUP TOT4K               AVERAGE     TOTAL
---------------------    --------  --------
...
PG P-LOCK LOCK REQ          65.26      1240
 SPACE MAP PAGES             6.95       132
 DATA PAGES                 16.11       306
 INDEX LEAF PAGES           42.21       802
PG P-LOCK UNLOCK REQ        58.26      1107
PG P-LOCK LOCK SUSP          8.84       168
 SPACE MAP PAGES             0.95        18
 DATA PAGES                  1.84        35
 INDEX LEAF PAGES            6.05       115
```

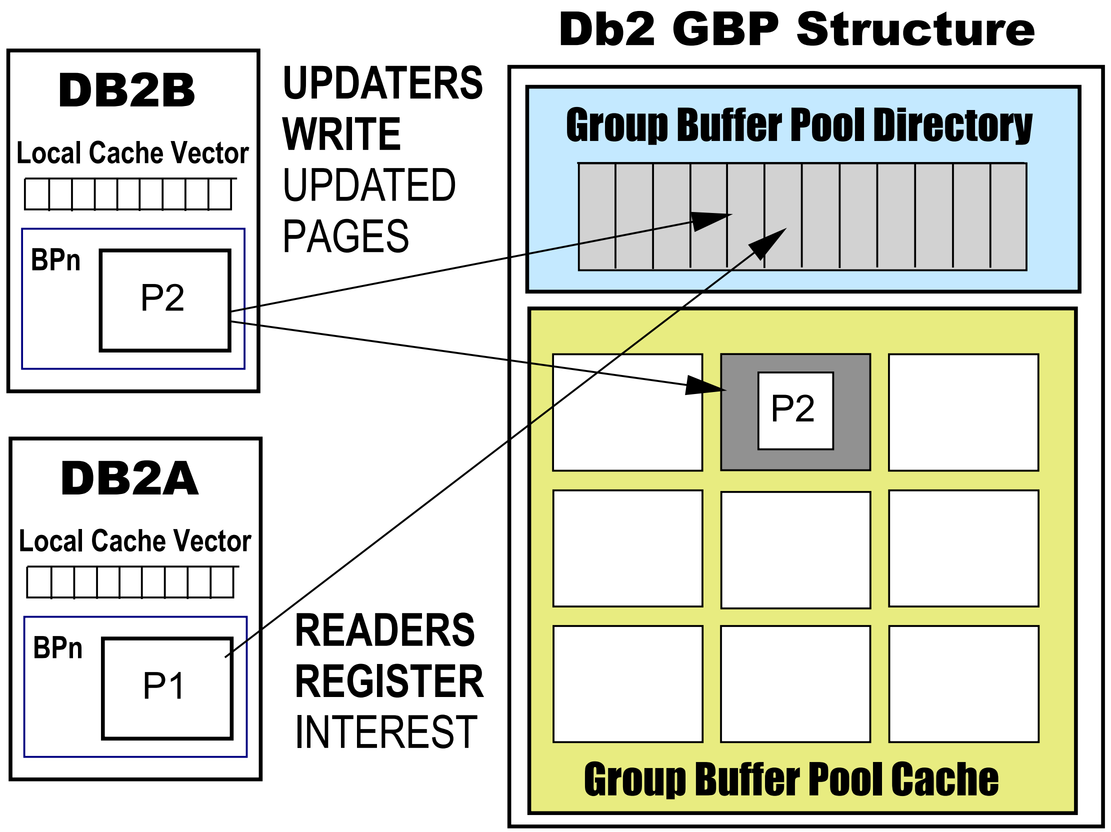
## Global contention …

- **Lock suspension report to identify ‘hot spots’**
  - For detailed analysis, start the following Db2 Performance traces for short periods of time during peak processing
    - TRACE(P) CLASS(30) IFCID (44,45,105,107,213,214,215,216,226,227)
  - Sample Db2 OMPM/PE report to generate a CSV file that can be easily loaded into a spreadsheet

