

Migrating to DB2 10 for z/OS

Jay Yothers Roger Miller
IBM Silicon Valley Lab

Session Code: B14
May 06, 2011 09:15 AM | Platform: DB2 for z/OS





Migration Agenda

Hardware & Software Prerequisites

Pre-migration planning

Migration paths available

- DB2 9 to DB2 10 "Standard release" Migration Process Overview
- V8 to DB2 10 "Skip release" Migration Process Overview

The Migration Process

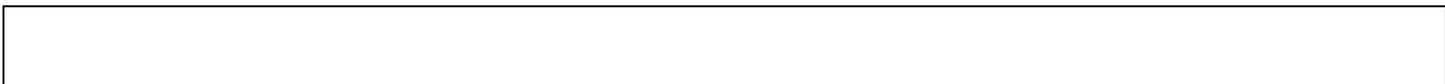
- New Release Migration Planning
- Migration to Conversion Mode (Running DSNTIJTC)
- Enable New-Function-Mode
- Catalog Restructure

Fallback - Returning to a prior release

Incompatibilities after migrating to DB2 10

Migration Considerations For Specific Features

Eliminate DDF Private Protocol





DB2 10 for z/OS

Hardware & Software

Prerequisites



Prerequisites – Hardware & Operating System

Processor requirements:

- zEnterprise z196, z10, z9, z990 and z890 processors supporting z/Architecture
 - **z800 or z900 processors NOT supported**
- DB2 10 for z/OS will probably require 5% more real storage for a workload compared to DB2 9 for z/OS



Software Requirements:

- z/OS V1.10 Base Services (5694-A01) at minimum
- DFSMS V1 R10 – DB2 Catalog becomes SMS managed
- Language Environment Base Services
- z/OS Version 1 Release 10 Security Server (RACF)
- IRLM Version 2 Release 3 (Shipped with DB2 10 for z/OS)
- z/OS Unicode Services and appropriate conversion definitions are required.
- DB2 Connect 9 Fixpack 1 is required; 9.7 Fixpack 3a recommended

This slide lists only the key highlights! Also, this applies whether you are migrating or installing.

Processors

DB2 for z/OS Version 10 operates on System z (z10, z9, z990, z890 and later) or equivalent processors running in 64-bit mode supported by z/OS V1R10. The processor must have enough real storage to satisfy the combined requirements of, DB2, z/OS and the appropriate DFSMS (tm) storage management subsystem components, access methods, telecommunications, batch requirements, and other customer required applications.

In addition, 64-bit virtual addressing support will probably require increased real storage for a workload compared to DB2 9 for z/OS.

DB2 Version 10 requires the function that is provided by z/OS Version 1 Release 10 Base Services (5694-A01) with the following base and optional elements:

- DFSMS Version 1 Release 10 features, part of the Systems Management optional feature of z/OS; specifically, DFSMSShm for archiving and DFSMSdss for concurrent copy in Utilities.
- Language Environment Base Services
- z/OS Version 1 Release 10 Security Server (RACF)
- IRLM Version 2 Release 3 (delivered with DB2 10)
- WebSphere Replication Server Version 9 Release 1 (5655-R55) and IBM Data Event Publisher Version 9, work with the following DB2 for z/OS versions:
 - DB2 9 for z/OS (5635-DB2)
 - DB2 Universal Database for z/OS Version 8 (5625-DB2)
- z/OS Unicode Services and appropriate conversion definitions are required. (As installed for V8 and DB2 9).
- Some of the basic operation of a DBMS is provided by utility functions, such as backup, recovery, reorganization, loading and unloading data, gathering statistics and checking data, indexes, and large objects. Customers should ensure that these functions are provided either by ordering DB2 Utilities Suite for z/OS Version 10, or by obtaining equivalent function elsewhere.

**DB2 for z/OS
Packaging**

**DB2 10 for
z/OS**

Base

- DB2 Base
- IRLM V2R3
- REXX
- MQSeries
- MQListener
- IMS Attach
- RACF Auth Exit
- Panels (English / Kanji)
- JDBC / SQLJ / ODBC

**Orderable
No-Charge
Features**

- z/OS Appl Connectivity to DB2 for z/OS

**Related
No-Charge
Product**

- DB2 Accessories Suite 5697-Q02
–See announcement 210-345 for
details

**Chargeable
Features**

- DB2 QMF V10
–DB2 QMF Enterprise Edition
–DB2 QMF Classic Edition

Detailed slides for the components of this chart follow.



DB2 10 for z/OS

Pre-migration planning

Run DSNTIJPM (DSNTIJPA) pre-migration job

Check for incomplete object definitions in the catalog

- Complete or drop all such objects **before** beginning migration
 - Avoids possible behaviour differences in DB2 10.

Run DSNTIJPM or DSNTIJPA, to identify them:

- DSNTIJPM ships with DB2 10 and should be run on previous releases to identify pre-migration catalog cleanup requirements.
 - DSNTIJPM may provide DDL or utility statements for the cleanup.
- DSNTIJPA is the same job and is shipped for both DB2 V8 & DB2 9, to maximize prepare time.



Important preparation

PDSE requirement

Enter "/" to select action	Dsorg	Recfm	Lrecl	Blksz
SYS2.DB2.V10.SDSNLINK	P0	U	0	32760
SYS2.DB2.V10.SDSNLOAD	P0-E	U	0	32760
SYS2.DB2.V10.SDSNLOAD2	P0-E	U	0	32760
SYS2.DB2.V10.SDSNMACS	P0	FB	80	3120

Migrate only from a system with an expanded BSDS (DSNJCNVB)

Old plans and packages V5 or before → REBIND

Plans containing DBRMs → packages
(PK62876 (9), PK85833 (9), PK79925 (V8), PM01821 (All))

ACQUIRE(ALLOCATE) → ACQUIRE(USE)

Old Plan table formats → DB2 V8 or 9 format (59 columns)

Old plans and packages V5 or before → REBIND

Plans containing DBRMs → packages PK62876

ACQUIRE(ALLOCATE) → ACQUIRE(USE) Please note that since ACQUIRE(ALLOCATE) is only for plans, once everything is packages, this will just happen automatically. Hence, no need to do anything to make this happen.

Old Plan table formats → DB2 V8 or 9 format (59 columns)

**Items deprecated in earlier versions - NOW eliminated**

Private protocol → DRDA (new help in DSNTP2DP, PK92339, PK64045)

XML Extender → XML type

DB2 MQ XML UDFs and stored procedures → XML functions

DB2 Management Clients feature → IBM Data Studio application & administration services

BookManager use for DB2 publications → Info Center, pdf

The DB2 Customization Center, DB2's plugin for the z/OS msys for setup installation initiative, is discontinued in DB2 Version 10 → install panels

See the list of deprecated functions from prior versions.

The above features are still included in DB2 9 and may be dropped from future versions. Note the direction indicated to the right of the arrows, as these are the functions provided to replace the existing function. If you are using any of these functions, then move to the new function.

See the Installation Guide section, "Functions that are deprecated" and the announcement material for more information on these changes.

http://www.ibm.com/common/ssi/rep_ca/8/897/ENUS206-098/ENUS206-098.PDF

Private protocol → DRDA (new help in DSNTP2DP, PK64045)

XML Extender → XML type

DB2 MQ XML user-defined functions and stored procedures → XML functions

DB2 Management Clients feature (DB2 Administration Server, Control Center, & Development Center) → IBM Data Studio application & administration services

msys for Setup DB2 Customization Center → install panels

BookManager use for DB2 publications → Info Center, pdf

The DB2 Customization Center, DB2's plugin for the z/OS msys for setup installation initiative, was deprecated in DB2 Version 9.1 for z/OS and is discontinued in DB2 Version 10.



Overview of the information center (IC)

BookManager format is not supported for the DB2 10 for z/OS library.

- We offer an on-line information center that contains the information for supported versions.
- As always, we support and provide PDF versions of our information.
- We also offer an installable IC that you can run on your local system or intranet server. For ordering information, see:

<http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.dzic.doc/installabledzic.htm>

The IC is the IBM strategic direction for delivering information.

- Easy to use! Easy to understand! Easy to find!

The IC is a browser-based information portal—an industry-wide trend.

The DB2 for z/OS IC has been available since V8 GA. Our IC is one of IBM's top visited ICs based on total number of hits!

Current information for V8 and DB2 9 is available at:

<http://publib.boulder.ibm.com/infocenter/imzic/>

BookManager format continues to be supported for the V8 and DB2 9 libraries. Although, documentation for the products in the DB2 Accessories Suite was never provided in BookManager format.

For DB2 10, we are investigating whether we can continue to provide messages for the IBM LookAt tool. An efficient way to look up messages and codes is by using the information center. For more information, see:
<http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.support.dzic.doc/html/searchmsg.htm>



Improvements to installation & migration information based on customer feedback

Pre-migration and migration checklists were added to help plan for and keep track of the migration process. The checklists include links to the pre-migration and migration steps. Access the checklists from the following locations:

- Introduction to migration from DB2 Version 8:
http://publib.boulder.ibm.com/infocenter/ecsimzic/v1r0/topic/com.ibm.db2z10.doc.inst/d_b2z_intro2migfromv8.htm
- Introduction to migration from DB2 Version 9.1:
http://publib.boulder.ibm.com/infocenter/ecsimzic/v1r0/topic/com.ibm.db2z10.doc.inst/d_b2z_intro2migfromv9.htm

Installation and migration steps clearly define tasks to complete. Related concept and reference information are included as links from the tasks.

Several new jobs simplify the setup and installation of DB2-supplied stored procedures and user-defined functions. The process is documented in [Installation step 20: Set up DB2-supplied routines \(optional\)](#) and [Migration step 25: Set up DB2-supplied routines \(optional\)](#).

For the DB2 10 development cycle, the information development team focused on addressing solutions to top customer pain points. Based on customer feedback, one of the areas we focused on improving is the installation and migration information.

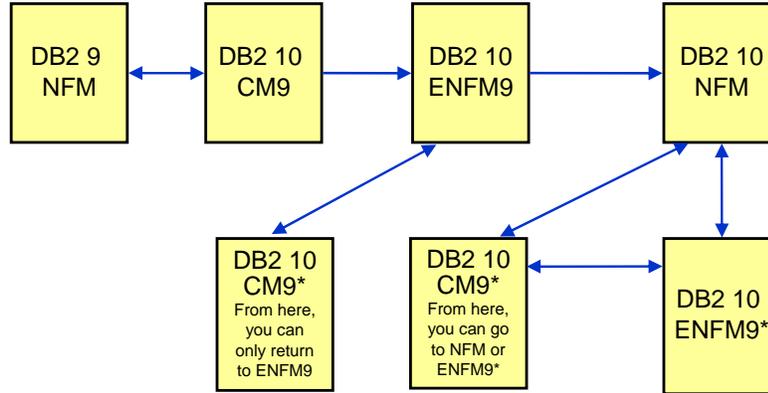


DB2 10 for z/OS Migration paths available



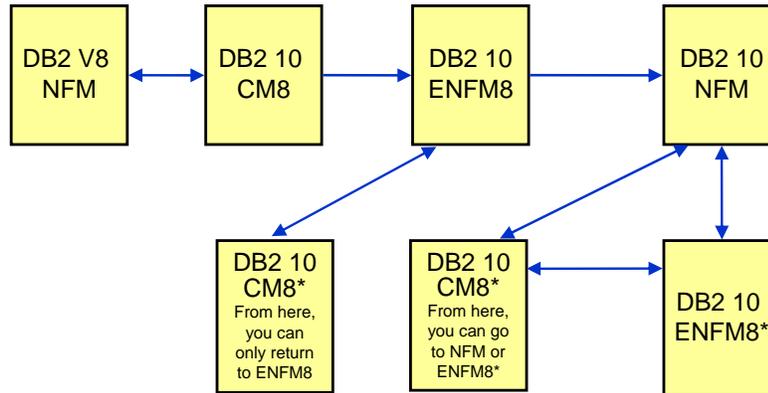
Migration and Fallback Paths when migrating (9 → 10)

- With DB2 10, you can always drop back to the previous stage
- Cannot fallback to DB2 9 after entry to DB2 10 (ENFM9), but can return to DB2 10 (CM9*)



Migration and Fallback Paths when migrating (V8 → 10)

- With DB2 10, you can always drop back to the previous stage
- Cannot fallback to DB2 V8 after entry to DB2 10 (ENFM8), but can return to DB2 10 (CM8*)





Considerations when “skipping” past DB2 9?

