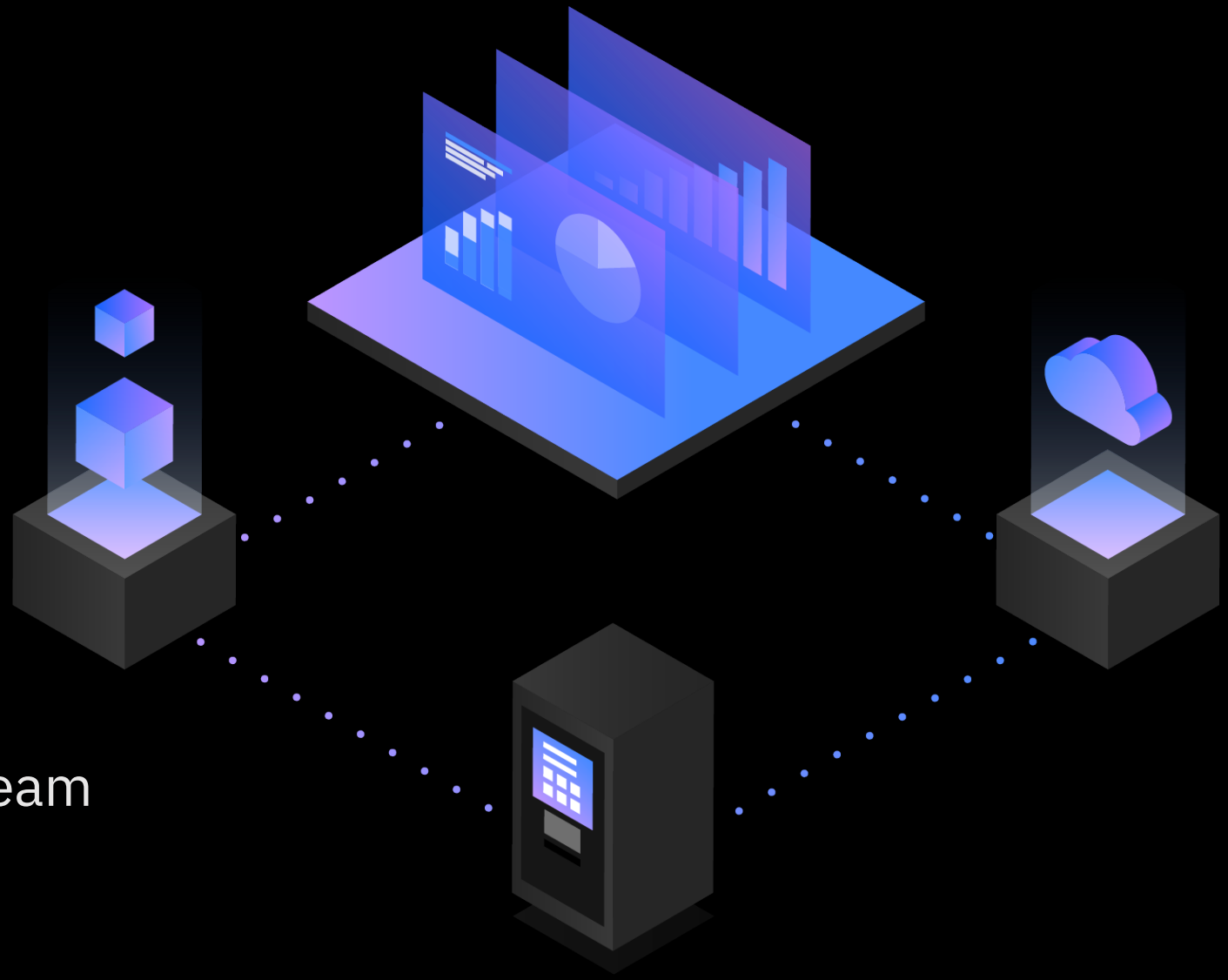
St. Louis Db2 User Groups

Data and AI

Db2 for z/OS Development SWAT: Db2 12 Function Level Overview

Anthony Ciabattoni
Executive IT Specialist
Db2 for z/OS Development SWAT Team



IBM

Agenda

- Db2 for z/OS into the future
- Db2 Continuous Delivery Review
- Summary of Catalog Changes
- Function Levels
- Questions

Disclaimer/Trademarks

THE INFORMATION CONTAINED IN THIS DOCUMENT HAS NOT BEEN SUBMITTED TO ANY FORMAL IBM TEST AND IS DISTRIBUTED AS IS. THE USE OF THIS INFORMATION OR THE IMPLEMENTATION OF ANY OF THESE TECHNIQUES IS A CUSTOMER RESPONSIBILITY AND DEPENDS ON THE CUSTOMER'S ABILITY TO EVALUATE AND INTEGRATE THEM INTO THE CUSTOMER'S OPERATIONAL ENVIRONMENT. WHILE IBM MAY HAVE REVIEWED EACH ITEM FOR ACCURACY IN A SPECIFIC SITUATION, THERE IS NO GUARANTEE THAT THE SAME OR SIMILAR RESULTS WILL BE OBTAINED ELSEWHERE. ANYONE ATTEMPTING TO ADAPT THESE TECHNIQUES TO THEIR OWN ENVIRONMENTS DO SO AT THEIR OWN RISK.

