Logistics and Questions

- **The presentation is available for download from the resources list**

- **The webcasts will be available on replay**

- **You can submit questions by typing into the questions area of your webcast control panel at least 15 minutes before the end of the webcast.**

- **Questions will be answered as time permits**

- **Any questions not answered due to time constraints will be answered after the webcast and added to the "Resources list"**

Disclaimer/Trademarks

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

DB2 12

Objectives

- **Introduce and discuss the new features of DB2 12 for z/OS**
- **Provide planning information for migration**
- **Understand the new performance features**

Agenda

- **Introduction**
- **Performance focus – traditional workloads**
- **Performance focus – enabling modern applications**
- **Migration**
- Application enablement
- Reliability, availability, scalability, security

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

INTRODUCTION

Goals

Application Enablement	• IDAA improvements to expand to new use cases • SQL/SQL PL improvements for next wave of applications	• Address key customer requirements to expand use of existing features • Mobile, hybrid cloud, and DevOps enablement
DBA Function	• Relieve table scalability limits • Simplify large table management	• Remove biggest 24x7 inhibitors • Security and compliance improvements
OLTP Performance	• 5-10% CPU reduction with use of in-memory features • 2x increase in Insert throughput for non-clustered	• Remove system scaling bottlenecks for high n-way systems • Serviceability, availability
Query Performance	• 20-30% CPU reduction for query workloads • Improve efficiency by reducing other resource consumption	• 80% UNION ALL performance improvement • Simplify access path management

Quick Hits

- **Scale and speed for the next era of mobile applications**
  - Over 11 Million Inserts per second measured when hitting "sweet spot"
  - 6 trillion rows in a single table, with agile partition technology
- **In Memory database**
  - Up to 23% CPU reduction for index lookup with advanced in-memory techniques
- **Next Generation application support**
  - 360 million transactions per hour through RESTful web API
- **Deliver analytical insights faster**
  - Up to 25% CPU saving for traditional query workloads
  - Up to 2x speed up for modern (complex OLTP or real-time analytics) workloads

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

PERFORMANCE FOCUS
TRADITIONAL WORKLOADS

Performance Enhancements

- **In-memory contiguous buffer pools**
  - Direct page access in-memory, greatly reduced GetPage overhead
  - Hash and LRU chains not maintained
  - Up to 8% CPU reduction measured for OLTP
  - PGSTEAL(NONE) – improved in DB2 12 to avoid LRU and hash chain management overheads
  - Overflow area (10% of VPSIZE, max of 6400 buffers)
    - Used in case objects do not fit
    - Automatically managed by DB2 using FIFO steal algorithm
    - Allocated when buffer pool is allocated, but only backed when used
- **In-memory index for fast traversal (see next slide)**
- **More granular Global Commit LSN and Read LSN**
  - Potential huge improvement in lock avoidance (data sharing)
  - Help space reuse for LOB insert

In-Memory Index Optimization

- **A new Index Fast Traverse Block (FTB) is introduced**
  - Memory optimized structure for fast index lookups
  - Resides in memory areas outside of the buffer pool
    - New zparm INDEX_MEMORY_CONTROL
    - Default=AUTO (min of 500 MB or 20% of allocated buffer pool storage)
  - UNIQUE indexes only, key size 64 bytes or less
- **DB2 automatically determines which indexes would benefit from FTB**
- **DISPLAY STATS command shows which indexes are using FTBs**
- **New SYSINDEXCONTROL catalog table**
  - Specify time windows to control use of FTBs for an index
- **New IFCIDs 389 and 477 to track FTB usage**

Simple Index Look-up: Faster & Cheaper

- **Up to 23% CPU reduction for index look up using DB2 12 In-memory index tree**

CPU Improvement (%) from Simple Lookup in DB2 12

6%

11%

16%

23%

2

3

4

5

Index Levels

Performance Enhancements ...

