



## What's New?





## What's New?

**Note**

Before using this information and the product it supports, be sure to read the general information under “Notices” at the end of this information.

**Sixth edition (October 2009)**

This edition applies to DB2 Version 9.1 for z/OS (DB2 V9.1 for z/OS), product number 5635-DB2, and to any subsequent releases until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Specific changes are indicated by a vertical bar to the left of a change. A vertical bar to the left of a figure caption indicates that the figure has changed. Editorial changes that have no technical significance are not noted.

© Copyright International Business Machines Corporation 2007, 2009.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>About this information</b> . . . . .	<b>v</b>
Terminology and citations . . . . .	v
How to send your comments . . . . .	vi
<b>Chapter 1. Introduction</b> . . . . .	<b>1</b>
<b>Chapter 2. Leveraging your enterprise for information on demand</b> . . . . .	<b>3</b>
Seamless integration of XML data and relational data. . . . .	3
Compatibility and leadership with SQL . . . . .	4
SQL consistency improvements . . . . .	5
Leverage existing application programming skills . . . . .	10
Enhancements to large object support . . . . .	10
SQL leadership: family firsts. . . . .	13
Tools that support your enterprise. . . . .	15
Accessing your enterprise data on demand with DB2 QMF . . . . .	15
Managing your enterprise with DB2 Tools . . . . .	18
<b>Chapter 3. Availability in Version 9.</b> . . . . .	<b>21</b>
Online REORG with no BUILD2 phase . . . . .	21
Faster replacement of one table with another . . . . .	21
Universal table spaces . . . . .	21
Better availability during REBUILD INDEX operations . . . . .	22
Improved availability with column and index renaming capabilities . . . . .	23
Modify EARLY code without an IPL . . . . .	23
ALTER TABLESPACE and index logging improvements . . . . .	23
Support for using SMS storage classes with DB2-defined data sets . . . . .	23
DB2 support for extended address volumes (EAV) . . . . .	24
<b>Chapter 4. Performance in Version 9 . . . . .</b>	<b>25</b>
Reduction in CPU processing time for utilities. . . . .	25
SQL optimization improvements . . . . .	25
Better data for the optimizer with histogram statistics . . . . .	25
Improved optimization techniques. . . . .	25
IBM Optimization Service Center for DB2 for z/OS . . . . .	27
IBM DB2 Optimization Expert for z/OS. . . . .	28
Indexing improvements . . . . .	29
Improved performance for varying-length rows . . . . .	30
Relief for sequential key insert . . . . .	30
Improved logging performance. . . . .	31
Improved data insert performance. . . . .	31
<b>Chapter 5. Security and regulatory compliance in Version 9.</b> . . . . .	<b>33</b>
Roles and network trusted contexts . . . . .	33
Improved auditing . . . . .	33
Support for Secure Socket Layer protocol . . . . .	33
More security options with INSTEAD OF triggers . . . . .	34
Support for AES encryption . . . . .	34
<b>Appendix. Planning for DB2 Version 9.1 for z/OS</b> . . . . .	<b>35</b>
Command changes in Version 9 . . . . .	35
New commands in Version 9 . . . . .	35
Changed commands in Version 9 . . . . .	35
Changes to utilities in Version 9 . . . . .	42
Utility changes in Version 9 . . . . .	42

	Other utility changes in Version 9 . . . . .	47
	SQL statement changes in Version 9 . . . . .	47
	New SQL statements in Version 9 . . . . .	47
	Changed SQL statements in Version 9 . . . . .	47
	New functions in Version 9 . . . . .	51
	Reserved words . . . . .	53
	Other SQL language changes in Version 9 . . . . .	58
	Catalog changes in Version 9 . . . . .	59
	New catalog tables in Version 9. . . . .	59
	Changed catalog tables in Version 9 . . . . .	61
	New and changed indexes in Version 9 . . . . .	65
	Performance monitoring and tuning changes in Version 9 . . . . .	68
	Performance changes in Version 9 . . . . .	68
	EXPLAIN table changes in Version 9 . . . . .	69
	New and changed IFCIDs in Version 9 . . . . .	80
	New IFCIDs in Version 9 . . . . .	80
	Changed IFCIDs in Version 9 . . . . .	81
	<b>Information resources for DB2 for z/OS and related products . . . . .</b>	<b>83</b>
	<b>How to obtain DB2 information . . . . .</b>	<b>89</b>
	<b>How to use the DB2 library . . . . .</b>	<b>93</b>
	<b>Notices . . . . .</b>	<b>97</b>
	Trademarks . . . . .	98
	<b>Glossary . . . . .</b>	<b>99</b>
	<b>Index . . . . .</b>	<b>101</b>

---

## About this information

This information provides an executive overview of new function in DB2® Version 9.1 for z/OS®. The topics in this information provide a framework for describing new function in DB2 for z/OS. New functions are categorized according to user benefits such as information on demand, availability, and performance.

In addition, this information summarizes changes that were introduced in this release for DB2 commands, DB2 utilities, SQL statements, the DB2 catalog, DB2 performance monitoring, and instrumentation facility component identifiers (IFCIDs).

This information assumes that your DB2 subsystem is running in Version 9.1 new-function mode. Generally, new functions that are described, including changes to existing functions, statements, and limits, are available only in new-function mode. Two exceptions to this general statement are new and changed utilities and optimization enhancements, which are also available in conversion mode unless stated otherwise.

### Who should read this information

This information is written primarily for people who are evaluating and planning for DB2 for z/OS.

---

## Terminology and citations

In this information, DB2 Version 9.1 for z/OS is referred to as "DB2 for z/OS." In cases where the context makes the meaning clear, DB2 for z/OS is referred to as "DB2." When this information refers to titles of DB2 for z/OS books, a short title is used. (For example, "See *DB2 SQL Reference*" is a citation to *IBM® DB2 Version 9.1 for z/OS SQL Reference*.)

When referring to a DB2 product other than DB2 for z/OS, this information uses the product's full name to avoid ambiguity.

The following terms are used as indicated:

**DB2** Represents either the DB2 licensed program or a particular DB2 subsystem.

**OMEGAMON®**

Refers to any of the following products:

- IBM Tivoli® OMEGAMON XE for DB2 Performance Expert on z/OS
- IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS
- IBM DB2 Performance Expert for Multiplatforms and Workgroups
- IBM DB2 Buffer Pool Analyzer for z/OS

**C, C++, and C language**

Represent the C or C++ programming language.

**CICS®** Represents CICS Transaction Server for z/OS.

**IMS™** Represents the IMS Database Manager or IMS Transaction Manager.

**MVS™** Represents the MVS element of the z/OS operating system, which is equivalent to the Base Control Program (BCP) component of the z/OS operating system.