ANY PERFORMANCE DATA CONTAINED IN THIS DOCUMENT WERE DETERMINED IN VARIOUS CONTROLLED LABORATORY ENVIRONMENTS AND ARE FOR REFERENCE PURPOSES ONLY. CUSTOMERS SHOULD NOT ADAPT THESE PERFORMANCE NUMBERS TO THEIR OWN ENVIRONMENTS AS SYSTEM PERFORMANCE STANDARDS. THE RESULTS THAT MAY BE OBTAINED IN OTHER OPERATING ENVIRONMENTS MAY VARY SIGNIFICANTLY. USERS OF THIS DOCUMENT SHOULD VERIFY THE APPLICABLE DATA FOR THEIR SPECIFIC ENVIRONMENT.

Trademarks
IBM, the IBM logo, ibm.com, Db2, IBM Z and z/OS are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Db2 for z/OS Into the Future

Continuous delivery brings new functions to market 2 - 4 times per year instead of every 3 years

As of 4Q2020, eight function levels have been delivered, plus nearly 100's new capabilities since Db2 12 GA

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 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, 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
    - 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 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, not on a delivery date

New Db2 for z/OS strategy for delivering new function …

- **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 + 1 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 - cumulative (skip level possible)
    - Db2 Catalog changes that are needed for some FLs
  - Function level (FL) - needs to be activated - cumulative (skip level possible)
    - Introduces new Db2 features and functionality
    - No impact or change in existing application behavior
  - APPLCOMPAT level (AC) - set by application - provides an "island of stability" for a given application
    - AC level in BIND/REBIND of package must be <= FL and overrides zParm
    - AC must be advanced to exploit new SQL function
    - Even if FL is regressed to an earlier level *new function continues to be used by that application*
  - clientApplCompat level - driver configuration keyword allows remote apps access to new functions
- **Minimum starting point for Continuous Delivery is Db2 12 FL=V12R1M500**

 http://ibm.biz/db2z12functionlevels

Db2 12 Catalog Changes

- **Catalog Changes**
  - Not all function levels require a catalog changes
  - Function level catalog changes are typically very minor in scope
  - Strong Recommendation to wait a period of time after *"elevating"* APAR is applied before updating the catalog
    - Maintenance cannot be backed out after applying catalog changes
  - Methodology
    - Execute catalog changes individually
    - Execute changes cumulatively/"rolled up"
    - Designed to be restarted if needed
      - ✓ Test in highly volatile non-production environment to gain confidence in restarting
  - Catalog changes and function level activations
    - Can occur together
    - "prefetch" catalog changes
      - ✓ Activate function levels at a later date

V12R1M500 & V12R1M501

- **V12R1M500**
  - ***No Catalog changes***
  - Entry level into “New Function Mode”
  - Take advantage of new Db2 12 features
- **V12R1M501**
  - ***No Catalog changes***
  - Very slight new feature to “get feet wet” with Db2 12 Continuous Delivery
  - LISTAGG built-in function, which produces a list of all values in a group

V12R1M502 - Pervasive Encryption with Db2

- ***Catalog change needed***
- **Pervasive Encryption**
  - Db2 can transparently encrypt non-active log data at rest without database downtime or requiring the administrator to redefine objects which could cause disruption to operations
    - Db2 active logs, delete/redefine
    - Db2 archive logs will start to encrypted after key labels have been defined (archive logs written to DASD)
    - Db2 Catalog/Directory, Online Reorg
    - Db2 User Tablespaces, Online Reorg
  - Utilizes new z/OS DFSMS data set encryption support delivered in z/OS 2.3 and z/OS 2.2 (with OA50569 and OA53951)
    - Extended format only
  - Db2 12 V12R1M502 adds additional controls to set up encryption policies using Db2 interfaces
    - CREATE/ALTER with KEY LABEL at STOGROUP/TABLESPACE
  - Don’t forget to give appropriate access to for Db2 professions for stand along utilities
    - Additionally, check with your vendors to verify support
  - On z/14 and z/15 customers are seeing good performance results typically < 2% but mileage may vary

V12R1M503 – Db2 AI for z/OS, Enablement for replication of system-temporal tables

- ***Catalog change needed***
- **Db2 AI for z/OS**
  - Empowers the optimizer in the Db2 for z/OS engine to determine the best-performing query access paths based on workload characteristics using machine learning capabilities
    - Automates SQL regression detection and resolution
  - Streamlines exception analysis in System Assessment
    - Exception detection from Db2 statistics with recommended actions
  - Delivers Distributed Connection Control
    - Extends protection for Db2 for z/OS subsystems to control access from dynamic IP addresses in addition to the existing static IP address protection. Clients can set and monitor threshold by dynamic IP addresses
  - Initially introduced via a function level
  - Now available via maintenance stream, not dependent on function level
- **Enablement for replication of system-period temporal tables and generated expression columns**
  - A new built-in global variable, SYSIBMADM.REPLICATION_OVERRIDE, provides support needed to replicate system-period temporal tables or in tables that contain generated expression columns
  - The global variable is intended for use on the apply side of the replication process by products that enable replication of Db2 data

