# IDUG Db2 Tech Conference, Rotterdam, Netherlands 2019

## JC Greatest Hits, War Stories, and Best Practices 2019 Part 1

**John Campbell**

*IBM Db2 for z/OS Development*

Session code:     A15

Thursday 24th October, 2019 08:30-09:30

Platform: Db2 for z/OS

**IDUG**

Leading the Db2 User
Community since 1988

# Objectives

- **Learn recommended best practice**
- **Learn from the positive and negative experiences from other installations**

# Agenda

- **Large size real memory page frames**

- **Combination of PGSTEAL(NONE) buffer pool with FRAMESIZE(2G)**

- **Pointer-Overflow Pairs (indirect references)**

- **Running out of basic 6-byte log RBA addressing range**

- **Db2 Connect and Continuous Delivery (Db2 12)**

- **Diagnosing and resolving slow-downs and hangs**

- **Hung Db2 threads**

# Large size real memory page frames

- **Benefit of large size (1M or 2G) real memory page frames**
  - The Translation Lookaside Buffer (TLB) is a cache used to speed up the conversion of virtual addresses into real addresses
  - With the introduction of 64-bit real and virtual addressing, the TLB coverage has dramatically shrunk, leading to performance degradation
  - Large size page frames help increase TLB coverage without having to enlarge the TLB size
  - Result: Better CPU performance by decreasing the number of TLB misses

- **Common problem**
  - LFAREA is grossly over-configured, which might result in a shortage of 4K size frames and lead to expensive breakdown of 1M size large frames, expensive page movement for 4K page fixes, premature paging, CPU burn, loss of LPAR
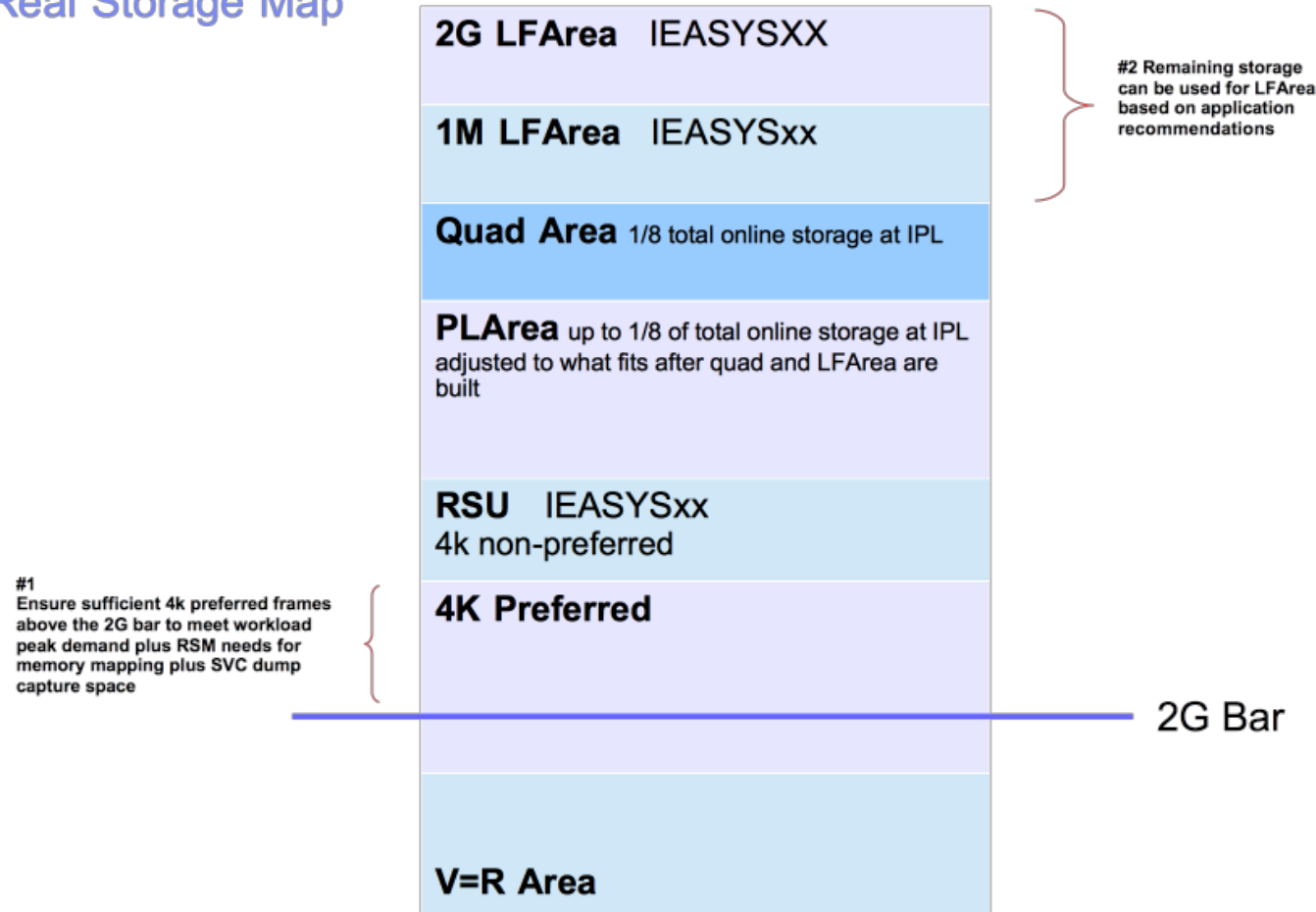
# Large size real memory page frames ...

