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=gpateam&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 means 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.