

IBM UK LTD
Product Introduction Centre
Data Management Team



DB2 UDB for z/OS Version 8

“Planning for Migration”

Ian Cook

DB2 Specialist

Email:- Ian_Cook@uk.ibm.com

Tele:- (Office) 0117-929 5049
(Mobile) 07802-458242

TABLE OF CONTENTS

INTRODUCTION	3
HEALTH CHECKING YOUR EXISTING CODE PAGE CONFIGURATION.....	4
DSNTIJPM	4
MANDATORY HARDWARE AND SOFTWARE PREREQUISITES	6
IMPORTANT MIGRATION CONSIDERATIONS:	7
ADDITIONAL KEY FUNCTIONAL REQUISITES	8
DOCUMENTATION REVIEW	9
DEPRECATED ITEMS WORTH NOTING	10
RELEASE INCOMPATIBILITIES.....	12
DSNTIJUZ CHANGES.....	13
PLANNING A MIGRATION STRATEGY	14
UNICODE IMPACT AND CONSIDERATIONS.....	15
UNICODE CATALOG CONSIDERATIONS.....	15
QUERIES MAY RETURN SLIGHTLY DIFFERENT RESULTS	15
UNICODE DATA MAY TAKE MORE SPACE.....	16
UNICODE TOOLING.....	16
CONVERSION FROM CHAR(8) TO VARCHAR(128).....	16
DB2 TOOLS CONSIDERATIONS.....	18
IBM TOOLS COMPATIBILITY WITH DB2 UDB FOR Z/OS VERSION 8.....	18
DATABASE RECOVERY AND REPLICATION MANAGEMENT.....	18
APPLICATION MANAGEMENT.....	19
DATABASE ADMINISTRATION	19
PERFORMANCE MANAGEMENT.....	19
LANGUAGE CONSIDERATIONS	20
ASSEMBLER	20
C/C++	20
COBOL.....	20
FORTRAN.....	20
JAVA.....	20
WEBSHERE.....	21
PL/1.....	21
REXX	21
SQL PROCEDURE LANGUAGE.....	22
GLOSSARY OF TERMS	23

Introduction

IBM DB2 Universal Database for z/OS Version 8 now becomes a general purpose 64-bit-exclusive database management system for large systems. The release breaks through many of the limits which have affected scalability and availability by delivering large address spaces with exploitation of 64-bit virtual addressing providing management of more data with bigger buffers in memory.

DB2 for z/OS Version 8 also enhances distributed and dynamic applications through the provision of long SQL statement support and breaks barriers to family compatibility and application portability by introducing long name support.

Increased application availability with schema evolution support now permits object changes without stopping data access to those objects while the changes are installed. Data partitioned secondary indexes provides access to your data through more partitions, enhancing availability and minimizing partition management, for historical data with support for rolling partitions.

As well as the rich collection of new features and functionality previewed above, DB2 for z/OS Version 8 also introduces some significant changes to the DB2 catalog. During the enable new function mode conversion process, 18 of the 22 catalog table spaces will be converted from EBCDIC to Unicode and long name support (VARCHAR (128)) will also be introduced.

As part of this release of DB2, a new multi-phased approach to migration will be introduced, to provide for a smooth transition from DB2 Version 7 through Version 8 compatibility mode and into Version 8 new function mode.

This paper highlights the prerequisites and early planning considerations which the DB2 for z/OS and OS/390 Version 7 user should be aware of, and implement, before beginning the migration to DB2 for z/OS Version 8.

A second document entitled "**DB2 UDB for z/OS Version 8 - The Migration Process**" provides a detailed account of the actual process of migrating a DB2 Version 7 subsystem through to DB2 for z/OS Version 8 new function mode. It is recommended that the user review and follow the implementation of the planning requirements highlighted in this document prior to reading the second document and embarking on a migration to DB2 Version 8.

Health checking your existing code page configuration

Prior to migrating your systems to V8, it is important to perform a code page health check to ensure no latent issues exist. Any discrepancies may cause problems in Version 8 once DB2 parses SQL in Unicode and starts performing code page conversions more frequently. It is important to check the following in each of your DB2 systems :

- Does your DB2 system have the correct CCSIDs specified? Do these match your terminal emulators and local applications like CICS? For example, we are aware that some customers have set up their DB2 systems with CCSID 500 when their terminal emulators are set up for their national language CCSID. The level of problems caused by this anomaly will increase in Version 8 and must be corrected prior to migration.
- Identify whether there are objects within the system with different CCSIDs. This is performed by a set of queries in job DSNTIJPM (details of job DSNTIJPM can be found below).