- **LFAREA – 1M/2G large frame area**
  - Fixed 1M/2G page frames
  - Defined in IEASYSxx parmlib member
  - 'Old' syntax (still supported)
    - LFAREA = (xM | xG | xT | x%)
    - Pct formula: (x% of online memory available at IPL) – 2G
    - Max LFAREA is (80% * online real memory available at IPL) – 2G
  - New syntax
    - LFAREA = (1M=(a [,b]) | 1M=(a% [,b%]) | 2G=(a [,b]) | 2G=(a% [,b%])
    - Pct formula: x% of (online memory available at IPL – 4G)
    - Max LFAREA is 80% of (online memory available at IPL time – 4G)
  - Only changeable by IPL
  - If the LFAREA is overcommitted, Db2 will use 4K and/or 1M size page frames

# Large size real memory page frames …

Real Storage Map

| Real Storage Map |
|---|
| **2G LFArea** IEASYSXX |
| **1M LFArea** IEASYSxx |
| **Quad Area** 1/8 total online storage at IPL |
| **PLArea** up to 1/8 of total online storage at IPL adjusted to what fits after quad and LFArea are built |
| **RSU** IEASYSxx 4k non-preferred |
| **4K Preferred** |
| |
| **V=R Area** |

#2 Remaining storage can be used for LFArea based on application recommendations

#1 Ensure sufficient 4k preferred frames above the 2G bar to meet workload peak demand plus RSM needs for memory mapping plus SVC dump capture space

2G Bar

# Large size real memory page frames …

- **Quad area**
  - 12.5% of online memory at IPL time

- **PLArea – Pageable 1M large frame area**
  - Pageable 1M page frames
  - Allocated on SCM-capable machines (zEC12/zBC12 and above)
    - If Flash Express/Virtual Flash Memory is installed, these large pages may be paged to and from SCM
    - If Flash Express/Virtual Flash Memory is not installed, then if those pages are ever paged out, they will be demoted to 4K size page frames and will remain 4K size until the next IPL
  - System-defined size
    - Approximately 12.5% of online memory at IPL time – adjusted to what fits after Quad and LFArea are built
  - Pageable 1M frames overflow into the LFAREA when PLArea is depleted

# Large size real memory page frames ...

- **Let's do some maths...**
  - Starting position

| | |
|---|---|
| Online memory (GB) | 50.0 |
| LFAREA (GB) | 0 |
| QUAD (GB) | 6.3 |
| 1MB PAGEABLE (GB) | 6.3 |
| 4KB FRAMES (GB) | 37.4 |

  - If you were to add 100GB to the LPAR and define it all as LFAREA

| | |
|---|---|
| Online memory (GB) | 150.0 |
| LFAREA (GB) | 100.0 |
| QUAD (GB) | 18.8 |
| 1MB PAGEABLE (GB) | 18.8 |
| 4KB FRAMES (GB) | 12.4 |

Probably not enough 4K frames to handle the 4K workload needs, including taking dumps quickly, without having to break down free 1M frames

Do not forget that Quad area and Pageable 1M area grow proportionally with additional REAL memory!