- **Avoid scheduling unnecessary prefetch**
  - Problem: when all pages are in memory, dynamic prefetch needlessly schedules prefetch
  - Wastes CPU, can cause "out of prefetch engine" condition
  - Attempts to solve this in the past failed - solved in DB2 12
  - Up to 6.8% CPU reduction for OLTP, 4.5% for query

Performance Enhancements ...

- **INSERT Algorithm 2 (see next slides)**

- **RLF control for static packages**

- **DB2 / DASD synergy enhancements – retrofit to V10, V11**
  - Exploit z/OS HyperWrite (PPRC log write accelerator)
    - Up to 30% log write latency reduction
  - Improved Integration with DS8870 Easy Tier multi-temperature management
    - Avoid I/O degradation after REORG

# INSERT Algorithm 2 Performance

- **Insert workloads are amongst the most prevalent and performance critical**

- **Performance bottleneck will vary across different insert workloads**

  - Index maintenance?

  - Log write I/O?

  - Space search (page p-lock, page latch contention)

  - PPRC disk mirroring

  - Network latency

  - etc

INSERT Algorithm 2 Performance …

- **DB2 12 may potentially deliver significant improvement for non-clustered insert (e.g., journal table pattern where both <u>concurrent</u> insert activity and <u>space search is the constraint</u> on overall insert throughput**

- **Applies to UTS with MEMBER CLUSTER (both with/without APPEND)**

- **Implemented advanced new insert algorithm to streamline space search**
  - Default is to use the new fast algorithm for qualifying table spaces
    - INSERT ALGORITHM zparm can change the default
    - INSERT ALGORITHM table space attribute can override zparm

- **Your mileage will vary**
  - Some insert workloads will see no improvement
  - Some specific insert workloads may see significant improvement

- **Will shift the bottleneck to the next constraining factor**

INSERT Algorithm 2 Performance – Shifting The Bottleneck …
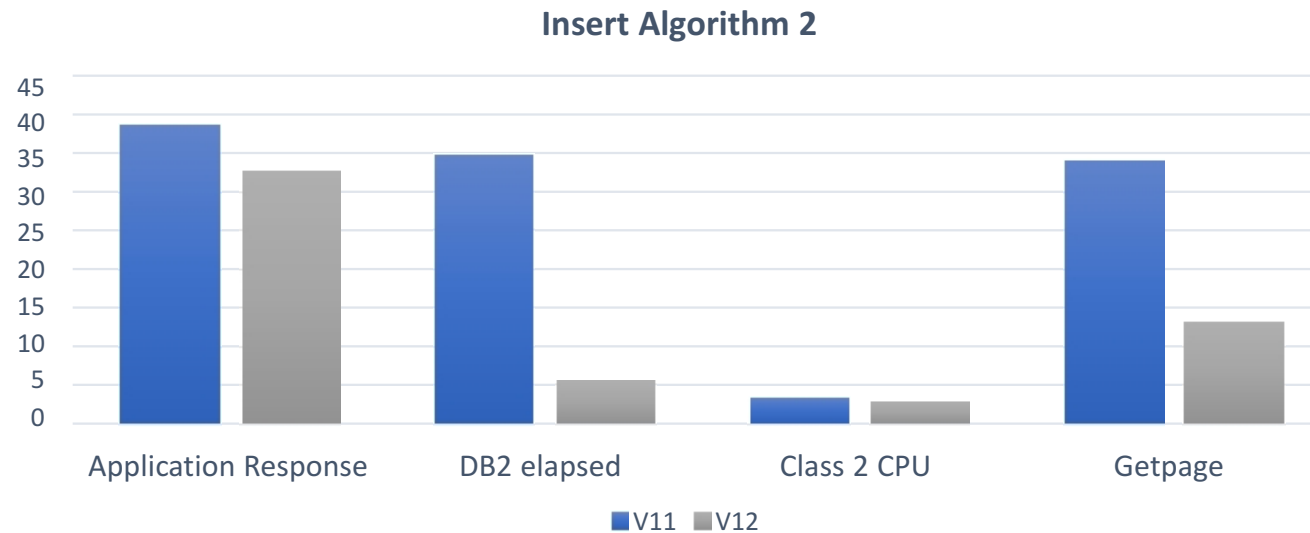
Insert Algorithm 2

