

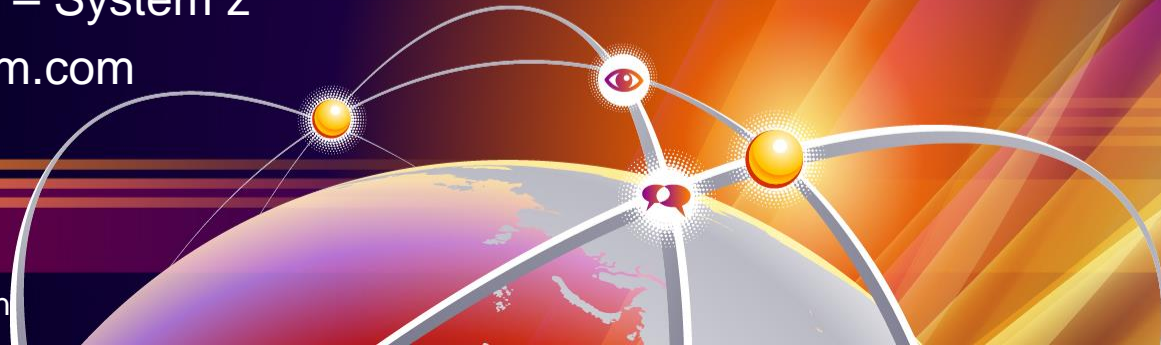
DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



DB2 12 Technical Preview

August 2016

Charles Lewis
DB2 for z/OS Advisor
IBM Analytics – System z
lewisc@us.ibm.com



Disclaimer

- Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.



Agenda

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

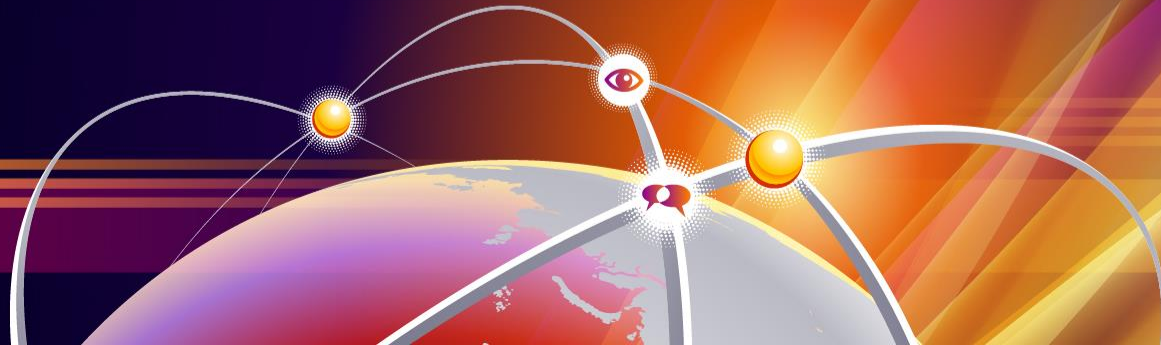


DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Introduction

Dr



DB2 12 Goals

Application Enablement

- IDAA improvements to expand to new use cases
- SQL/SQLPL 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 inserts

- 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

DB2 12: Quick Hits

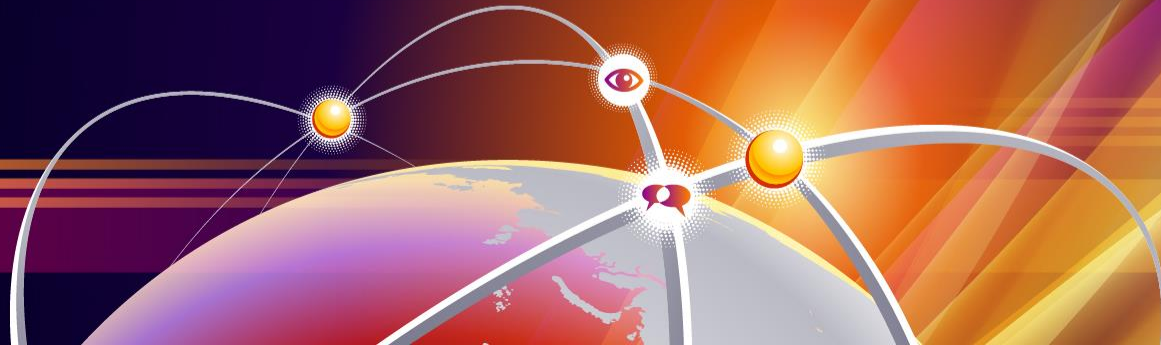
- **Scale and speed for the next era of mobile applications**
 - Over 1 million inserts per second measured, will scale higher
 - 256 trillion rows in a single table, with agile partition technology
- **In memory database**
 - Up to 23% CPU reduction for index lookup with advanced in-memory techniques
- **Next Generation application support**
 - 360 million transactions per hour through RESTful web API
- **Deliver analytical insights faster**
 - Up to 2x speedup for query workloads, 100x for targeted queries



DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Performance Focus – Traditional Workloads



DB2 12 Performance Enhancements

- **In-memory contiguous buffer pools**
 - Direct row access in-memory, avoid GetPage overhead
 - Up to 8% CPU reduction measured for OLTP
 - PGSTEAL(NONE) – improved in DB2 12 to avoid LRU and hash chain overheads
 - Overflow area is used in case objects don't fit
 - Automatically managed by DB2
 - Allocated when BP is allocated, but only backed when used
- **In-memory index for fast traversal**
- **More granular Global Commit LSN**
 - Potential huge improvement in lock avoidance (data sharing)
 - Also will help space reuse for insert



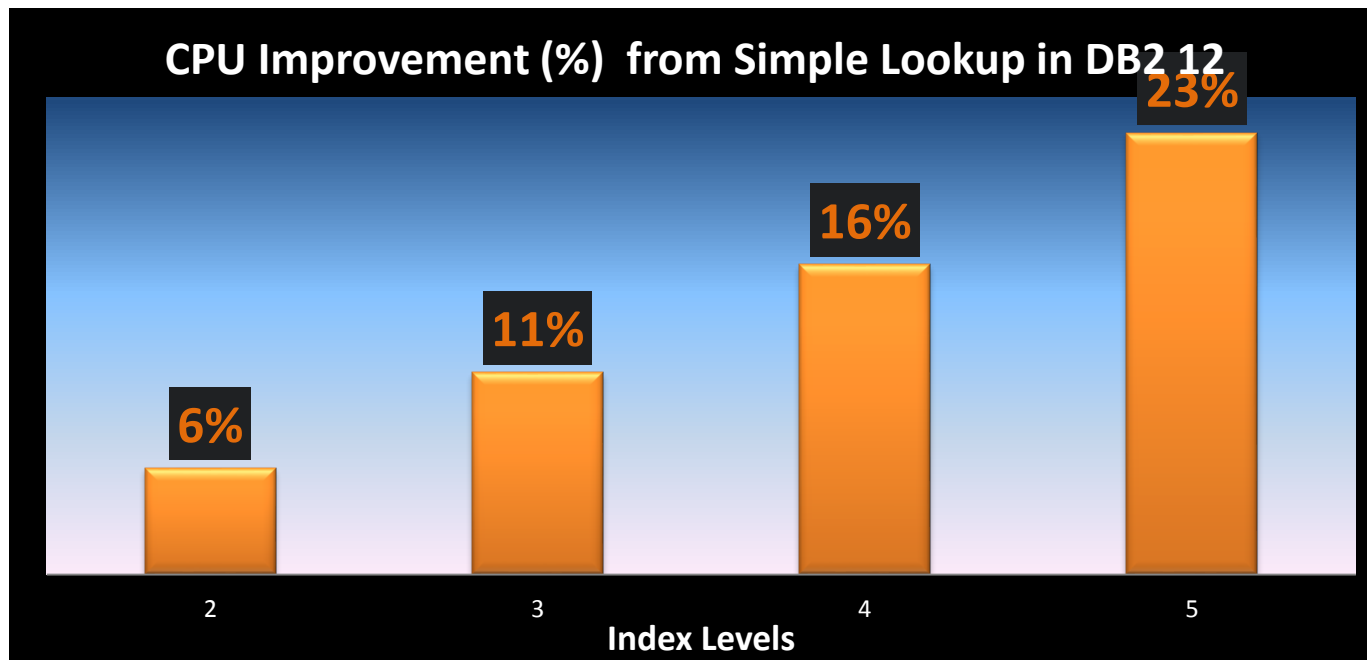
In-Memory Index Optimization

- 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 or 500 MB or 20% of allocated BP 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



DB2 12 Performance Enhancements...