## RACF®

Represents the functions that are provided by the RACF component of the z/OS Security Server.

---

## How to send your comments

Your feedback helps IBM to provide quality information. Please send any comments that you have about this book or other DB2 for z/OS documentation. You can use the following methods to provide comments:

- Send your comments by e-mail to [db2zinfo@us.ibm.com](mailto:db2zinfo@us.ibm.com) and include the name of the product, the version number of the product, and the number of the book. If you are commenting on specific text, please list the location of the text (for example, a chapter and section title or a help topic title).
- You can send comments from the Web. Visit the DB2 for z/OS - Technical Resources Web site at:

<http://www.ibm.com/support/docview.wss?&uid=swg27011656>

This Web site has an online reader comment form that you can use to send comments.

- You can also send comments by using the feedback link at the footer of each page in the Information Management Software for z/OS Solutions Information Center at <http://publib.boulder.ibm.com/infocenter/db2zhelp>.

---

## Chapter 1. Introduction

DB2 Version 9.1 for z/OS (Version 9) delivers exciting new innovations, in key areas, that help you realize the business value of your company's information.

Read on for a brief overview of the innovations that are delivered in this release of DB2 for z/OS.

### **Decreased total cost of ownership**

The total cost of ownership for maintaining an information management infrastructure is determined by many factors, including the costs of hardware and software and the number of staff required.

| With Version 9, reductions in your total cost of ownership can be realized through  
| savings in hardware and software, and productivity for people. The productivity  
| comes through improved SQL, the addition of XML, and the reduction in database  
| administration tasks. System z9<sup>®</sup> or System z10<sup>™</sup> offerings such as zIIP and new  
| Business Class and Enterprise Class Processors are economical and provide extra  
| capacity and reductions in DB2 costs for some distributed, parallel query, and  
| utility index workloads. To ease the challenging tasks that face database  
| administrators, some tasks have been automated or eliminated. As a result, a  
| database administrator can manage more terabytes of data and take on more  
| business tasks.

### **More power to leverage enterprise for information on demand**

Version 9 delivers a number of capabilities that help you leverage your enterprise for information on demand through increased programmer productivity, increased compatibility in the DB2 family of products, and improvements for managing enterprise data.

Native support for pureXML<sup>®</sup> increases productivity of programmers and provides unprecedented scalability and performance. With Version 9 at the core of a service-oriented architecture, you can have seamless integration of XML and relational data. Enhancements to SQL increase family compatibility and leadership. Improvements and additions in the QMF<sup>™</sup> family of products improve access to your enterprise data. New and enhanced DB2 Tools improve management of your enterprise data. Most of these improvements also provide benefits for key enterprise application partners (for example SAP, PeopleSoft, and Siebel) and their customers.

### **Increased availability through database definition on demand**

Version 8 delivered online schema change capabilities that increased database availability with less disruption.

Version 9 expands those capabilities to deliver data definition on demand. For example, improvements in online reorganization of table spaces for a few partitions is improved greatly. One of the important changes is the ability to replace one table with another table quickly. Another important change is the ability to rename a column or an index. Other capabilities let you modify, alter, create, and rebuild

objects and code either online or by other means that are less disruptive to the availability of your data.

## **Better performance**

Complex applications include both transactions and reporting. Performing both transactions and reporting well is imperative to running an enterprise efficiently. Version 9 delivers a number of enhancements that boost transaction performance and help reduce your total cost of ownership through reduced CPU time, improved query performance, faster disk access, and improved logging and insert performance.

Reductions in CPU time are observed for LOAD and REORG utilities and varying-length data. In addition, queries that are written in the native SQL procedural language are eligible to run on zIIP. Also, several improvements in disk access can reduce the time for sequential disk access.

Key improvements for reporting include optimization enhancements that improve query and reporting performance and ease of use. Improved query performance SQL improvements include more queries, more consistency with the database management industry and within the DB2 family (which contributes to ease of use and the ability to port applications to DB2 for z/OS).

## **Regulatory compliance**

Regulatory compliance, security and auditing are in the headlines and growing much more important. Version 9 helps you respond to increasing needs to protect security, to assure integrity, and to comply with regulations such as the Sarbanes-Oxley Act, the Health Insurance Portability and Accountability Act (HIPAA), and Payment Card Industry (PCI) Security Standards. Improved access control with network trusted context and roles allows more precise control of security. Improved filtering makes auditing more usable. Secure Socket Layer (SSL) data encryption on networks is more secure.

## **Increased synergy with System z®**

As in Version 8, Version 9 takes advantage of the latest improvements in System z9 and System z10 hardware and software to provide better performance, improved value, more resilience, and better function.

For example, remote native SQL procedures are now enabled for processing on the IBM System z9 or System z10 Integrated Information Processor (zIIP). DB2 synergy with System z9 and System z10 continues with a range of I/O improvements in channels, in disks, and in DB2. Tests with Version 8 that involve parallel access on one channel have shown faster response times for sequential access; similar improvements are anticipated for Version 9. For example, one early performance test for sequential performance achieved 183 MB per second. Version 9 also takes advantage of new System z9 and System z10 hardware support with a new decimal floating-point data type that lets you use decimal floating-point numbers with greater precision. Flashcopy can be used for DB2 database backup and restore operations. Other improvements for Version 9 on System z9 or System z10 include added security and encryption and Unicode collation.

---

## Chapter 2. Leveraging your enterprise for information on demand

DB2 for z/OS delivers functions and support that increase compatibility within the DB2 family and let you more easily leverage your enterprise for information on demand.

Increases in family compatibility translate to increases in productivity for programmers who work in a standard application environment and increases in the portability of your applications. Enhancements that address these areas include pureXML support in the DB2 database and compatibility and family leadership through various new and improved SQL functions and statements.

In addition to enhancing programmer productivity and application portability, improvements in DB2 QMF, and an array of DB2 Tools and new tools offerings such as the Optimization Service Center and DB2 Accessories Suite, let you access your data and manage your enterprise like never before.

Most of the new and improved functions deliver benefits for both business partners and customers.

---

### Seamless integration of XML data and relational data

XML is an important innovation that enables business-to-business communication of data, regardless of differences in the systems that receive and work with the data.

With XML, you can easily send semi-structured data across the Web without losing the relationship between data within a document. Because the content of an XML document is independent from the formatting instructions, data can be rendered appropriately in a variety of output formats. Rich support of XML within a database management system provides advantages that include more efficient storage, query, and indexing capabilities.

Support for XML capabilities and functions span the entire DB2 family. Version 8 of DB2 for z/OS and Version 8 of DB2 for Linux<sup>®</sup>, UNIX<sup>®</sup> and Windows<sup>®</sup> provide basic support for storing, retrieving, and querying XML documents. DB2 9 for Linux, UNIX and Windows continues the work by delivering rich support of XML. Now, this release of DB2 for z/OS expands on similar support by delivering seamless integration of XML data and relational data in the DB2 database.