# Large size real memory page frames …

- **Db2 can exploit fixed 1M and 2G large size page frames for page-fixed buffer pools**
  - Requirements
    - Buffer pools must be defined as PGFIX=YES
    - 1M size page frames requires Db2 10, z10 and above
    - 2G size page frames requires Db2 11, zEC12 and above
  - Db2 11 supports FRAMESIZE parameter (4K, 1M, 2G) at the BP level for flexibility
- **Db2 11 can also exploit fixed 1M page frames for the log output buffer (OUTBUFF)**
- **Db2 can exploit pageable 1M frames for buffer pools control blocks (PMBs)**
  - PMBs consume only 4-5% of buffer pool allocations but they are referenced very heavily
  - This is where customers get the most benefit
  - Requirements
    - Db2 10 with APAR PM85944 or Db2 11
    - Buffer pools can be defined as either PGFIX=YES or PGFIX=NO
    - Buffer pools can be backed by either 4K, 1M or 2G size page frames

# Large size real memory page frames …

- **Estimating 'optimal' LFAREA**
  - Total of
    - (Sum of VPSIZE*page size from candidate local buffer pools) * 1.05
    - Plus 20MB for z/OS usage
    - Plus log output buffer size (OUTBUFF) if running Db2 11
    - Plus non-Db2 usage e.g., Java heap sizes
    - Plus any over flow from PLArea (Pageable Large Area)
- **Recommendations**
  - Define LFAREA based on what you can actually afford after considering your total real memory demands for 4K frames above the 2G bar
    - Must consider operating system memory needs
      - RSM requirement for memory mapping (approximately 1/64 total online real memory at IPL)
    - System address spaces memory usage (Db2, CICS, etc.)
    - Must also include enough spare 4K size frames for taking dumps quickly
  - Specify LFAREA as an absolute number value as opposed to a percentage value

# What happens in case of shortage of 4K frames?

- **2 types of 4K requests**
  - Preferred requests
    - Used by non-swapable address spaces, can be fixed long term
    - Db2 uses preferred requests
  - Non-Preferred requests
    - Used by swapable address spaces, used short term and not fixed
- **Pageable 1M large frame area**
  - Pageable 1M frames can be broken down to satisfy demand for 4K preferred or non-preferred requests
- **Quad area**
  - Quad frames can be broken down to satisfy demand for 4K <u>non-preferred requests only</u>
  - z/OS V2.2 only:
    - Several customers have reported excessive paging after migration to z/OS V2.2
    - Analysis showed an abundance of available Quad frames that are not used by RSM
    - Fixed by APAR OA50945 (HIPER)

# What happens in case of shortage of 4K frames? …

- **LFAREA and INCLUDE1MAFC option**
  - Available frame count (AFC) is used to determine when storage management should begin paging frames
  - Pre-z/OS V2R2:
    - By default, AFC does not include LFAREA 1M pages
      - 1M fixed frames are preserved for fixed 1M requests
      - In case of shortage of 4K frames, paging can occur even with an abundance of available 1M fixed frames
    - INCLUDE1MAFC on LFAREA parameter means AFC includes the LFAREA 1M pages
      - New keyword added by APAR OA41968
      - 1M size fixed frames can be broken down to satisfy demand for 4K <u>non-preferred requests only</u>
      - Paging is delayed
  - z/OS V2R2 → Two options when defining the LFAREA: INCLUDE1MAFC(YES|NO)
    - YES is the default
  - Recommendation is to use INCLUDE1MAFC(YES) so that unused 1M pages can be broken down to satisfy demand for 4K non-preferred requests