Chart showing V11 (blue) and V12 (gray) values:
- Application Response: V11 ≈ 39, V12 ≈ 33
- DB2 elapsed: V11 ≈ 35, V12 ≈ 5.5
- Class 2 CPU: V11 ≈ 3, V12 ≈ 2.5
- Getpage: V11 ≈ 34, V12 ≈ 13

Legend: V11 V12

INSERT Algorithm 2 Performance - DB2 11 PMR Recreate …

Throughput (insert/sec): 2708, 65417 (24x)

TOTAL CPU per commit (us): 9453, 146 (1/64)

■ 11 ■ 12
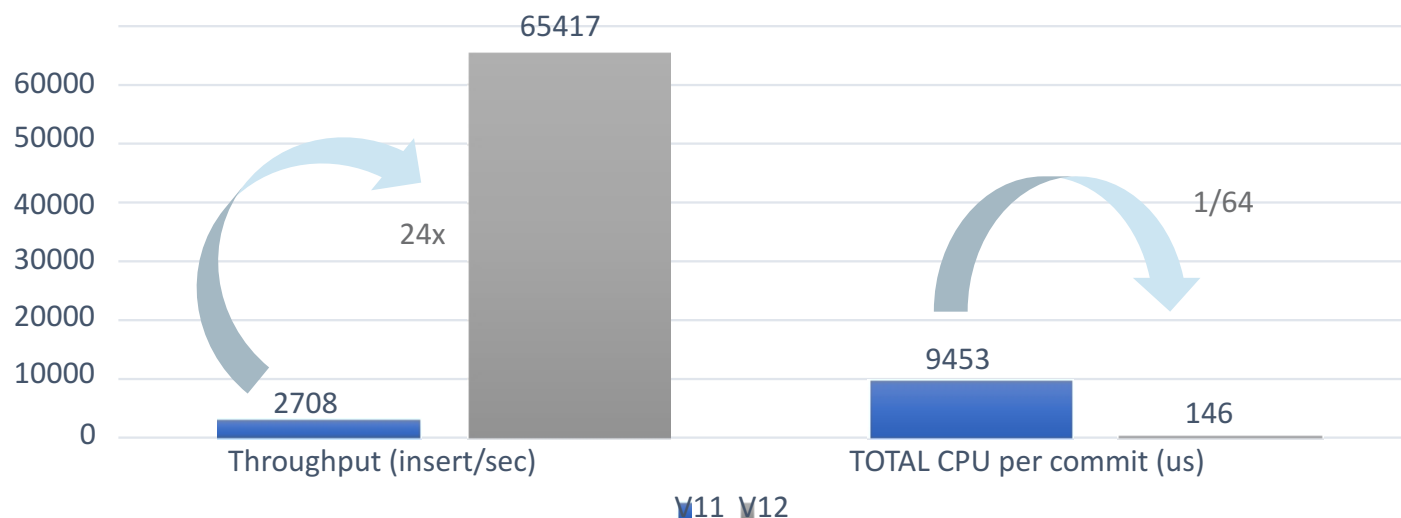
UTS PBG with Member Cluster, RLL, with 400 bytes per row, one index,
800 concurrent threads, 10 insert per commit

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

DB2 12

18

Performance Enhancements ...

- **Buffer Pool advisory mode to simulate larger buffer pools – rollback to V11**
  - Low CPU and real memory overhead
  - Statistics provided to indicate I/O savings
  - Retrofit to V11
- **Streamlined Claim/Declaim processing**
  - Avoid re-claim overhead across multiple commit scopes serially reusing persistent thread running RELEASE(DEALLOCATE)
  - Online REORG and other drainers can still break in

Other Performance Enhancements ...