Functional changes are provided in the following areas:

- XML document storage
- XML document retrieval
- Application development
- Database administration support
- Performance benefits through indexing support

#### XML document storage

This release of DB2 for z/OS provides fully integrated storage of XML data in the DB2 database system, which lets your client applications access and manage the XML data by leveraging DB2 functionality. The XML column data type is provided

for storing XML data in DB2 tables. Most SQL statements support the XML data type. As a result, you can perform many common database operations with XML data, such as creating tables with XML columns, adding XML columns to existing tables, creating indexes over XML columns, creating triggers on tables with XML columns, and inserting, updating, or deleting XML documents. A decomposition stored procedure is also provided. With this stored procedure, you can extract data items from an XML document and store those data items in columns of relational tables.

### **XML document retrieval**

You can use SQL to retrieve entire documents from XML columns in a way that is similar to retrieving data from any other type of column. When you need to retrieve portions of documents, you can specify XPath expressions, through SQL with XML extensions (SQL/XML).

### **Application development**

Application development support of XML in DB2 for z/OS enables applications to combine XML, relational data access, and storage. The following programming languages support the new XML data type:

- Assembler
- C or C++ (embedded SQL or DB2 CLI)
- COBOL
- Java™ (JDBC or SQLJ)
- PL/I

### **Database administration support**

Database administration support includes:

- An XML schema repository for all XML schemas that are required to validate and process XML documents that are stored in XML columns or decomposed into relational tables
- The ability to use DB2 for z/OS utilities to perform operations on XML objects in a way that is similar to handling LOB objects

### **Performance benefits through indexing support**

This release of DB2 for z/OS offers performance benefits through indexing support, which is available for data that is stored in XML columns. Using indexes over XML data can improve the efficiency of queries that you issue against XML documents.

---

## **Compatibility and leadership with SQL**

This release of DB2 for z/OS delivers functional changes in SQL that increase productivity of programmers through family compatibility and leadership.

Version 8 of DB2 for z/OS took big steps toward improving consistency of SQL across the DB2 family by providing more functions that are common between DB2 for Linux, UNIX and Windows. This release of DB2 for z/OS and DB2 9 for Linux, UNIX and Windows moves even more SQL function from the unique set to the common set. With the increase in common SQL, programmer productivity also increases. This release of DB2 for z/OS also introduces several new SQL functions that are firsts in the DB2 family.

## SQL consistency improvements

Increased consistency with SQL across the DB2 family is good news for programmers who write cross-platform applications, because porting and writing application programs becomes easier.

The following SQL functions, statements, and clauses are consistent across the DB2 family.

### **SELECT FROM UPDATE or SELECT FROM DELETE function**

The insert within select feature that was introduced in Version 8 of DB2 for z/OS has been expanded in Version 9 to support the retrieval of columns from rows that are modified by a SELECT FROM DELETE or a SELECT FROM UPDATE statement.

Now, one SQL call to DB2 modifies the table contents and returns the resultant changes to the application program. In addition, an application can now be coded to perform a destructive read from a table when a SELECT FROM DELETE statement is included. This feature is particularly useful when a table is used as a data queue.

### **INSTEAD OF triggers**

The introduction of INSTEAD OF triggers reduces the complexity in your application programs.

In previous versions of DB2 for z/OS, operations that required different rules for read and write access (for example, encryption and decryption) required users to build into their applications an awareness that the object for read access is a view, and that the object for write access is a base table.

INSTEAD OF triggers are defined on views only, and are used to process insert, update, and delete operations (through trigger logic) rather than relying on the INSERT, UPDATE, or DELETE statement to activate the trigger. With the INSTEAD OF trigger, an application doesn't need to include the complexity that specifies which operations are performed against views and which operations are performed against the base table because the activated trigger makes the operations seem as if they are performed against a view.

Examples of using INSTEAD OF triggers include updates through join queries or encoding and decoding data from the database within a view.

### **BIGINT data type and function**

This release of DB2 for z/OS introduces a new SQL data type and function, BIGINT (big integer), that supports big integers and extends the set of currently supported exact numeric data types (SMALLINT and INTEGER).

A big integer is a binary integer that has a precision of 63 bits. The BIGINT data type can represent 63-bit integers, and is compatible with all numeric data types. The BIGINT function returns a big integer representation of a number or a string representation of a number. You can store or load big integer values into DB2 tables, and manipulate the data in a variety of ways. In addition, you can use the CAST specification to increase the portability of your applications.

### **BINARY data type and function**

This release of DB2 for z/OS provides expanded support for binary string data by introducing the BINARY data type and function.

The VARBINARY (varying-length binary string) data type and function are also introduced. The VARBINARY data type and function are not yet supported by other members of the DB2 family.

The BINARY data type represents a fixed-length binary string. When fixed-length binary string distinct types, columns, and variables are defined, the length attribute is specified, and all values have the same length. A binary string column is used for storing non-character data, such as encoded or compressed data, pictures, voice, and mixed media. A binary string column can also hold structured data for use by distinct types, user-defined functions, and stored procedures. The BINARY function returns a BINARY (fixed-length binary string) representation of a string of any type or a row ID type.

#### **Related concepts**

“VARBINARY data type” on page 14

### **File reference variables**

This release of DB2 for z/OS adds support for two new file reference variables, LOB and XML.

A *file reference variable* is a host variable that is defined in a host language (for example C or COBOL) to contain the file name that directs file input and output for a large object (LOB).

DB2 support for file reference variables allows a large LOB or an XML value to be inserted from a file or selected into a file rather than a host variable; the application does not need to acquire storage to contain the LOB or XML value. File reference variables also facilitate the movement of LOB or XML values from the database server to a client application or from a client application to a database server without going through the working storage of the client application. In addition, file reference variables bypass host language limitations on the maximum allowed size for LOB values that are located in working storage. The following host languages support file reference variables:

- C and C++
- COBOL
- PL/I
- Assembler
- REXX™

### **INTERSECT keyword in a fullselect**

The new INTERSECT keyword specifies the intersection set operator in a fullselect operation between two result tables.

Two types of INTERSECT operations are available: INTERSECT ALL and INTERSECT DISTINCT. If you specify INTERSECT ALL, the result consists of all rows that are in both the first result table and the second result table, and redundant duplicate rows are included. If you specify INTERSECT DISTINCT, the result consists of all rows that are in the first and second result tables, and the redundant duplicate rows are eliminated. In either case, each row of the result exists in the first and second result tables.

### **EXCEPT keyword in a subselect**

The new EXCEPT keyword specifies the difference set operator in a fullselect operation between two result tables.

