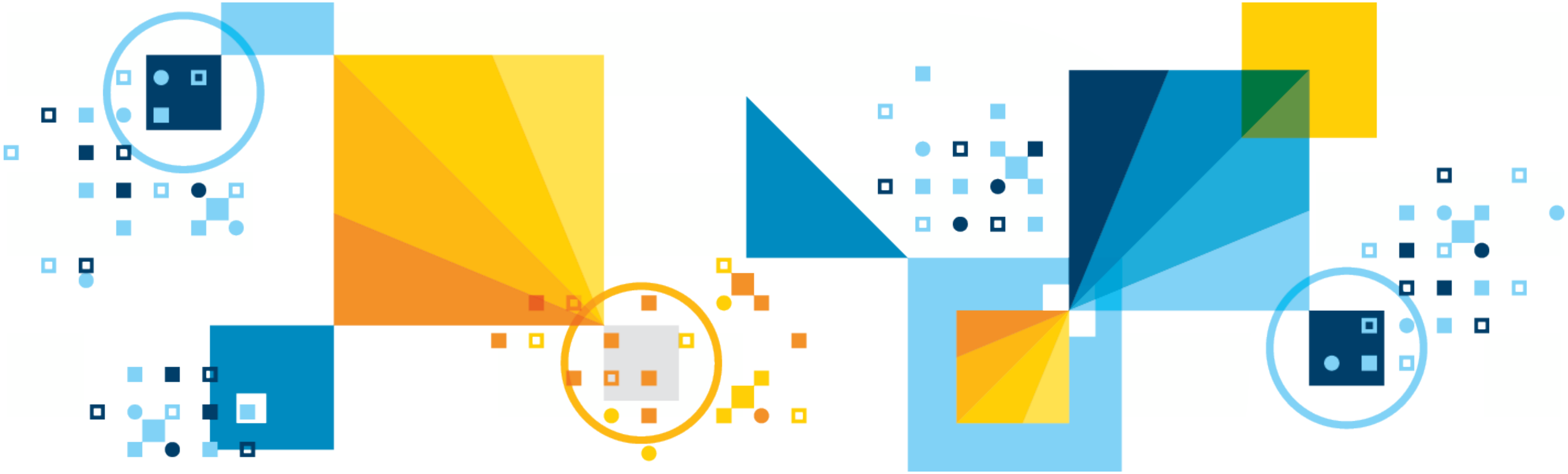
Db2 12 for z/OS Migration Planning and Early Experiences

John Campbell
Distinguished Engineer
Db2 for z/OS Development



Objectives

- Share lessons learned, surprises, pitfalls
- Provide hints and tips
- Address some myths
- Provide additional planning information
- Provide usage guidelines and positioning on new enhancements
- Help customers migrate as fast as possible, but safely

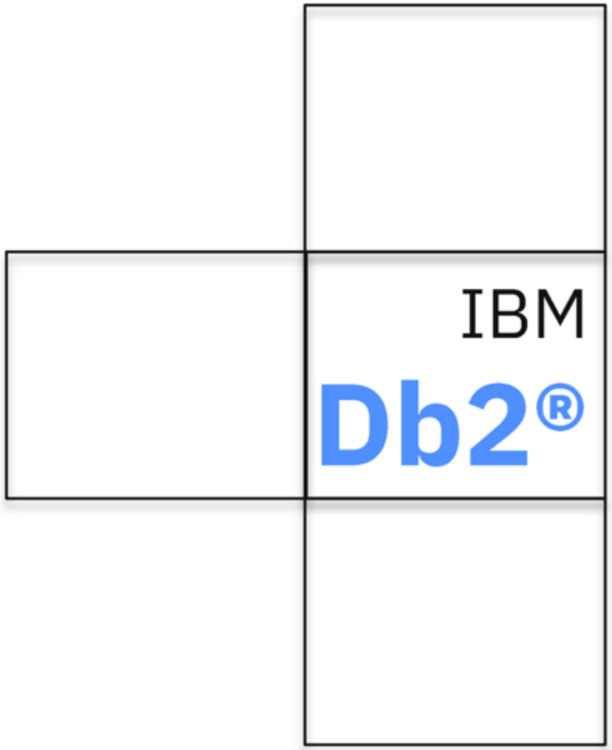
First … reintroducing the IBM Db2 Family

"Data is the world's next natural resource. The value of the data and extracting the information and insights will change how we make every decision."

Ginni Rometty

First … reintroducing the IBM Db2 Family …

- **Going forward, we are rebranding our data solutions to focus on the Db2 as the common denominator to represent our family of hybrid data management offerings**
  - Emphasize the common code base and common lineage across our solutions
  - Emphasize the application portability across our solutions from private to public and hybrid cloud deployments
  - Build on the value of our highly trusted and respected industry leading brand
- **Modern, elemental feel (think *periodic table*)**
  - Connotes the fundamental importance of hybrid data management
  - Places all emphasis on the capital “D” for data - more specifically our clients’ data



IBM

Db2®

First … reintroducing the IBM Db2 Family …

Db2

	Previous Name	New Name
Database	Db2 for Linux, Unix and Windows (Db2 LUW)	**Db2**
	Db2 for z/OS	**Db2** for z/OS
	Db2 on Cloud	**Db2** Hosted
	dashDB for Transactions	**Db2** on Cloud
	BLUSpark	Project EventStore*
Data Warehouse	dashDB/dashDB for Analytics	**Db2** Warehouse on Cloud
	dashDB local	**Db2** Warehouse
	IDAA	**Db2** Analytics Accelerator

- **Db2 when referring to the product in general e.g. Db2 for z/OS**
- **Db2 when referring to a specific release e.g. Db2 12 for z/OS**

Agenda

- **Part 1**
  - – Db2 11 for z/OS prerequisites for migration to Db2 12 for z/OS
  - – Db2 12 for z/OS Migration – Quick Hits
  - – Maintenance recommendations for early adopters of Db2 12 for z/OS
  - – Db2 12 for z/OS Risk Mitigation
  - – Understand Continuous Delivery starting with Db2 12 for z/OS
  - – Understanding new function levels
  - – Db2 12 for z/OS Greatest Hits
  - – Fast Un-clustered INSERT
  - – RTS enhancements

Agenda …

- **Part 2**
  - Fast Index Traversal
  - Data dependent vs. numeric based pagination syntax
  - More use of list prefetch
  - Increase in log record size after converting BSDS in Db2 11 and entry to Db2 12
  - Dynamic Plan Stability
  - More granular global commit LSN and global read LSN
  - SQLCODE -109 Issue
  - Enhanced SQL MERGE
  - DRDA Fast Load
  - UTS Relative Page Number (RPN)
  - INSERT Partition
  - Asynchronous CF Lock structure duplexing
  - Setting initial Statistics Profile
- **Summary**

Db2 11 for z/OS prerequisites for migration to Db2 12 for z/OS

- **Ensure catalog consistency**
  - REPAIR DBD TEST/DIAGNOSE + CHECK DATA/LOB/INDEX + DSNTESQ + ….
- **Run pre-migration check queries and act on the reported findings**
  - DSNTIJPM (Db2 12 for z/OS) or DSNTIJPC (APAR PI58254 for Db2 11 for z/OS)
- **Apply fallback SPE PTF to all data sharing members**
  - APAR PI33871 / II14794
- **Make sure Db2 11 for z/OS PTF level is reasonably current especially if exploiting mixed release coexistence with data sharing and all maintenance is applied related to Db2 12 for z/OS migration**
  - Use SMP/E Fix categories
    - IBM.Migrate-Fallback.DB2.DB2 12 and
    - IBM.Coexistence.DB2.SYSPLEXDataSharing

Db2 11 for z/OS prerequisites for migration to Db2 12 for z/OS …

- **Convert BSDS to 10 byte log RBA before leaving Db2 11 for z/OS NFM**
  - – For data sharing, convert single member at a time
  - – Things to consider before converting the BSDS (DSNJCNVT)
    - Stop the Db2 for z/OS subsystem that owns the subject bootstrap data set
    - Any utility (e.g., RECOVER, REORG) running on alternate Db2 member in data sharing that reads from peer BSDS must be terminated
      - ✓ Special considerations for Data Replication
        - » Stop any data replication process to ensure BSDS is successfully renamed and replaced
        - » Best practice is to stop data replication process first, then stop the Db2 subsystem
    - RACF user ID running DSNJCNVT must have read/write access on the new BSDSs, and read access on the old BSDSs
  - – After converting the BSDS, will see increased logging volume (3 <-> 40%)
  - – There will be further increase in log record size after entry in Db2 12 for z/OS because of 7-byte RID values and independent of using UTS PBR RPN
  - – Need to consider increasing size/number of active log pairs to maintain recommended 6 hours of recovery log data across active log configuration
  - – Need to reevaluate the size of the archive log DASD pool to ensure 48 hours’ worth of recovery log data can be kept

Db2 11 for z/OS prerequisites for migration to Db2 12 for z/OS …

- **Avoid autobind on pre-DB2 10 for z/OS plans and packages under Db2 12 for z/OS**
  - Avoid painful lesson related to plans
  - Thread break-in capability delivered in Db2 11 for z/OS may help for packages when performing rolling migration
  - Recommend explicit rebind under Db2 11 for z/OS NFM before leaving for Db2 12 for z/OS
    - Resolve any potential authorization issues that may exist
    - Use plan management for packages to keep a backup copy
- **Remember to set ZPARM ABIND=COEXIST if planning to use mixed release coexistence (Db2 11, Db2 12)**
- **Latest News …**
  - New APAR PI87675 - Re-migration autobinds are disabled
    - Both for plans and packages
    - Both for ABIND=COEXIST and ABIND=YES which now behave the same
      - ✓ Will no longer perform re-migration autobinds
    - Complete solution for Db2 11 for z/OS and Db2 12 for z/OS

Db2 11 for z/OS prerequisites for migration to Db2 12 for z/OS …

- **FREE inactive package copies (access plan management) before first REBIND under Db2 12 for z/OS**
- **Upgrade EXPLAIN tables to Db2 12 for z/OS format (should be at least Db2 11 for z/OS version)**
  - Can be done in Db2 11 for z/OS NFM with fallback SPE applied
  - Use of sample batch job DSNTIJXA with REXX DSNTXTA can help
- **Apply PTFs for APARs PI69589 (Db2 11 for z/OS) & PI69584 (Db2 12 for z/OS)**
  - Reduce catalog contention during “online” migration to Db2 12 for z/OS
- **Plan for activation of Db2 12 for z/OS EARLY code**
  - Activation via IPL or Command –REFRESH DB2,EARLY