- **Improvements in pool management to simplify and remove the scalability inhibitors**
  - EDM pools
  - LOB/XML storage
- **Other Improvements**
  - Reducing DGTT DECLARE overhead
  - Remove log force write from identity column and sequence caching in data sharing

System Scaling Enhancements

- **Large n-way scaling**
  - Improved efficiency on LPARs with high number of CPs
  - Log latch contention reduction: testing shows up to 41% CPU reduction and 6% throughput improvement for high contention cases
  - Buffer Pool scaling improvements:
    - LC23 reduction, PLO avoidance
    - 5-30% CPU improvement when accessing hot pages
  - IRLM latch contention reduction
  - EDM DBD and skeleton pool scalability improvements
- **Optimizations for new hardware**
  - Exploitation of z13 decompression enhancement
  - Internal structure changes for cache efficiency, more processor prefetch
- **Raise total buffer pool size limit to 16 TB**
- **>4G size active log datasets**

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

High level performance expectation

- **System and OLTP performance**
  - 2-3% CPU reduction without Index In-Memory feature
  - 5-10% CPU reduction by exploiting Index In-Memory feature
  - Further reduction is possible with contiguous buffer pools,   and / or persistent RELEASE(DEALLOCATE)
- **Query performance**
  - Wide range of improvement
    - Typically 0-20% without new access path
    - Typically 10-40% with new access path
    - Up to 90% reduction is observed in our evaluations
- **Concurrent insert against UTS and MEMBER CLUSTER**
  - 5-10% CPU reduction
  - Throughput improvement if current bottleneck is space search or page contentions

Instrumentation Enhancements

- **More granular wait times for IFCIDs 316 (dynamic) and 401 (static)**
  - Accumulated wait time due to global contention for locks (broken out by type)
- **Enhance IFCIDs 53/58 statement level section for PREPARE**
  - Similar to INSERT/UPDATE/DELETE
- **Enhance SQL performance tracing adding RDI Section Number in IFCIDs 53/58**
- **Add batch job STEP name in correlation header**
- **Add REFRESH TABLE to counts in DSNDQXST (SQL data section)**
- **Add workfile, tempfile usage information to Accounting trace**
- **Enhanced precision for IFCID 199 dataset I/O stats - microseconds**

Access path (plan) stability

- **Dynamic SQL plan stability**

- **Static plan stability usability**

- **Preserve local dynamic statement cache at rollback**

- **Integrated RUNSTATS with optimizer**
  - Automated update of statistics profiles by optimizer

- **Statistics profile support**
  - Automated update for Index DDL
  - Inline stats usage of profiles

- **Simplify creation of all tables required by Explain**
  - New ADMIN_EXPLAIN_MAINT stored procedure

Dynamic SQL Plan Stability

- **Problem:**
  - Unstable performance of repeating dynamic SQL statements
  - Environmental changes can result in change in access path or performance regression, and this can be tough to manage
    - RUNSTATS
    - Applying software maintenance
    - DB2 release migration
    - zparm changes
    - Schema changes
- **Static SQL has several advantages**
  - Access path established at BIND time
  - Static plan management gives advanced management functions
- **Objective: extend static SQL advantages to dynamic SQL**

Dynamic SQL Plan Stability ...

- **Base infrastructure**
  - Opaque parameter CACHEDYN_STABILIZATION
  - Capture via -START DYNQUERYCAPTURE STBLGRP(APP1) ...
    - Command with / without monitoring
    - Global variable
  - FREE
  - EXPLAIN (current, invalid)
  - Invalidation
  - LASTUSED (identify stale statements)
  - Instrumentation (query hash, explain, cache + catalog hit ratio)
  - APPLCOMPAT is part of matching criteria
- **Save query and cache structures to catalog for subsequent reuse on cache mis**

Dynamic SQL Plan Stability ...