Two types of EXCEPT operations are available: EXCEPT ALL and EXCEPT DISTINCT. If you specify EXCEPT ALL, the result consists of all rows that are only

in the first result table, and redundant duplicate rows are included. If you specify EXCEPT DISTINCT, the result consists of all rows that are only in the first result table, and the redundant rows are eliminated. In either case, each row in the result is a row from the first result table that does not have a matching row in the second result table.

## **Native support for SQL procedures**

In this release of DB2 for z/OS, new support for native SQL procedures simplifies the definition and use of SQL procedures by eliminating the need for generating a C program.

In previous versions of DB2 for z/OS, a C program must be generated from transformed SQL statements and SQL control statements that are contained in the SQL procedure. The resulting C program is then run like an external stored procedure. Now, when you create a native SQL procedure in new-function mode, the procedural statements are converted to a representation that is stored in the database directory, as is true for other SQL statements. The parameter list procedure options are stored in the database catalog tables as in prior releases. When a native SQL procedure is called, the representation is loaded from the directory, and the DB2 engine runs the procedure. A number of additional functions and extensions provide consistency with the SQL Standard and with the rest of the DB2 family. Examples of these improvements include support for:

- Changing name resolution within a procedure body
- Using delimited identifiers, including lowercase characters, for SQL condition names, SQL labels, SQL variables, and SQL parameters
- Nested compound statements, including a compound statement within the body of a condition handler
- Versioning and managing source code
- Deploying of native SQL procedures to multiple servers
- Debugging of native SQL procedures

## **Nested compound statements in SQL procedures**

SQL procedures now support nested compound statements.

Compound statements introduce a block of SQL statements in an SQL procedure. Before Version 9, the body of an SQL procedure could contain a single compound statement (which could contain other SQL statements, but not another compound statement), or a single SQL procedure statement other than a compound statement. As a result, a condition handler could not contain a compound statement, either.

Now, with support for nested compound statements in SQL procedures, you can:

- Use a compound statement within a condition handler
- Use nested compound statements to define different scopes for SQL variables, cursors, condition names, and condition handlers.

## **Expanded support for not logging table spaces**

This release of DB2 for z/OS supports suspension of logging for base table spaces, XML table spaces, and the indexes that are associated with them.

Version 8 of DB2 for z/OS provides support for suspension of logging in global temporary tables, LOB table spaces, during the LOAD REPLACE operation, and during an insert of a declared temporary table. Now, you can specify the logging attributes, LOGGED or NOT LOGGED, at the table space level when you alter or create tables and table spaces, and when you create tables or auxiliary tables. You can also suppress logging during the online LOAD RESUME process.

Because the existing logging facilities in DB2 are already finely tuned, suspension of logging does not generally improve the performance of your system. However, the ability to suspend record logging is useful in a variety of situations in which data is being duplicated and loss of concurrency and recoverability is not a concern. In those cases, if the data is lost, you can re-create or regenerate it from the original source instead of using an image copy and applying log records. Examples of tables for which suspension of logging is advantageous include materialized query tables, summary tables, tables to which data is propagated, and temporary tables that are populated with the result set from a query as an intermediate step in an application. An additional benefit of suspended logging is improved scalability, particularly for operations that insert large volumes of data with the INSERT statement.

## **OLAP specifications for RANK, DENSE\_RANK, and ROW\_NUMBER**

Online analytical processing (OLAP) specifications provide the ability to return ranking, row numbering, and existing aggregate function information as a scalar value in the result of a query.

You can include OLAP specifications in an expression, in a select-list, or in the ORDER BY clause of a select-statement. The result to which the OLAP specification is applied is the result table of the innermost subselect that includes the OLAP specification. This release of DB2 for z/OS delivers OLAP specifications for RANK, DENSE\_RANK, and ROW\_NUMBER.

### **RANK and DENSE\_RANK**

RANK and DENSE\_RANK specify that the ordinal rank of a row within the specified window is computed. Rows that are not distinct with respect to the ordering within the specified window are assigned the same rank. You can define the results of ranking with gaps in the sequential rank numbering by using the RANK specification, or without gaps, by using the DENSE\_RANK specification. Common examples of using RANK or DENSE\_RANK specifications include:

- Ranking sales figures (for example, determining which stores in a chain have the lowest and highest sales)
- Ranking employees within a department or division according to various indicators
- Creating "top-*n*" queries (for example, retrieve the five employees that have the highest salaries)

### **ROW\_NUMBER**

ROW\_NUMBER specifies that a sequential row number is computed for the row that is defined by the ordering, starting with 1 for the first row. If the ORDER BY clause is not specified in the window, the row numbers are assigned to the rows in an arbitrary order, as the rows are returned (but not according to any ORDER BY clause in the select-statement). You can use ROW\_NUMBER to number the result rows of a query. Row numbers also enable easy formulation of queries for computing histogram statistics (quantile computations), and they enable formation of other OLAP specifications (for example, moving sums, moving averages, and so on).

### **COLLATION\_KEY function**

The new COLLATION\_KEY function supports culturally correct and case-insensitive collation of Unicode data.

The COLLATION\_KEY function processes a Unicode UTF-16 input string and a collation name, and returns a varying-length binary sort key. The result of the COLLATION\_KEY operation can be compared to the result of another COLLATION\_KEY operation on another string to determine their order within the specified collation name. The attributes of the collation name specify the collation characteristics (for example locale attribute, treatment of accent and case, and so on) for the sort key, and conform to the conventions of the z/OS-supported Unicode conversion services.

You can also use the COLLATION\_KEY function to perform case-insensitive (or *caseless*) comparisons of string expressions by specifying attributes on the collation name that either ignore case or ignore a combination of case and other attributes such as spaces, punctuation, and symbols.

### **Capability to create an index on an expression**

In this release of DB2 for z/OS, an extension to the CREATE INDEX statement lets you create an index on an expression (also known as an *index on expression*).

An expression can be a column reference, a built-in function invocation, or a general expression with certain restrictions. Unlike a simple index, the index key of an index on expression is composed by concatenating the result (also known as a key-target) of the expression that is specified in the ON clause. An index that is created on an expression lets a query take advantage of index access (if the index is chosen by the optimizer) and avoid a table space scan.

### **Automatic creation of a database, a table space, and all system-required objects**

The function of the CREATE TABLE statement now supports automatic (implicit) creation of a database or a table space.