Db2 12 for z/OS Migration – Quick Hits

- **Minimum OS level lifted from z/OS V1R13 to V2R1**
- **Minimum hardware level lifted from z10 to z196/z114**
- **Replication**
  - Db2 12 for z/OS (with APAR PI70998) and Db2 11 for z/OS require the Q Capture and Capture programs from IBM InfoSphere Data Replication for Db2 for z/OS Version 10.2.1
  - Q Apply and Apply programs at architecture level 1001 will work with both Db2 11 for z/OS and Db2 12 for z/OS
  - APAR PI70998 for Db2 for z/OS
  - APAR PI66768 for IIDR 10.2.1 Q and SQL
  - APAR PI61562 for CDC

Db2 12 for z/OS Migration – Quick Hits …

- **Db2 Connect**
  - Any level of Db2 Connect drivers should work with Db2 12 for z/OS, both before and after new function is activated 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 of Db2 for z/OS FL=V12R1M**501** 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 Modification 2 Fix Pack 2, or later
  - New clientApplCompat 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 V12R1M500 is required to access Db2 12 for z/OS Server capability shipped after GA at function levels beyond Db2 12 for z/OS FL=V12R1M**500**

Db2 12 for z/OS Migration – Quick Hits …

- **Changes to Utilities Suite installation**
  - Requires registration in SYS1.PARMLIB(IFAPRDxx)
    - CBPDO is being sunset, and SystemPac is the strategic direction
    - Any separately orderable product using only F or J FMIDs has to be changed to use an E or H base FMID
  - Documented in Db2 Utilities Suite program directory

```
PRODUCT OWNER('IBM CORP') NAME('DB2 UTIL SUITE') ID('577-AF4')
VERSION(12) RELEASE(1) MOD() FEATURENAME('V12R1') STATE(ENABLED)
```

  - Failure to register Utilities Suite results in utility errors

```
DSNU3333I 012 14:35:50.01 DSNUGPRS - THE DB2 UTILITIES SUITE FOR Z/OS HAS NOT BEEN ENABLED
DSNU3330I 012 14:35:50.09 DSNUGPTS - THE xxxxxxxx UTILITY HAS RESTRICTED FUNCTION
IT IS PART OF THE DB2 UTILITIES SUITE FOR Z/OS WHICH HAS NOT BEEN ENABLED
```

Db2 12 for z/OS Migration – Quick Hits …

- **REORG MAPPING TABLE format must allow for 7-byte RID values**
  - – SQL DDL changes
    - Column 'SOURCE_RID'   CHAR(5) -> CHAR(7)
    - Column 'TARGET_XRID' CHAR(9) -> CHAR(11)
  - – No toleration logic in Db2 11 for z/OS NFM
    - Db2 11 for z/OS NFM REORG running with the Db2 12 for z/OS mapping table format will fail
  - – Db2 12 for z/OS in FL=V12R1M100 REORG tolerates Db2 11 for z/OS format mapping table format
  - – Db2 12 for z/OS, in FL=V12R1M100 and V12R1M5nn, REORG only supports the Db2 12 for z/OS mapping table format
- **BIF_COMPATIBILITY system parameter still supported**

Db2 12 for z/OS Migration - Quick Hits …

- **RACF changes related to IDAA**
  - RACF ACCESS(CONTROL) on MVS.VARY.TCPIP.DROP(OPERCMDS)
- **HVSHARE should be 510 TB (default)**
  - Db2 12 for z/OS requires 1 TB of 64-bit shared (private) storage in z/OS (same as Db2 11 for z/OS)
  - Virtual, not real
  - Monitor use with IFCIDs 217 and 225
- **Plan for real memory increase**
  - Trend continues … using larger size REAL memory to deliver performance improvements
  - Expect ~ 15% increase
  - Expect up to 30% increase if taking advantage of new in-memory function
    - Largest percentage from use of Fast Traverse Block (FTB) area – 20% increase on allocated VPSIZE
- **Consider current zIIP utilization**
  - Trend to extend zIIP offload continues
    - REORG and LOAD RELOAD phase
    - SQL query parallelism (child task eligibility 80% -> 100%)

Db2 12 for z/OS Migration - Quick Hits …

- **Increased space requirement for RID Pool as a result of RID value increase 5 -> 8-byte value**
  - Internally Db2 for z/OS uses a normalized 8-byte RID value to allow for future expansion
  - More RID blocks will be used for the same query because each RIDLIST holds fewer RIDs
  - RID Pool memory usage will be roughly 60% higher (for smaller lists it will be up to 2x higher)
  - May have to increase MAXRBLK (RID Pool size) by up to 60%
  - Data Manager logical limit (RIDMAP/RIDLIST) reduced from 26M (26,602,376) RIDs to 16M (16,625,976) RIDs
  - More RID Pool overflow to workfile is to be expected
- **Deprecation of Basic Row Format (BRF)**
  - zparm SPRMRRF is now hidden in Db2 12 for z/OS
  - ROWFORMAT keyword option for REORG/LOAD to convert a pageset between BRF/RRF has been removed from the documentation
    - Still supported from a utility syntax perspective
  - New objects created will always be RRF
  - Existing pagesets in BRF will continue to be supported for the time being

Db2 12 for z/OS Migration - Quick Hits …

- **Deprecation of Basic Row Format (BRF) - How to recover an object in BRF format**
  - For an image copy that is in BRF format, user can either:
    - Run DB2 RECOVER utility using that image copy
      - ✓ At the end of RECOVER completion, it will update the catalog/directory to reflect the state of the restored data
      - ✓ So if the image copy contains BRF data, the tablespace meta data definition will be updated to BRF at the end of RECOVER completion
    - Or use DB2 UNLOAD utility to unload the raw data from that image copy
      - ✓ The UNLOAD utility works perfectly fine still on BRF and RRF pagesets
      - ✓ Then use the LOAD utility to load the data back, into whichever format the target table is
  - Short of providing the instructions for user on how to enable/disable the now hidden RRF zparm, there is no direct user control on creating a BRF table space or partition
    - If there is ever a need to do that, user can create the tablespace or partition as usual, and then run REORG ROWFORMAT BRF to convert the object into BRF
    - ROWFORMAT keyword option on LOAD/REORG is no longer documented, but the option still remains functional as before

Db2 12 for z/OS Migration - Quick Hits …

- **Invalidation of prepared SQL statements in dynamic statement cache**
  - Prior to Db2 12 for z/OS, RUNSTATS would always invalidate prepared statements dependent on the object that the utility was run against
  - In Db2 12 for z/OS, RUNSTATS by default will not invalidate the prepared statements (incompatible change)
    - Use new INVALIDATECACHE YES option to force the invalidation of prepared statements
  - Invalidation of prepared statements will still occur when
    - RUNSTATS ... INVALIDATECACHE YES
    - RUNSTATS after SQL DDL (CREATE/DROP INDEX) and statistics profile updated
    - RUNSTATS … UPDATE(NONE) REPORT(NO)
    - For other utilities, if the object was in an invalid state before the utility began e.g., rebuild pending or reorg pending

Maintenance recommendations for early adopters of Db2 12 for z/OS

- **Early adopters of new releases and/or new functions should be more aggressive about applying preventative service**
- **Apply preventative maintenance every 3 months**
  - Use RSU instead of PUT to be less aggressive on applying non-HIPER maintenance
  - Sample strategy based on two 'major' and two 'minor' releases
    - Refresh of the base every 6 months ('major')
    - Each base upgrade should be based on latest quarterly RSU
      - ✓ Ensure that RSU-only service is installed by adding the SOURCEID (RSU*) option in the supplied APPLY and ACCEPT jobs
    - In addition, two mini packages covering HIPERs and PEs in between ('minor')
- **Review Enhanced HOLDDATA on a weekly basis**
  - Expedite critical fixes to production after 1-2 weeks in test
  - Others can be deferred until the next major or minor maintenance drop

Different APAR numbers across Db2 11 and Db2 12 for z/OS

- **Short term solution was to make the connection visible, bi-directional, either in the SYSROUTED FROM or SYSROUTED TO section of the Outline/Overview of the respective APAR**

```
APAR   PI76204   OUTLINE


ENTER   DISPLAY-ITEM   PAGES   ENTER   DISPLAY-ITEM   PAGES
 S    SUMMARY              3     X    SUBMITTER TEXT     5
 E    RESPONDER TEXT       0     P    PIN ITEM           0
 I    INTRSTD PARTIES      1     K    TRACKING           2
 C    CONSTANT             1     Z    SCRATCH-PAD        1
 F    FEEDBACK             0          STRUCTURE          2
 DUP  FIRST DUPLICATE APAR   (OR DUP/NNN)   ORG  ORIGINAL APAR
 PTF  FIRST PTF REQUESTED.   (OR PTF/NNN)
                                        LAST PAGE


PTF RQSTD:
APAR FIXED BY:
DUPLICATES:
SYSROUTED FROM:
SYSROUTED TO: PI76206
APPLICABLE PE-PTF'S:
```

Different APAR numbers across Db2 11 and Db2 12 for z/OS ...