- **Key limitations**
    - Literal concentration and temporal stabilization not currently included
    - REBIND support not included
        - No PLANMGMT/SWITCH/APREUSE

Static Plan Stability: Usability

- **BIND REPLACE of same version to keep LASTUSED**

- **FREE PACKAGE improvements**
  - Selectively FREE either original or previous
  - Choose to FREE if invalid only
  - Allow FREE of inactive package copies while application is running

- **REBIND PACKAGE improvements**
  - SWITCH to include APREUSESOURCE option so as to choose ORIGINAL or PREVIOUS copy as source for APREUSE
  - Addresses issue of switch accidentally causing invalid copy to become current
  - APREUSESOURCE option avoids the two step process of
   - REBIND SWITCH followed by
    - REBIND APREUSE

RUNSTATS Enhancements for SQL Performance

- **RUNSTATS (for access path selection benefit)**
  - CLUSTERRATIO formula improvements
  - Statement cache invalidation
    - Optional for RUNSTATS (new default)
    - For other utilities ONLY if objects in pending state before utility executed
  - Profile support for inline statistics
  - Automated COUNT for FREQVAL
    - Allow DB2 to collect the skewed values
      - Up to top 100 or until no skew for remaining values
  - Optimizer to automatically update statistics PROFILE with RUNSTATS recommendations
    - DSNACCOX to recommend RUNSTATS after profile update
      - Specify USE PROFILE on RUNSTATS to collect current statistics recommendations