If either of these raise an issue at your site, you must contact DB2 Level 2 Support for advice on how to resolve this prior to migration to V8. We recommend you run the checks as soon as you start planning your V8 migration in order to schedule any appropriate changes.

DSNTIJPM

The DSNTIJPM job will now be shipped with DB2 Version 7 as a sample member called DSNTIJP8, via PQ84421. It should be run on your DB2 for OS/390 and z/OS Version 7 systems prior to migration to DB2 for z/OS Version 8. The job queries the DB2 catalog to identify conditions that will need to be remedied before you attempt to migrate to DB2 Version 8, or that need to be attended to after you have migrated to DB2 Version 8.

The outputs from the DSNTIJPM / DSNTIJP8 job are:

- A number of reports detailing the results of the various queries executed.
- A data set of 'ALTER INDEX' statements that can be input to DSNTIAD to convert indexes from TYPE 1 to TYPE 2.
- A data set of 'REBUILD INDEX' utility statements that can be input to DSNUPROC to rebuild indexes after they have been converted from Type 1 to Type 2.

Currently, the DB2 for z/OS Version 8 job DSNTIJPM (DB2 Version 7 equivalent job provided by APAR PQ84421, is DSNTIJP8), checks for the following:

- Occurrences of type 1 indexes, which have been deprecated since V6 and are disallowed by CATMAINT for V8.
- DB2 catalog tables on which DATA CAPTURE is enabled.
- Partitioned table spaces that use selective partition locking (SPL).
- Partitioned table spaces that have a truncated limit key.
- Stored procedures that use LANGUAGE COMPJAVA.
- Stored procedures that use the DB2 stored procedures address space.
- Use of the DSNWZPR module by DSNWZP.
- Existence of the V7 sample database.
- Evidence of multiple CCSIDs in the same encoding scheme.

Mandatory hardware and software prerequisites

As DB2 for z/OS Version 8 is an exclusive 64-bit virtual release of the product, there are some key hardware and software prerequisites required to support the environment, as listed below:

- A processor that supports the z/Architecture, e.g., z800, z900, z990, or a comparable processor.
- A system image, running a minimum release of z/OS 1.3 (n.b., z/OS 1.3 requires WLM Goal Mode).
- As DB2 for z/OS Version 8 is an exclusive 64-bit virtual release, the system image must be IPLed in 64-bit mode.
- IRLM V2 R2 is being introduced with DB2 for z/OS Version 8 and has only been tested and approved for use on that DB2 release. Recommendation is that for other (pre Version 8), levels of DB2 for z/OS and IMS should continue to run with IRLM V2 R1, unless there is a note of IRLM V2 R2 support in their program directory.
- 64-bit virtual addressing support will probably require increased real storage for a workload compared to the same workload running under DB2 for z/OS and OS/390 Version 7.
- Note that the requirement for 64-bit virtual addressing means that z/OS Bimodal Migration Accomodation software cannot be used with DB2 for z/OS Version 8.
- z/OS Unicode Conversion Services must be setup and activated on the z/OS system, or z/OS must have OA04069 installed for automatic configuration of Unicode Conversion Services. The process of setting up these services is detailed in a later section.
- Before migrating the first member of a data sharing group to Version 8, check the service levels of any coupling facilities running with coupling facility control code (CFCC) at CFLEVEL=7 or CFLEVEL=8:
 - Coupling facilities at CFLEVEL=7 should have service level 1.06 or higher.
 - Coupling facilities at CFLEVEL=8 should have service level 1.03 or higher.
 - If these service levels are not installed, data corruption can occur. No service level requirements exist for the other CFLEVELs.

Important migration considerations:

As well as the hardware and software prerequisites listed in the previous section, the following must be considered when planning for migration to DB2 Version 8:

- There will be NO skip release migration option, as was available between Version 5 and Version 7.
- Migration to Version 8 will ONLY be possible from a Version 7 subsystem started with the fallback SPE PQ48486 installed, hence check that your current Version 7 maintenance level contains the fallback SPE.
- DB2 Version 8 will check for the fallback SPE at startup, by checking for a code level mismatch with the current catalog. If its not installed, DB2 Version 8 does not start and a message is issued.