V12R1M504 - Prevent the creation of new deprecated objects

- ***No Catalog change needed***
- **Enabling function level V12R1M504 *'can prevent'* the creation of new depricated objects:**
  - Any packages used to execute DDL that is bound with APPLCOMPAT(V12R1M504) or higher will fail when attempting to perform DDL on a deprecated object
    - Synonyms
    - Segmented (non-UTS) tablespaces
    - Partitioned (non-UTS) tablespaces
- **CURRENT APPLICATION COMPATABITY special register can be used**
  - SET CURRENT APPLICATION COMPATABILTY='V12R1M503';
  - Remote applications can use Db2 System Profile tables to SET CURRENT APPLICATION COMPATABILITY='V12R1M503';
- **Customers need a regimented cook-book for creating new tablespaces**
  - UTS - PBG – Partitioned by Growth is a great replacement for the classic segmented tablespaces
  - UTS - PBR – Partitioned by Range is a natural evolution from moving from classic partitioned
  - UTS - PBR RPN – Partitioned by Range Relative Page Number for supersized UTS PBR
- **Customers need a plan to take-action for existing tablespaces**
  - Develop a plan and a strategy to migrate from classic segmented & table controlled partitioned non-UTS
  - Develop a plan and a strategy to correct previous incorrect decisions to move to UTS PBG
    - Enable partition level independence/parallelism & eliminate technical debt and protect against CPU burn

V12R1M504 - Universal Tablespace Strategy and Execution

- **General recommendations for use of UTS PBG tablespace**
  - Only use UTS PBG tablespace as the alternative and replacement for classic segmented tablespace
  - A table greater than ~64GB in size should be created as a UTS PBR tablespace
  - Good reasons to limit number of partitions - should have as few partitions as possible - ideally only 1
  - DSSIZE and SEGSIZE should be consistent with the target size of the object e.g.,
    - Small size object: DSSIZE = 2GB and SEGSIZE = 4
    - Medium size object: DSSIZE = 4GB and SEGSIZE = 32
    - Large size object: DSSIZE = 64GB and SEGSIZE = 64
  - REORG at the table space level unless do not have sufficient DASD space for sort
  - Setting system parameter REORG_DROP_PBG_PARTS = DISABLE?
    - If required to prune back the number of partitions
      - Use online system parameter to temporarily enable for controlled use
    - Better still, in Db2 12, use the DROP_PART YES option of REORG

V12R1M504 - Universal Tablespace Strategy and Execution …

- **Primary driver for the developing UTS PBG tablespace was the removal of the 64GB limit for classic segmented tablespace & avoid the disruptive migration to classic partitioned tablespace**
- **Some considerations**
  - All indexes are going to be NPIs
  - Limited partition independence for utilities (REORG, LOAD)
  - Partitioning not used for query parallelism
  - Degraded insert performance (free space search) as the number of partitions grow
  - REORG Considerations
    - REORG PART will fail for a full UTS PBG partition if FREEPAGE or PCTFREE are non-zero
    - Setting system parameter REORG_DROP_PBG_PARTS = ENABLE could lead to operational issues if the number of PARTs are pruned back
      - No point-in-time recovery prior to the REORG that prunes partitions
      - Cannot use DSN1COPY to move data between Db2 systems
- ***Should not be using UTS PBG as the design default for all tables (with large number of partitions)***

Hidden ROWID support to partition

- **ROWID can be used as a partition column**
- **Benefits**
  - Allows table to be partitioned where no natural
    partitioning key exists or candidate partitioning keys
    do not provide a good spread across partitions
  - Transparent to the application
- **Application impact if ROWID cannot be hidden**
  - APARs to support to define a hidden ROWIDs
    - PI76972, PI77310, PI77302 (Db2 12)
    - PI77718, PI77719, PI77360 (Db2 11)

```
CREATE TABLE  PRDA.ZJSCNTP0
      (  CLIENT    VARGRAPHIC(3) NOT NULL,
         WI_ID     VARGRAPHIC(12) NOT NULL,
         LENGTH    SMALLINT,
         DATA      VARCHAR(1000),
         ROW_ID    ROWID NOT NULL
         IMPLICITLY  HIDDEN generated always
      )  PARTITION BY (ROW_ID)
          (PARTITION 1 ENDING AT (X'0FFF'),
           PARTITION 2 ENDING AT (X'1FFF'),
           PARTITION 3 ENDING AT (X'2FFF'),
           PARTITION 4 ENDING AT (X'3FFF'),
           :
           PARTITION 14 ENDING AT (X'DFFF'),
           PARTITION 15 ENDING AT (X'EFFF'),
           PARTITION 16 ENDING AT (MAXVALUE))
```

V12R1M505 - REBIND PHASE IN

- ***Catalog change needed***
- **Problem statement:**
  - Pre V12 DBA cannot change bind options for packages in execution
    - SIX vs S package lock
    - End of UOW, held cursors, KEEPDYNAMIC(YES)
    - Work around: BIND COPY but difficult to switch to new collection ID
  - DBA cannot switch to previous access path at statement-level granularity for packages in execution
    - REBIND SWITCH works on whole package
- **Existing PLANMGNT(EXTENDED):**
  - CURRENT (copy ID 0) in SYSPACKAGE, SYSPACKDEP, SPT01
  - ORIGINAL (copy ID 2) in SYSPACKCOPY, SYSPACKDEP, SPT01
  - PREVIOUS (copy ID 1) in SYSPACKCOPY, SYSPACKDEP, SPT01