DB2 9 for z/OS education

- Migration Planning workshop materials
- Redbooks
- Transition class

Check for DB2 9 release incompatibilities

- DB2-managed stored procedures -> WLM-managed SPs

Familiarity with the new modes

- CM*, ENFM* introduced in DB2 9

Deprecated items

- Create simple table space -> segmented or partition by growth

Performance benefits

Storage changes

- Temp databases & table spaces -> work files (favor 32K)

Utilities updates

Key enhancements e.g. pureXML, Native SQLPL



**Migration considerations to REMEMBER**

- V8 system started migration to DB2 10, can only fallback to V8.
- V8 system started the migration to DB2 10, then performed fallback to V8, can not then migrate to DB2 9.
- V8 system started the migration to DB2 9, then performed fallback to V8, can not then migrate to DB2 10. Finish migration to 9 NFM, then to 10.
- V8 system migrating to DB2 10, can not use DB2 9 new functions until DB2 10 NFM is reached.
- data sharing group that started migrating from V8 to DB2 10 can not have any DB2 9 members.
- DB2 9 system that has started the migration to DB2 10 can only fallback to DB2 9.
- Data sharing group that started migration from DB2 9 NFM to DB2 10 can not have any V8 members.

Some migration considerations are:

- A V8 system that has started the migration to DB2 10 can only fallback to V8.
- A V8 system that has started the migration to DB2 10 and then performed a fallback to V8 can not migrate to DB2 9.
- A system that migrates from V8 NFM to DB2 10 can not use DB2 9 new function until DB2 10 NFM is reached.
- A data sharing group that migrated from V8 NFM to DB2 10 skipping DB2 9 can not have any DB2 9 members.

- A DB2 9 system that has started the migration to DB2 10 can only fallback to DB2 9.
- A data sharing group that migrated from DB2 9 NFM to DB2 10 can not have any V8 members.



DB2 10 for z/OS

The Migration Process

New Release Migration Planning

Now lets look at the actual migration process.



Preparing your current DB2 for Migration to DB2 10 CM

Apply the Fallback SPE APAR, **PK56922** and any prerequisite fixes

- **Your DB2 (V8 or DB2 9) systems *MUST* be at the proper service level**
- **See Info APARs II14477 (9) or II14474 (8)**



Non-Data Sharing

- Current DB2 (V8 or DB2 9) must be started with the SPE applied, or migration to DB2 10 will terminate.

Data Sharing

- Before migrating a member to DB2 10, all other started DB2 (V8 or DB2 9) members must have the fallback SPE applied.
- The fallback SPE must be on all active DB2 (V8 or DB2 9) group members for DB2 10 to start.



Important – Apply SPE to ALL Data Sharing Members Before Starting Migration!

DB2 Version 10 Migration Considerations

Before attempting to migrate to DB2 Version 10, make sure that the release being migrated from is at the proper service level. A migration to DB2 Version 10 places the DB2 subsystem into DB2 Version 10 Conversion Mode. In Conversion Mode, no new DB2 Version 10 function is available for use that would preclude a successful fallback to the previous release.

Ensure you apply the fallback SPE APAR, PK56922 to your current DB2 (V8 or DB2 9) subsystems.

Code Level Checking:

Non-Data Sharing

At DB2 startup time, the code level of the starting DB2 will be checked against the code level required by the current DB2 catalog. If the starting DB2 has a code level mismatch with the catalog then the DSNX208E message will be issued and DB2 will not start. A code level mismatch indicates that the starting DB2 is at a level of code that is down level from what it needs to be for the current catalog.

If the catalog has been migrated to DB2 Version 10 then the starting DB2 must be an allowed DB2 Version with the appropriate fallback SPE on.

Before attempting to migrate to DB2 Version 10, it's necessary to run with a maintenance level on the previous DB2 Version with the fallback SPE prior to attempting to migrate to DB2 Version 10. If the fallback SPE was not on the previous DB2 Version and DB2 started, before attempting to migrate to DB2 Version 10, then the DB2 Version 10 migration will be terminated.

Data Sharing

At DB2 startup time, the code level of the starting DB2 will be checked against the code level required by the current DB2 catalog and against the code level of the other DB2s that are active. If the starting DB2 has a code level mismatch with the catalog or any of the other DB2s that are running then either the DSNX208E or DSNX209E message will be issued and DB2 will not start.

A code level mismatch indicates that the starting DB2 is at a level of code that is down level from what it needs to be for the current catalog or that one or more of the already running DB2s are down-level from where they need to be.

Before attempting to migrate to DB2 Version 10, all started DB2 subsystems must have maintenance through the DB2 Version 10 fallback SPE on before any attempt is made to migrate any member to DB2 Version 10. If the fallback SPE is not on all active group members then DB2 Version 10 will not start and you'll not be able to attempt the DB2 Version 10 migration. Message DSNX208E or DSNX209E will be issued in these cases.

It's recommended that only one DB2 subsystem be started at DB2 Version 10 in Data Sharing for migration processing. Once the migration to DB2 Version 10 completes, the other group members can then be brought up to DB2 Version 10 at any time.

During a migration to DB2 Version 10, the other group members may be active. DSNTIJTC processing will get whatever locks are necessary for the processing it needs to do. The other active group members may experience delays and/or timeouts if they try to access the catalog objects that are being updated or locked by migration processing.

So what about coexistence?

DB2 V8 NFM and DB2 10 CM8 will coexist OR DB2 9 NFM and DB2 10 CM9 will coexist.



DB2 10 for z/OS

The Migration Process

***Migration to Conversion Mode
(Running the DSNTIJTC job)***



Migration to Conversion Mode

Run the **DSNTIJTC** job to migrate:

- From DB2 for z/OS Version 8 to DB2 10 for z/OS Conversion Mode (**CM8**)
- From DB2 9 for z/OS to DB2 10 for z/OS Conversion Mode (**CM9**)
 - Authorization check (INSTALL SYSADM)
 - Verify Catalog is at correct level for migration - this check varies with each DB2 release
 - New release DDL
 - Additional Catalog updates
 - Update Directory header page and BSDS/SCA with new release information

```

DSNU000I 194 14:58:16.06 DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = RELODCAT
DSNU1044I 194 14:58:16.08 DSNUGTIS - PROCESSING SYSIN AS EBCDIC
DSNU050I 194 14:58:16.49 DSNUGUTC - CATMAINT UPDATE
DSNU750I 194 14:58:16.59 DSNUECMO - CATMAINT UPDATE PHASE 1 STARTED
DSNU777I 194 14:58:16.59 DSNUECMO - CATMAINT UPDATE STATUS - VERIFYING CATALOG IS AT CORRECT LEVEL FOR MIGRATION.
DSNU777I 194 14:58:17.29 DSNUECMO - CATMAINT UPDATE STATUS - BEGINNING MIGRATION SQL PROCESSING PHASE.
DSNU777I 194 14:58:43.15 DSNUEXUP - CATMAINT UPDATE STATUS - BEGINNING ADDITIONAL CATALOG UPDATES PROCESSING.
DSNU777I 194 14:58:43.16 DSNUEXUP - CATMAINT UPDATE2 STATUS - BEGINNING SYSCOPY TABLE SPACE MIGRATION PROCESSING.
DSNU777I 194 14:58:43.16 DSNUECMO - CATMAINT UPDATE STATUS - UPDATING DIRECTORY WITH NEW RELEASE MARKER.
DSNU752I 194 14:58:43.18 DSNUECMO - CATMAINT UPDATE PHASE 1 COMPLETED.
DSNU010I 194 14:58:43.23 DSNUGBAC - UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE=0
  
```

DSNTIJTC job - Migration to DB2 10 Conversion Mode (CM8 or CM9)

The DSNTIJTC job invokes the CATMAINT UPDATE utility in order to migrate the current (V8 or DB2 9) catalog and directory to DB2 10 for z/OS. The end result of this migration process is a current DB2 Version 10 catalog.

In DB2 Version 10, DB2 will continue to enforce a restriction that there must be no outstanding utilities started from prior releases when running CATMAINT UPDATE on a non data-sharing system, otherwise message DSNU790I will be issued.

The CATMAINT UPDATE utility performs some or all of the following tasks during any given migration:

- Adds entries in the catalog and directory for new catalog objects:
 - Table Spaces
 - Tables
 - Indexes
 - Table Check Constraints
 - Referential Constraints
- Adds new columns to existing catalog tables.
- Adds additional values to existing catalog columns.
- Changes the definitions of existing catalog tables.
- Makes additional updates to the migrated catalog and directory.
- Additional updates.

Unique to each migration are additional changes that need to be made to the existing catalog and directory.

SMS is now a prerequisite of DB2 10. During the process of migrating from V8/DB2 9 to DB2 10 conversion mode, all the new indexes and new table spaces in the catalog and directory will be created as SMS-controlled.



SMS now a prerequisite



Important – All new indexes and new table space data sets WILL be SMS controlled.

“SMS now a prerequisite of DB2 10”.

So be prepared!

During migration from (V8 or DB2 9) to DB2 10 conversion mode, all the new indexes and new table spaces in the catalog and directory will be created as SMS-controlled with the extended addressability (EA) attributes.

DSNTIJSS provides SMS classes for customers without SMS in use.

- Environment created by DSNTIJSS is ONLY for DB2 Catalog and Directory data sets.
 - Other DB2 data sets such as logs and BSDS not covered.

Useful SMS reference material includes:

- z/OS DFSMS Introduction.
- z/OS DFSMS Implementing System Managed Storage.
- z/OS DFSMS Storage Administration Reference.
- The Redbook: Maintaining your SMS environment



SMS is now a prerequisite of DB2 10.

During the process of migrating from V8/DB2 9 to DB2 10 conversion mode, all the new indexes and new table spaces in the catalog and directory will be created as SMS-controlled.

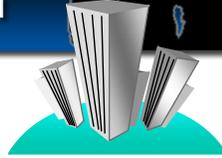
The environment created by the DSNTIJSS is only for DB2 Catalog and Directory data sets, which MUST be SMS managed in DB2 10. Other DB2 subsystem data sets such as logs and the BSDS are not accounted for in this environment.

Useful SMS reference material includes:

- z/OS DFSMS Introduction.
- z/OS DFSMS Implementing System Managed Storage.
- z/OS DFSMS Storage Administration Reference.
- The Redbook: Maintaining your SMS environment.

DB2 Catalog Evolution

The DB2 catalog continues to grow with every DB2 release.