- **Single APAR solution for a single problem for new APARs has now been delivered (June 2017)**
  - Default now is a single APAR for the same problem across Db2 11 for z/OS and Db2 12 for z/OS
  - Assumes the same “basing” – so there will still be some exceptions
  - Will not be applied respectively

Db2 12 for z/OS Risk Mitigation

- **Regression testing is critical piece to keep “fires away from production”**
  - Test all critical and custom processes, and scale them up
  - Run performance measurements and establish Db2 11 for z/OS baseline for comparison
  - Go / No Go decision for Db2 12 for z/OS migration of production system should be based on positive results from proper testing
    - Be prepared to postpone migration as opposed to forcing in Db2 12 for z/OS
  - Practice migration fallback from Db2 12 for z/OS to Db2 11 for z/OS and back to Db2 12 for z/OS
  - Design fallback strategy and practice it in pre-production environments
- **Minimize change and use of new function in and around when Db2 12 for z/OS is first introduced into production**
- **For production systems, stay on FL=V12R1M100 for at least a month to prove running smoothly**
  - Leaves back door open to go back to Db2 11 for z/OS NFM in an emergency
- **Make sure very current on preventative service for these specific Db2 12 for z/OS functions**
  - Fast Index Traversal
  - Active Log Dataset Size > 4G
  - UTS PBR RPN
  - INSERT ALGORITHM 2 (aka “Fast Insert” or “Smart Insert”) for fast un-clustered insert

Old Db2 for z/OS Strategy for delivering new function

- **We deliver most of our new function in a new release ~every 3 years**
- **Db2 for z/OS is on 3 year cycles, but many of our customers are on 4 year cycles, hence the interest in skip release migrations**
- **We develop or retrofit a very limited number of new features in the service stream, but only if urgent and generally low risk**
- **Deployment of new releases is seen as a disruption by our customers**
- **Many of our customers want new features delivered much faster**
- **Industry and customer trend is to move away from monolithic code delivery towards continuous delivery model**
- **IBM is moving towards continuous delivery model**
- **Time for us in Db2 for z/OS to change**

New Db2 for z/OS Strategy for delivering new function

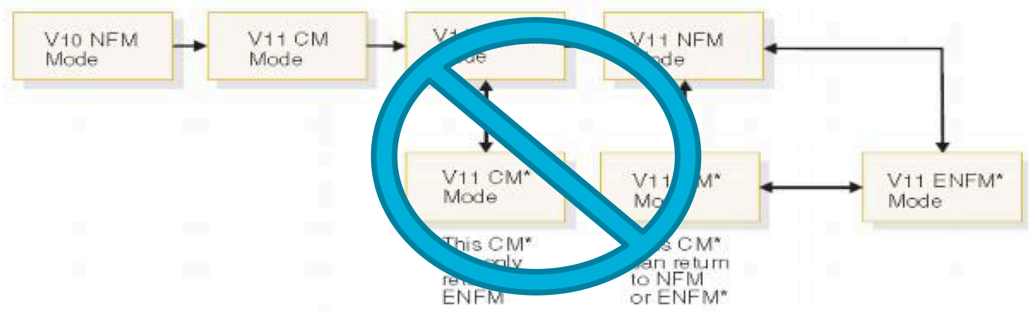
- **We are dedicating ourselves to going forward on a continuous delivery model**
  - Radical internal changes are required within Db2 for z/OS Development to do this
- **Db2 12 for z/OS is the starting point after GA**
  - There will be significantly higher volume of continuously delivered items
- **Customers will see a single maintenance stream for Db2 12 for z/OS, with the new function delivered into that**
  - The function will be designed to be easily consumable
- **Point releases or versions will be a very rare exception**
  - There are reasons why we might want to have a point release or new version
  - e.g., adopt a new compiler, extend control structures, enable an architecture level set
- **Db2 for z/OS Development will have relentless focus on maintaining continuous production level reliability for you in the service stream**
- **We are dedicated to doing this**
  - We will control the input to “the pipe”, the size and risk of the items
  - Increased internal focus on function and performance regression testing
  - We will deliver new function when the quality is right, and not based on a promised date for delivery

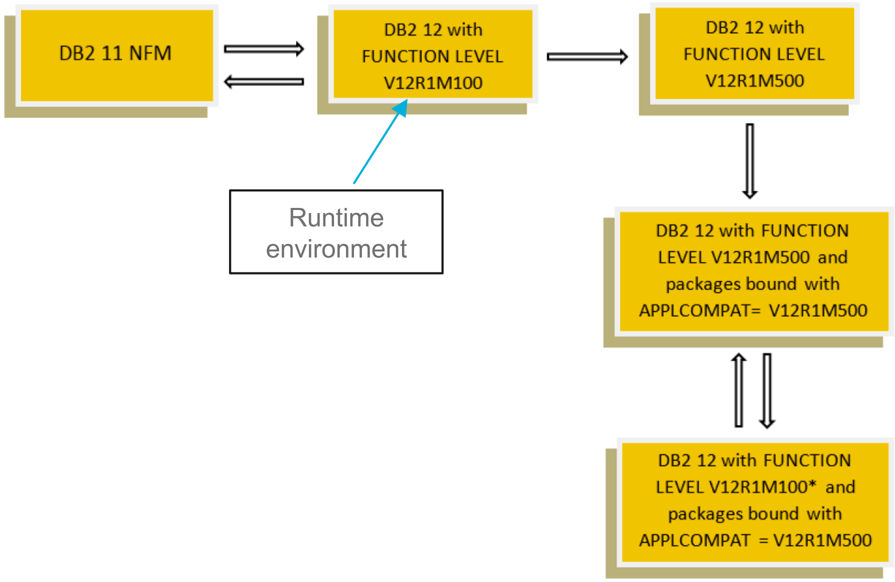
Understand Continuous Delivery starting with Db2 12 for z/OS

- **With Continuous Delivery, there is a single delivery mechanism for defect fixes and enhancements**
  - PTFs (and collections of PTFs like PUTLEVEL and RSU) → same as today
- **With Continuous Delivery, there are four Db2 for z/OS levels**
  - Maintenance level (ML) – lifted by applying maintenance
    - Also known as code level - contains defect and new enhancement fixes
    - Most new functions are shipped disabled until the appropriate new function level is activated
  - Catalog level (CL) - vehicle to enable new FL - accumulative (skip level possible)
    - Db2 Catalog changes that are needed for some FLs
  - Function level (FL) – needs to be activated - accumulative (skip level possible)
    - Introduces new Db2 for z/OS features and functionality
    - No impact or change in existing application behaviour
  - APPLCOMPAT level (AC) – set by application - provides an “island of stability” for a given application
    - Determines SQL function level of applications – can increase FL of the application (and fallback)
    - AC must be advanced to exploit new SQL function
    - AC level in BIND/REBIND of package must be <= FL and rules over FL
    - Freezes new SQL syntax even if FL is later moved back to earlier level
- **Minimum starting point for Continuous Delivery is Db2 12 for z/OS FL=V12R1M500**

Understanding new function levels

- **CM / ENFM / NFM no longer used**
- **Function Level V12R1M100**
  - Similar to CM / BNFA
  - Db2 12 for z/OS engine and catalog / directory
  - DSNTIJTC (CATMAINT) to get there
  - Fallback to Db2 11 for z/OS NFM possible
- **Function Level V12RM15nn**
  - Similar to NFM /ANFA
  - New functionality available
  - Command –ACTIVATE FUNCTION LEVEL(V12R1M5nn) to get there
  - Fallback to Db2 11 for z/OS NFM no longer possible (PIT recovery would be required)





DB2 11 NFM

DB2 12 with FUNCTION LEVEL V12R1M100

DB2 12 with FUNCTION LEVEL V12R1M500

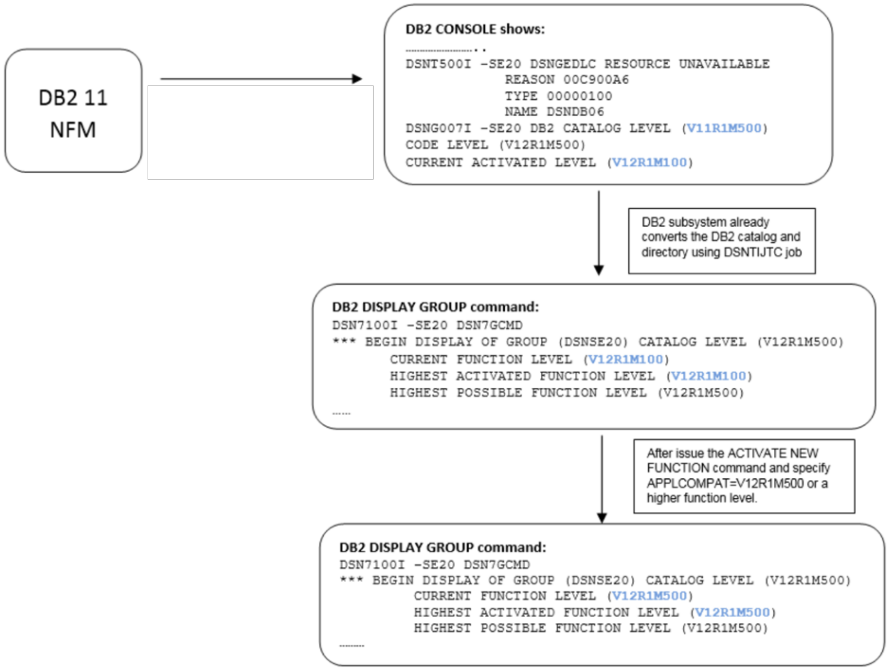
Runtime environment

DB2 12 with FUNCTION LEVEL V12R1M500 and packages bound with APPLCOMPAT= V12R1M500

DB2 12 with FUNCTION LEVEL V12R1M100* and packages bound with APPLCOMPAT = V12R1M500

Understanding new function levels …

1. **Set zparm APPLCOMPAT to V12R1M100**
   - Activated Level = V12R1M100
   - Keeps the application running at the current level
2. **Apply new maintenance (PTFs) to Db2 for z/OS libraries**
   - Maintenance Level (Code Level) = V12R1M500
   - New function exists but not active
3. **CATMAINT to update Catalog Level**
   - Catalog Level = V12R1M500
   - Db2 12 for z/OS Catalog and Directory updated
   - Fallback to Db2 11 for z/OS possible
4. **-ACTIVATE FUNCTION LEVEL command**
   - Function Level = V12R1M500
   - New functions available
   - No fallback possible to Db2 11 for z/OS
5. **Bind packages to set APPLCOMPAT for DB2Connect**
   - Activated Function Level = V12R1M500
   - Set zparm APPLCOMPAT to V12R1M500
   - Applications can use new SQL syntax



DB2 CONSOLE shows:

```
…………………………..
DSNT500I -SE20 DSNGEDLC RESOURCE UNAVAILABLE
            REASON 00C900A6
            TYPE 00000100
            NAME DSNDB06
DSNG007I -SE20 DB2 CATALOG LEVEL (V11R1M500)
CODE LEVEL (V12R1M500)
CURRENT ACTIVATED LEVEL (V12R1M100)
```

DB2 subsystem already converts the DB2 catalog and directory using DSNTIJTC job

DB2 DISPLAY GROUP command:

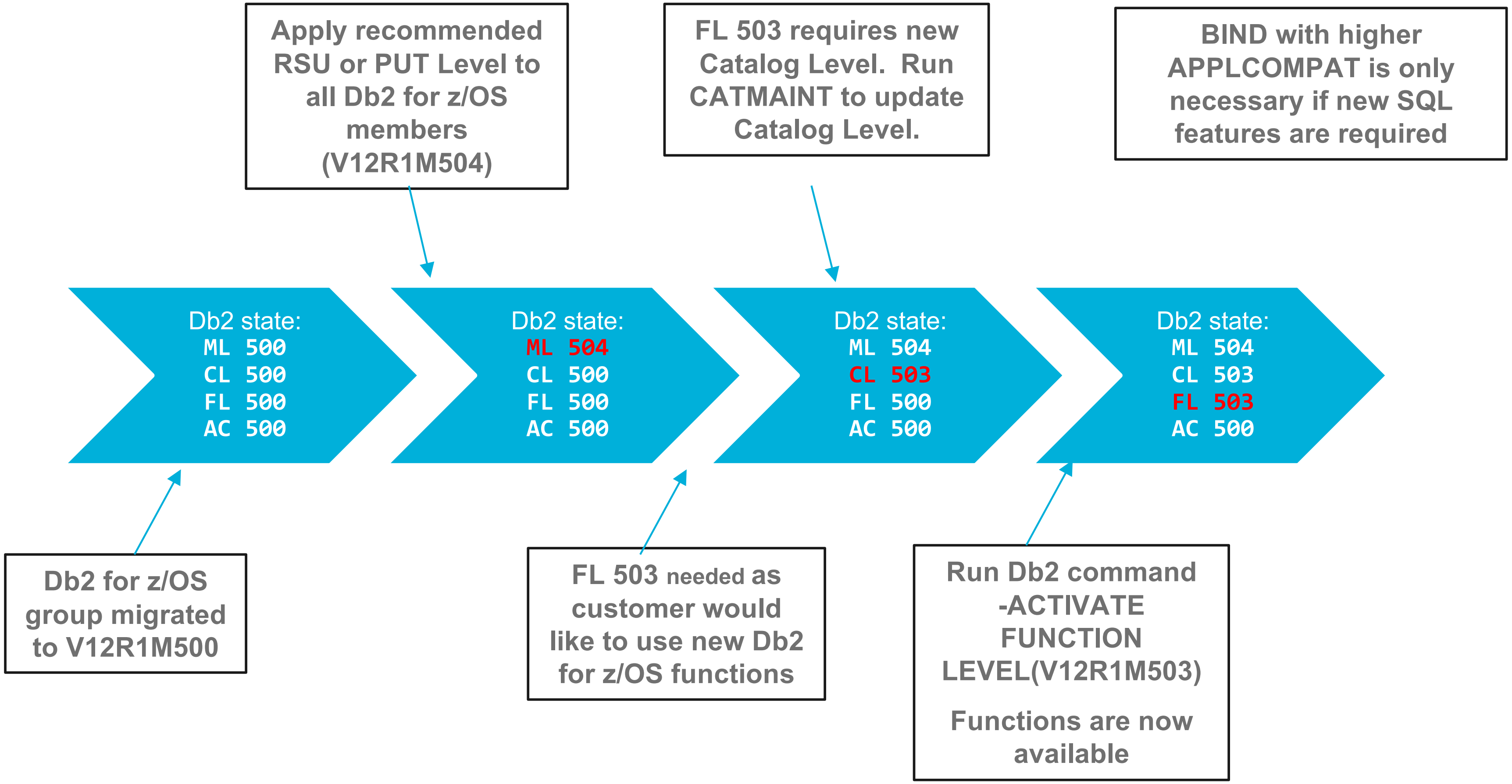
```
DSN7100I -SE20 DSN7GCMD
*** BEGIN DISPLAY OF GROUP (DSNSE20) CATALOG LEVEL (V12R1M500)
      CURRENT FUNCTION LEVEL (V12R1M100)
      HIGHEST ACTIVATED FUNCTION LEVEL (V12R1M100)
      HIGHEST POSSIBLE FUNCTION LEVEL (V12R1M500)
……
```

After issue the ACTIVATE NEW FUNCTION command and specify APPLCOMPAT=V12R1M500 or a higher function level.

DB2 DISPLAY GROUP command:

```
DSN7100I -SE20 DSN7GCMD
*** BEGIN DISPLAY OF GROUP (DSNSE20) CATALOG LEVEL (V12R1M500)
        CURRENT FUNCTION LEVEL (V12R1M500)
        HIGHEST ACTIVATED FUNCTION LEVEL (V12R1M500)
        HIGHEST POSSIBLE FUNCTION LEVEL (V12R1M500)
………
```

Example of how to get to a new function level



Db2 for z/OS group migrated to V12R1M500

Db2 state:
ML 500
CL 500
FL 500
AC 500

Apply recommended RSU or PUT Level to all Db2 for z/OS members (V12R1M504)

Db2 state:
ML 504
CL 500
FL 500
AC 500

FL 503 needed as customer would like to use new Db2 for z/OS functions

FL 503 requires new Catalog Level. Run CATMAINT to update Catalog Level.

Db2 state:
ML 504
CL 503
FL 500
AC 500

Run Db2 command -ACTIVATE FUNCTION LEVEL(V12R1M503)

Functions are now available

Db2 state:
ML 504
CL 503
FL 503
AC 500

BIND with higher APPLCOMPAT is only necessary if new SQL features are required

Change in strategy for APPLCOMPAT

- **No need to force the rebind all packages with a new, higher APPLCOMPAT level**
- **APPLCOMPAT will now have many more versions to support many Function Levels**
- **Must still rebind a package with a higher APPLCOMPAT level in order to exploit new SQL DML, SQL DDL, SQL DCL, and XML function**
  - Applications can only use new SQL if the packages are bound with the necessary and required Application Compatibility (APPLCOMPAT)
    - Packages can only be bound with an APPLCOMPAT less or equal to the current FL

- **Still recommended best practice to regularly rebind all packages to**
  - Benefit from latest run time performance improvements
  - Gain exposure to new access path selection improvements
  - Benefit from defect fixes
  - Reduce exposure to latent issues seeded previously

Is APPLCOMPAT a ‘sticky’ option on BIND/REBIND?

- **BIND REPLACE does not reuse any bind option from the existing package if the option is not explicitly specified**
- **SQL statements can be totally different so BIND REPLACE is considered a new bind**
- **REBIND and BIND COPY are the only subcommands that reuse the existing/source package's options**
- **This is true in all Db2 for z/OS releases and not just Db2 12 for z/OS**

Setting CURRENT APPLICATION COMPATIBILITY special register

- **Db2 11 for z/OS**
  - Value can be >= APPLCOMPAT LEVEL of the executing package but not > current Db2 version
- **Db2 12 for z/OS**
  - Value has to be <= APPLCOMPAT level of the executing package, independent of the current Db2 Function Level

Function Level Adoption – Best Practices 1/2

- **PTFs (RSUs…) are applied that may increase the Maintenance Level (ML) of a Db2 for z/OS subsystem**
- **After system is stable on maintenance**
  - Execute (If Any) Catmaint
    - After execution of Catmaint, the subsystem can only be started with a ML that supports the catalog
  - Activate Function Level (FL)
    - After activating a new FL, the subsystem can only be started with a ML that supports the FL
    - New function not related to SQL DML, DDL and DCL syntax is available
    - REBIND of packages with any APPLCOMPAT would pick up optimizer enhancements
    - Non-stabilized dynamic SQL would pick up optimizer / other non-APPLCOMPAT related enhancements

Function Level Adoption – Best Practices 2/2

- **After Function Level is considered stable - allow new application feature rollout**
  - REBIND DBA packages to allow new DDL to be utilized
  - REBIND application static packages with higher APPLCOMPAT to exploit SQL DDL/DML new functions/behaviors
  - REBIND dynamic packages with higher APPLCOMPAT to allow new SQL functions to be used
  - REBIND distributed packages (***in separate collection) with higher APPLCOMPAT to allow new SQL functions to be used
    - Switch applications to use new distributed package collection
  - Leverage PLANMGMT extended
    - Use REBIND SWITCH (PREVIOUS) to restore static SQL packages to prior runtime structures
    - Use REBIND SWITCH (PREVIOUS) for dynamic SQL packages would restore prior APPLCOMPAT
    - ***switching to prior collid for distributed dynamic would restore APPLCOMPAT

Db2 12 for z/OS Greatest Hits

- **Fast Index Traversal (FTB)**
- **Dynamic Plan Stability**
- **Granular global commit LSN and read LSN**
- **Enhanced SQL MERGE**
- **SQL pagination syntax LIMIT / OFFSET**
- **Online ALTER to increase DSSIZE**
- **Lifting partition size limit (1 TB)**
- **Insert Partition**
- **TRANSFER OWNERSHIP**
- **LOB compression**
- **DRDA Fast Load**
- **Asynch CF lock Duplexing**
- **REORG (and LOAD) use of statistics profiles**

Fast Un-clustered INSERT

- **Insert workloads are amongst the most prevalent and performance critical**
- **Performance bottleneck will vary across different insert workloads**
  - Index maintenance?
  - Log write I/O?
  - Data space search (space map and page contention, false leads)
  - Format write during dataset extend
  - PPRC disk mirroring
  - Network latency
  - etc
- **Common that Index insert time may dominate and mask any insert speed bottleneck on table space**

Fast Un-clustered INSERT …

- **Officially referred to as “Insert Algorithm 2 (IALG2)”**
- **Some times referred to as “Smart Insert” or even “Fast Insert”**
- **May potentially deliver significant improvement for un-clustered inserts (e.g., journal table pattern) where both**
  - Heavy concurrent insert activity (many concurrent threads)
  - Space search and false leads on data is the constraint on overall insert throughput
- **Applies to any UTS table space defined with MEMBER CLUSTER**
  - Applies to both tables defined as APPEND YES or NO
- **Implemented advanced new insert algorithm to streamline space search and space utilisation**
  - Eliminates page contention and false leads
  - Default is to use the new fast insert algorithm for qualifying table spaces
    - DEFAULT_INSERT_ALGORITHM system parameter can change the default
    - INSERT ALGORITHM table space attribute can override system parameter