# Large size real memory page frames – Monitoring

- **Useful commands**
  - Find out how many real memory page frames of each size are being used
    - Especially useful when running multiple Db2 subsystems on the same LPAR

  - Db2 10 -DISPLAY BUFFERPOOL(BPx) SERVICE(4)
    - See DSNB999I message

```
DSNB999I +D2V1 DSNB1DBP SERVICE( 4 ) OUTPUT
DSNB999I +D2V1 4K PAGES 0
DSNB999I +D2V1 1M PAGES 1476
```

  - Db2 11 -DISPLAY BUFFERPOOL(BPx) DETAIL(*)
    - See DSNB546I messages

```
DSNB546I  +PDJ1 PREFERRED FRAME SIZE 1M
          793600 BUFFERS USING 1M FRAME SIZE ALLOCATED
DSNB546I  +PDJ1 PREFERRED FRAME SIZE 1M
          2956400 BUFFERS USING 4K FRAME SIZE ALLOCATED
```

# Large size real memory page frames – Monitoring …

- **Useful commands …**
  - MVS DISPLAY VIRTSTOR,LFAREA
    - Show total LFAREA, allocation split across different size page frames, what is available
    - See IAR019I message

```
IAR019I  10.48.56 DISPLAY VIRTSTOR
 SOURCE =  C2
 TOTAL LFAREA = 14336M , 0G
 LFAREA AVAILABLE = 0M , 0G
 LFAREA ALLOCATED (1M) = 14329M
 LFAREA ALLOCATED (4K) = 0M
 MAX LFAREA ALLOCATED (1M) = 14329M
 MAX LFAREA ALLOCATED (4K) = 0M
 LFAREA ALLOCATED (PAGEABLE1M) = 7M
 MAX LFAREA ALLOCATED (PAGEABLE1M) = 7M
 LFAREA ALLOCATED NUMBER OF 2G PAGES = 0
 MAX LFAREA ALLOCATED NUMBER OF 2G PAGES = 0
```

```
IAR019I  10.45.57 DISPLAY VIRTSTOR
 SOURCE =  00
 TOTAL LFAREA = 7782M , 0G
 LFAREA AVAILABLE = 2025M , 0G
 LFAREA ALLOCATED (1M) = 5377M
 LFAREA ALLOCATED (4K) = 374M
 MAX LFAREA ALLOCATED (1M) = 5377M
 MAX LFAREA ALLOCATED (4K) = 526M
 LFAREA ALLOCATED (PAGEABLE1M) = 6M
 MAX LFAREA ALLOCATED (PAGEABLE1M) = 2031M
 LFAREA ALLOCATED NUMBER OF 2G PAGES = 0
 MAX LFAREA ALLOCATED NUMBER OF 2G PAGES = 0
```

*Undersized LFAREA resulting in (small) lost CPU savings*

*Oversized LFAREA resulting in a shortage of 4K size frames and expensive breakdown of 1M size large frames – may also lead to premature paging if INCLUDE1MAFC=NO*

# Large size real memory page frames – Major changes in z/OS V2.3

- **1MB LFAREA - Large Frame Area**
  - No longer physical range
  - Managed dynamically in non-reconfigurable memory above 2G bar
  - Capped by LFAREA (1M= ) (IEASYSxx)
  - INCLUDE1MAFC=NO is ignored
- **PLAREA – Pageable Large Area**
  - No longer physical range
  - Managed dynamically in non-reconfigurable memory above 2G bar
  - No cap
- **2GB LFAREA Large Frame Area – Unchanged in z/OS V2.3**

# Large size real memory page frames – Major changes in z/OS V2.3 …

- **New display in z/OS V2.3 with large size page statistics**

Pageable 1MB stats including number of 1M still available. If often low, consider adding memory.