DB2 Version	Table Spaces	Tables	Indexes	Columns	LOB Columns	Table Check Constraints
V1	11	25	27	269	0	N/A
V3	11	43	44	584	0	N/A
V5	12	54	62	731	0	46
V6	15	65	93	967	0	59
V7	20	84	118	1212	2	105
V8	22	85	132	1265	2	105
DB2 9	28	104	165	1643	3	119
DB2 10	95 (104-9)	134	233	1922	18	119

Does not include objects for XML Schema Repository

Complete release summary of DB2 10 changes (CM + ENFM)

60(2+11+47) new table spaces, 7 old table spaces dropped.

18(3+11+4) new tables

44(22+11+11) new indexes

33(25+8) new columns (on old tables)

Full details

Here is the list of changes during DSNTIJTC to migrate to DB2 10 CM9

25 new columns

22 new index

2 new table spaces

3 new tables

no change in check constraints or new check constraints.

changing the meaning of the catalog columns: there is nothing to change in terms of DSNTIJTC, nothing is done in DSNTIJTC to change the meaning.

it is just a change of how other db2 components use the columns and decide what values to put in or how to interpret the values..

DSNTIJTC doesn't actively change the values from A to B.

During the ENFM, we create more objects:

8 additional new columns

11 new LOB table spaces

11 new aux tables

11 new aux indexes

47 new table spaces

4 new tables

11 new indexes

7 old table spaces dropped



DB2 10 for z/OS

The Migration Process

Enable New-Function-Mode

ENFM and NFM Considerations



Attention: You cannot return to the previous release (DB2 V8 or DB2 9) once you enter ENFM.

Do NOT move to ENFM until you are certain that you will not need to return.

HOWEVER

The code base for DB2 10 ENFM and NFM is the same as for CM.

You can return to CM* from ENFM or NFM if necessary.



Moving to Enabling New Function Mode

DSNTIJEN job:

- CATENFM START places DB2 in Enabling New Function Mode.
- Data sharing groups must only have DB2 10 CM members.

ENFM will handle the “**Catalog Restructure**” changes

Existing catalog and directory table spaces will be dropped and the tables will be moved to the new SMS-controlled table spaces.

- DB2-defined indexes on these tables in the catalog and directory will also become SMS-controlled.
- User-defined catalog indexes continue as before.

New columns will be added.

A number of new Indexes will also be created

Data sharing groups must only have DB2 10 members - The first time DSNTIJEN runs, DB2 saves the RBA or LRSN of the system log in the BSDS.

The DSNTIJEN job invokes the CATENFM and REORG (with SHRLEVEL REFERENCE) utilities in order to convert a DB2 10 conversion mode catalog to ENFM level and complete the catalog restructure changes, which includes:

- Move the existing catalog tables to the new PBG table spaces. Each table with its own table space. These table spaces use row-level locking and all the tables are in reordered-row format.
- Convert SCTR, SYSLGRNX, SYSUTIL and SYSUTILX from EBCDIC to UNICODE.
- Merge SYSUTIL and SYSUTILX tables.
- Drop the old catalog table spaces.

Catalog and directory data sets under SMS control

During the ENFM process, some of the existing catalog and directory table spaces are dropped and the tables moved to the new SMS-controlled table spaces. The DB2-defined indexes on the tables in these catalog and directory table spaces also become SMS-controlled during the ENFM process while the user-defined catalog indexes stay the same way as before.

During ENFM, we create more objects:

- 8 additional new columns
- 11 new LOB table spaces
- 11 new aux tables
- 11 new aux indexes
- 47 new table spaces
- 4 new tables
- 11 new indexes
- 7 old table spaces dropped



DB2 10 for z/OS

The Migration Process

“Catalog Restructure”

Resolving concurrency issues occurring among Binds. Rebinds (auto) are not scheduled remember.

SQL statements are in SYSPKSTMT. Column is Binary (for bit data). Pre V8 starts in EBCDIC, since then Unicode. Both in table makes it difficult, with the SQL also split among rows.

64 gig limit on SPT01. APAR PK80375 will permit compressing of SPT01. Possible 85% gain. However LOBs do not have compression.

Removing links for concurrency.

DSNTIJEN has currently 9 TS changes (5 with links, SYSUTILX and others).

Catalog Restructure enhancement summary

- DB2 (SMS) managed catalog and directory data sets
- New CLOB/BLOB columns to the catalog
 - Merge records that store SQL statements' text
- Reduce catalog contention
 - Removal of links
 - Change to row-level locking
- Convert some catalog and directory table spaces to partition-by-growth (PBG)
 - With MAXPARTS = 1 DSSIZE 64 G
- Combine SYSUTIL and SYSUTILX into a single table SYSUTILX



Catalog Restructure is comprised of the following functions:

1. Reduce catalog contention by removing links and change to row-level locking
2. Make management of catalog and directory data sets easier by using z/OS Storage Management Subsystem (SMS).
3. Convert some catalog and directory table spaces to partition-by-growth (PBG), with MAXPARTS = 1
4. Add new LOB columns to the catalog (tables SYSINDEXES, SYSPACKSTMT, SYSVIEWS, SYSTRIGGERS, DBDS, SPT01) to merge records that store SQL statements' text.
6. Combine SYSUTIL and SYSUTILX into a single table SYSUTILX.
7. For documentation purpose, summarize catalog changes from other line items.

Similar to V8 and DB2 9, DB2 10 will also have an enabling-new-function mode process. This line item primarily focuses on the major catalog restructure changes that take place in this process. This line item will also address all the other catalog changes that take place during migration to DB2 10 conversion mode.



DB2 managed data sets - Benefits

Minimize user's effort to maintain data sets.

- No need to allocate data sets and extended pieces (A002, A003 etc...)
- No need to allocate data sets as part of the migration (next release)
 - We still have DSNTIJIN in this release because there are new catalog indexes to be created during migration and we cannot make them DB2 managed until leaving CM.
- No need to allocate data sets for the shadow for online REORG.
- No need to estimate the size for the data sets (had to provide space allocations in the DEFINES)
 - DB2 will use a default for the primary and a sliding scale for secondary.
 - Minimize the chance of running out of extends before reaching the maximum data set size.
- SMS will determine which data set goes to which volume.
- Minimize outage due to improper data set allocations.

New fields in installation/migration panels (CLIST)

- SMS information (data class, storage class, management class) stored in ZPARM.

Convert all the catalog and directory table spaces and DB2-defined indexes to SMS-controlled.



Row level locking and removing links

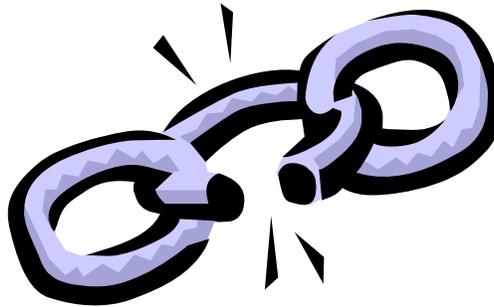
SYSDBASE, SYSPLAN, SYSDBAUT, SYSVIEW, SYSGROUP and DBD01 had links

These table spaces used page level locking because of the links.

SPT01, SYSOBJ, and SYSPKAGE are also processed in ENFM.

All of these table spaces will be removed and the tables within each will be moved to new PBG table spaces

- Row level locking
- New row format
- Partition-by-growth
- One table per table space
- Referential Integrity in place of links
- DSSIZE 64G
- MAXPARTS 1



- Links removal.
- Add new foreign key to the catalog tables that originally had link to a parent table.



Row level locking and removing links - Benefits

Row level locking minimizes timeout/deadlock for concurrent catalog access:

- DDL
- DML
- Grant/revoke
- Commands (DISPLAY, BIND/REBIND/FREE etc)
- Utilities (e.g. RUNSTATS)

Less or no SQL timeout when running BIND at the same time therefore increased availability for your applications.

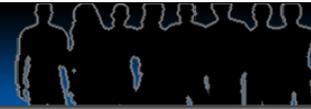
Reduced catalog contention:

- Concurrent Binding and running of DDL
- Multiple binds in parallel

Improve catalog concurrency has benefits for all

- Application Developer/ Architect binding and running applications
- Production DBA responsible for performance and running RUNSTATS.
- Development DBA responsible for database design and running the DDL.
- System Programmer responsible for system performance and running RUNSTATS.

With this enhancement contention on the catalog should be reduced allowing multiple binds, DDL and utilities to run without deadlocks or timeouts.



Eliminate 64GB limit on Catalog & Directory – e.g. SPT01

DB2 10 will relieve SPT01 space problems

SPT01 can grow beyond 64GB, when system in DB2 10 NFM.

LOBs used for package data inline LOB compression

Hence Application Developer creates packages using:

- Creates Packages using BIND PACKAGE.
- Creates Packages using CREATE TRIGGER.
- Creates Packages using SQL procedures.
- Binds a package, using plan stability.

DB2 extends beyond 64G when needed.



Binds and rebinds don't fail for lack of SPT01 space.

Aspects of a successful solution:

Binds and rebinds don't fail for lack of SPT01 space.

DB2 10 will relieve SPT01 space problems

SPT01 can grow beyond 64G, when system in DB2 10 NFM. Hence:

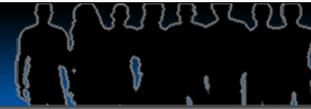
Application Developer creates packages using BIND PACKAGE.

- Creates packages using CREATE TRIGGER.
- Creates packages using SQL procedures.
- Binds a package, using plan stability.

DB2 extends SPT01 beyond 64G when needed.

When the SPT01 table space approaches 64G:

- Turn on compression for SPT01
- Reduce the size of the inlined LOB



DB2 10 for z/OS

Fallback

Returning to a prior release or mode



Returning to a previous release from DB2 10

Fallback is only supported from DB2 10 CM.

Migrate to DB2 10 CM9 then you can ONLY fallback to DB2 9 NFM

Migrate to DB2 10 CM8 then you can ONLY fallback to DB2 8 NFM

Fallback SPE (PK56922) **must** be applied beforehand.

Plans and packages bound (or rebound) in DB2 10 CM are automatically rebound in DB2 V8 NFM or DB2 9 NFM.

Objects with functional dependencies are indicated by an 'O' in the IBMREQD catalog columns. None should exist.

- Frozen on return to DB2 V8 or DB2 9.

Fallback to DB2 V8 NFM or DB2 9 NFM is only supported from DB2 10 CM.

Falling back is the process of returning to a supported previous release of DB2, after a successful migration of the catalog and directory to DB2 10 for z/OS Conversion Mode (CM). Supported releases are DB2 V8 for z/OS, and DB2 9 for z/OS.

Prerequisite changes to the fallback release should be installed before migrating to DB2 10 for z/OS; that is, to ensure fallback to the previous release, it is necessary to apply the required maintenance and the fallback SPE to the fallback release (DB2 V8 or DB2 9) before migrating to DB2 for z/OS Version 10. (The changes to the fallback release are necessary to tolerate the migrated catalog, directory, and log records thus allowing fallback after use of the new facilities).

The new facilities of this release require extensions to DB2. The extensions are upward compatible so that those objects which are not dependent on the new facilities are acceptable to the fallback release. Objects that are dependent on the new facilities are not acceptable to the fallback release. To distinguish the two classes of objects, DB2 will record the use of its new facilities. The recording is in the form of a release identifier (also known as the release dependency indicator) so that the scheme can also be used in subsequent releases.