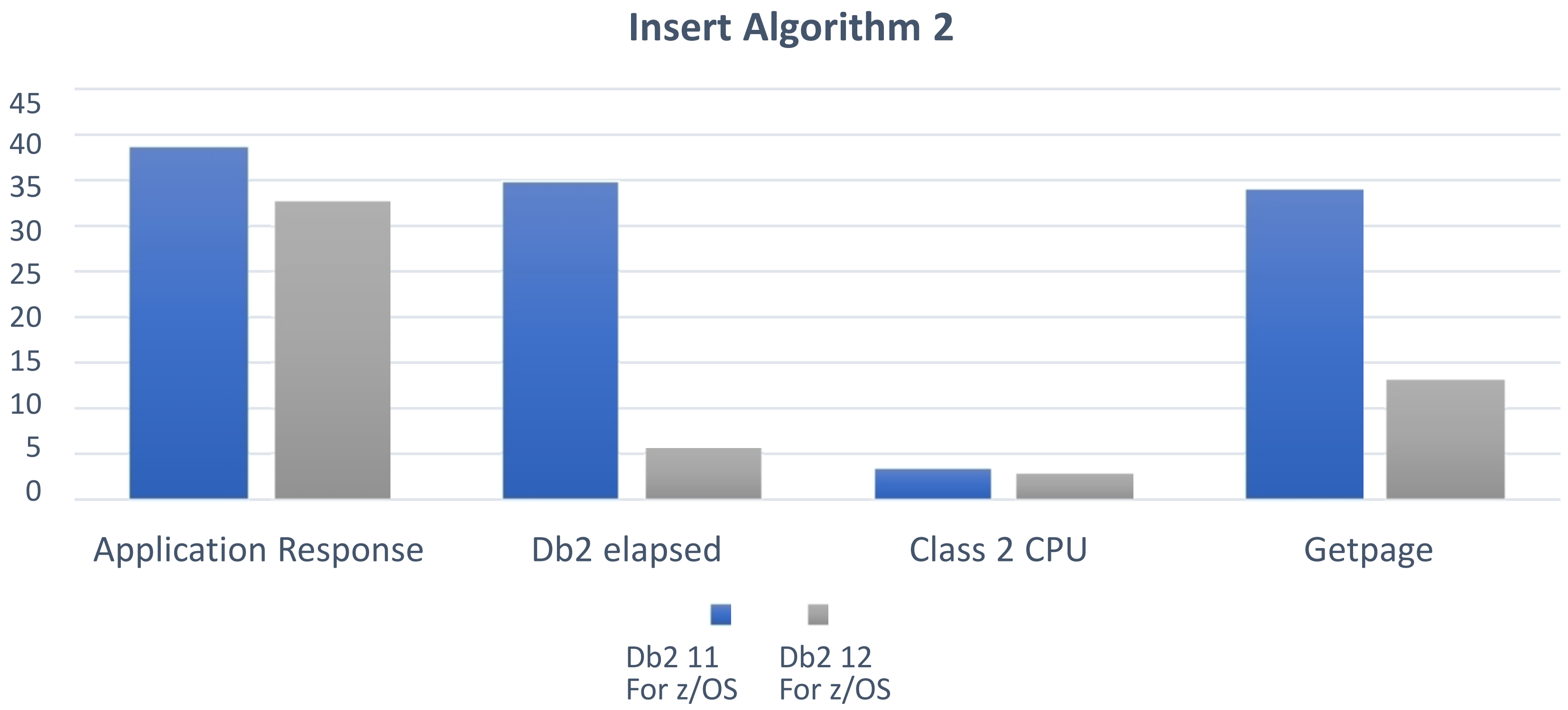
Fast Un-clustered INSERT …

- **Your mileage will vary**
  - Some insert workloads will see no improvement and is to be expected
  - Some specific insert workloads may see significant improvement
- **Will shift the bottleneck to the next constraining factor**
- **LOAD SHRLEVEL CHANGE can also use Fast Un-clustered INSERT**
- **Fast Un-clustered INSERT will not be used when lock escalation occurs or use of SQL LOCK TABLE**
- **Available after new function activation (FL=V12R1M5nn)**

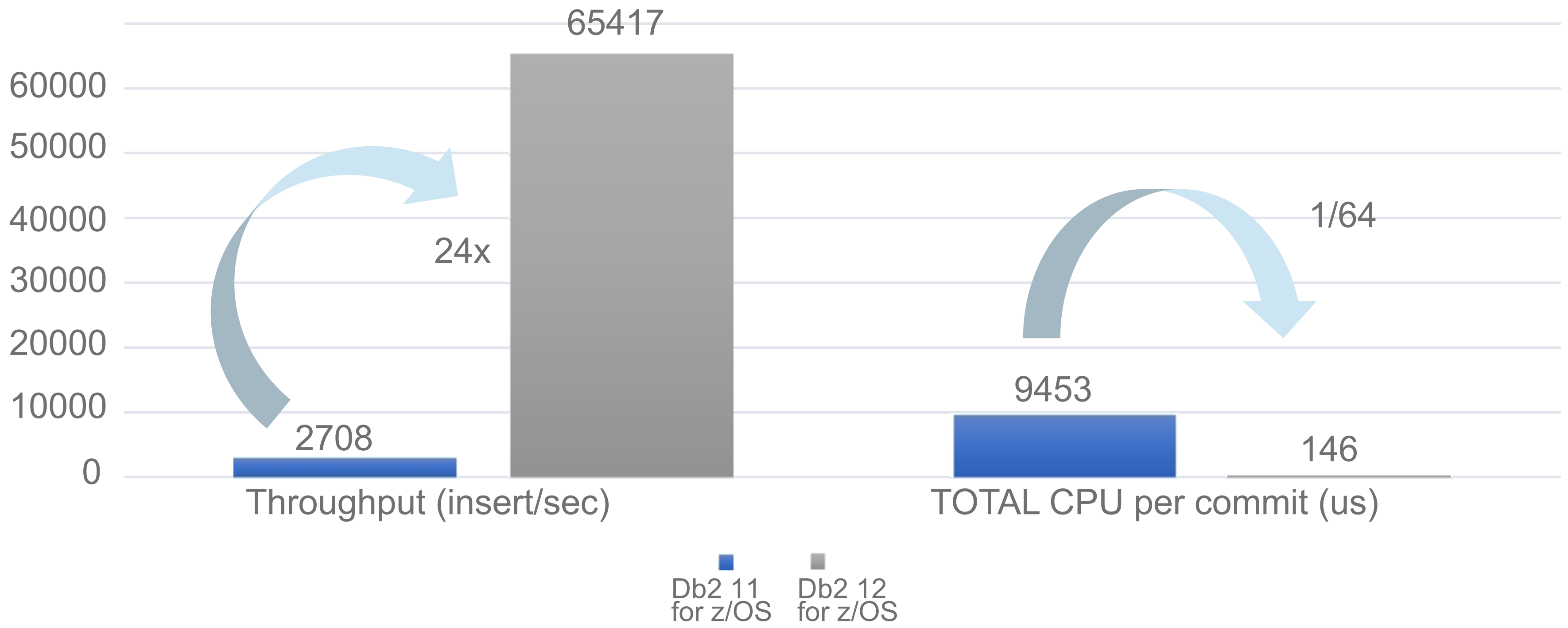
Fast Un-clustered INSERT …

- **Recommended preventative maintenance for robustness and serviceability as at 2nd September 2017**
  - APAR PI74870 - PTF UI44392
  - APAR PI75781 - PTF UI44974
  - APAR PI67997 - PTF UI41372
  - APAR PI77866 - closed
  - APAR PI68022 - closed
  - APAR PI80532 - closed
  - APAR PI77866 - closed
  - APAR PI81731 - still open
  - Several other APARs now open: PI83083, PI83519, PI85653, PI85018
- **Target date to make all the subject maintenance available is October 2017**
- **Current point-in-time recommendation**
  - Change system wide default - set system parameter DEFAULT_INSERT_ALGORITHM = 1 (old basic insert algorithm)
  - One size probably does not fit all tablespaces
  - Not much difference/improvement for short running transactions
  - Use INSERT ALGORITHM 2 (new fast insert algorithm) selectively at individual table space level to override system wide default

Fast Un-clustered INSERT – Shifting The Bottleneck …



Fast Un-clustered INSERT - Db2 11 for z/OS PMR Recreate …



UTS PBG with MEMBER CLUSTER, RLL, with 400 bytes per row, one index, 800 concurrent threads, 10 insert per commit

RTS enhancements

- **New messages DSNT535I and DSNT536I e.g.,**

```
DSNT535I  =D2E1 DSNIRTST 2 ATTEMPTS TO EXTERNALIZE IN-MEMORY STATISTICS TO REAL-TIME
STATISTICS TABLES FAILED DURING THE PAST 30 MINUTES 'BECAUSE A RESOURCE WAS UNAVAILABLE:
TYPE 00000304 NAME DSNDB06 .SYSTSISS.X'0000650D'.X'07'
```

- **New column GETPAGES added to both SYSIBM.SYSTABLESPACESTATS & SYSIBM.SYSINDEXSPACESTATS**
  - Very valuable
  - Records number of getpage requests since release migration, last REORG, last LOAD REPLACE or since object creation
  - Do not rely on the value whilst running in mixed release coexistence
- **Temporal (system-period data versioning)**
  - Requires FL=V12R1M5nn
  - SQL DDL changes performed by CATMAINT
  - Activated by ALTER TABLE … ADD VERSIONING clause -> SYSIBM.SYSTABLESPACESTATS & SYSIBM.SYSINDEXSPACESTATS
  - No indexes provided – must RYO to speed up your SQL queries
  - MAXPART 1 is ‘hard wired’ for history tables
  - Must develop procedures for cleanup of history tables and associated housekeeping

Fast Index Traversal

- **In memory index performance optimisation**
- **One of the most important performance features in Db2 12 for z/OS**
- **Used for fast index lookup by avoiding expensive index B-tree traversal**
- **Access must be random (index traversal) pattern to benefit**
- **SELECT, INSERT, DELETE, UPDATE, … can all benefit**
- **Separate Fast Traversal Block (FTB) memory area allocated outside of bufferpool**
  - Uses a concatenated structure, containing copy of non-leaf pages only, uses relative structure
- **Does not use bufferpool**
  - Non-leaf pages (except root page) are not fixed in the bufferpool
  - Pages are eligible for stealing and can be LRUed out of the bufferpool when the index non-leaf pages are stored in FTB memory
- **Improved performance**
  - Fast traverse block is L2 cache aware B-Tree like structure
  - Each page is equal to one cache line in size (256 bytes)
- **ESP customer example with**
  - 9.1% CPU reduction with 3 level index, 22.9% CPU reduction with 4 level index
- **Your mileage in terms of CPU reduction will vary**

Fast Index Traversal …

- **zparm INDEX_MEMORY_CONTROL = AUTO, DISABLE, x (MB)**
  - AUTO = 20% of total allocated bufferpool size (min 10 MB)
- **Each Db2 member will determine independently the good candidate indexes (daemon)**
  - Index must be unique
  - INCLUDE COLUMNS supported
  - Index entry length (key + additional columns) has maximum size of 64 bytes
  - Re-evaluates every 2 minutes and adjusts priority queue
    - Index traversal (+)
    - Index only access (++)
    - Index leaf page splits (----)
    - Index lookaside (-)
  - Internal threshold then applied
- **Control by SYSIBM.SYSINDEXCONTROL**
  - Indicate preference for specific indexes
  - Disable for specific indexes