LFAREA 1M stats including total size (cap)

LFAREA 2G stats

```
F AXR,IAXDMEM
IAR049I DISPLAY MEMORY V1.0
PAGEABLE 1M STATISTICS
  4824.0MB : TOTAL SIZE
  4585.0MB : AVAILABLE FOR PAGEABLE 1M PAGES
     3.0MB : IN-USE FOR PAGEABLE 1M PAGES
     3.0MB : MAX IN-USE FOR PAGEABLE 1M PAGES
     1.0MB : FIXED PAGEABLE 1M FRAMES
LFAREA 1M STATISTICS – SOURCE = IEASYS23
    64.0MB : TOTAL SIZE
    62.0MB : AVAILABLE FOR FIXED 1M PAGES
     2.0MB : IN-USE FOR FIXED 1M PAGES
     2.0MB : MAX IN-USE FOR FIXED 1M PAGES
LFAREA 2G STATISTICS – SOURCE = IEASYS23
     0.0MB : TOTAL SIZE = 0
     0.0MB : AVAILABLE FOR 2G PAGES = 0
     0.0MB : IN-USE FOR 2G PAGES = 0
     0.0MB : MAX IN-USE FOR 2G PAGES = 0
```

# Combination of PGSTEAL(NONE) buffer pool with FRAMESIZE(2G)

- **Incompatible change when migrating from Db2 11 to Db2 12**
- **For Db2 11, Db2 can use 2G size frames for PGSTEAL(NONE) buffer pools**
- **For Db2 12, PGSTEAL(NONE) buffer pool will use contiguous buffer pool feature and will NOT use 2G size frames**
  - Request to use 2G size frames is not honoured
  - Buffer pool will still be allocated, but in 4K size frames
  - DSNB548I message will be issued when
    - Allocating buffer pool which has PGSTEAL(NONE) and FRAMESIZE(2G) specified
    - ALTER BUFFERPOOL command changes either attribute with the result being PGSTEAL(NONE) FRAMESIZE(2G)
- **Why be concerned?**
  - If the size of contiguous buffer pools are very large, this can lead to shortage of 4K frames on the LPAR, with consequences
    - Penalty of page movement or paging I/O overhead with corresponding CPU burn in RASP address space
    - Worst case the LPAR will crash out!
- **Recommendation**
  - If using combination of 2G size large frames with PGSTEAL(NONE) buffer pools under Db2 11, then switch to using 1M size frames before leaving Db2 11 NFM

# Pointer-Overflow Pairs (indirect references)

- **What is a pointer-overflow pair and how is it created?**
  - Row increases in size as a result of update
  - Row no longer fits in the space available in the data page
  - Row is then moved to a new data page location (overflow record)
  - Pointer to the overflow location placed in the original spot
- **Problem**
  - Unique to data sharing and GBP-dependent object
  - 'Timing window' where an ISO(UR) scanner running on a different Db2 member failed to get a row in an overflow record

# Pointer-Overflow Pairs (indirect references) …

- **Db2 solution introduced with APAR PM82279**
  - Synchronous write force the new data page location to CF GBP structure preceded by force sync log write
  - Performance penalty which could be significant
  - Aggravating factor
    - May not be enough free contiguous free space for the overflow record in the new data page location, so the data page is compacted
    - After compaction, the data page still does not have enough committed free space to insert the overflow record, the page is released and move on to find another data page location
    - Even though the overflow record was not inserted into this compacted page, another synchronous write force the new data page location to CF GBP structure occurs preceded by force sync log write
- **Recommendation: generously allocate PCTFREE FOR UPDATE to limit number of new rows inserted and overflow records inserted into a data page, allow for updated rows to expand, and reduce the number of pointer-overflow pairs**

# Running out of basic 6-byte log RBA addressing range

- **Background**
  - Increasingly common for the basic 6-byte log RBA (256 TB) addressing range to be exhausted for Db2 subsystems
  - Tiny number of installations close to exhausting the basic 6-byte LRSN addressing range for a data sharing group
  - BSDS must be converted to extended 10-byte RBA format before migrating to Db2 12

- **Problem areas**
  - After BSDS conversion to extended 10-byte log RBA, non-data sharing Db2 subsystems will accelerate with increased velocity towards end of basic 6-byte log RBA addressing range!
  - After converting the BSDS, Db2 stops generating the DSNJ032I warning messages, even if there is imminent danger of reaching the 6-byte RBA soft limit (non-data sharing) or 6-byte LRSN soft limit (data sharing) for table spaces or index spaces in 6-byte basic format
  - Many installations embarked on an aggressive but unnecessary "crash project" to reorganise Catalog/Directory and application database objects to convert to extended 10-byte of RBA or LRSN format