```
LOCKING
         REPORT
         LEVEL(SUSPENSION)
         DDNAME(LORPTDD1)
         SPREADSHEETDD(SPSHDD)
         ORDER(DATABASE-PAGESET)
```

## Db2 group buffer pool structures

- **What are they used for?**
  - Register buffers for cross-invalidation (XI)
    - 'List' option provided for prefetch
  - Write changed buffers, send XI signals
  - H/W instruction to test vector bits for buffer XI
  - Fast refresh of XI'ed buffers
    - Store-in cache by default
      - 'No-data' option provided
    - Force-at-commit database write protocol used for writes to CF



## Cross invalidations

- **Two reasons for cross invalidations**
  - Perfectly normal condition in an active-active data sharing environment
  - Directory entry reclaims – condition you want to tune away from
    - CPU overhead and I/O overhead if there is not enough directory entries
    - Extra CF access and Sync I/O Read
- **-DISPLAY GROUPBUFFERPOOL(*) TYPE(GCONN)**

```
DSNB787I -    RECLAIMS
                FOR DIRECTORY ENTRIES                  = 0
                FOR DATA ENTRIES                       = 0
              CASTOUTS                                 = 0

DSNB788I -    CROSS INVALIDATIONS
                DUE TO DIRECTORY RECLAIMS              = 0
                DUE TO WRITES                          = 0
                EXPLICIT                               = 0
```

**ROT 1/3**

*Reclaims for Directory Entries should be minimised*

*Cross Invalidations due to Directory Reclaims should be zero*

## GBP reads

```
GROUP BP14                          QUANTITY  /SECOND  /THREAD  /COMMIT
----------------------------------  --------  -------  -------  -------
...
SYN.READ(XI)-DATA RETURNED           1932.00     0.09     0.01     0.00
SYN.READ(XI)-NO DATA RETURN         39281.6K  1823.66   236.31    79.22
SYN.READ(NF)-DATA RETURNED          22837.00     1.06     0.14     0.05
SYN.READ(NF)-NO DATA RETURN          6955.8K   322.93    41.85    14.03
```

**ROT 2/3**

*Sync.Read(XI) miss ratio should be < 10%*

TOTAL SYN.READ(XI) (A) = SYN.READ(XI)-DATA RETURNED
+ SYN.READ(XI)-NO DATA RETURN

Sync.Read(XI) miss ratio = SYN.READ(XI)-NO DATA RETURN / (A)

- **Local BP search → GBP search → DASD I/O**
- **SYN.READ(NF) = Local Buffer Pool miss**
- **SYN.READ(XI) = Local Buffer Pool hit but cross-invalidated buffer**
  - Most data should be found in GBP → if not, GBP may be too small or pages have been removed because of directory entry reclaims

## GBP writes

- **Changed Pages Sync Written to GBP / force write**
  - Commit
  - P-lock negotiation
- **Changed Pages Async Written to GBP**
  - Deferred write
  - System checkpoint
- **Pages in Write-Around**
  - Applies only to Pages Async Written to GBP
  - Db2 conditionally enables & disables the GBP write-around protocol
    - Turned on at GBP level threshold 50%, GBP Class threshold 20%
    - Turned off at GBP level threshold 40%, GBP Class threshold 10%
    - Pages are written directly to DASD instead of to the GBP
    - Cross invalidation signals sent to local BPs after DASD write I/O is complete

```
GROUP BP14    CONTINUED     QUANTITY  /SECOND  /THREAD  /COMMIT
-----------------------     --------  -------  -------  -------
WRITE AND REGISTER          54896.00     2.55     0.33     0.11
WRITE AND REGISTER MULT       255.5K    11.86     1.54     0.52
CHANGED PGS SYNC.WRTN         408.3K    18.96     2.46     0.82
CHANGED PGS ASYNC.WRTN       1713.4K    79.55    10.31     3.46
...
PAGES IN WRITE-AROUND           0.00     0.00     0.00     0.00
```

## GBP castout

- **GBP castout thresholds are similar to local BP deferred write thresholds**
  - Encourage Class_castout (CLASST) threshold by lowering its value
    - More efficient than GBP_castout threshold (notify to pageset/partition castout owner)
      - CLASST threshold check by GBP write
      - GBPOOLT threshold check by GBP castout timer (10sec default)
  - Default thresholds lowered in V8
  - V11: Class castout threshold below 1%