There are two types of release dependencies: functional dependencies and design dependencies. Functional dependencies can apply to any type of object including plans and packages and are the result of the use of new external function. Design dependencies apply to plans and packages and are the result of SQL optimization enhancements. For example plans or packages that use sysplex parallel query processing will have design release dependencies but no functional release dependencies.

Some objects in this catalog that have been affected by function in this release might become frozen objects after fallback. Frozen objects are unavailable, and they are marked with a release dependency indicator. To record the release dependency indicators, every catalog table has a CHAR(1) column named IBMREQD. In later releases, the use of the IBMREQD column was extended to record functional release dependencies. The release dependency indicators are:

Y denotes a built-in row of the catalog as in Version 1 Release 1 or 2.

N denotes a row that is not built-in and which describes an object that is not dependent on any new external function of the release.

L denotes a row that is not built-in and which describes an object that is dependent on new external function(s) of DB2 for z/OS Version 8.

M denotes a row that is not built-in and which describes an object that is dependent on new external function(s) of DB2 9 for z/OS.

O denotes a row that is not built-in and which describes an object that is dependent on new external function(s) of DB2 10 for z/OS.



Fallback Considerations

Buffer pool enhancements

- Fall back process unchanged
- Any BPs changed to PGSTEAL=NONE will revert to their previous values
 - i.e. LRU if PGSTEAL was never altered
 - Otherwise the value from the last alter (under either release)
- On remigration, any BP which was set to PGSTEAL=NONE before fallback will revert to PGSTEAL=NONE.

Autonomic diagnosis & tuning for query performance issues

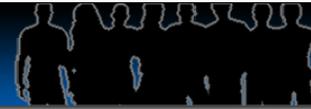
- Fall back process unchanged.
- No special procedure for the explain tables.
 - These tables in DB2 10 explain schema and UNICODE would simply work with the fallback subsystem as it was before after applying the SPE fallback APAR and completing the preparatory steps.

Buffer pool enhancements - Fallback

The fall back process is unchanged from the prior release. Upon fallback, the PGSTEAL attributes of any buffer pools which were changed to PGSTEAL=NONE will revert to their previous values, i.e. LRU if PGSTEAL was never altered to a different value, otherwise the value from the last alter (under either release) before the alter to PGSTEAL=NONE. Upon remigration, any buffer pool which was set to PGSTEAL=NONE before fallback will revert to PGSTEAL=NONE.

Autonomic diagnosis & tuning for query performance issues - Fallback

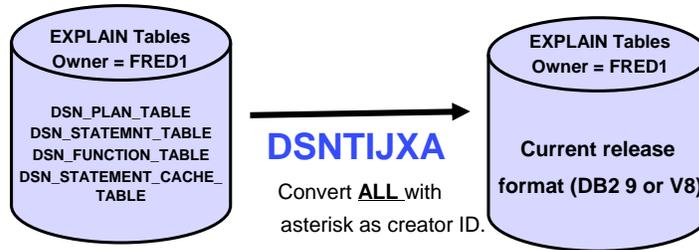
The fall back process is unchanged from the prior release. In this process, there is no special procedure to be done for the explain tables. These tables in DB2 10 explain schema and UNICODE would simply work with the fallback subsystem as it was before after applying the SPE fallback APAR and completing the preparatory steps.



DB2 10 for z/OS

***Migration Considerations
For Specific Features***

Migrate EXPLAIN tables to DB2 10 schema encoded in UNICODE - Prepare
DB2 10 EXPLAIN tables must be in latest schema format and in UNICODE.



Important - migrating V8 to DB2 10, you will not be able to convert DSN_STATEMENT_CACHE_TABLE to DB2 10 format until DB2 enters DB2 10 new-function mode. (BIGINT not supported in V8)

- However, DB2 10 will tolerate DSN_STATEMENT_CACHE_TABLE V8 format.
- Once DB2 is in DB2 10 new-function mode, rerun job DSNTIJXA to convert DSN_STATEMENT_CACHE_TABLE to DB2 10 format.



Part of the Autonomic diagnosis & tuning for query performance issues enhancements

Migrating EXPLAIN tables

This process migrates the explain tables to DB2 10 schema encoded in UNICODE. The explain tables migration process described here only covers migration from V8 NFM or DB2 9 NFM to DB2 10 CM, then to DB2 10 NFM. If customers have a DB2 subsystem in V7 or earlier, it is recommended to bring the system up to at least V8 NFM level by following the recommended migration procedure.

The migration process is divided into several phases:

1. Preparatory phase

- DB2 is assumed at DB2 9 NFM level. (The steps for migrating from V8 NFM is similar, see step 5 below).
- Customers are recommended to migrate the explain tables to the current release of explain table schema format, that is either DB2 9 or V8, as early as possible to be ready for release to release migration. Customers can skip this step if they have migrated to the latest explain schema already.
- A sample, customizable job to migrate the explain tables is provided to ease this process. Use job DSNTIJXA to bring all explain tables belonging to a specified creator ID into current release format. To convert all non-compliant explain tables regardless of the creator, specify an asterisk as the creator ID. Note: If you are migrating to DB2 10 from DB2 V8, you will not be able to convert DSN_STATEMENT_CACHE_TABLE to DB2 10 format until DB2 enters DB2 10 new-function mode. Until that time, DB2 10 will tolerate DSN_STATEMENT_CACHE_TABLE in V8 format. Once DB2 is in DB2 10 new-function mode, rerun job DSNTIJXA to convert DSN_STATEMENT_CACHE_TABLE to DB2 10 format.



Enhanced Monitoring support – Migration impact on SYSPACKSTMT

DB2 10 CM, a new STMT_ID column created (but NOT materialized) as INT NOT NULL.

During ENFM, STMT_ID column altered to BIGINT and a new system sequence object SYSIBM.DSNSEQ_PKS_STMT_ID also created as BIGINT.

ENFM REORG phase, populates STMT_ID column for any existing rows in SYSPACKSTMT table.

After ENFM, SYSPACKSTMT will be out of sync with SPT01 until REBIND of the existing package is done.

During BIND phase, the STMT_ID column is populated via the SYSIBM.DSNSEQ_PKS_STMT_ID sequence object for the new package.

Rebind of any existing pre-DB2 10 package for static SQL statements is required once the ENFM process is completed successfully.

- SYSPACKSTMT then in sync with SPT01
- STMT_ID column contains a unique value for each row in the SYSIBM.SYSPACKSTMT table.

When the statement identifier shows in the messages and traces for static statements, the user then can correlate the identifier to the STMT_ID column in the SYSIBM.SYSPACKSTMT table to identify the particular SQL statement in question.

The migration impact on the SYSIBM.SYSPACKSTMT table

A new STMT_ID column is created in Conversion Mode(CM) as INT NOT NULL. The column is not materialized in CM. The default value is zero for the column.

During ENFM processing, the STMT_ID column is then altered as BIGINT data type. A new sequence object SYSIBM.DSNSEQ_PKS_STMT_ID is created as BIGINT. This system sequence object is created to be used for the STMT_ID column, any attempted reference on this system-generated sequence object is not allowed and resulted in with SQLCODE -20142.

In ENFM REORG phase, the STMT_ID column is populated for any existing rows in the SYSIBM.SYSPACKSTMT table. After ENFM process is completed successfully, SYSIBM.SYSPACKSTMT catalog table will be out of sync with DB2 directory residing in SPT01 until REBIND of the existing package is done.

During BIND phase after ENFM process, the STMT_ID column is then populated via the SYSIBM.DSNSEQ_PKS_STMT_ID sequence object for the new package, the sequence value starts at n+1 and increments by 1 where n is the number of the existing rows in the SYSIBM.SYSPACKSTMT table. After a FREE package, n is actually the current contents of the sequence.

Rebind of any existing pre-DB2 10 package for static SQL statements is required once the ENFM process is completed successfully for any user who wants to utilize the enhanced messages and traces that provide the statement identifier information. Once the package is rebound or bound, SYSIBM.SYSPACKSTMT table is in sync with DB2 directory residing in SPT01 and the STMT_ID column contains a unique value for each row in the SYSIBM.SYSPACKSTMT table. When the statement identifier shows in the messages and traces for static statements, the user then can correlate the identifier to the STMT_ID column in the SYSIBM.SYSPACKSTMT table to identify the particular SQL statement in question.



Migration impact on SYSIBM.DSN_PROFILE_TABLE_IX2_ALL index

Index on **SYSIBM.DSN_PROFILE_TABLE**

- DB2 10 adds columns, ROLE , PRDID, and GROUP_MEMBER

No migration considerations for V8 to DB2 10.

After migrating update format of **SYSIBM.DSN_PROFILE_TABLE**:

- Use SPUFI to alter/add the new columns to the table and index, so that it includes the new keys in the required sequence
- Job DSNTIJOM.
- Job DSNTIJXA to run REXX exec DSNTXTA, specifying schema name SYSIBM. DSNTXTA.

To utilize the new DB2 10 system profile monitoring functions related to connections and threads, the new index must be defined.

- If not, a DSNT745 for START PROFILE command issued
- A user who does not use the new DB2 10 system profile monitoring functions can use profile monitoring properly with the old DB2 9 index. However, any usage of the new DB2 10 system profile monitoring functions in the profile table requires the new DSN_PROFILE_TABLE_IX2_ALL definition.

Profile tables contain information about how DB2 executes or monitors a group of statements. **SYSIBM.DSN_PROFILE_TABLE** is used to specify which SQL statements are monitored or tuned by each monitor profile. Each row defines a particular monitor profile.

DB2 10 adds three columns, ROLE , PRDID, and GROUP_MEMBER, to the index SYSIBM.

DSN_PROFILE_TABLE_IX2_ALL. There is no migration consideration for customers who migrate from V8 to DB2 10 because these objects are not created in V8. After migrating from DB2 9 to DB2 10, customers who have an existing SYSIBM.DSN_PROFILE_TABLE can use any of the following techniques to bring it into current release format:

- Use SPUFI or a similar tool to process SQL statements to alter/add the new columns to SYSIBM.DSN_PROFILE_TABLE and to drop and recreate SYSIBM.DSN_PROFILE_TABLE_IX2_ALL so that it includes the new keys in the required sequence
- Use job DSNTIJOM to alter/add the new columns to SYSIBM.DSN_PROFILE_TABLE and drop and recreate SYSIBM.DSN_PROFILE_TABLE_IX2_ALL so that it includes the new keys in the required sequence.
- Use job DSNTIJXA to run REXX exec DSNTXTA, specifying schema name SYSIBM. DSNTXTA will then alter/add the new columns to SYSIBM.DSN_PROFILE_TABLE and drop and recreate SYSIBM.DSN_PROFILE_TABLE_IX2_ALL so that it includes the new keys in the required sequence.

When the user wants to utilize the new DB2 10 system profile monitoring functions related to connections and threads, the new index definition must be used. A DSNT745 code will be issued for START PROFILE command if the new index definition is not created along with the new DSN_PROFILE_TABLE definition with columns ROLE and PRDID. A user who does not use the new DB2 10 system profile monitoring functions can use profile monitoring properly with the old DB2 9 index. However, any usage of the new DB2 10 system profile monitoring functions in the profile table requires the new DSN_PROFILE_TABLE_IX2_ALL definition.



XML Schema Validation in the engine

The UDF version of DSN_XMLVALIDATE is deprecated in DB2 10, and will be removed in the release that follows.