# Running out of basic 6-byte log RBA addressing range …
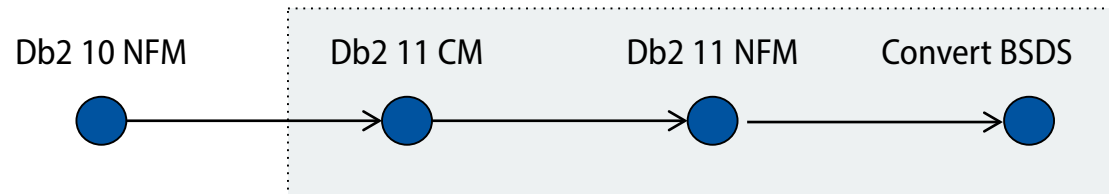
- **Recommendations**
  - Non-data sharing - getting to extended 10-byte log RBA format
    - Must convert the Catalog, Directory and all application objects via REORG to extended 10-byte log RBA format
    - Must convert BSDS of the problem Db2 subsystem to extended 10-byte log RBA format
    - Leave converting the BSDS to the very end of the overall conversion process, otherwise will accelerate towards end of basic 6-byte log RBA addressing range with increased velocity
  - Data sharing - getting to extended 10-byte log RBA format for a specific Db2 member
    - Just convert BSDS of the problem Db2 member
    - No need to convert Catalog, Directory and application objects via REORG to extended 10-byte LRSN format
  - Data sharing – getting to the extended 10-byte LRSN format
    - Must convert the Catalog, Directory and application objects via REORG to extended 10-byte extended LRSN format
    - Must convert BSDS of each Db2 member to extended 10-byte extended log RBA/LRSN format
    - Convert the BSDS at the start of the overall conversion process to get potential incremental performance benefit from "LRSN spin" avoidance as Catalog, Directory and application objects are converted to extended 10-byte LRSN format

# Reaching the limit conversion strategies

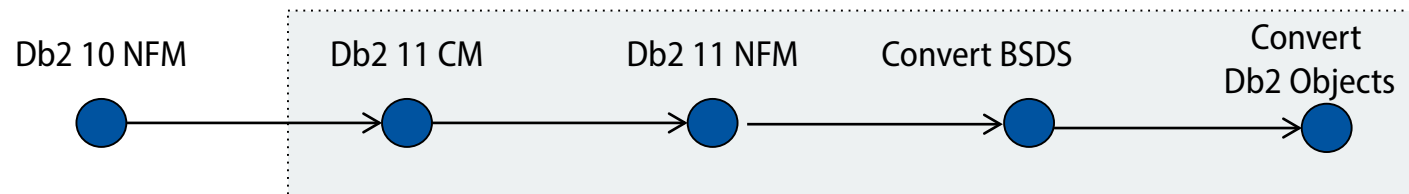### Non Data Sharing – End log RBA problem

Db2 10 NFM        Db2 11 CM        Db2 11 NFM        Convert Db2 Objects        Convert BSDS
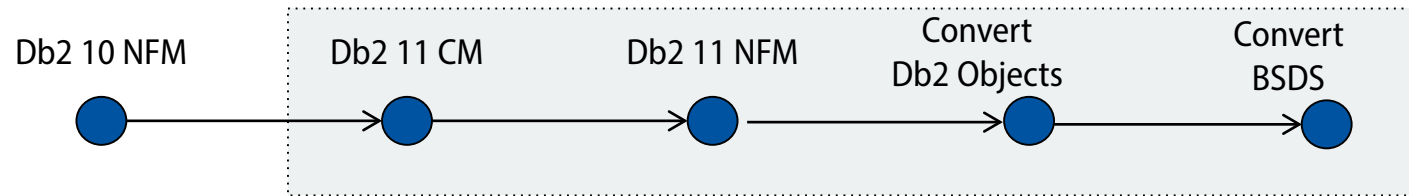
### Data Sharing – End log RBA problem only on one or more members