**Implicit creation of a database, a table space, and all system-required objects:** A database, a table space, and all system-required objects are created implicitly if you do not name a table space or a database in the IN clause of a CREATE statement. The attributes of the table space (for example, table space type, the underlying data sets, and use of data compression) are determined by table space installation parameters. If a table space is implicitly created, the following system-required objects are also created:

- The enforcing primary key index
- The enforcing unique key index
- The ROWID index (if the ROWID column is defined as GENERATED BY DEFAULT)
- A LOB table space, auxiliary table, and auxiliary index

**Implicit creation of a table space:** A segmented table space is created implicitly in DSNDB04 if you do not name a table space or a database in the IN clause of a CREATE statement.

### **IBM Spatial Support for DB2 for z/OS**

The new IBM Spatial Support for DB2 for z/OS allows you to generate and analyze spatial information about geographic features and to store and manage the data on which this information is based.

Spatial Support for DB2 for z/OS provides of a set of spatial data types, user defined functions, and stored procedures for spatial related queries. With Spatial Support, you can:

- Invoke spatial queries for local and remote clients to answer questions based on geographic relationships.
- Create spatial indexes on spatial columns, which can improve query performance when you use spatial predicate functions.
- Manage geographic coordinate systems, spatial indexes, and spatial column usages through stored procedure interfaces.

In addition, Spatial Support includes an ODBC program that you can use to enable and disable the spatial feature. This program can invoke each of the stored procedures through a set of command line arguments.

## Leverage existing application programming skills

This release of DB2 for z/OS improves support for end-to-end application development by connecting new languages and environments to the scale and value of the existing infrastructure.

Increased consistency across the DB2 family lets you leverage the existing skills of your application programmers through support of a wider range of tools, environments, and languages. The IBM Rational® and WebSphere® product lines provide part of the connection, with products like Rational Data Architect, Rational Application Developer, and WebSphere Information Integration. DB2 clients provide more support for new environments and new languages.

This release of DB2 for z/OS supports key database technologies, including SQL and native SQL procedures, XML, service-oriented architecture (SOA), and various Web services.

A wide variety of development frameworks provide language support and application development and deployment tools for Web-based applications that work with the DB2 family. These frameworks also support both traditional programming languages and newer, open source languages. For example, development communities exist for the following traditional programming languages:

- COBOL, PL/I, REXX, C, C++, Fortran, assembler, and APL2®
- Java support, including JDBC and SQLJ
- Microsoft® language products, including .NET, Visual C# for .NET, and Visual Basic for .NET

Examples of Open Source development support include:

- Eclipse
- PHP and Zend Core for IBM
- Perl
- Python
- Ruby on Rails
- TOAD for DB2

**Note:** The IBM Data Server Driver for JDBC and SQLJ has been upgraded to Version 3.57, which is the same level that DB2 for Linux, UNIX, and Windows, and IBM Informix® Dynamic Server provide.

## Enhancements to large object support

This release of DB2 for z/OS offers many improvements for handling large objects (LOBs) through extensions to SQL statements and enhancements to utilities and performance.

## SQL enhancements for large objects

This release of DB2 for z/OS provides enhancements to the FETCH statement for LOB and XML data, and includes support for LOB file reference variables.

### Enhancements to the FETCH statement for LOB and XML data

In previous versions of DB2, applications that worked with LOBs had two primary methods for fetching LOB data: Fetching data into a pre-allocated buffer and using a LOB locator to retrieve a handle on the data. Fetching data into a preallocated buffer has the potential for causing virtual storage constraint problems, especially LOBs that have a large maximum length. In addition, an application that uses LOB locators that commit infrequently or do not explicitly free the locators can use considerable amounts of DB2 resources. The introduction of XML objects in Version 9 presents additional complications because XML columns do not have locators, and when a table containing an XML column is created, no maximum length is defined for the XML object.

Version 9 introduces the WITH CONTINUE clause on the FETCH statement to address the problems that are encountered when an application fetches LOB and XML data. You can use the WITH CONTINUE clause in an application to retrieve LOB and XML columns in multiple pieces without using a LOB locator, and continue a FETCH operation to retrieve the remaining data for LOB and XML columns when truncation occurs. The application manages the buffers and reassembles the pieces of data. Two common uses for FETCH CONTINUE include:

- Dynamic allocation of appropriate storage size: You can use the initial FETCH statement to fetch data into a preallocated buffer of a moderate size. If the returned data item is too large to fit in that buffer, you can use the length information that is returned to allocate the right amount of storage, and use one FETCH CONTINUE statement to retrieve the remaining data.
- Streaming data through a single, fixed-size buffer: If more data remains after a FETCH operation, you can use as many FETCH CONTINUE statements as necessary to retrieve the data and use the same buffer area. In this case, the data in the buffer must be processed after each FETCH or FETCH CONTINUE operation.

### File reference variables for large objects

A *file reference variable* is a host variable that is defined in a host language (for example C or COBOL) to contain the file name that directs file input and output for a large object (LOB). This release of DB2 for z/OS adds support for a LOB file reference variable.

With DB2 support for file reference variables, large LOB values can be inserted from a file or selected into a file rather than a host variable; the application does not need to acquire storage to contain the LOB value. File reference variables also facilitate the movement of LOB values from the database server to a client application or from a client application to a database server without going through the working storage of the client application. In addition, file reference variables bypass host language limitations on the maximum allowed size for LOB values that are located in working storage. The following host languages support file reference variables:

- C and C++
- COBOL
- PL/I
- Assembler

- REXX

## **Utilities enhancements for large objects**

This release of DB2 for z/OS provides enhancements for handling large objects (LOBs) in the following utilities: LOAD, UNLOAD, REORG, CHECK LOB, and CHECK DATA.

### **Improved LOB handling for LOAD and UNLOAD utilities**

Because of enhancements to the LOAD utility, an input field value can contain the name of the file that contains a LOB column value. The LOB column value is loaded from that file.

Enhancements to the UNLOAD utility let you store the value of a LOB column in a file and record the name of the file in the unloaded record in the base table.

### **Improved REORG of LOB table spaces**

Before Version 9, you cannot access LOB data during the REORG operation, and LOBs are moved within the existing LOB table space, which prevents physical space from being reclaimed from the LOB data set. Improvements in Version 9 overcome those previous limitations. Now, the original LOB table space is drained of writers. All LOBs are then extracted from the original data set and inserted into a shadow data set. When this operation is complete, all access to the LOB table space is stopped (the readers are drained) while the original data set is switched with the shadow data set. At this point, full access to the new data set is enabled, and an inline copy is taken to ensure recoverability of data.

### **CHECK LOB and CHECK DATA utilities**

The options SHRLEVEL REFERENCE and SHRLEVEL CHANGE have been added to both CHECK LOB and CHECK DATA utilities. These options dramatically reduce the amount of time during which the data is in the read-only state when you run CHECK LOB and CHECK DATA utilities.