Enhanced statistics profile management
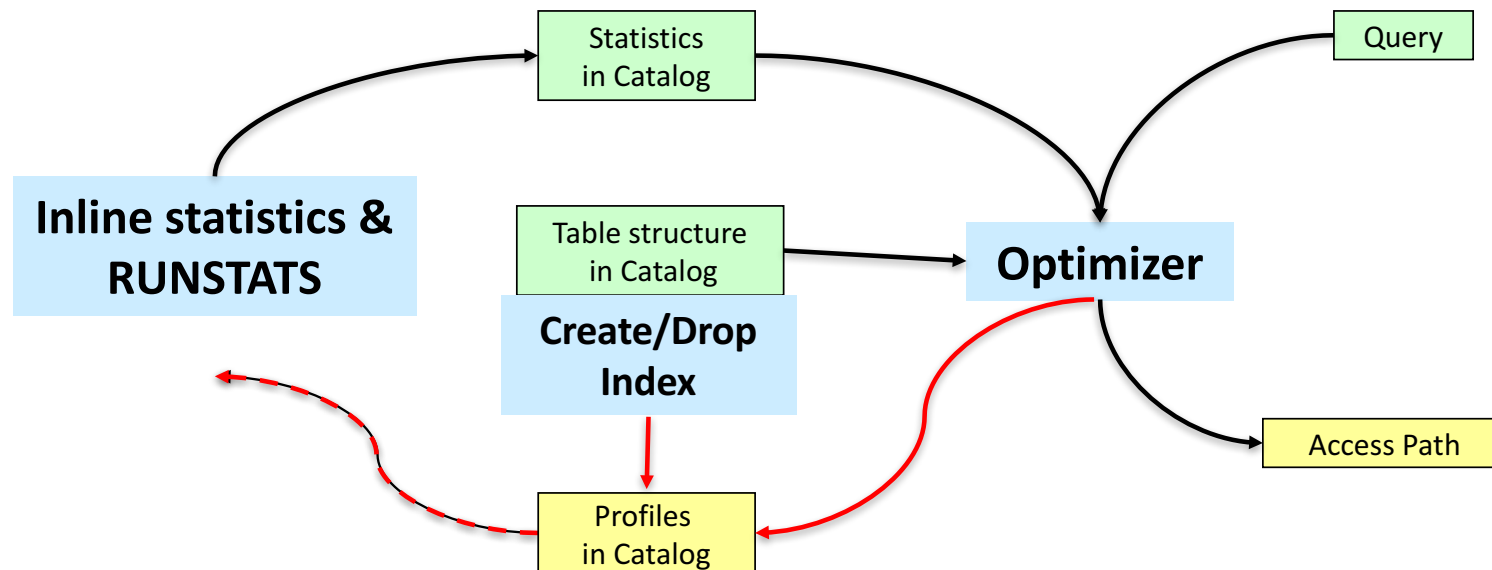

Inline statistics & RUNSTATS

Statistics in Catalog

Table structure in Catalog

Create/Drop Index

Profiles in Catalog

Optimizer

Query

Access Path

All new with V12

PERFORMANCE FOCUS
ENABLING NEW APPLICATIONS

Query Performance Enhancements

- **Up to 25% CPU improvement for traditional query workloads**

- **Up to 2x improvement for modern SQL applications**
  - Performance improvements for next generation SAP applications
  - Real-time analytics, complex OLTP workloads

- **100% zIIP eligibility for parallel query child tasks**

- **Modern applications contain more complex SQL patterns (targeted in DB2 12), and more sorting, joins, stage 2 predicates etc.**
  - These complex patterns are less common in traditional OLTP/batch

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

Query Workload CPU and Elapsed time Improvement from V11 (%)
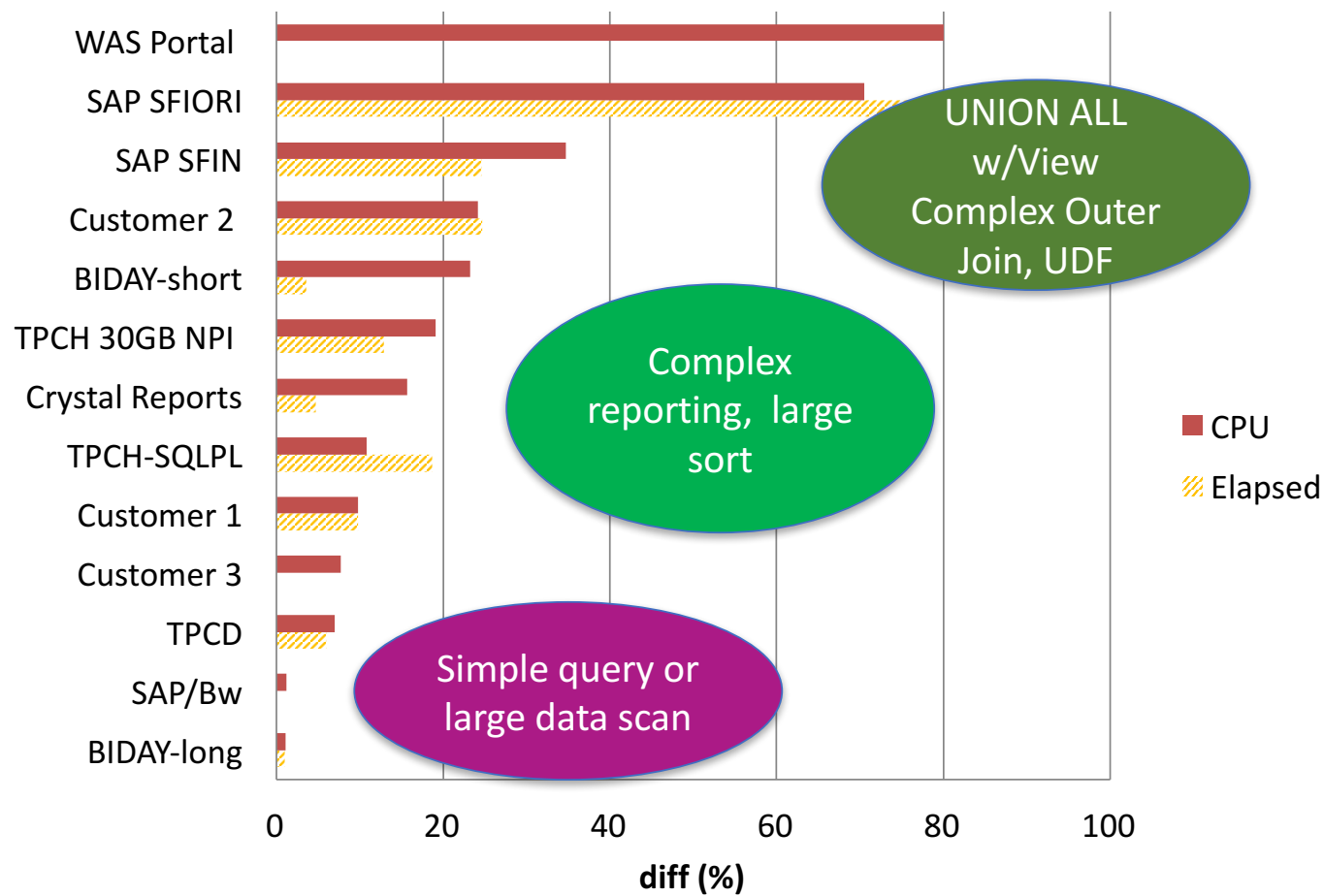
Chart categories (top to bottom): WAS Portal, SAP SFIORI, SAP SFIN, Customer 2, BIDAY-short, TPCH 30GB NPI, Crystal Reports, TPCH-SQLPL, Customer 1, Customer 3, TPCD, SAP/Bw, BIDAY-long