Fast Index Traversal …

- **How does an index come into FTB area?**
  - Daemon task
    - zIIP eligible
    - Runs every 2 minutes
    - System agent correlation identifier: 014.IFTOMK00

```
DSNV497I  -DB2A SYSTEM THREADS -
DB2 ACTIVE
NAME     ST A   REQ ID           AUTHID   PLAN     ASID TOKEN
...
DB2A     N  *     0 014.RTSTST00 SYSOPR            004C     0
V490-SUSPENDED 17081-10:05:25.83 DSNB1TMR +00000EBF UI38562
DB2A     N  *     0 014.IDAEMK00 SYSOPR            004C     0
V490-SUSPENDED 17081-10:01:16.95 DSNB1TMR +00000EBF UI38562
DB2A     N  *     0 014.IFTOMK00 SYSOPR            004C     0
V490-SUSPENDED 17081-10:05:22.32 DSNB1TMR +00000EBF UI38562
DB2A     N  *     0 010.PM2PCP01 SYSOPR            004C     0
V490-SUSPENDED 17081-10:05:26.51 DSNB1TMR +00000EBF UI38562
...
```

Fast Index Traversal …

- **Monitor**
  - -DISPLAY STATS(IMU) or –DISPLAY STATS(INDEXMEMORYUSAGE) LIMIT(*) command

```
DSNT783I  -DB2A
DBID PSID DBNAME   CREATOR        INDEXNAME      LEVEL PART  SIZE(KB)
---- ---- -------- -------------- -------------- ----- ----- --------
0256 0005 SZI10D    § § § § § § §   SZI10X        0002  00001 00000025
0261 0005 SZI20D   A2345678901234 SZI20X         0002  00001 00000025
0262 0005 SZI30D   SYSADM         X2345678901234 0002  00001 00000025
0263 0005 SZI40D   SYSADM         SZI40X         0002  00001 00000025
******* DISPLAY OF STATS TERMINATED *********************************
DSN9022I  -DB2A DSNTDSTS 'DISPLAY STATS' NORMAL COMPLETION
```

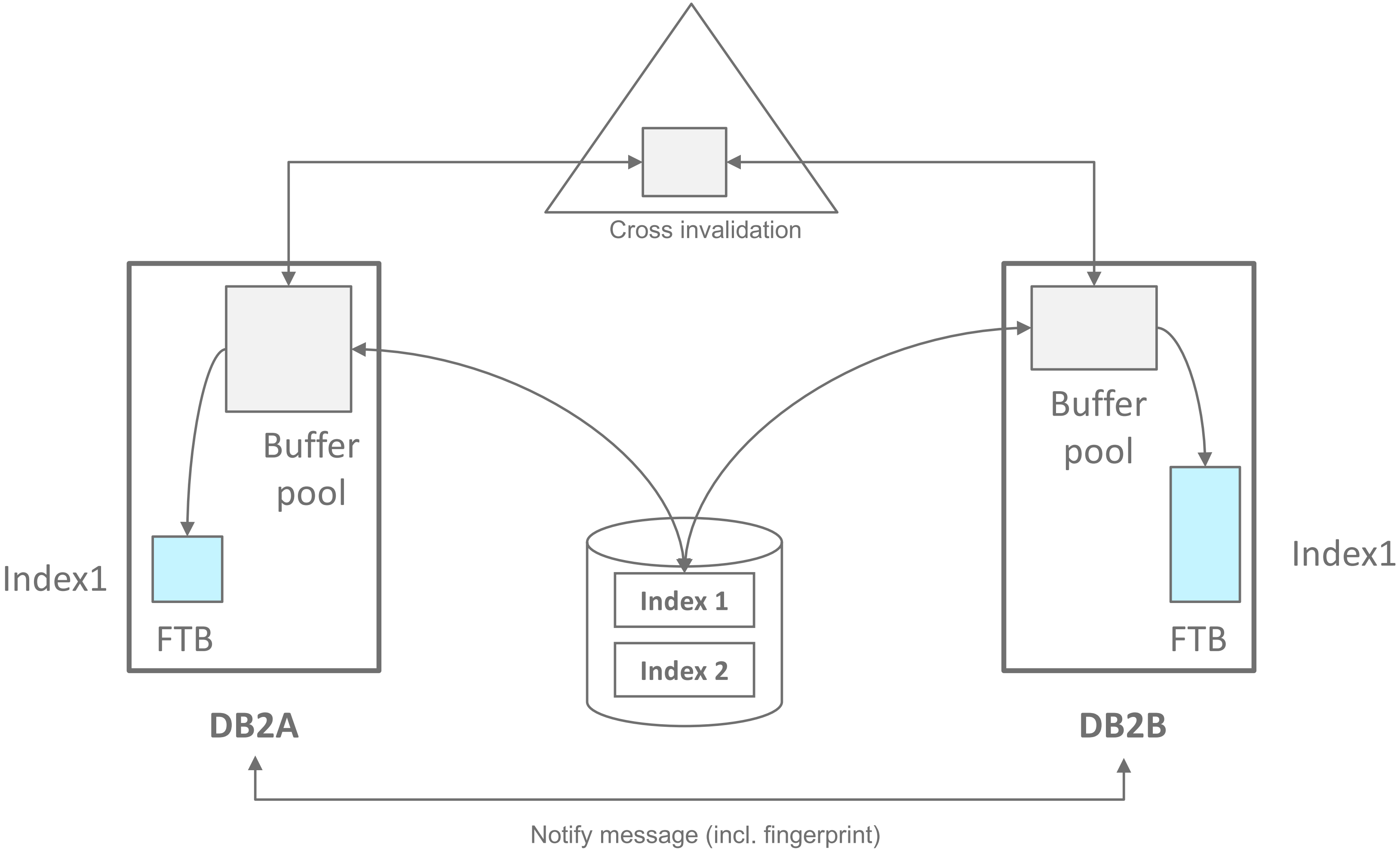
  - Trace
    - -START TRACE (PERFM) DEST(SMF) IFCID(477)
    - -START TRACE (STAT) DEST(SMF) CLASS(8) IFCID(389)

Fast Index Traversal …

- **Free FTB area for an index**
  - Pageset close
  - SQL mass delete
  - ALTER INDEX, RECOVER INDEX, REBUILD INDEX
  - Trick: ALTER INDEX from COPY YES to COPY NO (and the other way around)

Fast Index Traversal …

- **Data Sharing considerations – high level picture**



Fast Index Traversal …

- **Migration**
  - Available in mixed release coexistence (Db2 11 and Db2 12 for z/OS) or Db2 12 for z/OS before new function activation (V12R1M100)
    - FTB only used while index object is not GBP-dependent
    - If index object becomes GBP-dependent, the FTB content will be deleted/bypassed
  - After new function activation (V12R1M5nn)
    - FTB can now also be used when index object is GBP-dependent

Data dependent vs. numeric based pagination syntax

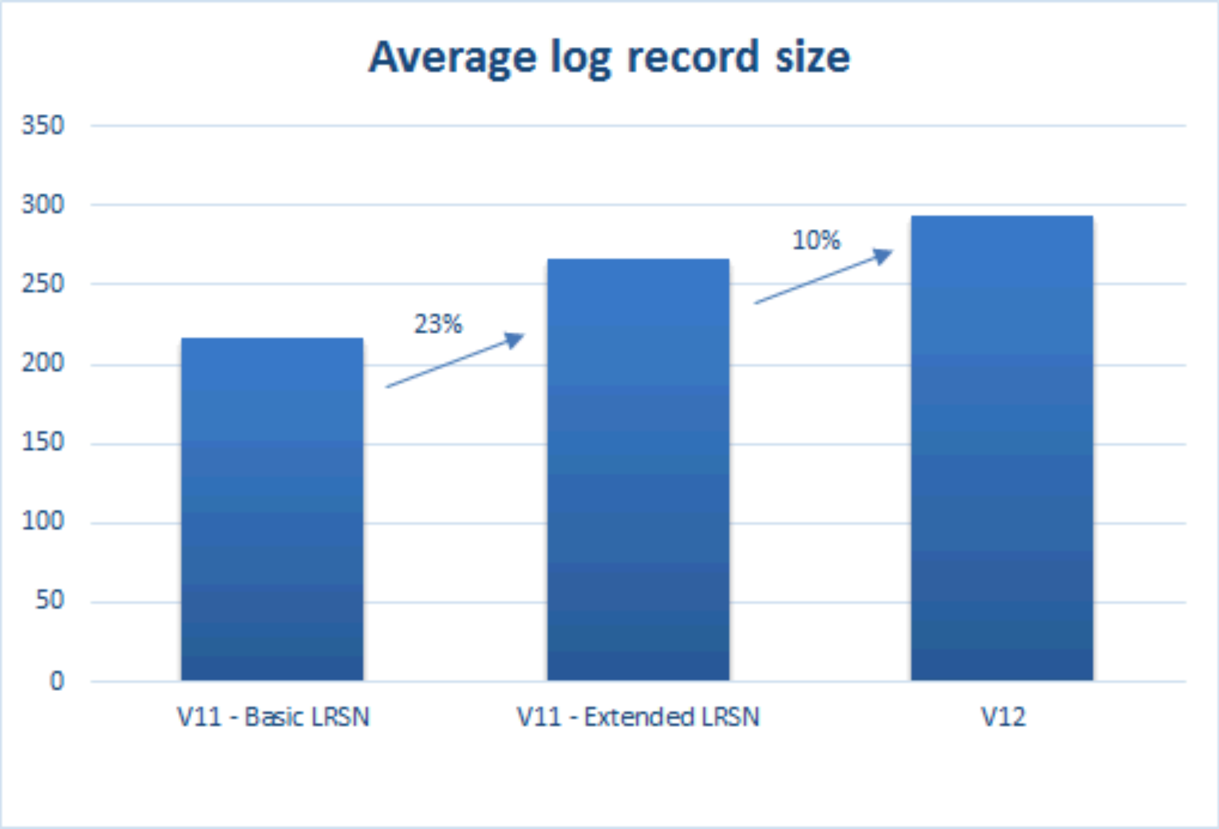
- **Data dependent pagination syntax e.g.,**
  SELECT … FROM … WHERE (LASTNAME, FIRSTNAME) >= (:lname, :fname)
  - Given correct index design
    - Can go directly to the needed rows
    - Exploits range-list index scan (ACCESSTYPE=‘NR’)
- **Numeric based pagination syntax e.g.,**
  SELECT … FROM … OFFSET 10 ROWS FETCH FIRST 10 ROWS ONLY
  - Will have to skip through the unneeded rows
  - If rows are deleted/inserted from other applications in between
    - May see the same rows twice or not see the rows at all
- **Many static scrollable cursors can be replaced by SQL pagination**
  - Result set is no longer materialized
  - Read-only applications will not create long running unit of recoveries
  - Performance can be improved
- **Works very well as advertised**