Db2 10 NFM        Db2 11 CM        Db2 11 NFM        Convert BSDS

### Data Sharing – End LRSN problem

Db2 10 NFM        Db2 11 CM        Db2 11 NFM        Convert BSDS        Convert Db2 Objects

# Db2 Connect and Continuous Delivery (Db2 12)

- **Db2 Connect – Situation until recently prior to applying PTF for APAR PH08482**
  - Any in-support level of Db2 Connect drivers should work with Db2 12 for z/OS, both before and after new function is activated (FL500) with no behavior change
  - Data server clients and drivers must be at the following levels to exploit Db2 for z/OS function-level application compatibility (APPLCOMPAT) of V12R1M501 or greater:
    - IBM® Data Server Driver for JDBC and SQLJ: Versions 3.72 and 4.22, or later
    - Other IBM data server clients and drivers: Db2 for Linux, UNIX, and Windows Version 11.1 M2 Fix Pack 2, or later
  - New ClientApplCompat (ODBC) and clientApplcompat (JDBC) property setting allows you to control the capability of the client when updated drivers ship changes to enable new server capability
    - You might want specific control of driver capability when:
      - Db2 client driver introduces new behavior currently not controlled by Db2 application compatibility
      - Change needs to be controlled at the application level to ensure compatibility with new behavior
  - ClientApplCompat/clientApplcompat setting of V12R1M500 is absolutely required to exploit Db2 12 for z/OS Server capability shipped after GA at function levels beyond Db2 12 for z/OS FL=V12R1M500
  - Db2 Connect Server gateway does NOT support ClientApplCompat/clientApplcompat

# Db2 Connect and Continuous Delivery (Db2 12) …

- **Db2 Connect – new behavior after applying PTF for APAR PH08482**
  - Makes setting of ClientApplCompat/clientApplcompat property optional
    - Customers no longer forced to have the setting
  - No changes are required in Db2 Connect level and configuration
  - All customers must upgrade to at least Db2 Connect V11.1 M1 FP1 or higher in order to run DRDA applications where packages have APPCOMPAT > FL500
  - Db2 Connect Server gateways will need to be upgraded to at least Db2 Connect V11.1 M1 FP1 to access packages using an APPLCOMPAT > FL500
    - Both the clients and the servers must be at V11.1 FP1 or higher
    - APAR PH15092 will allow down level clients to connect through a server at V11.1 FP1 or higher
  - When ClientApplCompat/clientApplcompat is set, Db2 will perform validation checking where there are changes in DRDA message flows i.e., check the underlying infrastructure and avoid application incompatibilities

# Db2 Connect and Continuous Delivery (Db2 12) …

- **Db2 Connect – new behavior after applying PTF for APAR PH08482 …**
  - Going out into the future when not setting ClientApplCompat/clientApplcompat there are consequences in terms of risk when DRDA flows change on existing applications
    - So far in Db2 12 there are no changes in DRDA flows, no changes are in plan, but at some point it will likely happen
    - Applications may break in the future when DRDA transactions run with packages where APPLCOMPAT > FL500
    - If an application breaks then Db2 Development will not provide server side support to allow these broken applications to run i.e., no more new DDF_COMPATABILITY zparm settings
  - So what are the your options
    1. Rebind driver packages in the NULLID collection and back level the APPLCOMPAT setting
       This is a "one size fits all" solution to fallback to an earlier APPLCOMPAT
    2. "Penalty Box" the problem applications
       - Switch the problem applications out to use the driver packages in a different collection which has a back levelled APPLCOMPAT setting, or
       - Switch all the good applications out into a new collection using driver packages with the new APPLCOMPAT setting and leave the problem applications still using the driver packages in the NULLID or different collection but with the driver packages running a back levelled APPLCOMPAT setting