- **Piecemeal list prefetch (still under test)**
 - Improved query performance for unclustered data, ISO(UR)
 - REORG avoidance feature
- **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 performance improvements
- RLF control for static packages
- SYSPACKAGE.LASTUSED becomes useful for determining when packages can be freed
- 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 Performance

- Insert workloads are amongst the most prevalent and performance critical
- DB2 12 delivers significant improvements for Non-clustered insert: journal table pattern
 - UTS with MEMBER CLUSTER (both with/without APPEND)
- 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



Performance Enhancements...

- Implement new PL/X compiler
- BP advisory mode to simulate larger BPs – rollback to v11
 - Low CPU and real memory overhead
 - Statistics provided to indicate I/O savings
 - Retrofit to V11
- Increase use of 1M size pageable page frames for DB2 working storage
 - 4.7% CPU decrease measured (no paging)
 - Zparm controlled – if DB2 is paging, then there is overhead for 1M paging.
 - Flash Express improves 1M paging
- Streamlined Claim/Declaim processing
 - Avoid re-claim overhead across multiple commit scopes for RELEASE(DEALLOCATE) threads
 - Online REORG and other drainers can break in



Performance Enhancements...

- Improvements in pool management to simplify and remove scalability inhibitors
 - EDM pools
 - LOB/XML storage
- Other improvements
 - Reducing DGTT DECLARE overhead
 - Remove log force write from IDENTITY columns and SEQUENCE caching in data sharing



DB2 12 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
 - BP 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 enhancements
 - Internal structure changes for cache efficiency, more processor prefetch
- **Raise Buffer Pool size limit to 16 TB**
- **>4G active log datasets**



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 threads with 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 observed in our evaluations
- **Concurrent insert against UTS and MEMBER CLUSTER**
 - 5-10% CPU reduction
 - Throughput improvement depends on whether 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
- Add workfile, tempfile usage information to Accounting Trace
- Enhanced precisions for IFCID 199 data set I/O stats - microseconds



Access path (plan) stability

- Dynamic SQL plan stability
- Static plan stability usability
- Preserve dynamic statement cache at rollback
- Integrated RUNSTATS with optimizer
 - Automated update of statistics profiles by optimizer
- Statistics profile support
 - Automated update for 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 sw 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 Plan Stability

- **DB2 12 plan – base infrastructure**
 - Opaque parameter CACHEDYN_STABILIZATION
 - Capture
 - 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
- **Key DB2 12 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 appl is running
- **REBIND PACKAGE** improvements
 - Addresses issue of switch accidentally causing invalid copy to become SWITCH to include APREUSESOURCE option so as to choose ORIGINAL or PREVIOUS copy as source for AP reuse current
 - APREUSESOURCE options avoids the two step process of
 - REBIND SWITCH followed by
 - REBIND APREUSE



RUNSTATS Enhancements for SQL Performance

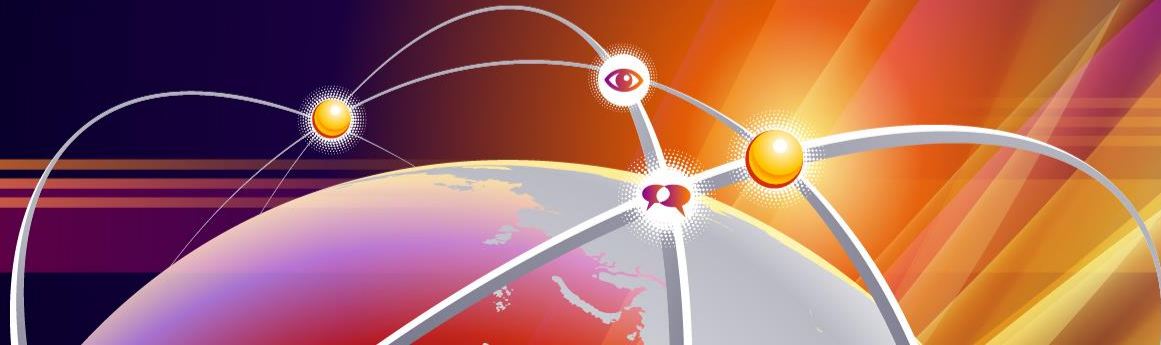
- **RUNSTATS (for access path selection benefit)**
 - Clusterratio formula improvements
 - Statement cache invalidation
 - Optional for RUNSTATS (new default)
 - For other utilities ONLY if objects in pending state before utility executed
 - Profile support for inline stats
 - Automated COUNT for FREQVAL
 - 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 stats recommendations



DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Performance Focus – Enabling new applications



DB2 12 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 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