The CHECK DATA utility checks table spaces for violations of referential and table check constraints and reports information about violations that it detects. CHECK DATA also checks for consistency between a base table space and the corresponding LOB or XML table spaces. The new SHRLEVEL REFERENCE and SHRLEVEL CHANGE options provide the following functions:

- CHECK DATA with SHRLEVEL REFERENCE specifies that applications can read from but cannot write to the index, table space, or partition that is to be checked during CHECK DATA processing.
- CHECK DATA with SHRLEVEL CHANGE specifies that applications can read from and write to the index, table space, or partition that is to be checked during CHECK DATA processing.

The CHECK LOB utility identifies any structural defects in the LOB table space and any invalid LOB values. You run the CHECK LOB utility before the CHECK DATA utility if a table space contains at least one LOB column. The new SHRLEVEL REFERENCE and SHRLEVEL CHANGE options provide the following functions:

- CHECK LOB with SHRLEVEL REFERENCE specifies that applications can read from but cannot write to the table space that is to be checked during CHECK LOB processing.

- CHECK LOB with SHRLEVEL CHANGE specifies that applications can read from and write to the table space that is to be checked during CHECK LOB processing.

### **Performance enhancements for large objects**

Several enhancements improve performance for retrieving data for small and medium LOBs and eliminate locks for LOB operations.

### **Optimization of data retrieval for small and medium LOBs**

In prior releases, processing large LOBs, particularly from a distributed perspective, has been optimized for retrieving larger amounts of data. Many applications effectively use locators to retrieve LOB data regardless of the size of the data that is retrieved. The mechanism involved incurs a separate network flow to determine the length of the data to be returned. Locators in this process remain active for the scope of the transaction, which can be longer than necessary for smaller amounts of data unless the locators are explicitly freed. As a result, valuable server resources are consumed.

Version 9 introduces the ability for a server to dynamically determine the most efficient mode in which to return LOB or XML data. When the dynamic data format is enabled, the life span of the locator is the scope of the cursor (a cursor-based locator) instead of the scope of the transaction. A new mechanism is also provided to allow the requester to retrieve sequential chunks of the LOB data while maintaining the position of the data (by means of the locator) at the server.

### **Increased performance through elimination of LOB locks**

LOB locks are used to serialize LOB table space access and to determine whether the previously deallocated LOB space can be reallocated. In prior releases, DB2 acquires a lock on the LOB value while performing insert, update, delete, and select operations, and during LOB space allocation. This release of DB2 for z/OS eliminates the acquisition of locks on every LOB operation including LOB locks that are used for space allocation. The requirement for a LOB lock is removed for insert, delete, update, and select operations. In addition, a LOB lock is no longer required to serialize the consistency between the value of the LOB and the column of the base row for an uncommitted read operation. As a result, lock escalation for LOB locks at the table space level is also eliminated. This enhancement improves the overall elapsed time for LOB data retrieval.

## **SQL leadership: family firsts**

This release of DB2 for z/OS provides leadership in SQL by offering a number of DB2 family firsts.

### **TRUNCATE TABLE statement**

The TRUNCATE TABLE statement provides an efficient mechanism for deleting all data rows in a designated DB2 table without activating delete triggers or altering the current table attributes in the DB2 catalog.

The statement also provides an IMMEDIATE option to permanently empty the designated DB2 table without issuing a commit, and it provides a REUSE STORAGE option to allow reuse of deallocated storage.

### **DECFLOAT built-in data type**

DECFLOAT (decimal floating point) is a new built-in SQL data type that has a maximum precision of 34 digits.

DECFLOAT data can be manipulated, stored, or loaded into DB2 tables.

A decimal floating-point value is an IEEE 754r (finite) number with a decimal point. The position of the decimal point is stored in each decimal floating-point value.

### **VARBINARY data type**

This release of DB2 for z/OS provides support for the VARBINARY data type and function.

The VARBINARY data type represents a varying-length binary string. The VARBINARY function returns a varying-length binary string representation of a string of any type.

#### **Related concepts**

“BINARY data type and function” on page 5

### **Enhancements to optimistic concurrency control and update detection**

Enhancements to optimistic concurrency control provide a faster, more scalable locking alternative to database locking for concurrent data access, and an additional related enhancement provides a mechanism to detect recent (daily, weekly, or monthly) database updates.

#### **Optimistic locking**

Optimistic locking minimizes the time for which a given resource is unavailable for use by other transactions.

Because DB2 can determine when a row was changed, it can ensure data integrity while limiting the time that locks are held. With optimistic concurrency control, DB2 releases the row or page locks immediately after a read operation. To ensure data integrity, DB2 also releases the row lock after each FETCH operation and takes a new lock on a row only for a positioned update or delete.

A new row change timestamp column that you specify on the CREATE TABLE and ALTER TABLE statement lets you implement optimistic concurrency control. The column is defined with one of two options:

- NOT NULL GENERATED ALWAYS FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP
- NOT NULL GENERATED BY DEFAULT FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP.

DB2 maintains the contents of the row change timestamp column. When you want to use this change token as a condition when making an update, you can specify an appropriate condition for this column in your WHERE clause.

#### **Daily, weekly, and monthly update detection**

Database administrators benefit from knowing the volume of updates within specific time ranges so that they can plan for data replication, create auditing scenarios, and so on.

A new expression, ROW CHANGE, returns a token or a timestamp that represents the last change to a row. Now, an application has the following options:

- Determine when a row was last changed (or changed within a range of dates or number of days) by using the ROW CHANGE TIMESTAMP expression

- Return a token as a BIGINT (big integer) value that represents a relative point in the modification sequence of a row by using the ROW CHANGE TOKEN expression

### **MERGE and SELECT FROM MERGE statements**

This release of DB2 for z/OS provides an easier and more efficient mechanism for merging data through the MERGE statement by using arrays of input.

In prior releases, merging data (from 100 transactions to a master table, for example) required many separate operations:

- 100 update operations to update existing rows in the master table
- Insert operation to insert zero to 100 rows from the transactions that do not currently exist in the master table

Now, the MERGE statement lets you update and insert many rows in a table from a single statement. You can embed the MERGE statement in an application program or issue it interactively. The statement is executable and can be dynamically prepared. In addition, you can use the SELECT FROM MERGE statement to return all the updated rows and inserted rows, including column values that are generated by DB2.

---

## **Tools that support your enterprise**

IBM offers many tools that help you to perform administration tasks, to access and integrate your information, and to effectively manage utilities, performance, recovery, and applications.

### **Accessing your enterprise data on demand with DB2 QMF**

DB2 QMF Version 9.1 supports the new function that is in DB2 V9.1 for z/OS.

This latest version of DB2 QMF offers a completely redesigned, cross-platform workstation and Web-based solution that provides on demand access to data, reports, and interactive visual solutions through a rich desktop application or an ordinary Web browser.