Additional key functional requisites

The following table identifies other key requisites required to support operation of specific functions of DB2 for z/OS Version 8:

Function	Product Name & Minimum Service Level	Program Number
Migration/Fallback/Coexistence with DB2 for z/OS Version 8	DB2 for OS/390 Version 7 (APAR PQ48486)	5675-DB2
Unicode data storage and manipulation capabilities in a 64-bit environment	z/OS Version 1 Release 3, or higher (APARs OW56073 and OW56074) z/OS Version 1 Release 5 users require PTF UQ05790	5647-A01
System level Point-in-Time (PIT) Backup and Recovery function (important - the RESTORE SYSTEM LOGONLY option has no dependency on z/OS v1.5)	z/OS Version 1 Release 5 and DFSMSHsm, DFSMSdss and FlashCopy (FlashCopy Version 2 is much faster and is recommended.)	5647-A01
Multilevel Security with Row Granularity (MLS)	z/OS Version 1 Release 5 RACF	5647-A01
Group Bufferpool (GBP) Caching z/OS	z/OS Version 1 Release 4 and Coupling Facility (CF) level 12	5647-A01
All WLM-managed Stored Procedures / Server Task Management	z/OS Version 1 Release 3, or higher with APAR OW56108	5647-A01
msys plug-in	z/OS Version 1 Release 3 DB2 Customization Centre plug-in for msys	5647-A01

Documentation Review

The following documentation should be reviewed prior to installing and/or migrating to DB2 for z/OS Version 8:

- Program Directory for IBM Database 2 Universal Database for z/OS Version 8 Program Number 5625-DB2 (FMID HDB8810, HIY8810, HIZ8810, HIR2220), which can be found on the library web site at: <http://www.ibm.com/software/data/db2/os390/library.html>
- Check Information/Access or the ServiceLink facility of IBMLink for the Preventive Service Planning (PSP) bucket for **UPGRADE DB2810** and the following **SUBSET ids**; HDB8810, HIY8810, HIZ8810, HIR2220, HDB881A, HDRE810, JDB8812, JDB8814, JDB8817, JDB881B, JDB881C and JDB881X.
- Review the two part Unicode paper "Introducing Unicode" produced by the Product Introduction Centre. This article was published in the IDUG Solutions Journal (December 2003 and March 2004).
- Review the accompanying document entitled "**DB2 UDB for z/OS Version 8 - The Migration Process**", which provides a detailed account of the new tiered migration process for migrating a DB2 Version 7 subsystem through to DB2 for z/OS Version 8 new function mode.
- Check out the following URL for the latest information on DB2 for z/OS Version 8 :
<http://www.ibm.com/software/data/db2/os390/db2zosv8.html>
- Review the redbook: *DB2 UDB for z/OS Version 8: Everything you ever wanted to know, ... and more*, SG24-6079-00.
- Review the redbook: *DB2 UDB for z/OS Version 8 Technical Preview*, SG24-6871-00.
- Recommend you also take a look at the presentations available at:
<http://www.ibm.com/software/data/db2/os390/presentations.html>

Deprecated items worth noting

A number of items have been deprecated as part of DB2 Version 8 and should be considered as part of the early planning phase and they are listed below. For additional details, please see the latest edition of the *DB2 Universal Database for z/OS Version 8 Installation Guide, GC18-7418*. A list of all unsupported objects in a catalog can be determined by running the queries that are found in the DSNTESQ and DSNTIJPM jobs (DSNTIJP8 is the equivalent job shipped with DB2 Version 7):