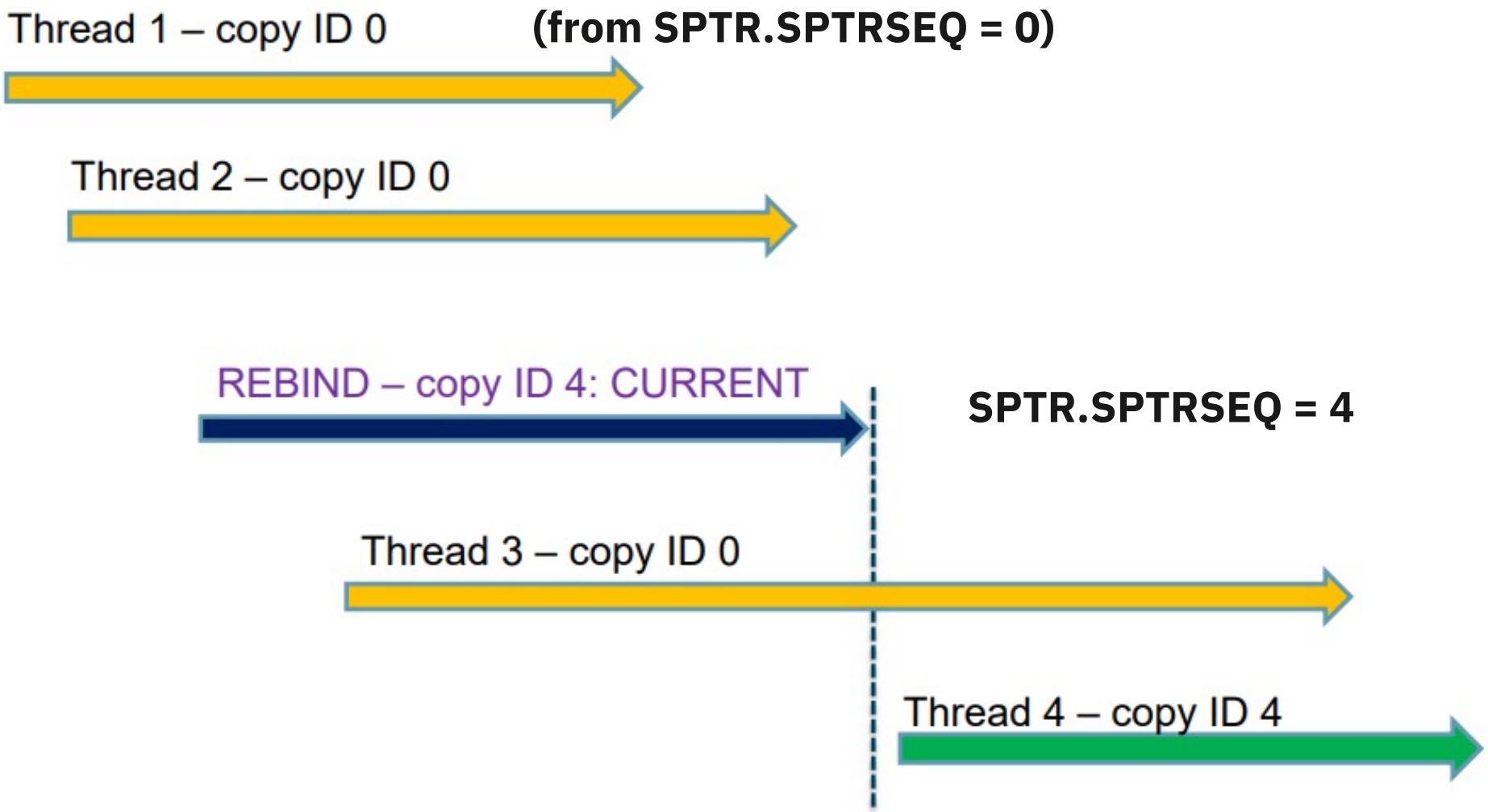
    * Note that SYSPACKSTMT only has the original rows for all those copies

V12R1M505 - REBIND PHASE IN ...

V12R1M505 enables rebinding a package concurrently with execution of the package and switching to previous access paths / runtime structures gradually without incurring an application outage:

- REBIND creates a new (soon to be) CURRENT copy (n) with the next copy ID (n+1)
  - Conditional SIX lock, then U package lock
  - ***Strong recommendation to apply APAR PH28693 which removes the "conditional SIX lock" on REBIND Phase In***
  - Copy ID 0, 4, 5, 6,.. , max=16, then wrap back to 0
  - Replicate copy ID n to copy ID 1/2 (previous/orig)
- Copy ID n can be used for execution at same time
- When copy ID n+1 rebind is committed, new threads executing the package will use this copy
  - Mark copy ID n as no longer current in SPT01
  - Mark copy ID n's PT as "old|do-not-use" in EDM pools
  - Move copy ID n to SYSPACKCOPY
- Reuse skeleton package table (SKPT) information
  - SPTR.SPTRSEQ column to indicate CURRENT copy ID

V12R1M505 - REBIND PHASE IN ...



V12R1M505 - REBIND PHASE IN ...