More use of list prefetch

- **Enhancement to the Optimizer cost model to more closely reflect the true cost (and benefit) of list prefetch**
- **Expected to see an increase in list prefetch (and potentially hybrid join)**
- **But not necessarily changes in the access plan where Db2 would previously have chosen a sort avoidance plan**
- **Db2 for z/OS trying to be careful not to select list prefetch (with sort) as an access path when there was an alternative access path that could use an index to avoid a sort i.e., for pagination type SQL**

Increase on log record size after converting BSDS in Db2 11 & entry to Db2 12

- **About 50 byte increase after converting BSDS under Db2 11 for z/OS NFM**
- **Further increase in log record size in Db2 12 for z/OS because of larger 7-byte RID values**
  - Increase is about 20 bytes for table space and about 28 bytes for index



Dynamic Plan Stability

- **Welcome new feature that will bring some relief in the area of performance management of dynamic SQL**
  - Goal is to provide consistent, more reliable performance
  - Sweet spot is short running SQL that is executed 1000s of times
  - Helps with high "turnover" periods in dynamic statement cache
- **In Db2 11 for z/OS a miss in dynamic statement cache requires a new full prepare e.g.,**
  - Db2 subsystem recycle
  - Release migration
  - RUNSTATS
- **In Db2 12 for z/OS can stabilize a query statement from the dynamic statement cache**
  - No new full prepare needed
  - Statement is loaded into the dynamic statement cache from the Catalog
  - Statement is invalidated by SQL DDL like a static SQL package
- **Can stabilize**
  - Specific dynamic query statement
  - Dynamic query statements with more than a certain amount of executions

Dynamic Plan Stability …

- **Change of APPLCOMPAT and/or special registers (DEGREE, OPTHINT, etc) will cause cache miss**
- **No REBIND capability to "repair" after invalidations**
  - Need to wait for new stabilization
- **Restrictions**
  - Display command has only local scope
  - No support for concentrated statements
  - No support for query statements against temporal and transparent archive
- **FREE stabilized dynamic query STBLGRP(x)**
  - Will also invalidate the statements in the dynamic statement cache
  - May result in a "storm" of full prepares
- **Stabilized dynamic query statements do consume more CPU than the equivalent static query statement**

More granular global commit LSN and global read LSN

- **Db2 for z/OS does not actually track "more current" value for each individual object**
- **Each member maintains two global lists of the 500 objects which have the worst (oldest) CLSN and read-LSN values**
- **Global lists built by a system task that wakes every 2 seconds (subject to change)**
- **Rebuilds it's own list**
- **Merges it with every other member's list to create the global list**
- **When it comes time to pick up an object's CLSN or read-LSN value**
  - Check the appropriate global list for the object
    - If it is on there, then we know what it's LSN is
    - If not, then use as an "alternate" LSN for the newest object (as object's LSN cannot be worse then this value)
  - Either way Db2 will compare the LSN picked pick up with the old global value (from SCA), and use that if it is better
- **Very nice enhancement that has great potential to improve lock avoidance and/or space reuse on LOB insert when the inevitable long running reader-UR is in play**

LOB compression

- **Requires zEDC hardware feature**
  - Will decompress existing compressed LOB if zEDC not available
  - Will not compress a LOB if zEDC not available
- **Inline LOB is completely separate from LOB compression**
  - LOB compression only applies to the the out-of-line portion
  - Split and compressed independently
- **Aimed at textual**
  - Not video and audio as these are already heavily compressed outside of Db2 for z/OS e.g., MP3 or MP4

SQLCODE -109 Issue

- **Problem:**
  - Non-documented and illegal use of SELECT ... INTO ... UNION ALL syntax
    - Customer complaints, can produce wrong results, defect
- **Solution:**
  - Loophole closed in Db2 12 for z/OS
  - Retrofitted back to Db2 11 for z/OS with APAR PI67611
  - New zparm: DISALLOW_SEL_INTO_UNION
    - NO (Db2 11 for z/OS default)
      - ✓ Allows usage of this illegal SQL syntax when such usage is encountered during execution of a BIND or REBIND command
      - ✓ Db2 will write an incompatibility trace record to IFCID 376
      - ✓ Use these trace records to identify and correct applications that are using the illegal SQL syntax
    - **YES** (Db2 12 for z/OS default)
      - ✓ Disallow usage of this illegal SQL syntax
      - ✓ Statements that include syntax will fail with SQLCODE -109
      - ✓ Running IFCID 376 under Db2 11 for z/OS will help identify problem applications
  - Need to deal with this potential issue before migration to Db2 12 for z/OS or change the Db2 12 for z/OS default

Enhanced SQL MERGE

- **Db2 12 for z/OS delivers ANSI compliant MERGE capability**
- **SQL MERGE is now very powerful**
  - Source can now include TABLE, VIEW and full Select
  - Additional predicates on MATCHED/NOT MATCHED
  - Can do DELETE
  - Can do multiple UPDATE, INSERT and DELETE phrases
    - But not on same row
    - Can accept SIGNAL and IGNORE
- **Benefits**
  - Development productivity
  - Improved performance
  - Application porting to Db2 for z/OS

Enhanced SQL MERGE …

- **But SQL MERGE is now so powerful …**
  - Input can be a SELECT (JOIN) returning many rows (millions, billions)
  - # UPDATEs, INSERTs and DELETEs could explode
  - Considerations
    - No intermediate commit points
    - Long rollback time
    - Lock escalation and impact on concurrency
    - No SQL pagination support

DRDA Fast Load

- **It is super fast**
- **Some complication to format the input records correctly**
- **Problem area is missing restart after failure**
  - Must terminate Utility
  - RECOVER and REBUILD objects
  - Restart the DRDA Fast Load

UTS PBR Relative Page Number (RPN)

- **Motivation**
  - Tremendous improvement in terms of availability and usability
    - DSSIZE can vary for different partitions
    - DSSIZE can now be increased for an individual partition with zero application impact
      - ✓ Immediate alter and no REORG required to increase DSSIZE
    - Note: A decrease in DSSIZE is still a pending alter and requires a full table space level REORG
  - Scalability
    - Maximum partition size increases to 1 TB
    - Maximum table size increases to 4 PB
    - Maximum number of rows in a table increases from 1.1 Tn to 280 Tn

UTS PBR Relative Page Number (RPN) …

- **Migration possible from either classic partitioned and UTS Partition By Range (PBR) table spaces**
  - Steps for conversion
    1. ALTER TABLESPACE … SEGSIZE n
       - ✓ If starting from classic partitioned
    2. ALTER TABLESPACE … PAGENUM RELATIVE
       - ✓ Table space put into AREOR state
    3. REORG TABLESPACE …
  - Base and XML table spaces can be migrated separately
    - Can “coexist” running with mixed RELATIVE/ABSOLUTE attributes
  - One-way ticket – no fallback to absolute page numbering (PAGENUM ABSOLUTE)
  - Extended Addressability (EA) must be used for UTS PBR RPN datasets
  - DASD space for large datasets can lead to problems (e.g. running out of volumes)
    - Datasets can only be spread across 59 volumes
    - For example, a 1 TB dataset will require 3390 Model 27 or above

UTS PBR Relative Page Number (RPN) …

- **Migration issues**
  - – Cannot convert to RPN or even create new RPN tablespace because cannot REORG them if you want inline part-level image copies to go to tape
    - New TAPEUNITS option should be available by end of 2017
    - See APAR PI75518 which is still open
  - – Pre-V6 range partitioned tablespaces with limit key values truncated at 40 bytes cannot be converted over
    - Should only affect a small number of customers
    - Problem is fenced and the conversion will not succeed
      - ✓ ALTER TABLESPACE PAGENUM RELATIVE fails with SQLCODE -650 RC 39
    - No target date at present time for providing relief

UTS PBR Relative Page Number (RPN) …

- **Other considerations**
  - – Indexes will increase in size because of larger 7-byte RID values
  - – SYSCOPY entry will be created for index even if the index is defined as COPY NO
  - – Recommend the index COPY/RECOVER for XXXL size NPIs
  - – Note: can no longer identify the partition number from the page number