To use the new built in scalar function DSN_XMLVALIDATE, an SQL package rebind is needed.

- Not rebinding the package, means the old UDF DSN_XMLVALIDATE will be invoked.

After rebinding the package,

- if DSN_XMLVALIDATE is unqualified or qualified with SYSIBM in the call, the new built in scalar function will be invoked;
- if DSN_XMLVALIDATE is qualified with SYSFUN in the call, the old UDF DSN_XMLVALIDATE will be invoked.

XML schema validation in the engine - Migration

The user defined function version of DSN_XMLVALIDATE is deprecated in DB2 10, and will be deleted in the release that follows. Deprecation in DB2 10 means we still document it and allow users to use it. To use the new built in scalar function DSN_XMLVALIDATE, an SQL package rebind is needed. If a user does not rebind the package, the old UDF DSN_XMLVALIDATE will be invoked.

After rebinding the package, if DSN_XMLVALIDATE is unqualified or qualified with SYSIBM in the call, the new built in scalar function will be invoked; if DSN_XMLVALIDATE is qualified with SYSFUN in the call, the old UDF DSN_XMLVALIDATE will be invoked.



Dynamic Statement Cache enhancements

New column, **LITERAL_REPL**, added to user table **DSN_STATEMENT_CACHE_TABLE**

- Install / Migration process will handle adding of column
- CHAR(1) NOT NULL WITH DEFAULT

'R'	Literals have been replaced with '&'
'D'	Literals have been replaced with '&' and may have an identical value for STMT_TEXT for another cached statement, but literal reusability matching criteria failed.
' '	CONCENTRATE STATEMENTS has a value of OFF (default value).

Dynamic Statement Cache enhancements - Migration

A new column, **LITERAL_REPL**, is added to the **user table DSN_STATEMENT_CACHE_TABLE**.

The DB2 10 Install / Migration process is modified to handle adding the new column to table **DSN_STATEMENT_CACHE_TABLE**.

The column **LITERAL_REPL** is CHAR(1) NOT NULL WITH DEFAULT.



UTS support for Member Cluster

Pre DB2 10, values "K" or "I" in the TYPE column of SYSTABLESPACE indicated a MEMBER CLUSTER table space structure.

- "K" = >64Gb and "I" = <64Gb

In DB2 10 new column MEMBER_CLUSTER added to SYSTABLESPACE

- Populated during ENFM process for a MEMBER CLUSTER table space.

Original values of 'I' or 'K' replaced with blank or L respectively.

- New column SYSTABLESPACE MEMBER_CLUSTER will have value of 'Y'.

SYSTABLESPACE.TYPE = I

Mode	Value of SYSTABLESPACE.TYPE	Value of SYSTABLESPACE.MEMBER_CLUSTER column
Before ENFM	I	Blank
After ENFM	Blank	Y

SYSTABLESPACE.TYPE = K

Mode	Value of SYSTABLESPACE.TYPE	Value of SYSTABLESPACE.MEMBER_CLUSTER column
Before ENFM	K	Blank
After ENFM	L	Y

Universal Table Space support for member Cluster - Migration

Prior to this line item, value "K" or "I" in the TYPE column of the SYSTABLESPACE catalog table indicates the table space has MEMBER CLUSTER structure.

A new column MEMBER_CLUSTER on the SYSTABLESPACE catalog table is added, and it will be populated during the ENFM of migration process for existing MEMBER CLUSTER table space.

The original value of 'I' or 'K' SYSTABLESPACE TYPE column will be replaced with blank or L respectively, and the new column SYSTABLESPACE MEMBER_CLUSTER will have value of 'Y'.



DB2 10 for z/OS

Eliminate DDF Private Protocol

Eliminate DDF PRIVATE protocol - Migration

The purpose of this enhancement is to formally remove private protocol support from DB2 10 and enhance the capabilities of DRDA protocol so customers can convert those applications which utilize private protocol today, to eventually use DRDA protocol. This line item will also provide capabilities in V8 and DB2 9 to assist our customers in migrating all of their plans and packages to DRDA protocol prior to an attempt to migrate to DB2 10 since no private protocol plan or package can exist in a catalog that is the source of a migration to DB2 10.

One of the first tasks of the DB2 10 DSNTIJTC would be to check whether any plans or packages exist in the catalog are bound with DBPROTOCOL(PRIVATE). If DBPROTOCOL(PRIVATE) objects do exist, DB2 10 DSNTIJTC will halt and indicate the existence of those objects prevented DSNTIJTC from completing. The customer will be instructed that they must perform actions to remove DBPROTOCOL(PRIVATE) objects back in V8 or DB2 9 before continuing with DB2 10 DSNTIJTC. If the DB2 10 DSNTIJTC does not find any private protocol objects in the catalog, DSNTIJTC will continue with its other migration steps.

Back in V8 or DB2 9, the V8 or DB2 9 toleration SPE to handle a DB2 10 CM catalog would detect that the catalog is at a partial DB2 10 CM level which would still have it not allowing the use of DBPROTOCOL(PRIVATE) bind option during any BINDs and REBINDs since the customer must complete the actions to migrate any remaining DBPROTOCOL(PRIVATE) plans and packages to DBPROTOCOL(DRDA). The customer would now complete the task of migrating all plans and packages to DBPROTOCOL(DRDA).

If a fallback to V8 or DB2 9 were to occur again, the V8 or DB2 9 toleration SPE to handle a DB2 10 CM catalog would still prevent the addition of any new DBPROTOCOL(PRIVATE) objects to the catalog.



DB2 9 Deliverable

DB2 9 for z/OS delivered the following functionality

- Warning when PRIVATE protocol used at BIND time
- Prevent PRIVATE being the default option
- IFCID records to determine applications using PRIVATE
- Analysis Tool to identify and convert plans and packages
 - Additionally identify Aliases required at remote sites

DB2 10 will no longer support **ANY** Private Protocol

- Customers should convert to DRDA prior to DB2 10
- Enhancements to V8 and DB2 9 DRDA support to aid conversion



Enhancements to V8 & DB2 9

The following is delivered to facilitate conversion

- Ability to Enable or Disable Private protocol (PK92339)
 - Via ZPARM PRIVATE_PROTOCOL
 - Enables testing to ensure private is eliminated
- Allow Private protocol to access V8/DB2 9 data sharing members
 - Until all members are at DB2 10 CM
- DRDA alias resolution processing to be enabled or disabled (PK64045)
- Provide Enhanced Tools to identify plans and packages requiring conversion (PK64045)
 - DSN1PPTP analyses IFCID157 trace records for input into DSNTP2DP
 - DSNTP2DP reports on applications using private protocol
- Program to scan SYSSTMT and SYSPACKSTMT for invalid syntax (PK64045) - DSNTPPCK

Provide a DSNTIJTC to switch a PLAN and/or PACKAGE bound with DBPROTOCOL(PRIVATE) to DBPROTOCOL(DRDA) when it is determined that a PLAN or PACKAGE does not have any remote location dependent statements or dynamic SQL statements
 ZPARM MACRO DSN6FAC

DSN1PPTP details

-START TRACE(PERFM) CLASS(30) IFCID(157) DEST(SMF)

will go through the SMF data file looking for IFCID 157 trace entries amongst the SMF 102 records and then put out the information contained in those trace entries to a JCL DD data set, IFCID157. The format of the IFCID157 file will be as follows:

LOAD	PLANNAME	CORRNAME	CORRNMBR	CONNECTTYPE	SECTN	R/W	CALLTYPE
	SERVING_LOCATION	PROGRAM/PACKAGE_					
YES	LI883A	LI883J5	'BLANK'	DB2 CAF		1	W
	AUXCALL	STLEC4B		LI883A			
YES	LI883B	LI883J5	'BLANK'	DB2 CAF		1	W
	AUXCALL	STLEC4B		LI883B			
YES	LI883C	LI883J5	'BLANK'	DB2 CAF		1	W
	AUXCALL	STLEC4B		LI883C			
YES	LI883D	LI883J5	'BLANK'	DB2 CAF		1	R
	OPENCALL	STLEC4B		LI883D			

The actual record length will be about 133 bytes.

The LOAD column in the IFCID157 file is being added so that the data from the IFCID157 trace can be then loaded into a DB2 table which can then be used by the DSNTP2DP via the PPINFOTABLE parameter specification to fill in any target locations for plans and packages.

DSNTP2DP will scan catalog and output from DSN1PPTP creates

Creates 2 part alias names for target systems for 3 part alias names found locally

Creates plan/package commands to convert protocol

Input parm PPINFOTABLE – table containing IFCID 157 trace info



DB2 10 Enhancements

Enhancements to DB2 10 are

- Remove PRIVATE PROTOCOL from DB2
- Attempt to load/execute object with DBPROTOCOL column = 'P' will be tolerated, but any reference to non-local data will fail with SQLCODE -904, reason code 00E3001E
- REBIND will always use DBPROTOCOL(DRDA) regardless of object's current DBPROTOCOL value
- BIND PACKAGE COPY will create DBPROTOCOL(DRDA) object regardless of source object of copy
- DRDA alias resolution will always be performed
 - SQL statement text sent to remote system will contain object names substituted for 3-part name alias references
 - Aliases will not be needed at remote locations
 - Occurs during dynamic PREPARE and EXECUTE IMMEDIATE and remote package bind

Currently, when processing an SQL statement from an application under control of a package or plan bound with DBPROTOCOL(DRDA), the SQL statement text is sent to a remote location as it was received from the application. Thus, if the SQL statement referenced an alias, and that alias resolved to a remote 3-part name, then the statement with the alias reference would be sent unchanged on to the remote location (identified by the alias) for processing. This is not the behaviour of private protocol, in that the alias reference in the SQL statement is resolved to the remote object name prior to the statement being sent to the remote location. Also, the private protocol behaviour occurs whether the statement is being processed via embedded dynamic SQL processing or embedded static SQL processing. By not resolving alias references into the SQL statement text during DRDA protocol processing, one might need an alias at the remote location to resolve the object reference. Also, the following alias definition scenario will not currently work with DRDA protocol:

- create an alias named ALIAS1 in location DB2A to point to table USER.TABLE at DB2B
- create an alias named ALIAS1 in location DB2B to point to table USER.TABLE at DB2A
- with private protocol, USER.TABLE is accessed at DB2B when specifying ALIAS1 at DB2A and USER.TABLE is accessed at DB2A when specifying ALIAS1 at DB2B
- with DRDA protocol, the ALIAS1 access on either system would result in an error because the alias reference would lead to a continuous cross-system loopback until the nesting of levels reached 16 levels

In DB2 10, aliases will be resolved with DRDA protocol during:

- a PREPARE of a dynamic SQL statement (if DEFER(PREPARE) is also specified, the SQL statement text will have the alias reference resolved but the statement will not be sent to the remote location for processing until a request to execute the statement is made by the application)
- an EXECUTE IMMEDIATE of a dynamic SQL statement
- a remote package bind of a static SQL statement



Migration Steps

Activate trace while running Private protocol applications

- Identify embedded dynamic SQL and remote sites referenced

Enable Alias Resolution

- To properly convert applications

Build Private Protocol Information Table

- Output from DSN1PPTP (DSNTIJPT installation job)

Run Protocol REXX Catalog Analysis Program (DSNTP2PD)

- Provides commands for plan and package management

Action Plan/Package Management commands

Test the application

- Check for errors from DRDA conversion
- SQLCODE -805 suggests embedded dynamic SQL
 - Fix using BIND PACKAGE COPY of indicated package

To properly convert any application that is currently using private protocol, alias resolution processing must be enabled for DRDA protocol. This is accomplished by either changing and rebuilding an existing DB2 subsystem parameters module or creating a new subsystem parameters module which includes setting the DRDA_RESOLVE_ALIAS parameter of the DSN6SPRM ZPARM macro to YES. This new or changed subsystem parameters module can then be loaded or reloaded via the -SET SYSPARM DB2 command.