```
GROUP BP14                        QUANTITY  /SECOND  /THREAD  /COMMIT
-----------------------------     --------  -------  -------  -------
PAGES CASTOUT                      2224.8K   103.28    13.38     4.49
UNLOCK CASTOUT                    58868.00     2.73     0.35     0.12
...
CASTOUT CLASS THRESHOLD           26835.00     1.25     0.16     0.05
GROUP BP CASTOUT THRESHOLD          594.00     0.03     0.00     0.00
GBP CHECKPOINTS TRIGGERED            45.00     0.00     0.00     0.00
```

| | *V7* | *V8/V9/V10/V11* |
|---|---|---|
| VDWQT (dataset level) | 10% | 5% |
| DWQT (buffer pool level) | 50% | 30% |
| CLASST (Class_castout) | 10% | 5% |
| GBPOOLT (GBP_castout) | 50% | 30% |
| GBPCHKPT (GBP checkpoint) | 8 | 4 |

## GBP castout …

- **As transaction and data volumes grow, the GBP can become stressed**
  - Pages written to GBP faster than castout engines can process
  - Group buffer pool congested with changed pages
  - Can cause group buffer pool full condition in extreme cases

```
 21.02.15 S0012052 *DSNB325A  -DP1A DSNB1CNE THERE IS A CRITICAL SHORTAGE OF SPACE IN GROUP BUFFER POOL GBP11
...
 21.07.37 S0012052  DSNB327I  -DP1A DSNB1CNE GROUP BUFFER POOL GBP11 HAS ADEQUATE FREE SPACE
```

- **Problems are often precipitated by update-intensive batch jobs or utilities run against GBP dependent objects**
  - Intense, sustained GBP page write activity can lead to a shortage of GBP memory
  - Automatic GBP ALTER via XES Autoalter can respond and increase GBP size up to SIZE
    - Provided there is sufficient headroom …

## GBP castout …

- **GBP shortages may impact application performance and ultimately become an availability exposure**
  - When the GBP fills up, Db2 starts pacing for the commit - slower response time

```
DSNB750I  -DP11 DISPLAY FOR GROUP BUFFER POOL GBP11 FOLLOWS
...
DSNB786I  -DP11   WRITES
                 CHANGED PAGES                            = 688259863
                 CLEAN PAGES                              = 0
                 FAILED DUE TO LACK OF STORAGE            = 3630
               CHANGED PAGES SNAPSHOT VALUE               = 44474
```

```
GROUP BP14                           QUANTITY  /SECOND  /THREAD  /COMMIT
-----------------------------        --------  -------  -------  -------
CASTOUT CLASS THRESHOLD              26835.00     1.25     0.16     0.05
GROUP BP CASTOUT THRESHOLD             594.00     0.03     0.00     0.00
GBP CHECKPOINTS TRIGGERED               45.00     0.00     0.00     0.00
WRITE FAILED-NO STORAGE                  0.00     0.00     0.00     0.00
```

  - After repetitive write failures page will be put on the LPL
    - If the failures are against an index, the entire index might be put on the LPL
  - Recovery actions are then necessary

**ROT 3/3**

*WRITE FAILED-NO STORAGE < 1% of TOTAL CHANGED PGS WRTN*

Reduce castout thresholds, and/or
Reduce GBP checkpoint timer and/or
Increase GBP size

## Group buffer pool tuning

- **-DIS GBPOOL(*) GDETAIL(*) TYPE(GCONN)**