DB2 12 Query Performance Focus

- Improve performance of
 - UNION ALL and outer join performance enhancements
 - Performance issues are similar with both types of query patterns
 - 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



DB2 12 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)



DB2 12 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



DB2 12 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

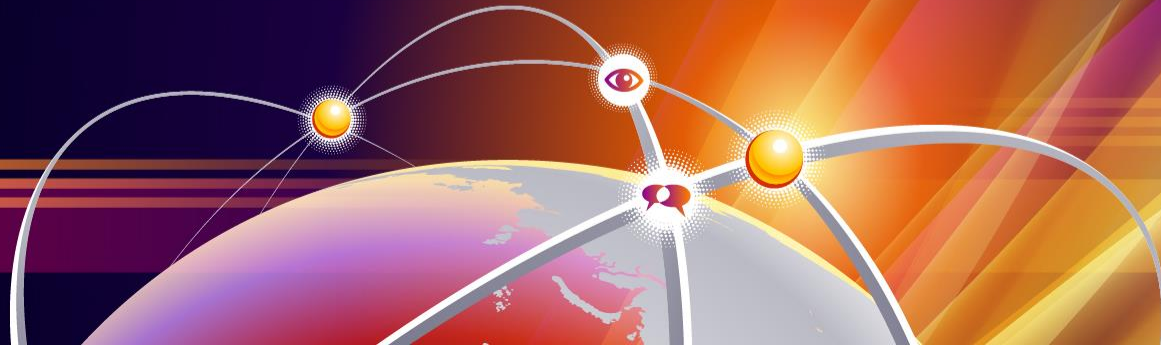


DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Application Enablement

Dr



DB2 for z/OS Adapting to Modern Application Development Paradigms

- Many modern application developers work with REST services and JSON data formats
- **DB2 Adapter for z/OS Connect** provides the means to do this

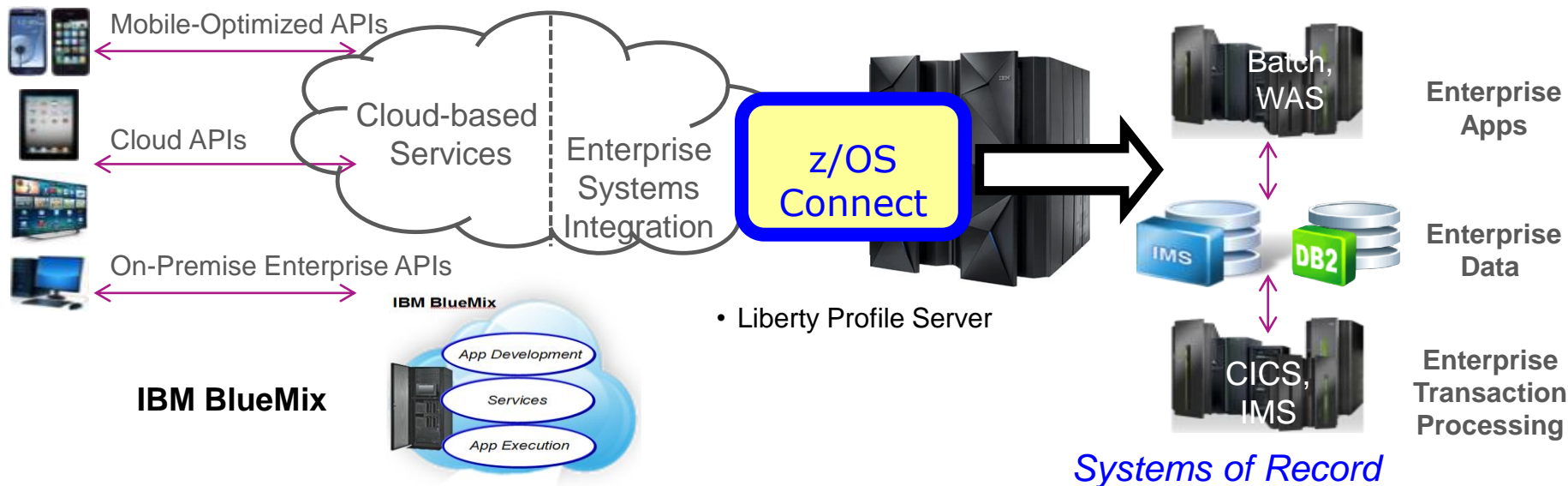
`http://10.25.14.17:259/service/db1/getcust?32537`

```
{  
  "customer":  
    {"ID": "32537",  
     "Name": "Bob Rady Enterprises"}  
}
```