- The migration will fail if you have any TYPE 1 indexes in your subsystem.
- The TYPE 2 keyword has also been deprecated in Version 8 for the CREATE and ALTER INDEX statement and is ignored by DB2 Version 8.
- Support for DB2-established stored procedure address spaces is deprecated in DB2 Version 8, hence you can no longer specify the NO WLM ENVIRONMENT option when you create or alter your stored procedure definitions. Existing stored procedures can still run in a DB2-established stored procedure address space; however, you should move your stored procedures to WLM environments as soon as possible.
- Support for a DB2-established data space for cached dynamic statements is removed and you can no longer specify the parameters EDMDSPAC or EDMDSMAX during install or migration. Instead, a new EDM Statement Cache parameter EDMSTMTC is included. On migration to DB2 Version 8, the value specified for EDMDSPAC will be used for EDMSTMTC. Important - the EDMPOOL is now three separate pools: two above the bar, namely, the DBD cache and the dynamic statement cache, and below the bar, the EDM pool housing the plans and packages, as in previous releases.
- Support for LANGUAGE COMPJAVA stored procedures has been removed and is no longer supported in DB2 Version 8, since Visual Age Java will no longer support compiled Java link library files. To migrate a LANGUAGE COMPJAVA stored procedure to LANGUAGE JAVA, take the following steps:
 - Use ALTER PROCEDURE to change the LANGUAGE and the WLM ENVIRONMENT. The EXTERNAL NAME clause must also be specified. Even if it isn't changed, DB2 needs to verify it. Here is an example:

```
ALTER PROCEDURE SYSPROC.JAVADV  
LANGUAGE JAVA EXTERNAL  
NAME 'display.display.main'  
WLM ENVIRONMENT WLMENVJ;
```
 - Make sure the WLM environment has been set up and the required JVM installed.
 - Make sure the .class file identified in the EXTERNAL NAME clause of the ALTER PROCEDURE is either contained in a JAR that has been installed to DB2 with an invocation of the INSTALL_JAR stored procedure, or that the .class file is in a

directory name in the CLASSPATH ENVAR of the data set named on the JAVAENV DD card of the WLM stored procedures address space JCL.

- Use of selective partition locking on partitioned table spaces is deprecated in DB2 Version 8. Although it is still supported for compatibility mode, the default behaviour changes once the DB2 catalog enters new function mode.

Release incompatibilities

Some release incompatibilities to observe and account for in your early planning schedule (for additional details, please see the latest edition of the *DB2 Universal Database for z/OS Version 8 Installation Guide, GC18-7418*):

- A couple of changes to defaults for utilities:
 - SORTKEYS is the default for the REORG, LOAD, and REBUILD utilities
 - SORTDATA is the default for the REORG utility
- Changed output for DISPLAY GROUPBUFFERPOOL command, which now displays both the operational and actual CFLEVEL. In Version 7, only the operational CFLEVEL was displayed.
- Several new SQL reserved words are introduced. Please refer to the *DB2 Universal Database for z/OS Version 8 SQL Reference, SC18-7426* for the complete list and adjust your applications accordingly.
- Possible adjustments to your trace applications to handle Unicode fields may be needed for better performance and less conversion cost. If you have trace applications that use statement-length fields, you might need to change them to use 4-byte statement length fields.
- Partitioned table spaces that have a truncated limit key, which can result in abend 04e-00c90101 if the same limit key value is used for different partitions due to the truncated limit key. This problem was corrected by PQ48024 on DB2 V6 and V7 but table spaces created prior to application of the fixing PTF should be corrected before migration to DB2 V8.
- The V7 sample database. Customers are instructed to verify migration to V8 compatibility mode by rerunning portions of the V7 IVP jobs. The IVPs in turn require the sample database, thus if the database is missing, customers need to recreate it - and run phase 1 and 2 IVPs in V7 before beginning migration to V8.
- Evidence of more than one code page within the same encoding scheme is a condition that should be resolved in any case but particularly before migrating to Version 8. See job DSNTIJPM (in DB2 V8) or DSNTIJP8 (in DB2 V7, PQ84421).

DSNTIJUZ changes

Please be aware that DB2 for z/OS Version 8 has introduced some changes in this area with regards to the usage of the DB2 supplied version of DSNHDECP module in SDSNLOAD.

You are now required to create a new DSNTIJUZ-generated load module called DSNHMCID in SDSNEXIT and SDSNLOAD. DSNHMCID is a data-only load module which contains the EBCDIC CCSIDs to be used for message token translation by DB2 facilities that run in allied address spaces, such as attaches and utilities. DB2 version 8 utilities and applications must have access to this module and you can provide access to DSNHDECM in one of the following ways:

- Permit the DSNHDECM module to reside in SDSNLOAD.
- Include the library SDSNEXIT before SDSNLOAD in the system link list.
- Verify that all jobs and tasks that use DB2 utilities or call DB2 application programs are updated to STEPLIB or JOBLIB to SDSNEXIT.

A detailed account of the changes can be found in the document entitled "**DB2 UDB for z/OS Version 8 - The Migration Process**" and a summary of the changes are listed below:

Summary of DSNTIJUZ changes

- You are now required to create a new DSNTIJUZ-generated data-only load module called DSNHMCID in SDSNEXIT and SDSNLOAD.
- DB2 can no longer start with the DB2-supplied DSNHDECP.
- The precompiler now returns code 8 and an error message when it uses the DB2-supplied DSNHDECP.
- DSNTIJUZ has new job steps to let you assemble and link edit the DSNHMCID module in the SDSNEXIT and SDSNLOAD libraries.