The DB2 QMF family of products includes:

- DB2 QMF for TSO/CICS
- DB2 QMF High Performance Option (HPO)
- DB2 QMF for Workstation
- DB2 QMF for WebSphere

DB2 QMF Version 9.1 provides the following enhancements:

- Support for DB2 Version 9.1 for z/OS, including XML and enhancements to SQL
- Drag-and-drop development of OLAP analytics, SQL queries, tabular reports, graphical reports, pivot tables, and data analysis views
- Drag-and-drop development of interactive visual solutions such as executive dashboards that are capable of graphically rendering data that is simultaneously mined from multiple data sources
- A rich security model that supports the personalization and dissemination of data and reports on a per-user or per-group basis
- A new object repository that simplifies the storage, access control, and distribution of QMF objects across the enterprise

- A re-engineered Eclipse-based desktop application that extends QMF to numerous workstation operating systems, including Windows, Linux, Solaris, and AIX®
- Full assimilation of the former QMF Visionary solution into the core DB2 QMF product, which extends the Visionary graphical objects to QMF reports and unifies the development of visual solutions within a single application
- A redesign of DB2 QMF for WebSphere, which extends key QMF functionality to browser-based users across multiple platforms

## **DB2 QMF for TSO/CICS**

DB2 QMF for TSO/CICS Version 9.1 offers significant enhancements in database processing, support for new DB2 V9.1 for z/OS data types, import and export, security, and new diagnostic capabilities.

Some of the highlights of Version 9.1 of DB2 QMF for TSO/CICS include:

- Enhancements in database processing include a new program parameter that takes advantage of DB2 multiple-row fetch and insert processing.
- Support for the following new DB2 V9.1 for z/OS data types:
  - BIGINT
  - BINARY
  - DECFLOAT
  - VARBINARY
  - XML
- Enhancements to import and export functions, including:
  - Support for new XML data format
  - Ability to export and import data using UNIX path names
  - Ability to export and import QMF objects and data using PDSE data sets
  - Easier specification of data set size
- Enhancements to security through the QMF CONNECT command, which now supports RACF mixed-case passwords
- Diagnostic support for several new DB2 V9.1 for z/OS SQL codes

## **DB2 QMF for Workstation**

The former DB2 QMF for Windows has been re-engineered and renamed DB2 QMF for Workstation.

The new Eclipse-based architecture enables DB2 QMF for Workstation to support popular workstation environments such as Linux, UNIX, Solaris, and AIX, in addition to Windows.

Other changes and enhancements to DB2 QMF for Workstation include:

- Full integration of the former DB2 QMF Visionary product into DB2 QMF for Workstation
- A new object repository that replaces the former Server Definitions File (SDF) and offers a variety of enhancements
- Several productivity and ease-of-use enhancements, including:
  - Enhanced support for OLAP
  - Re-engineered table editing capabilities
- A distributed architecture that:

- Enables users to access data sources remotely by using network or Web-service connections over HTTP/HTTPS
- Enables administrators to optionally consolidate database access on a designated server or collection of servers
- Blends the benefits of a zero client-side administration while retaining the flexibility and performance of a rich desktop application

## **DB2 QMF for WebSphere**

DB2 QMF for WebSphere provides dramatic improvements in performance and usability. Key enhancements include:

- A redesigned user interface with improved performance and reduced server response time, and a layout that is similar to DB2 QMF for Workstation
- Full support for capabilities that were formerly provided in DB2 QMF Visionary
- A fully independent Web-based QMF administration console that allows QMF administrators to manage the QMF for WebSphere application through an ordinary Web browser
- Support for new and existing data types and new SQL extensions in DB2 V9.1 for z/OS, including:
  - BIGINT
  - BINARY
  - DECFLOAT
  - VARBINARY
  - XML

## **DB2 QMF High Performance Option**

Key enhancements for Version 9.1 of the DB2 QMF High Performance Option (HPO) include:

- Leverage of DB2 multi-row fetch and insert for increased performance
- Support for DSQLLIB to allocate libraries, providing support that is comparable to that of QMF for TSO
- A new option to automatically replace an existing stored procedure
- Incorporates a stored procedure interface that takes a query and form as input, and returns the resulting report as a result set
- Adds support for the new BIGINT DB2 data type

## **Packaging options for DB2 QMF**

Two packaging options for DB2 QMF are available: DB2 QMF Enterprise Edition and DB2 QMF Classic Edition.

The DB2 QMF Enterprise Edition provides the entire DB2 QMF family of technologies, which enables enterprise-wide business information across end-user and database platforms. DB2 QMF Enterprise Edition consists of the following components:

- DB2 QMF for TSO/CICS
- DB2 QMF High Performance Option (HPO)
- DB2 QMF for Workstation
- DB2 QMF for WebSphere

The DB2 QMF Classic Edition supports end users that access DB2 for z/OS databases solely from traditional mainframe terminals and emulators, including

IBM Host On Demand. This edition consists of DB2 QMF for TSO/CICS.

## Managing your enterprise with DB2 Tools

DB2 Tools for z/OS help reduce manual tasks, maintain high availability, and perform information replication and integration functions.

Together, the array of tools that IBM offers can help you meet performance standards and control IT costs. You can combine a selection of tools from any category to build a complete, flexible, and affordable solution that is tailored to meet your specific needs. A wide variety of tools are available and ready to support all major new Version 9 functions.

### Database administration

Database administration tools meet common database service and support requirements to streamline the management of DB2. DB2 Database Administration tools include:

- IBM DB2 Administration Tool for z/OS
- IBM DB2 Administration Toolkit for z/OS the SAP Edition
- IBM DB2 Audit Management Expert for z/OS
- IBM DB2 Object Comparison Tool for z/OS
- IBM DB2 Storage Management Utility for z/OS
- IBM DB2 Thread Expert for z/OS
- IBM Data Encryption for IMS and DB2 Databases for z/OS

### Utilities management

Utilities management tools provide a full set of utilities that handle unload, load, and many other tasks. Numerous enhanced utilities help improve productivity, performance, and availability. Utilities management tools include:

- IBM DB2 Automation Tool for z/OS
- IBM DB2 Automation Toolkit for z/OS the SAP Edition
- IBM DB2 Cloning Tool for z/OS
- IBM DB2 High Performance Unload for z/OS
- IBM DB2 Utilities Suite for z/OS

**Important:** In this version of DB2 for z/OS the DB2 Utilities Suite is available as an optional product. You must separately order and purchase a license to such utilities, and discussion of those utility functions in this publication is not intended to otherwise imply that you have a license to them. See Part 1 of *DB2 Utility Guide and Reference* for packaging details.