Cloud/Mobile integration with DB2 and z/OS Connect

Getting REST and JSON into your mainframe environment in a way that enables you to best take advantage of the assets that exist there:



Provides a common and consistent entry point for mobile access to one or many backend systems

Simplifies front-end functions by allowing them to pass RESTful and JSON rather than be aware of or involved in data transformation

DB2 12 Application Related Enhancements

- DRDA Fast Load – callable command for fast streaming of data into DB2
- System profiling enhancements
 - Auto start of system profiles
 - Set global variables based on profiles (e.g. transparent archiving)
- New **MODIFY DDF PKGREL(BNDPOOL)** option
 - Pool high performance DBATs at connection termination
- DSNULI now supported for IMS Attach
- Long list of SQL and XML improvements



DB2 12 DRDA Fast Load

- **Problem:**
 - DB2 provides the DSNUTILU stored procedure to load data from a client
 - But this is difficult to use, application must transfer data to z/OS file
- **Solution:**
 - DB2 Client API (CLI and CLP) for remote load into DB2
 - Easy and fast loading of data from file that resides on client
 - Internal format (SAP), as well as delimited and spanned (LOB data)
 - Applicable for some “cloud” use cases
 - Overlap network operations with data ingest on the DB2 server
 - Measured results show as fast or faster than DB2 LOAD utility
- **Plans to use this feature for fast write from Spark**



DB2 12 SQL Enhancements

- Simple VALUES using dynamic SQL – v10,v11
- JDBC/ODBC type2 performance enhancements
- ODBC driver improvements:
 - TIMESTAMP with TIMEZONE support
 - Multi context support using ASSOCIATE/DISSOCIATE THREAD
 - Ability to preserve dynamic statement cache after rollback
- SQL Pagination syntax support – OFFSET n and LIMIT
- Support prepareAttribute literal replacement as BIND option



SQL Enhancements...

- Keep views, UDFs intact on DDL for underlying tables
- Increase maximum number of tables referenced in view, UDF, statement
- Enhanced MERGE support
- SQL pagination improvements
- New built in functions
 - HEX2BIN (v11)
 - MEDIAN, PERCENTILE_CONT, PERCENTILE_DISC
 - COUNT(DISTINCT x)
 - GENERATE_UNIQUE – new optional length parm
 - HASH functions (CRC32, MD5, SHA1, SHA256)



Enhanced Merge Support

- DB2 z/OS initial support for MERGE statement with limited functionality was delivered with Version 9:
 - Limited to UPDATE and INSERT and only one of each
 - Focused on use of host variable column arrays to provide multiple rows of input data
- In DB2 12, DB2 z/OS MERGE statement will be aligned with behavior defined in SQL Standard and DB2 family.
 - Source data as a *table-reference*
 - Multiple MATCHED clauses
 - Additional Predicates with [NOT]MATCHED
 - Support DELETE operation
 - Allow IGNORE and SIGNAL



SQL Pagination

- With the growth of web and mobile applications, application developers are looking for more efficient ways to develop well performing applications.
- Numeric-based pagination
 - `SELECT * FROM tab OFFSET 10 ROWS FETCH FIRST 10 ROWS ONLY`
- Data-dependent pagination
 - Existing syntax
`WHERE (LASTNAME = 'SMITH' AND FIRSTNAME >= 'JOHN') OR (LASTNAME > 'SMITH')`
 - New equivalent syntax
`WHERE (LASTNAME, FIRSTNAME) > (SMITH, JOHN)`



SQL Enhancements...

- Unicode columns in EBCDIC tables
- Piecemeal DELETE – allow for interim commits
 - E.g. Delete from T1 where C1 > 7 **“FETCH FIRST 5000 ROWS ONLY”**
 - AUTONOMOUS SQL PL procedure is an existing tactical solution
- MQ UDF enhancements: allow MQ message header to be passed
- BiTemporal enhancements
 - Auditing enhancements – v11
 - Inclusive/inclusive for business time
 - Logical transaction for system time
 - Temporal RI



Piece-wise Modification of Data

- Mitigate the effects of locking and logging when potentially millions of rows could be affected by a simple statement like:
"DELETE FROM T1 WHERE C1 > 7"
- Solution
 - Support fullselect as the target of a DELETE statements where the fullselect allows for the FETCH FIRST N ROWS ONLY to be specified.
 - DELETE FROM (SELECT * FROM T1 WHERE C1 > 7 **FETCH FIRST 5000 ROWS ONLY**)



SQL Enhancements...