Planning a migration strategy

Due to the new three phase migration path with DB2 Version 8, additional considerations should be taken into account when considering when to migrate to V8 and how to roll it out across your systems. These include:

- **How long will you want to run in compatibility mode?**
This will vary from site to site but it is expected that customers will run for at least a month in *CM* in order that they verify all processing, including month and quarter end, before moving to new function mode from which there is no fallback. However, this needs to be balanced against the fact that most people don't like to keep their production and development systems out of sync for too long.
- **How soon do you require the new functions?**
Bear in mind that new functions cannot be exploited until your subsystem / group is actually in new function mode, following successful processing of the DSNTIJNF job. If you have a business requirement for early development using new functions of DB2 Version 8, then you may wish to consider running the installation process to produce a new subsystem / group, which immediately produces a new function mode environment.
- **How soon after GA do you want to migrate to Version 8?**
Bear in mind the discovery of defects is higher, the closer to *GA* you take the new code level. Hence, the sooner you migrate to a new version the more prepared you need to be for raising defects and working with the Support to resolve them. In addition, the newer the version, the more important it is to stay current on maintenance so you need to ensure you have a good strategy for applying Hipers and preventative maintenance.

Below is a suggested plan for migration of a data sharing group of perhaps 3 members:

- Migrate two of the members to DB2 Version 8 *CM* to gain a full assessment of the risks involved. With two members in *CM*, you are exploiting the DB2 Version 8 data sharing code.
- One week later migrate the third member to DB2 Version 8 *CM*.
- Run through your key application processes for a period which allows validation of your monthly, quarterly and perhaps year-end processes.
- If everything is stable after this period, plan to rollout to production.

Unicode impact and considerations

Now lets assess the impact Unicode will have on your environments and review some considerations:

Unicode catalog considerations

Once a DB2 subsystem/group has successfully converted to V8 new function mode, the DB2 catalog is predominantly Unicode (UTF-8). During enabling-new-function mode processing, most of the existing DB2 catalog information is converted from EBCDIC to Unicode (UTF-8). In the enabling-new-function mode, a catalog table space that has already been converted to Unicode will store its data in Unicode. A catalog table space that has not yet been converted to Unicode during the enabling-new-function mode process will still store its data in EBCDIC.

As a result of this, there are some differences and considerations that you should be aware of. It is recommended that you read the 'Introducing Unicode' paper for a full explanation and details but they are summarised briefly here:

- When displaying the Unicode catalog data, in most cases the data will be displayed correctly as the appropriate conversion will be performed automatically by either DB2 (based on the application encoding scheme) or the DRDA receiver.
- An exception to this is FOR BIT DATA columns, such as those containing SQL statement text, which will be displayed in Unicode unless an explicit conversion is performed.
- Basically, the statements in the DB2 catalog remain in the encoding scheme that prevailed when they were inserted. All statements inserted by DB2 Version 7 remain in EBCDIC. Statements inserted in DB2 Version 8 are EBCDIC or Unicode, depending on the NEWFUN precompiler options NO or YES respectively. You can tell by looking to see if the plan or package is Version 8 dependent.

Queries may return slightly different results

- ORDER BY returns rows in Unicode order, in which the first 128 code points happen to match the first 128 code points of ASCII (numbers first). Hence, Unicode has a different collating sequence from EBCDIC.
- Also be aware that upper case comes before lower case.
- Both factors may impact ORDER BY and range predicates, although ranges or sorts within a case or within the numbers may not be affected.

In addition to the conversion to Unicode, V8 expands the string columns in the DB2 catalog to VARCHAR(128) which means that there may be less columns displayed on a screen when using facilities such as SPUFI. You could consider using the SUBSTR function to select only the first part of each column, but bear in mind that functions such as SUBSTR work at a byte level rather than counting characters. For example, when selecting SUBSTR(NAME,1,8) FROM

SYSIBM.SYSPACKAGE DB2 may not return all 8 characters should the name include any characters which are stored as more than one byte. Therefore, you may wish to use the new SUBSTR_OCTET function which would produce the full package name in this example, as SUBSTR_OCTET is a character-based function, rather than byte based.