- **Max copyID = 16**
  - If all in use/allocated by threads
  - if all in use/allocted by threads, then REBIND fails (RC00E30307)
- **Daemon (once/day) checks for no more threads using non-current copy IDs and deletes it**
  - Conditional SIX lock (to encourage RELEASE(DEALLOC) break-in
  - If fails, compare bindtime to lowest allocation time among active threads
- **Old, do-not-use copies in EDM pool can be removed when referenced count = 0 or LRUed**
- **AUTO BIND always works on CURRENT copy (SPTR.SPTRSEQ)**
- **EXPLAIN PACKAGE COPY(CURRENT) works on CURRENT copy (SPTR.SPTRSEQ)**
- **QPAC_COPYID in IFCID239 stores copy-id for package execution**

V12R1M505 - REBIND PHASE IN ...

- **REBIND Switch**
  - Replicate copy ID n+1 from 1 (previous) or 2 (orig), CURRENT=Y
  - Set CURRENT=N for copy ID n
    - Set do-not-use flag for n in EDM pools
  - Replicate copy ID 1 or 2 from if PLANMNT(EXTENDED|BASIC)
- **Catalog change**
  - SYSPACKDEP.DTYPE has a new values for copy IDs 3-16
  - Package lock name change: optional
    - Copy ID as part of lock name

V12R1M506 – Alternative BIF Syntax, Addition DROP TABLE functionality

- ***No Catalog change needed***
- **BIF Syntax**
  - Compatible with other DBMSs
  - Simplify porting of applications to Db2 for z/OS and leverages the z/OS platform for growth
- **DROP TABLE automatically drops underlying explicitly created Universal Table Space or LOB Table Space**
  - Prior to function level the drop was disallowed on explicitly created tablespaces
    - Only a single drop is needed previously it was a multi-step process

Use Cases	Current Behavior	New Behavior
Drop of a table residing in an explicitly created UTS table space	DROP TABLE of the base table is not allowed (SQLCODE -669). DROP TABLESPACE required	Table is dropped, UTS table space is dropped
Drop of a ***system-period temporal*** or ***archive-enabled*** table where the ***history*** or ***archive*** table's containing table space is explicitly created UTS table space	DROP TABLE of the base table is not allowed (SQLCODE -669). DROP TABLESPACE of the history, archive table's table space first, then proceed to drop the base table	The drop of the base table is not blocked. The history/archive table and its containing table space are automatically dropped
Drop of an auxiliary table via a DDL statement (DROP TABLE of an auxiliary, a base or ALTER TABLE DROP COLUMN of a LOB column)	LOB table space remains	Auxiliary table is dropped, LOB table space is dropped
REORG TABLESPACE of a PBG UTS that removes empty partition where the table has a LOB column and the associated LOB table space is explicitly created	LOB table space remains	Empty partition is removed, associated LOB table space is dropped

V12R1M507 – Dynamic Profile Support

- ***Catalog change needed***
- **Use Profile Support**
  - *SYSIBM.SYSSTATFEEDBACK*
  - When STATFDBK_SCOPE= ALL/STATIC/DYNAMIC in Db2 ZPARMs
    - During every BIND/REBIND/PREPARE the optimizer looks if current statistics
      - ✓ To generate sufficient candidates for access path selection
      - ✓ If any discrepancy is detected, SYSSTATFEEDBACK catalog table is updated with recommendation for a given object
    - Db2 places information about statistics seen by the optimizer, in the course of generating query access plans, as being absent or inconsistent (regarding the latter, consider that one can insert statistical information into catalog tables on one's own, perhaps causing an inconsistency - a value in one column that should reflect the value in another column, but doesn’t)

V12R1M507 – Dynamic Profile Support …

- **Use Profile Support …**
  - *SYSIBM.SYSTABLES_PROFILES*
    - STATFDBK_PROFILE = YES in Db2 ZPARMS
    - Profile is populated based on ***current statistics*** and data collected in the SYSIBM.SYSSTATFEEDBACK table
      - ✓ Obsolete or “dead” statistics if present will be used to create a profile
      - ✓ Strongly recommend cleaning up and removing obsolete, “dead” statistics prior to enabling profile support in V12R1M507
  - When profile support is turned on, RUNSTATS statistics will be generated based on information *SYSIBM.SYSTABLES_PROFILE*
  - As a defensive measure prior to using dynamic profile support backup current statistics and prepare to repopulate if needed

V12R1M508 - Migration of Multi-Table TS to PBG UTS

- ***No Catalog change needed***
- **V12R1M508 enables the ability to migrate multi-table tablespace to a PBG UTS via a pending Db2 alter and a materializing Db2 Online REORG:**
  - Create new target PBG UTS target tablespaces
    - New target tablespace needs to be created in advance
    - CREATE TABLESPACE NEWTS1 IN DB1 MAXPARTIONS 1 DEFINE NO;
    - CREATE TABLESPACE NEWTS2 IN DB1 MAXPARTIONS 1 DEFINE NO;
      - ✓DEFINE NO
      - ✓MAXPARTITIONS 1
      - ✓LOGGED/NOT LOGGEWD attribute needs to match source TS
      - ✓CCSID values need to match source TS
  - Alter existing multi-table tablespace
    - ALTER TABLESPACE DB1.TS1 MOVE TABLE TB1 to TABLESPACE DB1.NEWTS1;
    - ALTER TABLESPACE DB1.TS1 MOVE TABLE TB2 to TABLESPACE DB1.NEWTS2;
  - Reorg existing ”alter pending” tablespace (all tables moved)
    - For each table moved:
      - ✓Linear increase time required to perform ALTER statement & REORG SWITCH
  - Drop “old” tablespace that previously was the multi-table, tablespace
    - Only after all tables have been moved to target and not tables exist in tablespace

V12R1M508 - Migration of Multi-Table TS to PBG UTS ...