- APPLCOMPAT support for DB2 12
- Global variable enhancements
 - Array Global Variables
 - LOB Global Variable
 - FETCH statement and global variables as a target
 - EXECUTE statement and global variables
 - OPEN statement and Global Variables
- SQLPL
 - SQLPL in triggers, including trigger versioning and debugging support
 - Support for constants
 - SQLPL obfuscation
 - Dynamic SQL in SQLPL UDFs and stored procedures
 - DBMS_OUTPUT for UDF tracing – v11



DB2 12 SQL PL Triggers

- Rich capability in trigger body
 - Allow **SQL PL control statements**
 - IF-THEN-ELSE, LOOP, REPEAT, ...
 - Allow more SQL statements, dynamic SQL, variables, handlers
 - Easier porting of triggers from other DBMSes
 - **DEBUGGER** support
 - **VERSIONs** support
 - Provides a better way to change a trigger without DROP, CREATE
 - Can change trigger online and maintain trigger activation order
- Richer capability means some performance overhead compared to an equivalent traditional trigger



XML Enhancements in DB2 12

Improve the performance of the XML queries by choosing optimal access:

76% / 77% reduction in the Class 1 / Class 2 time respectively.

Improve the performance of XMLTable that conducts the pivot-like operation to XML data with name-value pair pattern:

90% C2 CPU improvement using customer's xml data/queries

Improve performance and developer productivity by allowing multiple update inside XMLModify function :

Up to 90% improvement in ET; up to 97% improvement in CPU time (when comparing to semantics equivalent single update)

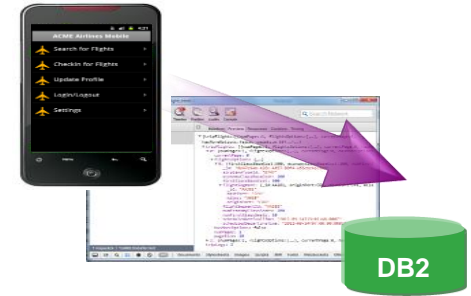
XSLTRANSFORM
function for XML
extender users

Allow for easy migration
to pureXML



DB2 JSON Support for New Era Application Requirements

- Store data from web/mobile apps in its native form
 - Many web applications use JSON for storing and exchanging information
 - JSON is often the preferred data format for mobile application back-ends
- Move from development to production in no time
 - Ability to create and deploy flexible JSON schemas
 - Gives power to application developers by reducing dependency on IT
 - no need to pre-determine schemas and create/modify tables
 - Ideal for agile, rapid development and continuous integration
- DB2 provides two ways for working with JSON
 - Java driver for JSON API
 - SQL extensions – DB2 11, with enhancements in DB2 12



JSON Enhancement in DB2 12

```
>>-JSON_VAL(--json-value--,--search-string--,--result-type--)-----<<
```

To extract and retrieve JSON data into SQL data types from BSON.

In DB2 12, we remove the requirement that 1st parameter has to be a BLOB column (already retrofit to V11 in PI39003)

Example (before):

```
JSON_VAL(column1,'PO.cust  
omer.@cid', 'i:na')
```

In V12, we support more as 1st parameter:

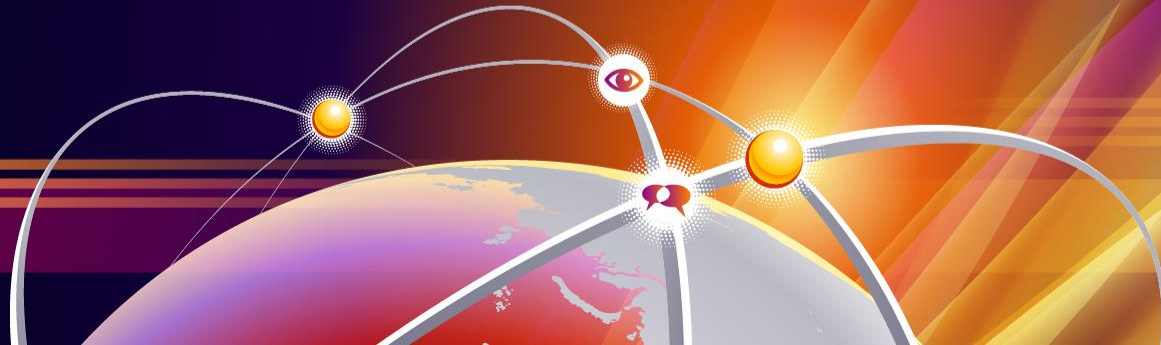
- view column
- CASE expression
- table expression with union all
- trigger transition variable
- SQL PL variable/parameter



DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Reliability, Availability, Scalability, Security



Lifting Partition Limits – Problem Statement

- Maximum number of partitions is dependent on DSSIZE and page size
 - E.g., if DSSIZE = 256 GB and page size = 4K, then maximum number of partitions is 64
- Running out of space in a partition is an application outage
- Altering DSSIZE, REORG must be run on entire table space
- DSSIZE is at table space level and not at partition level
 - All partitions inherit the same DSSIZE set at table space level
 - No ability to have differing partitions sizes
 - REBALANCE must run against multiple partitions
- Maximum table size limited to 16TB



Lifting Partition Limits – Solution

- **New UTS PBR tablespace structure called ‘UTS PBR RPN’**
 - Relative page numbers (RPN) instead of absolute page numbers
 - Simplicity, usability, availability, scalability
 - Remove dependency between number of partitions and partition size
 - 7-byte RIDs (2 byte part number, 5 byte page number)
 - Up to 256 trillion rows in a single table
 - New REORG mapping table format, optional until new function enabled
 - Support up to 1TB partition size
 - Max table size increased from 16TB (4K page size) to 4PB
 - Architected to go even larger
 - Increasing DSSIZE is supported at partition-level
 - New DSSIZE support for indexes



Lifting Partition Limits – Considerations

- **Conversion / Exploitation:**
 - Zparm to control whether creation of range partitioned uses relative page numbering
 - PAGENUM RELATIVE/ABSOLUTE option on CREATE & ALTER TABLESPACE
 - Conversion is pending alter - requires tablespace-level online REORG
 - Online alter to increase DSSIZE – immediate, non-disruptive
 - Online alter to decrease DSSIZE - pending alter requiring table space -level REORG
- **Log record formats changed to support 7 byte RIDs**
 - Improved serviceability, DSN1LOGP now formats partition number explicitly
 - New log record format written as soon as migrate to DB2 12, requiring fallback toleration support in V11
 - Not just for RPN pagesets!



Online Schema

- Insert partition
- Online deferred ALTER INDEX COMPRESS YES
 - Previously placed indexes in RBDP
- Option to defer column-level ALTERs
 - Materialize through online REORG
 - Avoid availability constraints and conflict with other deferred alters



Insert Partition

- **Problem statement**

- Large range-partitioned tables often have hot spots and rebalancing across entire set of partitions is onerous
- Partitioning scheme chosen in the past may no longer be optimal

- **Solution**

- Ability to insert a new partition with specified limitkey value
 - ALTER TABLE ADD PARTITION ENDING AT xxx
- Split existing partition, distribute data between new and old
- Online change through pending alter
- Only affected partition(s) need to be REORGed
- No PIT recovery prior to materializing REORG



Pending Column Level Alter

- **Problem statement**
 - Some column-level alters result in application impact
 - E.g. indexes placed in RBDP
 - Immediate alters conflict with pending alters
 - Additional REORGs required to materialize pending alters to avoid DDL failures
- **Solution**
 - Allow existing immediate alters to become pending alters, UTS only
 - All pending alters accumulated and materialized through online REORG
 - New zparm
 - ALWAYS IMMEDIATE: Existing behavior for existing alters
 - ALWAYS PENDING: Current immediate alters are converted to pending alters
 - ALTER COLUMN type to avoid RBDP on indexes



Security & General Enhancements

- LOB compression
 - zEDC hardware requirement
- Improved LOB handling for ISO(UR) queries to avoid SQLCODE +100
- TRANSFER OWNERSHIP
- Permit installation, migration without INSTALL SYSADM authority to limit data access
- New UNLOAD privilege
- Support long, camel-cased DBRM names
- IFI 306 log records returned in the correct version