Unicode data may take more space

It is important to note that with Unicode, the storage size no longer always equals the displayed size. You may need to allow extra space for the DB2 catalog data sets due to the potential increase in storage size for some characters. The amount will depend on your existing character set and, in extreme cases with certain characters, Unicode can increase disk storage costs by up to three times, with an associated processor memory increase.

In most places in the world, there will be no expansion at all. In a few places, the so called national characters (#, \$ and @) may expand by 1 byte. These characters are not in common use except by vendors. In Asia, characters may double in size. Experiences with testing customer DB2 catalogs so far have shown that noticeable expansion is attributable to the change to VARCHAR, and not names.

Unicode tooling

Even if you do not store application data as Unicode, it should be noted that Unicode data will be flowing internally, hence there will be a need for customers to be able to readily interpret SQL text in IFCIDs, catalog columns, DBRMs, etc. Data retrieved by SPUFI and DSNTEP2 is converted from Unicode to EBCDIC by DB2, based on the application encoding scheme, before being placed in their host variables.

However, there will sometimes be a need to have your tools, utilities and aids able to show SQL text in EBCDIC format as per the DSNHECP setting. With some data now also in Unicode, the IBM DB2 tools will perform the appropriate conversion so that the data is shown in a displayable form. Visual Explain can now also be used for displaying all SQL statement text. In addition, the DBRM can be downloaded in binary to a workstation, and displayed with a workstation tool like Notepad.

A test usermod has been developed to allow the ISPF Browser to convert specific areas of a file which are Unicode to display in EBCDIC. It is packaged as a usermod and a PDF document exists as a guide as to how to use the Unicode support. The guide is also packaged in the usermod and once the usermod is installed, the guide can be downloaded as a binary object from member ISRUCGDE in the SISPSAMP library.

Conversion from CHAR(8) to VARCHAR(128)

It is important to understand what actually happens during conversion from CHAR to VARCHAR. Some people think that when the ENFM process does that conversion, that it removes trailing

blanks, but actually it DOES NOT. The reason why it does not remove them is because it does not know how many of those trailing blanks are real. It is the same reason that we cannot use a PADDED index on a varchar column for index only access of application data.

As a result, converting from CHAR(8) to VARCHAR(128) ends up with each column being 8 bytes long, including whatever blanks we had to add to make it 8 when it was inserted, with a length field of 8. New rows will be inserted with the true length. All of this is not unique to the DB2 catalog, as this is the way schema evolution works when converting from CHAR (or GRAPHIC) to VARCHAR (or VARGRAPHIC).

DB2 Tools considerations

It is essential that you check your existing tools products with the vendor(s), to ensure that they are compatible with, and support, DB2 for z/OS Version 8 in either toleration or exploitation mode depending upon your requirements.

Also review any in-house-developed tools or application programs as these may well require changes for DB2 Version 8 compliance to tolerate the DB2 catalog changes for Unicode and long names.

The following lists highlight a subset of the various IBM tools products and the levels of which that are required for compatibility with DB2 for z/OS Version 8. Please refer to the following IBM Data Management Tools websites for a comprehensive list of products and their current status with regards toleration and exploitation of DB2 UDB for z/OS Version 8:

<http://ibm.com/software/data/db2imstools/>

http://www.ibm.com/support/docview.wss?rs=203&q1=DB2+for+z%2fOS+Version+8&q2=tools+compatibility&uid=swg21162152&loc=en_US&cs=utf-8&lang=en

IBM Tools compatibility with DB2 UDB for z/OS Version 8

The following list highlights a subset of the tools from the IBM product set which currently tolerate and/or exploit the new functions and features available in DB2 UDB for z/OS Version 8.

- **Tolerate (Tol)** means that the product executes as it did on a previous release of DB2 but does not utilise any of the new DB2 Version 8 functions.
- **Exploit (Exp)** means that the product utilises the new functions and features available in DB2 Version 8.