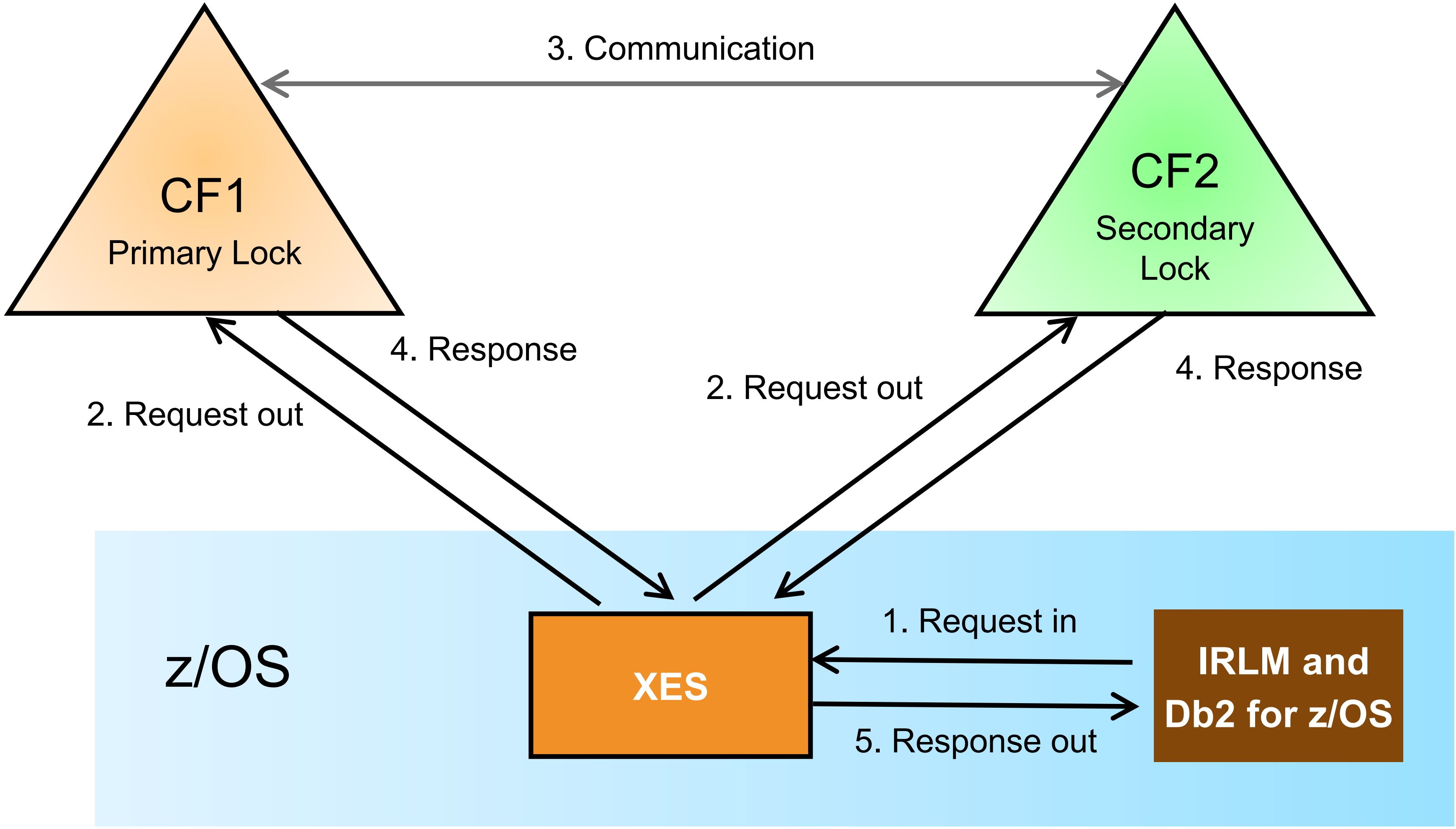
Insert Partition

- **Insert Partition "in the middle" where it is required**
- **UTS PBR only, BUT no requirement for RPN**
- **Restriction: no LOB or XML**
- **ALTER … ADD PART ENDING AT (…) is a pending alter**
- **Necessary REORG can be limited to a minimum subset of partitions (only affected partitions)**
- **Be aware that logical partition numbers have to be translated to physical partition numbers**
  - New physical partition is added at the end i.e., A00n+1
  - New logical partition is added in the middle and logical partitions are appropriately renumbered
  - Awkward consideration with utilities - range of parts - as it is based on physical partition numbers
- **Do not have to take care of adjacent partitions which possibly reach their space limit**
- **Once you determine the limit key for the new inserted partition, the procedure for handling "partition full" conditions is very easy to automate**
  - Add new partition
  - Run REORG against the new and adjacent partitions

System Managed Duplexing (SMD) of CF Lock Structure – Challenges

- **Required for highest availability in Db2 for z/OS data sharing environments**
  - Single and Multi-site z/OS Parallel Sysplex environments with no failure isolated CFs or external CFs
  - Without SMD, the failure of the ‘wrong CF’ may result in a group-wide outage
    - LOCK1 or SCA can only be dynamically rebuilt into an alternate CF if all the Db2 for z/OS members survive the failure
- **Existing synchronous SMD of LOCK1 structure can be expensive in terms of increased host CPU resource consumption, degraded application elapsed time performance, and aggravated global lock contention**
  - All types of requests are duplexed
  - Duplexed request can consume 3x-4x host CPU cost vs. simplex structure
  - Synchronous lock requests are converted to asynchronous requests to limit host CPU penalty
  - CF service times will increase which will elongate transaction response times and batch processing elapsed time, and possibly aggravate global lock contention
  - Performance impact will vary
    - Dependent on locking intensity of respective application workload
    - Stretched distance for Multi-site data sharing group

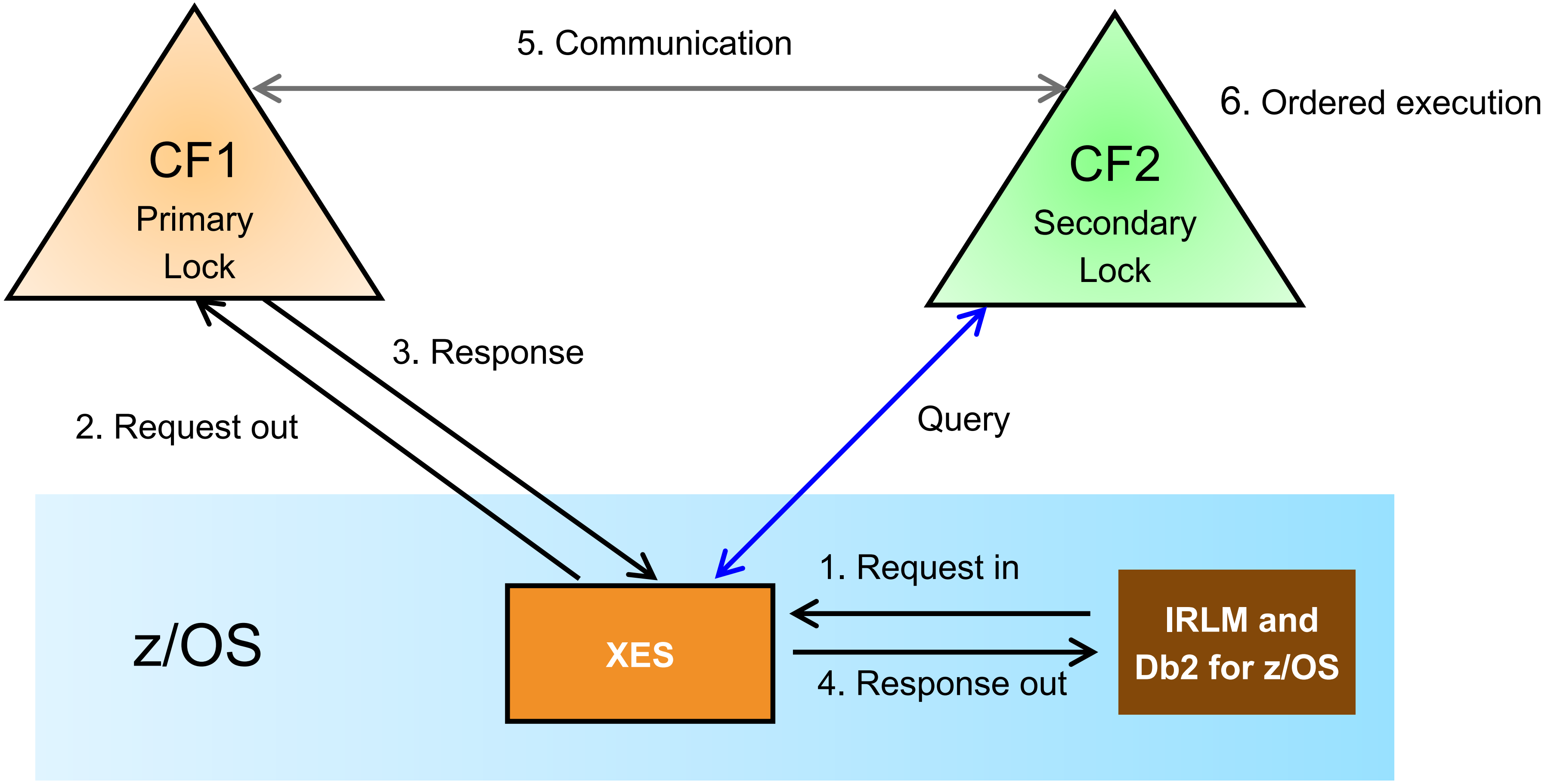
Synchronous CF lock structure duplexing – how it works today



Asynchronous CF Lock structure duplexing new in Db2 12 for z/OS

- **Reduces overhead for system managed duplexing of CF LOCK1 and SCA structures**
- **Secondary structure updates are performed asynchronously with respect to primary updates**
- **Db2 for z/OS will sync up with z/OS to ensure data integrity i.e., all modify locks have been "hardened" in the secondary lock structure before the corresponding undo/redo record for the update is written to the Db2 for z/OS active log on DASD**
- **The physical log writer performs the "synch" call to query the secondary, and it happens whenever log records get physically written to DASD which can be earlier than commit**
- **Increases the practical distance for multi-site sysplex operations while duplexing of CF LOCK1 structure**
- **Requirements:**
  - IRLM V2R3 Function Level 40 with PTFs
    - Db2 12 for z/OS FL=V12R1M100 with PTF for APAR PI66689
    - IRLM V2R3 with PTF for APAR PI68378
  - CFCC firmware support for CFLEVEL 21 Service Level 02.16 (z13)
  - z/OS V2R2 SPE with PTFs for APARs OA47796 and OA49148
  - CF to CF connectivity via coupling links

Asynchronous CF lock structure duplexing – how it works



Asynchronous CF Lock structure duplexing new in Db2 12 for z/OS …

- **Benefits**
  - Cost of lock structure duplexing is significantly lower than before
    - Host CPU for lock requests decreases
    - IRLMs receive responses sooner
  - Existing sites using synchronous SMD should see lower host CPU cost and better elapsed times
  - More environments can now achieve higher availability in all-ICF configurations with SMD
    - Reduce risk with asynchronous SMD and lower cost all round
      - ✓ Hardware maintenance
      - ✓ Capital cost for extra frames
  - Processor technology refresh applies to both host GCP and ICF engines
- **But it is not free**
  - Will have to acquire ICF engines and coupling links for CF to CF connectivity
  - CF Utilisation is significantly higher for async SMD relative to simplex case, but it is much less than sync SMD
    - Expected to be higher than simplex because there is simply more work for the CF to do

Asynchronous CF Lock structure duplexing new in Db2 12 for z/OS …

- **Performance Summary comparing async SMD relative to simplex**
  - ITR degraded by 13%
  - Response time and ETR are comparable
  - z/OS host CPU resource consumption is higher
  - CF CPU resource consumption is significantly higher

Setting initial STATISTICS PROFILE

- **It is important to clean up any (SYSCOLDIST) statistics that you do not intend to regularly collect before first BIND/REBIND, PREPARE or EXPLAIN after entry to Db2 12 for z/OS**
- **These statistics could be stale or inconsistent today because they are not being regularly collected**
- **Statistics profile is created on first BIND/REBIND/PREPARE/EXPLAIN after entry to Db2 12 for z/OS**
- **After the initial create, cannot tell from the subject statistics profile what statistics are the ones that were the older/inconsistent statistics**

Summary

- **Share lessons learned, surprises, pitfalls**
- **Provide hints and tips**
- **Address some myths**
- **Provide additional planning information**
- **Provide usage guidelines and positioning on new enhancements**
- **Help customers migrate as fast as possible, but safely**