Eliminate DDF Private Protocol – Coexistence support

We will see (V8 + DB2 10) and / or (DB2 9 + DB2 10) scenarios

- Some members will still be able to support Private Protocol inbound access.

Enabling an earlier release of DB2 to continue to use private protocol to access the data sharing group (in coexistence mode)

- Implement member-specific routing to DB2 10 coexistence data sharing group (i.e. LULIST)
- DB2 requester will attempt connect to LUNAMEs in LULIST until success (at least one member of coexistence group must be either V8 or DB2 9)
- If more than one member of coexistence group is V8 or DB2 9, then requester will balance requests based on DB2 SNA Sysplex TPN exchange

Eliminate DDF PRIVATE protocol - Coexistence

Coexistence of multiple releases of DB2 is unchanged from the prior release. However, as a V8 or DB2 9 data sharing group undergoes a gradual migration to DB2 10CM, some members will still remain able to support private protocol inbound access while others will not. To enable an earlier release of DB2 to continue to use private protocol to access the data sharing group while some members still permit inbound private protocol and also make sure that the requester only accesses those members which still allow inbound private protocol, the following changes will be made as part of this line item:

- Code will be added in DB2 10 which cause a DB2 to register its version level with the WLM sysplex routing services interface.
- Code will be added in V8 and DB2 9 via the DB2 10 toleration SPE and DB2 10 to have the DB2 utilize the version level information retrieved from WLM sysplex routing services and return that information to any requester via the reply to the DB2 sysplex TP (VTAM Transaction Program) request
- Code will be added to the V8 and DB2 9 requesters that will cause it to issue a DB2 sysplex TP request to the LUNAME(s) of the location to ascertain the levels of the server systems if it had previously received a private protocol VTAM communications connection request failure with VTAM sense code X'10086021' and based on the reply from the DB2 sysplex TP request, the following will occur:
 - if at least one of the server systems is not DB2 10 or the server list does not contain any version level indicators (no members are running DB2 10), then the requester will then route the private protocol request to a remaining non-DB2 10 system
 - if all server systems are DB2 10, then the requester will return an SQLCODE -904 with reason code X'00D31026', resource type X'1001' (System Conversation), and resource information being of the format LU.MODE.RTNCD.FDBK2.RCPRI.RCSEC.SENSE where the LU will be the LUNAME to which a conversation was being established, MODE will be the VTAM logmode that was used in the connection attempt, and RTNCD.FDBK2.RCPRI.RCSEC.SENSE will 00.0B.0004.0008.10086021



Are you ready for DB2 10?

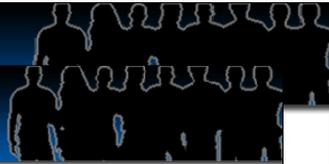
- Check prerequisites
- Migration planning workshop
- Plan gains, testing, memory, and performance
- Build detailed migration plan
- Contact vendors
- Get all the parts out of the box
- Check information APAR
- Apply required service
- Run premigration checks DSNTIJPA (or M) early and often
- Resolve incompatible changes
- Get rid of private protocol
- Convert to packages from DBRMs in plans
- Upgrade plan table formats to Unicode V8 or DB2 9 level
- Get ready for SMS
- Save performance and access path information

This is a checklist on getting ready for DB2 10. You will want to get this work done before starting to work with the new code. When this checklist is completed, you can move to the checklists in the DB2 Installation Guide.

See the checklist in the DB2 10 Upgrade paper.

<ftp://ftp.software.ibm.com/software/data/db2/zos/presentations/migration/>

<ftp://ftp.software.ibm.com/software/data/db2/zos/presentations/migration/upgrading-to-db2z10-miller.pdf>



- Run checks against your V8 / DB2 9 system
- Review documentation
- Migration planning
- Resolve inconsistencies and release incompatible changes
- Establish a project plan and team
- Assess new features & training plans
- Complete any prerequisite projects, e.g. tools
- Collect performance baselines

- Establish test plans
- Determine conversion & coexistence (data sharing) goals
- Keep your IBM team informed

Questions and Answers about DB2 10 for z/OS

March 28, 2011

This article is a pointer to resources and information about DB2 10 for z/OS. Customers have asked many questions on webcasts and at conferences. Many of the pointers have more extensive information to respond to the questions.

Even though this product is generally available, changes are likely to occur for some of these answers. 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.

How can I get more detailed information about DB2 10? Are there any IBM Redbooks publications out now? How can I get the DB2 books?

More details are provided on the web:

DB2 main web page <http://www.ibm.com/software/data/db2/zos/>

DB2 10 web page Get white papers from this page, such as business value, temporal data, or see what customers have said about DB2 10.

<http://www.ibm.com/software/data/db2/zos/db2-10/>

DB2 books, Information Center

<http://www.ibm.com/support/docview.wss?uid=swg27019288>

DB2 best practices web page. These best practices present advice on the optimal way to use DB2 for z/OS to satisfy key business data processing needs. These presentations and articles are authored by leading experts in IBM's development and consulting teams.

<https://www.ibm.com/developerworks/data/bestpractices/db2zos/>

DB2 for z/OS IBM Redbooks publications <http://www.redbooks.ibm.com/cgi-bin/searchsite.cgi?query=DB2+AND+z/OS&SearchOrder=4&SearchFuzzy=>

DB2 10 booklet

<ftp://public.dhe.ibm.com/common/ssi/ecm/en/imm14075usen/IMM14075USEN.PDF>

The first DB2 10 IBM Redbooks publications are DB2 10 for z/OS Technical Overview and Extremely pureXML, discussing XML in DB2 9 and 10. Watch for Performance Topics, anticipated in the second quarter of 2011.

You can also learn more at upcoming conferences, including Share, IDUG, and IOD. Get information from the recent conferences from their web sites.

IDUG	www.idug.org
IOD	www.ibm.com/software/data/2010-conference/
Share	www.share.org
IBM Education	www.ibm.com/training/us/db2zospath
DB2 events	www.ibm.com/software/data/db2/zos/events.html

We'll be getting more information out at the conferences: Share, GSE, IOD, and IDUG. See the IDUG conference content if you have a membership. Register to join IDUG if you don't. Register for the virtual IOD conference.

Here are some sessions from the IDUG EMEA 2010 track for DB2 10:

- A01 - DB2 10 for z/OS Overview
- A02 - DB2 10 for z/OS Migration Planning
- A03 - DB2 for z/OS Trends and Future Direction
- A04 - DB2 9 & 10 for z/OS Enhancements for System Programmers
- A05 - DB2 10 for z/OS Security Enhancements
- A06 - DB2 10 Availability Enhancements
- A07 - DB2 10 - What's in it for SAP and Enterprise Applications
- A08 - What's new from the optimizer in DB2 10 for z/OS?
- A09 - DB2 for z/OS and Websphere Update
- A10 - Hash Access to DB2 Data - Faster, Better, Cheaper
- A11 - DB2 and System z Synergy
- A12 - DB2 10 Application topics – A sneak preview
- A13 - DB2 10 for z/OS Performance Preview
- A14 - Utilities Update - DB2 10 for z/OS
- A16 - The Next Level: Data Warehousing with DB2 10 and zEnterprise

How does DB2 10 use the zEnterprise?

Faster CPUs , more CPUs, and more memory means better DB2 performance and scalability. LSPR measurements of DB2 9 with the zEnterprise z196 show substantial reductions when compared to z10 processors. This transaction workload showed a range of 1.3 to 1.6 times CPU time reduction, with the best CPU reductions when more processors per LPAR are used. Including the larger number of faster processors (80 vs 64) and DB2 10 can mean fewer footprints. This has excellent synergy with DB2 10, which removes many single system scaling inhibitors.

Compression hardware improvements in zEnterprise are expected to increase DB2 data compression performance. Larger cache (192 MB in level 4) is expected to benefit DB2 workloads. DB2 for z/OS can take an advantage of cache optimization on zEnterprise. Translation Lookaside Buffer changes are expected to improve DB2 10 performance for 1MB page sizes. The 1 MB hardware page sizes improve DB2 performance and are only on z10 and zEnterprise. Hybrid architecture delivers new opportunities for DB2 query performance acceleration with IBM Smart Analytics Optimizer.

Combined with DB2 10 improvements CPU reduction, buffer pool management, relief for virtual storage constraint and latch contention, DB2 applications can enjoy significant cost reduction and scalability improvement on zEnterprise.

Should I upgrade to DB2 10? When and how should I migrate to DB2 10?

The answer to upgrading to DB2 10 is a definite Yes. The question is not so much whether to upgrade as when and how to upgrade. If you are running DB2 9 today, then DB2 10 is in your near future, giving you more room to grow, with higher limits, lower CPU, and more for less. If you are running DB2 V8 today, then you have a choice of jumping to DB2 9 or directly to DB2 10. Both choices are fully supported with migration and fallback.

While DB2 10 is expected to be better than prior versions, it will have maturity and service delivery like other software, with more defects at first, then fewer as the software matures. Determining when the software is ready for a specific customer and when the customer is ready for the software depends upon the specific customer resources for testing, prior experience, and the value for the improvements versus the need for stability. Many customers depend upon tools or other software, and having that software work with DB2 is a prerequisite. Customers who migrate early will need more robust test and service processes.

See the Upgrade to DB2 10 paper at the end of this presentation for much more on this topic.

<ftp://ftp.software.ibm.com/software/data/db2/zos/presentations/migration/db2-10-migration-planning-idug-emea-2010-nov-miller.pdf>

When will DB2 10 for z/OS be generally available or GA?

DB2 10 for z/OS was generally available world wide as of October 22, 2010.

<http://www.ibm.com/software/data/db2/zos/db2-10/>

When will DB2 V8 go out of support?

DB2 end of service is scheduled for April 2012, as announced in August 2010. Customers should be getting ready for their next migration if they are on DB2 V8. Any customers who are on V7 or lower should be migrating to V8 as soon as possible, leaving the versions which are out of service. Extended service can be provided for a fee, if you can't migrate and need support. Contact your IBM representative. Click on Product Support Lifecycle from this page to see the service cycle:

http://www-947.ibm.com/support/entry/portal/Planning/Software/Information_Management/DB2_for_z-OS

<http://www.ibm.com/common/ssi/cgi-bin/ssialias?infotype=AN&subtype=CA&appname=g pateam&supplier=897&letternum=ENUS910-169&pdf=yes>

V8 has been withdrawn from marketing. If we're still on V7, what are our options?

The withdrawal from marketing means that new customers cannot order V8. As an existing customer, you can order and migrate to DB2 V8. Talk to your IBM representative to get DB2 software or migration support. Ask about the Migration Planning Workshop as well.

How should I start getting ready for DB2 10?

Are you ready for DB2 10? Are you running in DB2 9 NFM or DB2 V8 NFM? Have you looked though the information APARs? Have you contacted your vendors? Do you have the required service applied to your current system?

Have you run the premigration job and worked to resolve the incompatibilities? Have you gotten rid of private protocol? Have you converted to packages instead of DBRMs directly in plans? Those are first tasks, along with planning what you expect to gain and the process for testing the new version.