Database recovery and Replication Management

- IBM DB2 DataPropagator for z/OS Version 8 (5655-I60) (Exp with PQ74674)
- IBM DB2 DataPropagator for z/OS Version 7 (Tol with PQ50953)
- IBM Application Recovery Tool V1R2 (5697-F56) (Exp with PQ84689)
- IBM DB2 Change Accumulation Tool Version 1 Release 3 (5655-F55) (Exp with PQ84974)
- IBM DB2 Object Restore for z/OS Version 2 Release 1 (5697-I20) (Exp with PQ84483)
- IBM Log Analysis Tool for z/OS Version 2 Release 1 (5655-E56) (Exp with PQ84373)

Application Management

- IBM DB2 Bind Manager for z/OS Version 2 Release 2 (5655-E43) (Exp)
- IBM DB2 Web Query Tool for z/OS Version 1 Release 3 (5655-E71) (Exp with Fix Pack 5)
- IBM DB2 Path Checker for z/OS Version 2 Release 1 (5697-I23) (Exp)
- IBM DB2 Table Editor for z/OS Version 4 Release 3 (5697-G65) (Exp with PQ80771 and Fix Pack 5)
- IBM DB2 Archive Log Accelerator Version 2 Release 1 (5697-I73) (Exp)
- IBM DB2 Test Database Generator for z/OS Version 2 Release 1 (5697-I02) (Exp with PQ84475)
- IBM DB2 Data Archive Expert for z/OS Version 1 Release 1 (5655-I95) (Exp with PQ84256)

Database Administration

- IBM DB2 Administration Tool for z/OS Version 5 Release 1 (5697-K90) (Exp)
- IBM DB2 High Performance Unload for z/OS Version 2 Release 1 (5655-I19) (Exp with PQ84688)
- IBM DB2 Object Comparison Tool for z/OS Version 3 Release 1 (5697-K40) (Exp)
- IBM DB2 Utilities Suite for OS/390 and z/OS Version 8 (5655-K61) (Exp)
- IBM DB2 Operational Utilities for OS/390 and z/OS Version 8 (5655-E63) (Exp)

Performance Management

- IBM DB2 Performance Monitor for z/OS Version 8 (5655-J50) (Exp with PQ81857)
- IBM DB2 Performance Expert for z/OS Version 2 Release 1 (5655-J49) (Exp with PQ81853)
- IBM DB2 Buffer Pool Analyzer for z/OS Version 2 Release 1 (5697-H82) (Exp with PQ81858)

Language considerations

Here we review various language considerations when migrating your application environments to Version 8. The following application development tools and programming languages can be used to build applications for DB2 for z/OS V8:

Assembler

High Level Assembler, which is part of the System Services element of z/OS, is supported for use with DB2 for z/OS Version 8.

C/C++

Any of the following can be used to develop applications with DB2 for z/OS Version 8:

- C/C++ (with or without Debug Tool), part of the Application Enablement Services optional feature of z/OS
- IBM C/C++ for MVS/ESA Version 3 Release 2 (5655-121)
- IBM SAA AD/Cycle C/370 Compiler Version 1 Release 2 (5688-216)

COBOL

Any of the following can be used to develop COBOL applications with DB2 for z/OS Version 8:

- ENTERPRISE COBOL Version 3 Release 2 (5655-653).
- IBM COBOL for OS/390 & VM Version 2 Release 2 (5648-A25).

Compiling a DB2 UDB for z/OS Version 8 program requires a current, supported COBOL compiler. Existing load modules continue to run, but we know that the V8 precompiler generates COBOL that some of the out-of-service compilers cannot compile.

Fortran

The following release of Fortran can be used to develop applications with DB2 for z/OS Version 8:

- VS Fortran Version 2 Release 6 (5668-806, 5688-087, or 5668-805)

Java

Applications or stored procedures written in Java, such as those using the JDBC or SQLJ interfaces to DB2, require IBM Developer Kit for OS/390, Java 2 Technology Edition (5655-D35) at run time.

N.b., DB2 for z/OS Version 8 will no longer support LANGUAGE COMPJAVA stored procedures.

The new IBM Universal Driver for JDBC and SQLJ is a single driver for Unix, Linux, Windows and z/OS platforms, which provides a number of benefits including the elimination of porting problems. You will probably wish to migrate your Java applications to the new driver gradually, having first carried out tests to determine any differences such as behavioural anomalies. More information can be found at <http://www.ibm.com/software/data/pubs/papers/sqlj/>

In December 2003, IBM also announced the Application Connectivity to DB2 for z/OS and OS/390 software, also known as the Type 4 JDBC Driver for Java. This feature is now available on DB2 for z/OS and OS/390 V7, and will also be provided as part of DB2 Version 8, delivering the development and connectivity support necessary for users to write Java applications natively on the z/OS platform without incurring the cost of a second DB2 license (this code was previously imbedded within DB2). For further information on this software, please see the announce letter at the following URL:

http://www.ibm.com/common/ssi/rep_ca/1/897/ENUS203-351/ENUS203-351.PDF

WebSphere

WebSphere Studio Enterprise Developer Version 5.0 (5724-B67). The Program Requirements are:

- IBM SDK for z/OS, Java 2 Technology Edition, Version 1.4 (5655-I56) with APAR PQ74985 (also known as Version 1.4.1) is the recommended level
- IBM SDK for z/OS, Java 2 Technology Edition, Version 1.3.1

For more information on Java 2 for OS/390 and z/OS including a link to software prerequisites, see:

<http://www.ibm.com/servers/eserver/zseries/software/java/>

PL/1

Any of the following can be used to develop PL/1 applications with DB2 for z/OS Version 8:

- IBM Enterprise PL/I for z/OS and OS/390 Version 3 Release 2 (5655-H31)
- IBM PL/I for MVS & VM Version 1 Release 1 (5688-235)

Use of the DB2 Precompiler Services requires the DB2 Coprocessor provided with IBM Enterprise PL/I for z/OS and OS/390 Version 3 Release 2 or later.

In previous versions of DB2, the precompiler-modified source was generated in codepage 037 (US English). Now in Version 8, it will be generated in the CCSID as depicted by the value presented in the new precompiler CCSID option. However, watch for older (unsupported) versions of PL/1 as they do not understand characters not in the 037 codepage, e.g. the OR compiler option. In later supported versions of PL/1 you can specify an alternate character to be interpreted as the logical OR operator. Enterprise PL/1 V3R2 has a CODEPAGE option, which is not available in the previous compilers and is used in CHAR and WIDECCHAR conversions and SAX parser processing, not in determining the code page that the source program is held in.

REXX

The following release of REXX can be used to develop applications with DB2 for z/OS Version 8:

- IBM TSO Extensions for MVS REXX, which is part of z/OS
- REXX Version 1 Release 3 (5695-013 or 5695-014)

SQL Procedure Language

A 'C' language compiler is required on z/OS to develop stored procedures using the SQL procedure language. Specific C compilers include:

- C/C++ (with or without Debug Tool), optional features of z/OS
- IBM SAA AD/Cycle C/370 Compiler Version 1 Release 2 (5688-216). Restriction: this compiler is not supported for SQL Procedures used with the SQL Debugger.

Glossary of Terms

Abbreviation / Acronym	Description
CF, Coupling Facility	A shared memory device used in a Parallel Sysplex environment for sharing data used by all participating systems. The shared (CF) 'memory' is exploited using coupling facility structures.
CICS, Customer Information Control System	IBM online transaction management system.
CM, Compatibility Mode	Migrating from DB2 Version 7 via CATMAINT process, places the DB2 subsystem in DB2 Version 8 Compatibility Mode, the first tier of the migration process.
Data sharing	The concurrent access of databases by two or more DB2 subsystems.
DB2	IBM's premier relational database management system.
ENFM, Enable New Function Mode	Migrating from DB2 Version 8 Compatibility mode to new function mode is accomplished through the Enable New Function Mode process. A serial process of changing the DB2 via the CATENFM process, readies the DB2 subsystem for DB2 Version 8 new function mode, the final tier of the migration process.
ESP, Early Support Program	A program run by the Product Introduction Centre, to validate the content and functionality of a new release of a software product.
GA, General Availability	Time when new product code becomes generally available to the public.
IMS, Information Management System	IBM hierarchical database management system.
ISV, Independent Software Vendor	A company that produces and/or sells software products.
NFM, New Function Mode	After successful completion of the enable new function mode process and the running of job DSNTIJNF, the subsystem or group will be in new function mode. At this point, all new functions of DB2 for z/OS Version 8 can be used.
OS/390	MVS operating system running under the 390 architecture. N.b., not a valid operating system in which to run DB2 Version 8.
Parallel Sysplex	IBM technology introduced in 1994, comprising of multiple processors connected together and processing workload. A development of Sysplex to allow high speed datasharing between S390 images.
PIC, Product Introduction Centre	Formerly known as the ISC.
SVL	DB2 Silicon Valley Laboratory.
USS, UNIX Systems Services	Fully UNIX-compliant operating system under MVS.
WLM, Workload Manager	Software used for managing workload scheduling and performance in a 390 environment.
XCF, Cross System Coupling Facility	Software used for messaging and communication between images in a Parallel Sysplex environment (not to be confused with the hardware coupling facility component).
z/OS	Latest MVS operating system running under the z/Architecture, supporting 64-bit addressing.