X-axis: diff (%) — 0, 20, 40, 60, 80, 100

Legend: CPU, Elapsed

Callouts: UNION ALL w/View Complex Outer Join, UDF; Complex reporting, large sort; Simple query or large data scan

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

DB2 12

33

High-level Performance Focus

- **Query (RDS) focus based upon new(er) workloads**
  - Complex views or table UDFs
    - UNION ALL
    - Outer joins
    - Join predicates with (Stage 2) expressions
  - CASE expressions, CAST functions, scalar functions
- **Query - General Bottlenecks**
  - Sort/workfile reductions
  - Reducing prepare cost and frequency
  - I/O performance
    - Reduce unnecessary prefetch scheduling

Query Performance Focus

- **Improve performance of**

  - UNION ALL and outer join performance enhancements

    - Performance issues are similar with both types of query pattern

    - Reduce materializations

      - Bypass workfile usage when materialization required
      - Trim unnecessary columns from materializations

    - Push predicates inside UNION ALL legs or OUTER JOIN query blocks

    - Push ORDER BY and FETCH FIRST into UNION ALL legs

    - Reorder OUTER JOIN tables to avoid materializations

  - Table UDFs

    - Improve merge similar to views

    - Indexability of join predicates

Query Performance Focus …

- **Improve performance of …**
  - Join predicates with Stage 2 expressions
  - Stage 2 to indexable
    - VARBINARY, COLLATION_KEY IOE
  - Expression evaluation
    - CASE, SUBSTR, etc
    - Expression sharing (SELECT list only)
    - Caching deterministic UDF results
  - Enabling parallelism
    - 100% zIIP offload for parallel child tasks
    - Reduce cost and resource consumption
  - Sort
    - Reduce workfile usage for GROUP BY/DISTINCT
    - Reduce key length for GROUP BY/DISTINCT and sparse index
    - Continued progress towards in-memory for smaller sorts (begun in V9)

Query Performance Focus …

- **Adaptive index**
  - Simple example of targeted use case
    SELECT * FROM TAB1 WHERE COL1 < ? AND COL2 < ? AND COL3 < ?;
    INDEXES: IX1 (col1), IX2 (col2), IX3 (col3)