```
 DSNB750I  -PR4B DISPLAY FOR GROUP BUFFER POOL GBP2 FOLLOWS
...
 DSNB757I  -PR4B MVS CFRM POLICY STATUS FOR DSNPR0B_GBP2     = NORMAL
                 MAX SIZE INDICATED IN POLICY             = 614400 KB
...
 DSNB758I  -PR4B     ALLOCATED SIZE                           = 614400KB
...
 DSNB759I  -PR4B     NUMBER OF DIRECTORY ENTRIES              = 384147
                   NUMBER OF DATA PAGES                   = 116010
...
DSNB786I  -PR4B   WRITES
                   CHANGED PAGES                          = 2882842576
                   CLEAN PAGES                            = 0
                   FAILED DUE TO LACK OF STORAGE          = 71
                 CHANGED PAGES SNAPSHOT VALUE             = 10642
 DSNB787I  -PR4B    RECLAIMS
                   FOR DIRECTORY ENTRIES                  = 2495178
                   FOR DATA ENTRIES                       = 3290663329
                 CASTOUTS                                 = 4018446743
 DSNB788I  -PR4B    CROSS INVALIDATIONS
                   DUE TO DIRECTORY RECLAIMS              = 586960
                   DUE TO WRITES                          = 877165837
                   EXPLICIT                               = 2
```

Note:
Make sure to collect Statistics Class 5 (IFCID 230)
→ Additional GBP stats including Reclaims and XIs

OMPE stores the GBP stats in two different tables:
DB2PM_STAT_GBUFFER
DB2PMSYSPAR_230

## Group buffer pool tuning …

- **GBP size is defined in CFRM policy**