- **Considerations:**
  - Increased number of open Db2 datasets
  - Sequential inline image copy is allocated for each target TS
  - Packages will be invalidated after materializing REORG
  - Inline statistics will not be gathered for either the source or target tablespaces during the REORG, and existing statistics for source tablespace will not be carried over to the target tablespaces
  - Existing table level statistics will continue to persist and will not be touched by the REORG
    - Minimal risk of access path regression for the resultant REBINDs or AUTO BINDS of packages invalidated for the moved tables
    - Execute a standalone RUNSTATS after REORG has moved objects from multi-table tablespaces to PBG UTS
    - REBIND with APREUSE(WARM) to minimize access path change from REBIND of the invalidated packages
    - AUTO BIND will also issue APREUSE(WARN)
      - ✓ If AUTO BIND is enabled (requires APAR PH15896)
    - Execute a standalone RUNSTATS after REORG has moved objects from multi-table table spaces to PBG UTSs

V12R1M508 - Migration of Multi-Table TS to PBG UTS …

- **Catalog Externals**
  - After the ALTER is materialized, the follow Catalog fields are updated for the moved table's tablespace info:
    - SYSTABLES.TSNAME = target TS name
    - Target SYSTABLESPACE.OLDEST_VERSION = SYSTABLES.VERSION
    - Target SYSTABLESPACE.CURRENT_VERSTION = SYSTABLES.VERSION
    - Target SYSTABLEPART.OLDEST_VERSION – SYSTABLES.VERSION
    - Source SYSTABLESPACE.NTABLES = decrement by # of tables moved out
    - Source SYSTABLESPACE.STATUS = 'T' (incomplete) if moving last table out
    - Note: Source TS version # is left as is in SYSTABLESPACE.CURRENT_VERSION (same behavior as DROP TABLE today)
    - During the materializing REORG, new SYSCOPY records are inserted as follows:

ICTYPE	STYPE	TTYPE	Explanation
W	M		REORG moved one or more tables out
A	T	M	ALTER TABLESPACE MOVE was materialized
F	W		Inline image copy created by REORG

    - For source tablespace:

ICTYPE	STYPE	DSNAME	Explanation
W	M	Source dbname.tsname	REORG moved a table in from source table space
C	L (Logged) O (Not Logged)	Target dbname.tsname	Data set created by REORG for the target table space
F	W	Image copy data set name	Inline image copy created by REORG

V12R1M509 (V12R1M504) - Huffman Compression

- ***Catalog change needed***
- **Huffman Compression introduced in Db2 12 Function Level 504**
  - Just like Lempel/Ziv, Huffman is a dictionary-driven compression and decompression algorithm
    - Fixed Length vs Huffman compression chosen at ZPARM
  - The entries in the dictionary are sorted by frequency of occurrence with the highest frequency get a shorter bit pattern to identify the entry in the dictionary
  - Prerequisites
    - z14 for the improved CMPSC instruction
    - Function Level 504 of Db2 12
  - No dependency on the Integrated Accelerator for zEDC technology on the z15
  - Seen up to 40% (avg. 20-30%) improvement compared to legacy fixed length compression
  - Wide range of variability in terms of CPU and elapsed time performance (+ / -)
  - Some cases may see CPU and elapsed time reduction e.g., sequential processing

V12R1M509 (V12R1M504) - Huffman Compression ...

- **Huffman Compression introduced in Db2 12 Function Level 509**
  - Huffman compression can now be used at a tablespace/partition granularity level
  - DSN1COMP enhancements made for Huffman
  - New columns added to the Db2 Catalog
    - SYSIBM.SYSTABLESPACE.COMPRESS
    - SYSIBM.SYSTABLPART.COMPRESS
    - SYSIBM.SYSTABLEPART.COMPRESSED_USED
  - The use of Huffman compression should be evaluated object by object
- **Recommendations**
  - Do not implement Huffman compression until the following is in place:
    - Applying regular scheduled drops of preventative service for Db2
    - Db2 provides object level control (Function Level 509)
  - Implement in a controlled and incremental rollout

V12R1M510 – Head Start to Migrate to Db2 VNext

- ***No Catalog change needed***
- **Db2 for z/OS Development has fully transitioned to a continuous delivery model**
  - Most new capabilities for Db2 for z/OS in the service stream
  - There will be changes that will require a new Db2 release
  - IBM has not, yet announced the name, content, or availability dates of Db2 VNext
  - Function level V12R1M509 is the last function to deliver new capabilities in Db2 12
- **V12R1M510 Does not introduce any new features/capabilities**