- Plan the migration process. Make sure that you have the prerequisites and current software that works with the new version. Check with your vendors. See the planning for migration presentations referenced below. Migration planning workshops are available at no charge for DB2 V8, DB2 9, and DB2 10 and can save you time, providing experience from other customers and IBM.
- Check for incompatible changes. As you go through the process, check for the incompatible changes and deprecated function in DB2 10 as well. That way, you can avoid some of these problems and make the changes in the normal course of work, so there is less effort when you upgrade to DB2 9 or DB2 10. Run the premigration jobs (DSNTIJP9 and DSNTIJPA) early and often.
- Read the Installation Guide, the Preventive Service Planning (PSP) bucket and Authorized Program Analysis Report (APAR) installation text. The Consolidated Service Test (CST) can provide a much more stable level of service for a broad stack of products. CST has worked for very small customers, very large customers, and those in between.
- Develop a test plan for your unique workload and environment. CST can help for more general processing work. Collect performance information and access path information. DB2 9 and 10 package stability can help you get back to the original access paths and are generally used in a release migration.
- Put plans in place to ensure that each step is tested and successful before continuing. Go through the next checklist in the DB2 upgrade paper and then use the more detailed checklists in the DB2 Information Center.

Do I need to REBIND? When?

REBIND is not required for migration to DB2 10, but REBIND is strongly recommended. Getting the best performance improvements and eliminating regression does depend upon rebind in most situations: getting current structures, better access paths, and reusing threads. Eliminating performance regression may depend upon REBIND. Storage constraint relief depends upon REBIND. Changing to use release deallocate requires a REBIND. The migration process has been smoother when customers REBIND, with fewer problems.

All plans containing DBRMs must be rebound. All packages that were last bound on V5 or lower must be rebound. Static SQL with DEGREE(ANY) for parallel processing should be rebound, or it will be sequential. Other REBINDs can be staged over weeks of time, and REBIND is only needed once per package for the

migration. Improvements in access paths can be very significant, such as stage 2 predicates that can become stage 1. REBIND in DB2 10 takes more CPU and elapsed time than in prior versions, but more concurrent REBINDs are possible in NFM. So please be kind. REBIND.

What else is needed to get performance out-of-the-box?

Some customers have needed to use RELEASE(DEALLOCATE) to get better performance for short running transactions. With the IRWW workload, the improvement from use of this parameter was about 10%.

RELEASE(DEALLOCATE) could not be used by distributed applications, before DB2 10, and improvements have been implemented to recycle the thread every 200 transactions. RELEASE(DEALLOCATE) is best for high volume batch or transactions with few SQL statements in each COMMIT. The transactions also need to be well-behaved for locking. Local transactions need some mechanism to end the thread, so that utilities, data definition changes and other processes can be performed.

Can you explain more about the changes for RELEASE(DEALLOCATE) in DB2 10?

RELEASE(DEALLOCATE) has been part of DB2 for a long time, but DB2 10 makes the function more useful. Using RELEASE(DEALLOCATE) requires much more memory. The dramatic memory improvements in DB2 10 that you can achieve with rebind makes it possible to use RELEASE(DEALLOCATE) more. This change saves significant CPU time for high volume transactions with few short running SQL statements, without changing applications or DDL. RELEASE(DEALLOCATE) works in CM, but does require a REBIND unless the packages already use RELEASE(DEALLOCATE). For DDF work, after rebinding with Release(Deallocate), the customer must issue the MODIFY DDF PKGREL(BINDOPT) command. By default JDBC is changed to use Release Deallocate. Additionally, even if packages are not rebound, a new DDF option called High Performance DBAT can be set that will cause resources to be periodically released after 200 Commits. So RELEASE(DEALLOCATE) is more applicable and safer to use in many more situations. RELEASE(DEALLOCATE) depends upon having very well debugged, well behaved applications that are careful with locking and commit frequently.

Some customers are getting a lot of performance improvements that can be made in CM on the first day. The improvements do require a REBIND in most situations, and that does mean testing, but DB2 version changes also take testing, so combining the work for a dramatic improvement will work for many customers. This change can be implemented for a few very high volume transactions and provide a great return.

The DB2 10 performance presentation shows a fairly common scenario. Using more CICS protected ENTRY (persistent) threads with RELEASE (DEALLOCATE) provides the opportunity for significant price/performance improvement. High-volume, short running distributed transactions can take advantage of CPU reductions, using RELEASE(DEALLOCATE). This is the scenario for a benchmark transaction that is run on DB2 9, then on DB2 10. This scenario uses some new function in DB2 9 to BIND or REBIND a package with access control management to allow three copies. These are fairly light CICS transactions that have been used for many DB2 transaction benchmarks.

In step 1, this application is moved to DB2 10 CM without a REBIND, and the result is a 3.7% reduction in CPU time.

In step 2, still in DB2 10 CM, a REBIND is performed but with exactly the same access path. With the REBIND, the CPU savings over DB2 9 was 7.4%, double that without the REBIND.

In step 3, moving to NFM, the CPU time is the same.

In step 4, these transactions are changed to use RELEASE(DEALLOCATE), saving an additional 10% of the CPU time compared to the prior RELEASE(COMMIT). This change could also be made in CM with step 2.

So this scenario demonstrates the runtime improvements and CPU value of REBIND and RELEASE(DEALLOCATE) for a high volume, short transaction.

What does DB2 10 do to enable RELEASE(DEALLOCATE)? Why can't I rebind packages now in DB2 9 with this option and realize less CPU?

RELEASE(DEALLOCATE) takes memory, which is not available for many customers on DB2 9 or V8. In DB2 9, you get RELEASE(COMMIT) for DDF work, even if packages are bound RELEASE(DEALLOCATE). DB2 10 allows distributed RELEASE(DEALLOCATE). DB2 10 also changes to end the thread every 200 transactions. If your distributed transactions are already bound with RELEASE (DEALLOCATE), then they would not need to be rebound, just change the DDF setting.

What negative effects do I need to be aware of in using RELEASE (DEALLOCATE)?

If your application is not well behaved, then you can get into problems, deadlocks, timeouts, and inability to run utilities. If your process is missing commits or takes gross table space locks, then it's not a good candidate for DEALLOCATE.

Did some customers find performance regression?

Most customers found longer BIND times in DB2 10. The default for access path management has changed in DB2 10 from none to EXTENDED. Customers who move from DB2 9 and have used access path management have some improvements. Customers who did not have access path management will find

increases in BIND CPU time from this change. If you want to reduce the time for BIND, then change the subsystem parameter back to none. Use access path management where you have noted problems with new access paths.

What is DBAT reuse? How does it better perform from DB2 Connect concentrator features?

DBAT is a DDF thread, which does not permit thread reuse in DB2 9 or earlier, and will reuse threads in DB2 10. Thread reuse with small, high volume transactions can improve CPU times significantly.

What happens to DSNZPARM COMPRESS_SPT01? I have seen recommendations to compress in DB2 9, but since compression in DB2 10 is unnecessary - is there something in the DB2 10 migration process to revert back to uncompressed?

The DB2 10 ENFM migration process changes the catalog and directory into the new format, including this change. Some customers do need compression to save space, and recent APARs enable most of the space savings.

Is the combination of hash access and unique index access just overhead or can they complement each other?

The access techniques can be complementary. If you have fully qualified keys with equal predicates, then the hash can be used. Otherwise, for range predicates as an example, the indexes are needed. Hash means that you can't cluster, so hash will be used where fully qualified key access is the primary access and where clustering is not needed. The best candidates for hash access are random single row access into a large table with a fixed, known size, with many rows per page, small variation in row sizes, and number of index levels greater than 4, not using index lookaside or index only access.

Is memory used moved above the 2GB bar evenly spread through DB2 addresses spaces or are it concentrated in DBM1, for instance?

Most of the savings are in DBM1, as that is the primary constraint for customers. REBINDs are required for the EDMPOOL improvements and some working storage. The shared area above the bar can help with other address spaces. Some ECSA for instrumentation is also moved above the bar.

I thought DB2 would never have another skip a release migration. Why the change for V8 to V10?

Skip release is costly for developers and complex, so it's not for every version. DB2 delivered skip for V5 to V7 and now for V8 to V10. I think that most customers are best served by staying current with DB2 versions, but some customers get behind. Skip migration is great for customers running V7 today or for those who just finished migrating to V8. Customers who plan to migrate to DB2 9 in 2011 should generally do so. The presentation and paper on migration have much more information.

Does a statement concentrator need to be enabled for caching of dynamic SQL with literals?

Caching for dynamic SQL statements is part of DB2 10 NFM. You need to allow caching to ignore literals with a prepare attribute or an ODBC or JDBC attribute.

Will EXPLAIN show hash access if chosen over index access?

EXPLAIN shows hash access. See the presentations and documentation on hash access. The IDUG EMEA conference in Vienna included a full session on hash access, with best practices for establishing the hash access. Please get it from the IDUG web site under content for that conference.

Can you please expand on hash access and how it is useful?

Hash access is a new direct access technique within DB2 10. DB2 uses the direct keys defined and an algorithm to access the data. This direct access is sometimes done in a single I/O making it very fast and efficient for common customer id and product id access types. The direct single I/O access will improve any application that performs these keyed access types that are sometime done millions of times a day.

How is the hash access path established?

The IDUG EMEA conference in Vienna included a full session on hash access, with best practices for establishing the hash access. Please get it from the IDUG web site under content for that conference.

Is there any improvement compared for MEMBER CLUSTER compared to DB2 9?

MEMBER CLUSTER is not permitted for universal table spaces (UTS) in DB2 9 and is for DB2 10. With DB2 10, you can ALTER to a UTS, then REORG to change the table space to MEMBER CLUSTER, using a pending ALTER. Insert performance improvements build upon those in DB2 9. Insert rates improve a lot and CPU time reductions are very large in DB2 10.

What's the recommendation for IRLM storage? PC = YES / NO?

PC=YES has been the only option since V8.

The DB2 10 overview indicates that most utility enhancements in DB2 10 are available in NFM, rather than CM. Previous DB2 releases provided utility enhancements in CM. Can you explain this change in behavior please?

Delivering new function in new function mode means a better ability to fall back cleanly and better consistency for DB2 and utilities. Customers are spending less time in Conversion Mode and getting more value in New Function Mode.

During PENDING ALTER, is the object is accessible?

Yes, objects are accessible, as long as applications commit to allow the switch for the online reorg.

With DB2 9 and 10, we hear more and more about stabilizing access paths through hints and many more options. Is this another way of saying that the cost based optimizer has somehow reached a limit and that optimizer capabilities are no longer sufficient?

The cost based optimization continues to be an active area for enhancements. DB2 9 and 10 have many query access path improvements, but access paths for transactions and making the maintenance and release migration process simpler are also important. For short running transactions, concerns about regression are key, but even short running transactions can be improved with hash access. More and more of the SQL is complex, running on larger sets of data, so improvements in optimization are as important as ever, even as the techniques are much more complex than in earlier versions.

Does use of the z/OS 1MB page size change the maximum DB2 data page size? Are we still stuck with 32K maximum, or are we going to be allowed to grow to a 1MB page size (and buffer pool) in DB2 10?

Page sizes are an overloaded term and often cause confusion. Hardware and z/OS have page sizes or frames of 4KB – and now 1MB for the z10 and z196. DB2 page sizes remain the same for indexes and data: 4KB, 8KB, 16KB, and 32KB.