```
STRUCTURE NAME(DSNDB2_GBP1)
          SIZE(1024M)
          INITSIZE(512M)
```

- **Additionally, you can specify a RATIO using the ALTER GROUPBUFFERPOOL command to indicate how many Directory Entries (ENTRIES) per Data Pages (ELEMENTS)**

APAR PH13045 introduces 2 changes for Db2 12 users:
Default value of RATIO: 5 → 10
Limit of RATIO on the ALTER GROUPBUFFERPOOL command: 255 → 1024

**NEW**

## Group buffer pool tuning …

- **Example based on DISPLAY GBPOOL output**

| GBPOOL | SIZE (MB) | ALLOC_SIZE (MB) | DIR_ENTRIES | DATA_PAGES | RATIO [(1)] | SUM VPSIZEs + DATA_PAGES |
|---|---|---|---|---|---|---|
| GBP0 | 150 | 90 | 74285 | 14857 | 5 | 34857 |
| GBP1 | 400 | 300 | 318108 | 45444 | 7 | 225444 |
| GBP2 | 600 | 550 | 225969 | 116010 | 1.9 | 356010 |
| GBP8K0 | 990 | 990 | 403011 | 106257 | 3.8 | 456257 |
| GBP8K1 | 600 | 500 | 554987 | 36998 | 15 | 436998 |
| GBP16K0 | 120 | 60 | 15792 | 3156 | 5 | 5156 |
| GBP32K | 220 | 120 | 48499 | 3030 | 16 | 73030 |

| GBPOOL | % COVERAGE | FAIL_LACK_OF_STG | DIR_RECLAIMS | XI_DIR_RECLAIMS |
|---|---|---|---|---|
| GBP0 | 213% | 0 | 0 | 0 |
| GBP1 | 141% | 0 | 0 | 0 |
| GBP2 | 63% | 71 | 2495178 | 586960 |
| GBP8K0 | 88% | 4231 | 35733124 | 12267527 |
| GBP8K1 | 127% | 0 | 0 | 0 |
| GBP16K0 | 306% | 0 | 0 | 0 |
| GBP32K | 66% | 0 | 0 | 0 |

## Group buffer pool tuning …

- **Targeted tuning #1: GBP with large number of directory reclaims and XI due to directory reclaims (but no or minimal write failures) → GBP2**
  - Tuning:
    - Keep number of data pages the same to avoid aggravating write failures
    - Increase SIZE and RATIO to cover the max number of directory entries that could ever be required (1 for each local buffer + 1 for each GBP data page)
      - Note: RATIO can be a decimal value with 1 digit after the decimal point (e.g. 5.2)
  - Changes required:

NEW DIR ENTRIES = SUM VPSIZE across all Db2 members + GBP DATA PAGES

NEW RATIO = NEW DIR ENTRIES / GBP DATA PAGES

NEW INITSIZE = (GBP DATA PAGES * PAGE SIZE (KB) + NEW DIR ENTRIES * 410 bytes / 1024) / 1024

NEW SIZE = 1.3 to 2x NEW INITSIZE

↑

The size of a Directory Entry can vary by CF level
For a rough estimate, use 410-430 bytes per entry on CF Level 17

## Group buffer pool tuning …

- **Targeted tuning #2: GBP with large number of write failures (with or without large number of directory reclaims and XI due to directory reclaims) → GBP8K0**
  - Tuning:
    - Adjust CLASST and GBPOOLT to trigger more frequent castout
    - Increase the number of data pages to reduce critical space shortages and write failures
    - Adjust size and ratio to still cover the max number of directory entries that could ever be required (1 for each local buffer + 1 for each GBP data page)
      - Note: RATIO can be a decimal value with 1 digit after the decimal point (e.g. 5.2)
  - Changes required:

NEW GBP DATA PAGES = 1.3 to 2x GBP DATA PAGES

NEW DIR ENTRIES = SUM VPSIZE across all Db2 members + NEW GBP DATA PAGES

NEW RATIO = NEW DIR ENTRIES / NEW GBP DATA PAGES

NEW INITSIZE = (NEW GBP DATA PAGES * PAGE SIZE (KB) + NEW DIR ENTRIES * 410 bytes / 1024) / 1024

NEW SIZE = 1.3 to 2x NEW INITSIZE

## Group buffer pool tuning …

- **Simplified tuning when using XES AUTO ALTER**
  - Very useful autonomic functionality to simplify GBP tuning
    - Tries to avoid Structure Full conditions
    - Tries to avoid Directory Reclaim conditions
  - Recommendations:
    - Set CFRM policy properties
      - ALLOWAUTOALT(YES)
      - FULLTHRESHOLD = 80-90%
      - SIZE = 1.3-2x INITSIZE
      - MINSIZE = INITSIZE
    - Periodically review GBP actual allocations
    - If SIZE is reached, it limits the effectiveness of XES AUTO ALTER → plan to update the CFRM policy and schedule a REBUILD to increase the structure

## Group buffer pool tuning …

- **Simplified tuning when using XES AUTO ALTER …**

  - Changes when allocated size reaches SIZE - but no or minimal write failures

    NEW DIR ENTRIES = SUM VPSIZE across all Db2 members + GBP DATA PAGES

    NEW RATIO = NEW DIR ENTRIES / GBP DATA PAGES

    NEW INITSIZE = (GBP DATA PAGES * PAGE SIZE (KB) + NEW DIR ENTRIES * 410 bytes / 1024) / 1024

    NEW SIZE = 1.3 to 2x NEW INITSIZE

  - Changes when allocated size reaches SIZE with large number of write failures

    NEW GBP DATA PAGES = 1.3 to 2x GBP DATA PAGES

    NEW DIR ENTRIES = SUM VPSIZE across all Db2 members + NEW GBP DATA PAGES

    NEW RATIO = NEW DIR ENTRIES / NEW GBP DATA PAGES

    NEW INITSIZE = (NEW GBP DATA PAGES * PAGE SIZE (KB) + NEW DIR ENTRIES * 410 bytes / 1024) / 1024

    NEW SIZE = 1.3 to 2x NEW INITSIZE

A more sophisticated approach would be to look at the peak rate of Changed Pages written to the GBP and calculate the average residency time → tuning target 30-60 seconds

## Group buffer pool tuning …

- **Increase of local buffer pool size on a 'healthy GBP'**

  - Tuning:
    - Keep number of data pages the same to avoid aggravating write failures
    - Increase size and ratio to cover the max number of directory entries that could ever be required (1 for each local buffer + 1 for each GBP data page)

  - Changes required:

NEW DIR ENTRIES = SUM VPSIZE across all Db2 members + GBP DATA PAGES

NEW RATIO = NEW DIR ENTRIES / GBP DATA PAGES

NEW INITSIZE = (GBP DATA PAGES * PAGE SIZE (KB) + NEW DIR ENTRIES * 410 bytes / 1024) / 1024

NEW SIZE = 1.3 to 2x NEW INITSIZE

## Questions?