V12R1M510 – Head Start to Migrate to Db2 VNext …

- **Verifies the preparation into Db2 VNext**
  - Essential pre-migration tasks have been completed
  - All Db2 12 function level are activated
  - All catalog updates for Db2 12 are applied
    - Db2 catalog is ready for the CATMAINT for Db2 VNext catalog VnnR1M100
  - All application packages used within the last 18 months are rebound in an appropriate Db2 release (Db2 11 or later) to ensure that they are supported by the next Db2 release
  - Essentially, activation of V12R1M510 fails if any of these conditions are not met, and activation of V12R1M510510 will be required before you can migrate to Db2 VNext
- **APAR PH33727**
  - Does not include the fallback SPE for migration to the next Db2 release
  - V12R1M510 does not verify that the fallback SPE is installed
  - Fallback SPE is expected be delivered by a future APAR

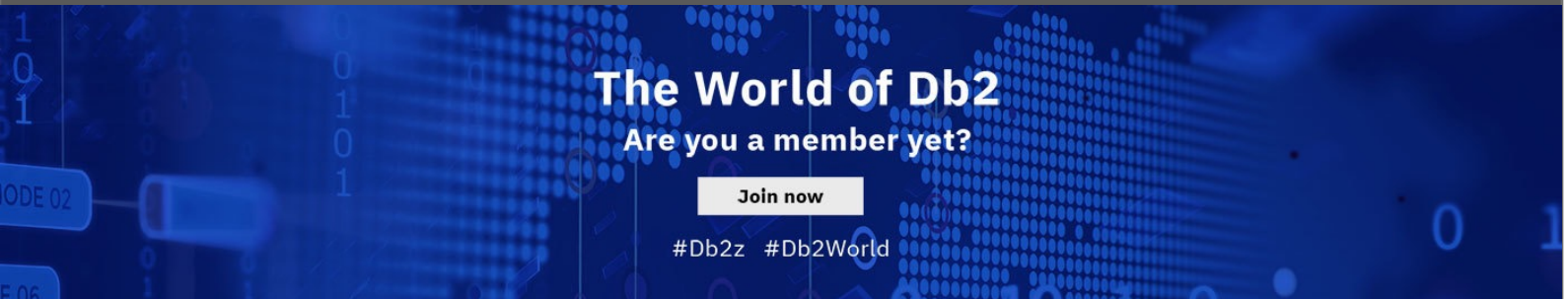
Db2 12 Migration resources

- **John Campbell's webcasts**
  http://ibm.biz/Db212TechnicalRoadshow
  - Db2 12 Technical Overview Parts 1 & 2
  - V12 Migration Planning and experiences
- **Db2 Master Class**
  - Held twice a year in June and September
  - EMEA - June 21, 2021 (9:00am London)
  - United States - September 27, 2021 (8:30am EST)
  - Held as virtual classes in 2021 because of COVID-19

  https://ibm.biz/Db2ZMasterClass2021

- **Join the World of Db2 for additional webcasts and materials**

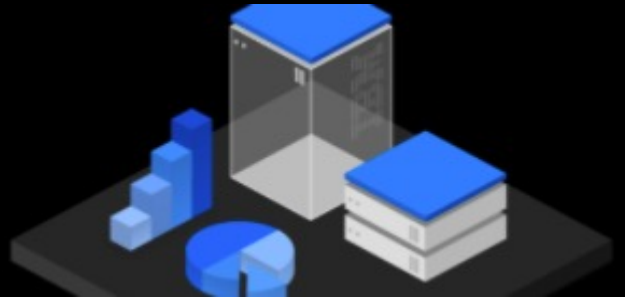
  www.worldofdb2.com



"It's all about robustness"

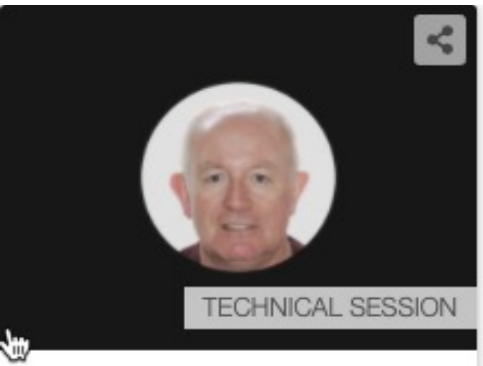
IBM Db2 for z/OS

An AI infused enterprise data server for Hybrid Cloud and business critical transactions