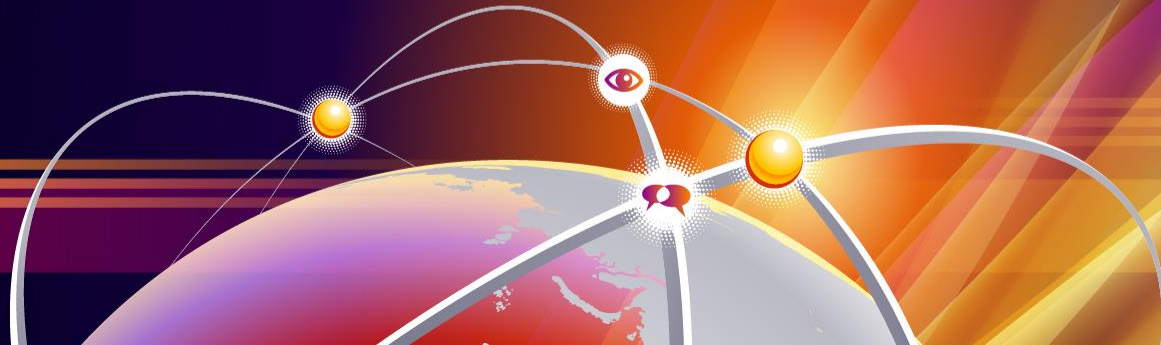


DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Data Sharing

Dr



DB2 12 Data Sharing Improvements

- Data sharing support for global transactions
- DDF shared session data across group
 - DDF transaction re-routing, Session Token for client fail-over
- Data sharing performance improvements:
 - Improved lock avoidance checking to reduce CF lock requests
 - In-memory indexes can reduce GetPages and CF GBP requests
 - Improved insert space search can avoid P-lock contention and streamline inserts
 - RUNSTATS and UNLOAD ISOLATION(UR) to avoid CF page registration



DB2 12 Data Sharing Improvements

- New data sharing peer recovery option
- Retry of automatic LPL and GRECP recovery
- Asynch CF Lock duplexing (not yet enabled)
 - Reduce overhead for system managed duplexing of CF LOCK1 and SCA structures
 - Secondary structure updates are performed asynchronously with respect to primary updates
 - DB2 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 active log on DASD
 - Increases the practical distance for multi-site Sysplex operations whilst duplexing of CF LOCK1 and SCA structures
 - Planned as 4Q 2016 deliverable with
 - z/OS 2.2 APAR
 - Z13 GA2 and CFCC 21

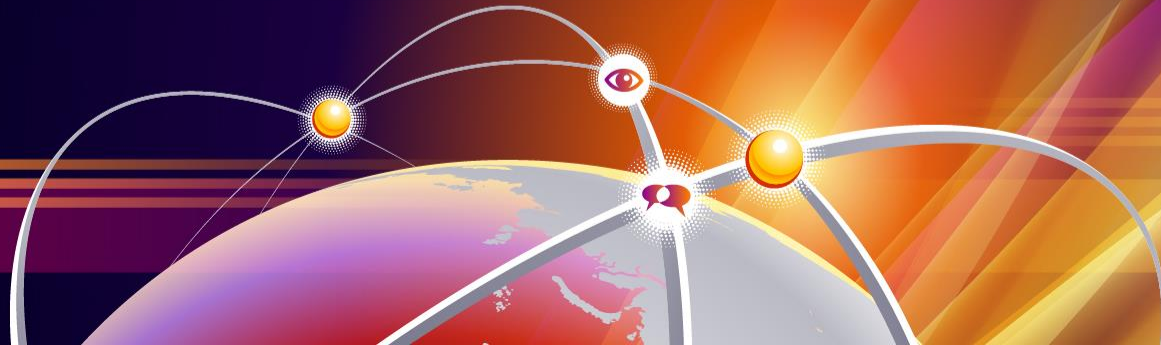


DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Migration

Dr



Migration Prerequisites – Hardware and Operating System

- **Processor requirements:**

- z196, or higher, processors running z/OS V2.1 or later
- DB2 12 will probably require increase 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 2.3 (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 NEW FUNCTION
 - 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 pagesets still supported, but zparm & 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 allocations locks released in a timely manner
 - PI39053 – Avoid locks from SET statement
 - PI40755 – Dynamic SQL release prepare locks earlier



Delivering New Capabilities Faster For Increased Productivity (aka DB2 Continuous Delivery)

- Moving forward DB2 for z/OS will be moving to a new delivery model
- What: John Campbell and Chris Crone webcast to introduce and discuss:
 - reasons for change
 - product strategy going forward
 - how corrective/preventative maintenance will be applied
 - how new function will be consumed and activated
 - what enhanced documentation will be provided to help understand and manage continuous deliver
- When: Tuesday, September 27, 2016 – 11 AM EDT
- To register for webcast: <http://ibm.biz/DB2zContinuousDelivery>



DB2 for z/OS – Ultimate Database for Cloud, Analytics and Mobile
Industry-leading performance, security, scale and reliability



Thank you!

Dr