Has the global buffer pool contention seen with very large buffer pools been addressed in V10? We see contention with pseudo close causing all the buffer pools to be scanned.

This overhead was addressed in DB2 10, avoiding buffer pool scans.

Does DB2 10 support private protocol?

DB2 10 has only DRDA. Warnings have been delivered for 15 years, and conversion is necessary before moving to DB2 10. See the upgrade presentation and paper.

Timestamps in DB 10 provide 12 digits beyond the second or picoseconds. Do existing applications need to change to accommodate all 12 digits instead of microseconds?

Existing applications don't need to change if they don't need to change from using microseconds. The new timestamp capabilities within DB2 10 provide the database designer with the flexibility to define timestamp precision as application requirements dictate. DB2 10 allows 0 to 12 digits for timestamps with a compatible default of 6. So you could have picoseconds with 12 digits or nanoseconds with 9 digits, more precise than earlier DB2 versions. You could define timestamps with 0 digits, if seconds are precise enough. Existing applications do not need to change if the current timestamp precision is adequate for the application. A general purpose application which will process any DB2 data needs to change to handle this variation and timestamps with time zones.

How small is a small LOB or SLOB?

A small LOB is able to fit on the same page size along with its other associated data column information. This avoids creating a separate table space for the LOB data. The desired is to fit these small LOBs within their 4k, 8k, 16k or 32k page sizes so that no additional I/O, memory, and CPU are needed to store or retrieve the LOB data. These small LOBs can be anywhere from hundreds of bytes to 31k, so that they fit on the same page with the rest of the row.

What are the DB2 10 improvements for BLOB and CLOB?

Many improvements were made in DB2 10 for LOBs. Inline LOBs, improved LOB streaming to and from DDF are key performance improvements. Indexes can be defined on the inline LOB. Data compression can be used for the online LOB portion. Better streaming and minimized LOB default sizes help improve performance when LOB materialization occurs. This is especially important for DDF applications because of the network impact of large objects. DEFINE NO for LOB and XML columns makes definition easier. REORG SHRLEVEL(CHANGE) for LOBs improves availability. Online REORG permits movement of rows between partitions when LOB columns exist, and allows DISCARD to delete associated LOB values. Non-NULL default values can be defined for inline LOBs. Now you can stream LOBs into one data set, rather than taking a data set for each LOB.

How do the utilities work with LOB sizes greater than 32K?

The utilities have been working with LOBs since DB2 V7 with APARs by having separate data sets or members for each LOB. New changes allow LOBs to span records instead of having separate data sets or members.

Is it feasible to convert Oracle or Sybase to DB2 10?

DB2 10 has even more DB2 family and other DBMS compatibility functions. We've seen many customers able to convert with DB2 9. The new timestamp capabilities and precision definitions, the stored procedure language enhancements and many other features that first came in DB2 LUW 9.7 are now offered in DB2 10 for z/OS. It is possible to convert Oracle to run on DB2 10 and z/OS. The Oracle complexities will determine the level of effort but DB2 10 has the majority of the functionality needed for a quick migration effort. SQL Skin works with DB2 for LUW to ease Sybase migration now, and might work with DB2 for z/OS in the future.

Will this change the DB2 Utilities?

Changes in utilities are an integral part of DB2 for z/OS. See utilities presentation on the IDUG web site.

Are there any changes on SQL HINTS topic included in DB2 10?

The infrastructure around the SQL BIND process is enhanced in DB2 10 not just the SQL Hints. DB2 10 provides the capability to provide hints globally to all users for an SQL statement and provide certain optimizer zparm settings such as parallelism star join and other BIND parameter settings for specific package or specific SQL statements. This provides extreme granularity and flexibility for

turning on specific parameters for certain databases, applications, tables or even certain SQL statements within your environment. DB2 10 provides unprecedented levels of package optimization, stability and control through the basic, extended and versioning of packages within your DB2 system. This feature provides the flexibility to choose the package with the desired access path and lock it down. By locking down the access path, administrators can error code out a REBIND that doesn't have the same access path that we desired and prevent it from going into our production environment. This helps guarantee performance and stability within our production systems.

Can solid state drives be used with hardware replication?

Yes. Here are some pointers:

<http://www.ibm.com/systems/storage/solutions/ssd/>

ftp://public.dhe.ibm.com/storage/whitepaper/disk/ds8000_taneja.pdf

ftp://service.boulder.ibm.com/storage/software/virtualization/clipper_svc_5.pdf

<ftp://ftp.software.ibm.com/common/ssi/sa/wh/n/tsw03044usen/TSW03044USEN.PDF>

<https://www.ibm.com/developerworks/mydeveloperworks/blogs/InsideSystemStorage/tags/announcements?lang=en>

Can you explain the common SDSNEXIT in DB2 10?

Allowing tailored names for DSNHDECP will permit many subsystems to share the SDSNEXIT data set. Having more than one name means that programs cannot just load the DSNHDECP module to use the data. Note the changes in DB2 9 to prepare for this.

http://publib.boulder.ibm.com/infocenter/dzichelp/v2r2/topic/com.ibm.db29.doc.inst/db2z_functionsdeprecated.htm

Why one would need to select from SYSLGRNX?

SYSLGRNX can be used to provide RECOVERY information (e.g. for estimation of recovery time, need for image copies and MODIFY. DROP performance can also be affected by large numbers of SYSLGRNX records.

Will R10 restrict BINDING a plan to a single DBRM?

BIND for a DBRM is always to a package in DB2 10. Every DBRM will be bound to a package, and plans will need one or more packages for execution. See the migration presentation for more.

What happens for existing DBRMs bound into plans?

The best practice is to convert all plans to use packages on DB2 V8 or DB2 9. If you don't convert plans to use packages, they will be rebound when the plans run, using a default collection id. See the next question.

Will a formal conversion method be provided to convert stand alone DBRMs to packages for DB2 10?

The process was provided in DB2 9 and V8, about a year ago. See the migration presentation for the process. The Packages Revisited chapter 4 discusses the changes and process. See Packages Revisited, SG24-7688,

<http://www.redbooks.ibm.com/abstracts/SG247688.html>

Does DB2 10 have any features like multirow fetch or insert that applications need to modify to improve performance?

Applications don't require changes to have performance features as dramatic as V8 multi row fetch and insert, but do have many improvements in SQL and XML in DB2 10. For example, the dynamic statement cache improvement needs to have an attribute set. See the overview presentation for the application improvements, ranging from temporal or versioned data, new timestamp options, and improved ability to port from other platforms, and many XML enhancements. Many of the improvements depend upon implementing the latest clients, drivers, and DB2 Connect 9.7 fixpack 3a. See the upgrade presentation and paper for more.

We have currently disabled RRF in our DB2 subsystems since we have encountered numerous problems with it. Is RRF usage a must from the start in DB2 10?

No, the option still exists in DB2 10, but RRF is the default, and required for most new improvements. Customers can use this parameter to control conversion to RRF, but should transition to RRF.

What are the major software/hardware requirements for DB2 V10?

See the migration presentation and paper and the general availability announcement. Check the Program Directory on the web.

http://www.ibm.com/common/ssi/rep_ca/0/897/ENUS210-380/ENUS210-380.PDF

<http://www.ibm.com/support/docview.wss?uid=swg27019288>

<http://publibfp.boulder.ibm.com/epubs/pdf/i1088290.pdf>

We are running IMS V9. In DB2 10 must we upgrade our IMS to 10 or 11? We are currently starting to upgrade to IMS v10.

IMS V9 is out of service and is not supported with DB2 10. Migrate to IMS 10 or 11 soon. See the announcements for prerequisites. IMS 12 is already running their Quality Partnership Program or beta.

http://www.ibm.com/common/ssi/rep_ca/5/877/ENUSZP10-0015/ENUSZP10-0015.PDF

What are the major conversion tasks for applications (primarily binds)?

Migration to DB2 10 has many steps similar to V8 and DB2 9. REBINDs are necessary to leverage many of the performance and memory enhancements. New Function Mode is necessary for other new enhancements. Review the migration presentation and Installation Guide to understand the feature requirements. REBINDs and adjusting for the function removed are the primary application programming tasks. See the upgrade presentation and paper.

Does BIND time increase in DB2 10?

BIND times do increase in moving to DB2 10. This increase is often offset by the improved concurrency for BIND. A substantial part of the increase has been the change to default to use of access path stability, which is an option.

Could you recap index include please? What is index include designed to do?

The new DB2 10 INCLUDE COLUMNS on a unique index feature provides the ability to add additional non-unique columns to a unique index definition. This new feature provides the ability to consolidate and eliminate other indexes on your tables leading to improved access and reduced I/Os by having fewer indexes to maintain. This also shortens DB2 access paths in some cases, from multiple index access to a single index access. This function is also part of DB2 for LUW.

Given potential data sharing member consolidation with storage relief, do you think open dataset might become more of a bottleneck/concern?

Some customers may find data set open a bottleneck, but most will not. DB2 9 handles roughly 60 thousand concurrent allocations for some customers today. DB2 9 has increased that number to about 100K. Changes in z/OS V1.12 improve allocation for very large numbers of data sets.

Will the DB2 catalog / directory have to be SMS managed? Will IBM supply a conversion guide or procedure?

Yes, the new data sets for the DB2 catalog change from using manual definition and extension to DB2 managed data sets under SMS control. See the migration presentation and paper for a start. The DB2 catalog and directory have some table spaces defined as DSSIZE 64G, which requires SMS managed storage with extended format and extended attributes.

Will dynamic statement cache handle literals for select, insert, update and delete?

The DB2 10 dynamic statement cache (DSC) enhancements consolidate all types of SQL statements and their literals. So if the same statement is used within the application with different literals DB2 will consolidate it within the DSC leaving more room for other statements and better caching overall. This behavior requires a parameter to be set, as literals can provide important information for optimization.

There was mention of compress SMF option. How does this differ from ACCUMACC?

ACCUMACC, commonly called rollup, combines multiple accounting records into one. The rollup technique is being improved to work better with packages, allowing more customers to use this option. Compression uses an SMF compression option, often reducing data volumes by 70% to 80%, while increasing CPU time by less than 1%. Customers can use either one or both techniques to reduce SMF accounting volume.

What is different between MLS and Row (and column) access control?

The primary design for MLS is strict security labels, similar to the needs for military confidentiality. MLS builds upon the concepts in z/OS and RACF. Row and column access control is more flexible, with capabilities similar to those of a view, but oriented to security.

COBOL and C Stored procedures perform better than Native Stored Procedures in DB2 9. Will native stored Procedures perform better than C and COBOL Stored procedures in DB2 10?

Native SQL procedures came very close to COBOL and C in IBM laboratory measurements on DB2 9, within about 15%. DB2 9 remote native SQL procedures used much less standard CPU than COBOL or C by using zIIP engines. A DB2 10 native SQL procedure workload has shown up to 20% CPU reduction using SET statements, IF statements and SYSDUMMY1, making them very close to COBOL and C for local calls, and able to use zIIP for remote calls.

How much more zIIP offload is expected in DB2 10? Is there a way to utilize zIIP more?

See the performance presentation for an overview. The primary changes in DB2 10 for additional use of zIIP are removing some parallel restrictions, expanded XML validation, prefetch reads and deferred writes, and RUNSTATS options other than column groups or inline RUNSTATS. Some customers will find that the new zAAP on zIIP facility will reduce complexity or make zIIP more useful.