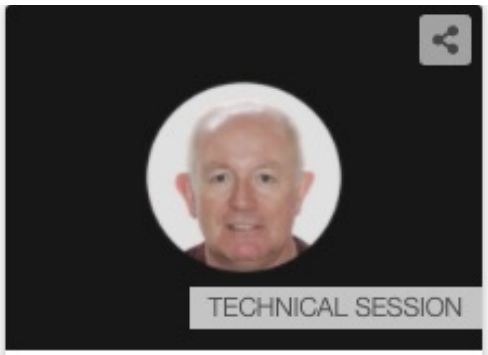
Get a head start for 2021

IBM



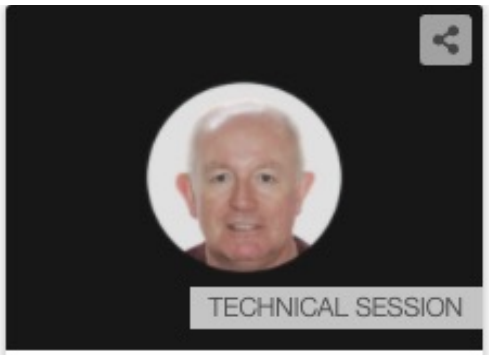
On Demand

Session 7. Primer on Insert Algorithm 2 with Db2 12



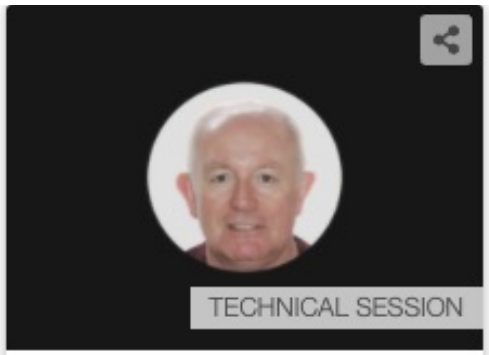
On Demand

Session 6. Database Housekeeping - recommended best practice for generating REORGs



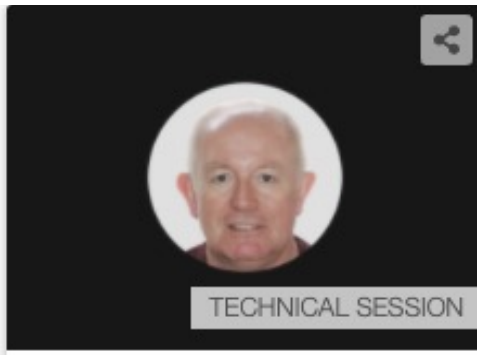
On Demand

Session 5. Point In Time Recovery in Db2 12 with RECOVER and SCOPE UPDATED



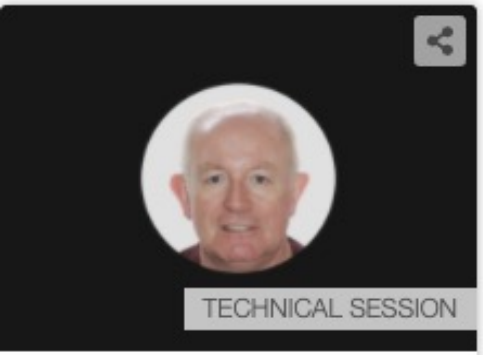
On Demand

Session 4. Saving CPU: Using thread reuse and BIND option RELEASE DEALLOCATE



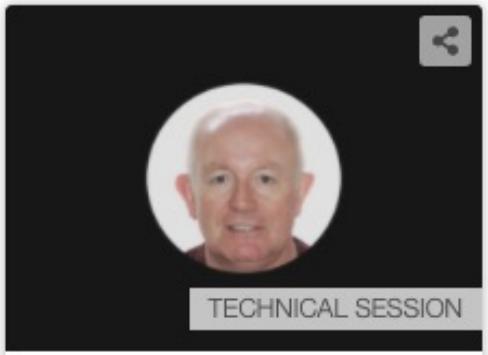
On Demand

Session 3. Latest news on in-memory performance optimization (FTB) in Db2 12 for z/OS



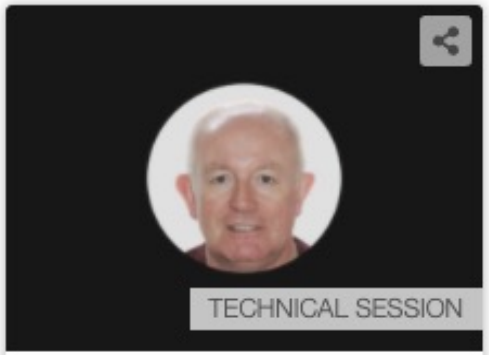
On Demand

Session 2. Dos and Dont's about Continuous Delivery



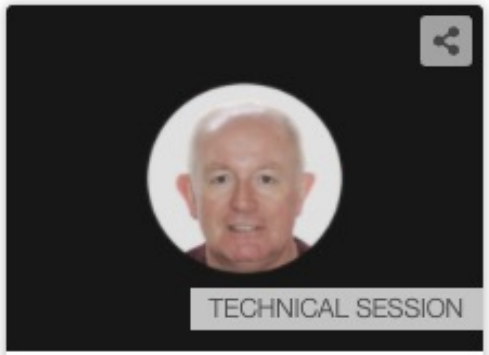
On Demand

Session 1. Planning your Db2 12 for z/OS migration strategy



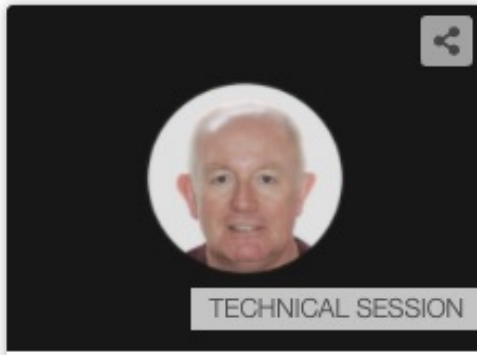
On Demand

Db2 for z.OS: Hot Topics and Best Practices with John Campbell PART 2



On Demand

Db2 for z/OS: Hot Topics and Best Practices with John Campbell PART 1



On Demand

Db2 for z/OS: Trends & Directions and Latest Updates

Questions

Thank You