# Db2 Connect and Continuous Delivery (Db2 12) …

- **General best practice recommendation**
    - When migrating to Db2 12 - all DRDA applications should continue to use the driver packages in the NULLID collection
    - These packages can have an APPLCOMPAT setting of V10R1, V11R1, V12R1M100 or V12R1M500 depending on where you are in the migration process
    - The APPLCOMPAT setting for the driver packages in the NULLID collection should not advance beyond V12R1M500
    - When specific applications and their application servers want to use new function requiring APPLCOMPAT setting > FL500, these application servers should switch away from using the driver packages in the NULLID collection to a new collection (e.g., V12R1M503) where the driver packages are bound with a higher APPLCOMPAT setting (e.g., driver packages bound with APPLCOMPAT V12R1M503 in collection V12R1M503)

# Diagnosing and resolving slow-downs and hangs

- **Very rare condition, with little or no customer operational experience and confidence to handle**
- **Techniques**
  - Issue Db2 and IRLM commands for each Db2 member and if a Db2 member/IRLM does not respond, then should take that Db2 member out even if it means using the z/OS CANCEL command
    - Sample commands to be issued
      - -DIS THD(*) SERVICE(WAIT) to each Db2 member
      - MODIFY xxxxIRLM,STATUS,ALLD to each IRLM
      - -DIS THD(*) SERVICE(WAIT) SCOPE(GROUP) to each Db2 member
  - Manually trigger rebuild of the LOCK1 structure into alternate CF based on the PREFERENCE LIST in CFRM policy
    - Issue SETXCF START,REBUILD,STRNAME=DSNxxxx_LOCK1,LOCATION=OTHER
    - Structure rebuild may clear the condition
    - However if the structure rebuild fails, find the connector which did not respond and IPL the non-responsive z/OS LPAR

# Hung Db2 threads

- **Things are hung, do not cancel … at least not yet**
- **Take a dump first, cancel changes the layout of the control blocks and may hide the problem**
- **Bring out the smallest size "hammer" first**
  - Cancel the thread in Db2
  - FORCEPURGE CICS TRAN or CANCEL BATCH JOB
  - Cancel Db2 via MODIFY irlmproc,ABEND,NODUMP
    - FORCE IRLM
    - FORCE MSTR
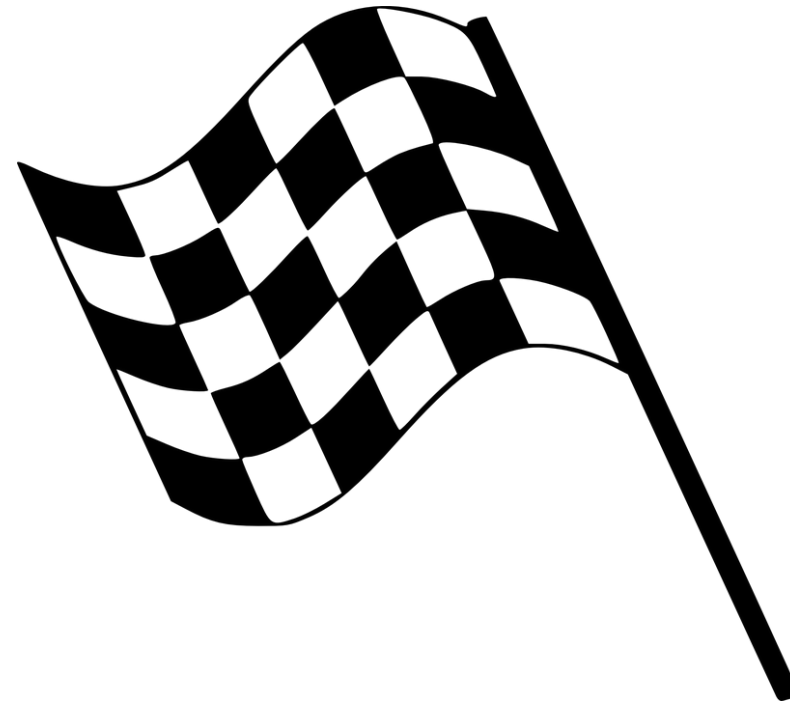    - FORCE DBM1
  - IPL z/OS LPAR

# Questions?

# John Campbell
IBM Db2 for z/OS Development
*campbelj@uk.ibm.com*

Session code:    A15

**IDUG**
Leading the Db2 User
Community since 1988