  - Filtering of the above query is dependent on literals at execution
    - Common pattern for search screens (with BETWEENs or LIKEs)
    - Common that 1 index is filtering unless a highly skewed value is searched

Query Performance Focus …

- **Adaptive index …**
  - Allow RID based plans (single index list PF or multi-index) to quickly determine filtering from index
    - Without requiring REOPT(ALWAYS)
    - For list prefetch or multi-index ORing
      - Earlier opportunity to fallback to tablespace scan if large % of table to be read
    - For multi-index ANDing
      - Reorder index legs from most to least filtering
      - Early-out for non-filtering legs, and fallback to rscan if no filtering
  - Optimizer to use uncertainty to determine risk of a single index plan
    - Quick evaluation done based upon literals used
    - Any further evaluation of filtering deferred until after 1 RID block retrieved
      - Ensuring that very short running queries do not incur overhead

MIGRATION

Migration Prerequisites – Hardware & Operating System

- **Processor requirements:**
  - z196, or higher, processors running z/OS V2.1, or later
  - DB2 12 will probably require increased real storage for a workload compared to DB2 11 for z/OS

- **Software Requirements:**
  - z/OS V2.1 Base Services, (5650-ZOS), or later
  - DFSMS V2.1, or later
  - Language Environment Base Services
  - z/OS V2.1 Security Server (RACF), or later
  - IRLM Version 2 Release 3 (Delivered with DB2 12)

- **Additional details:**
  - http://www.ibm.com/common/ssi/rep_ca/1/897/ENUS215-371/ENUS215-371.PDF

Migration & Catalog

- **Single phase migration process**
  - No ENFM phase
  - New function activated through new command
    - -ACTIVATE FUNCTION LEVEL with "V12R1Mnnn" where nnn >= 500
  - APPLCOMPAT rules, fallback rules continue to apply
- **BSDS conversion to support 10 byte log RBA is pre-requisite**
- **No pre-V10 bound packages**
  - Get rid of 31-bit runtime, some performance improvements
- **BRF is deprecated**
  - BRF page sets still supported, but zparm and REORG options are removed
- **Temporal RTS tables**
  - Defined in catalog, enablement is optional

Online Migration Improvements

- **Pause statistics externalization during migration to reduce contention (part of fallback SPE)**
    - Real time statistics
    - Stats feedback
    - SYSPACKAGE.LASTUSED

- **Catalog/directory lock duration reduction APARs (impacts online migration, catalog REORGs)**
    - PI43662 – Accelerator resources released in a timely manner
    - PI43916 – DB2 plan allocation locks released in timely manner
    - PI39053 – Avoid locks from SET statement
    - PI40755 – Dynamic SQL release prepare locks earlier

Top DB2z Social Media Channels   #DB2z

- **Join the World of DB2 www.worldofdb2.com**

- **Follow @IBMDB2 on Twitter https://twitter.com/IBMDB2**

- **Join DB2z LinkedIn Group**

- **https://www.youtube.com/user/IBMDB2forzOS**

- **DB2z on Facebook**
  - https://www.facebook.com/IBMDB2forzOS/

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

DB2 12

Register Now coming DB2 Webcasts #DB2z - http://ibm.biz/DB2z2017WebcastSeries

- **DB2 for z/OS RESTful API enabling the Mobile Economy - 11th April**

- **DB2 12 Migration Planning and Very Early Experiences PART 1 - 16th May**

- **DB2 12 Migration Planning and Very Early Experiences PART 2 - 17th May**

- **DB2 12 for z/OS Data Sharing Enhancements 6th June**

- **DB2 for z/OS and FlashCopy: Practical Use Cases - 13th June**

- **Using Larger Real Memory Size with DB2 for z/OS to reduce CPU - June 27th**

DB2 12 — The ultimate enterprise database for business-critical transactions and analytics

DB2 12