### Performance management

Performance management tools help you keep DB2 running at peak levels even under heavy demand. Performance management tools include:

- IBM DB2 Buffer Pool Analyzer for z/OS
- IBM DB2 Optimization Expert for z/OS
- IBM DB2 Performance Toolkit for DB2 on z/OS the SAP Edition
- IBM DB2 Query Monitor for z/OS
- IBM DB2 SQL Performance Analyzer for z/OS
- IBM Tivoli OMEGAMON XE for DB2 Performance Expert on z/OS
- IBM Tivoli OMEGAMON XE for DB2 Performance Monitor on z/OS

## Recovery management

Recovery management tools manage a variety of image copy and backup and recovery tasks, including change accumulation, that improve the flexibility of DB2 and increase the value and availability of copied data for DB2. Recovery management tools include:

- IBM DB2 Archive Log Accelerator for z/OS
- IBM DB2 Change Accumulation Tool for z/OS
- IBM DB2 Log Analysis Tool for z/OS
- IBM DB2 Object Restore for z/OS
- IBM DB2 Recovery Expert for z/OS
- IBM Application Recovery Tool for IMS and DB2 Databases

## Application management

Application management tools help manage DB2 applications and offer cost-effective global access to information and Web-enabled transactions.

Application management tools include:

- IBM DB2 Bind Manager for z/OS
- IBM DB2 Data Archive Expert for z/OS
- IBM DB2 Path Checker for z/OS
- IBM DB2 Table Editor for z/OS
- IBM DB2 Test Database Generator for z/OS

## Business analysis

Business analysis tools help business analysts leverage new business intelligence from their database data. Business analysis tools include:

- IBM DataQuant for z/OS
- IBM QMF for z/OS
- IBM DB2 Web Query Tool for z/OS

## Information integration

Information integration tools offer enhanced data replication and data event publishing capabilities as part of the full Information Integration Solutions portfolio. Information integration tools include:

- IBM WebSphere Classic Data Event Publisher for z/OS
- IBM WebSphere Classic Federation Server for z/OS
- IBM WebSphere Classic Replication Server for z/OS
- IBM WebSphere Data Event Publisher for z/OS
- IBM WebSphere Data Integration Classic Connector for z/OS
- IBM WebSphere DataStage<sup>®</sup> for z/OS
- IBM WebSphere QualityStage for z/OS
- IBM WebSphere Replication Server for z/OS

## Related information

 [DB2 and IMS Tools Web page](#)



---

## Chapter 3. Availability in Version 9

This release of DB2 for z/OS offers improved availability through a new DB2 command and extensions to existing SQL statements and online utilities.

---

### Online REORG with no BUILD2 phase

In this release of DB2 for z/OS, the BUILD2 phase has been eliminated to improve availability of your data.

When you reorganize a table space with SHRLEVEL CHANGE in previous versions of DB2, the data that is undergoing reorganization is not available to applications during the relatively lengthy SWITCH and BUILD2 phases.

---

### Faster replacement of one table with another

This release of DB2 for z/OS provides the ability to replace tables faster through support for clone tables.

With clone table support, you can generate a copy of a current table, in the same table space, that has the same attributes, structure, and data as the original table. After you create a clone table, you can insert or load data into the clone table and exchange the clone table name with the current table name.

Extensions to the ALTER TABLE statement support the clone table function and let you:

- Create clone tables with the ADD CLONE option. Clone tables can be partitioned or non-partitioned, and you can make image copies of a clone table.
- Drop clone tables with the DROP CLONE option.
- Exchange current data with clone data with the EXCHANGE option, which provides a fast replacement of the original data, and is an alternative to the online LOAD REPLACE capability.

Web-based applications that need maximum availability can benefit from the option to implement a pair of tables that have cloned structures. Copies for application testing and auditing can be easily created. Clone table support also provides the unique ability to quickly change table names. As a result, applications can quickly and almost transparently switch between dual mirror tables. When the data in the clone table needs to become active to an application, you can use the EXCHANGE option of the ALTER TABLE statement to switch the table name, which provides fast replacement of the original data.

---

### Universal table spaces

Version 9 introduces a new type of table space: a *universal table space*.

A universal table space is a table space that is both segmented and partitioned. Two types of universal table spaces are available: the partition-by-growth table space and the range-partitioned table space.

A universal table space offers the following benefits:

- Better space management relative to varying-length rows: A segmented space map page provides more information about free space than a regular partitioned space map page.
- Improved mass delete performance: Mass delete in a segmented table space organization tends to be faster than table spaces that are organized differently. In addition, you can immediately reuse all or most of the segments of a table.

## Partition-by-growth table space

Before Version 9, partitioned tables required key ranges to determine the target partition for row placement. Partitioned tables provide more granular locking and parallel operations by spreading the data over more data sets. Now, you have the option to partition according to data growth, which enables segmented tables to be partitioned as they grow, without the need for key ranges. As a result, segmented tables benefit from increased table space limits and SQL and utility parallelism that were formerly available only to partitioned tables, and you can avoid needing to reorganize a table space to change the limit keys.

You can implement partition-by-growth table space organization in several ways:

- You can use the new MAXPARTITIONS clause on the CREATE TABLESPACE statement to specify the maximum number of partitions that the partition-by-growth table space can accommodate. The value that you specify in the MAXPARTITIONS clause is used to protect against run-away applications that perform an insert in an infinite loop.
- You can use the MAXPARTITIONS clause on the ALTER TABLESPACE statement to alter the maximum number of partitions to which an existing partition-by-growth table space can grow. This ALTER TABLESPACE operation acts as an immediate ALTER.

## Range-partitioned table space

*A range-partitioned table space* is a type of universal table space that is based on partitioning ranges and that contains a single table.

The new range-partitioned table space does not replace the existing partitioned table space, and operations that are supported on a regular partitioned or segmented table space are supported on a range-partitioned table space. You can create a range-partitioned table space by specifying both SEGSIZE and NUMPARTS keywords on the CREATE TABLESPACE statement.

With a range-partitioned table space, you can also control the partition size, choose from a wide array of indexing options, and take advantage of partition-level operations and parallelism capabilities. Because the range-partitioned table space is also a segmented table space, you can run table scans at the segment level. As a result, you can immediately reuse all or most of the segments of a table after the table is dropped or a mass delete has been performed.

---

## Better availability during REBUILD INDEX operations

The online REBUILD INDEX utility has been extended to support read and write access for a longer period of time during the utility operation.

As a result, applications have greater access to data while indexes on that data are being rebuilt. This complements the Version 8 support in which insert, update, and delete operations are supported on indexes that are non-unique while the index is in the process of rebuilding.

---

## Improved availability with column and index renaming capabilities

Two new capabilities allow you to rename a column without dropping and rebuilding a table, and to rename an index through a catalog operation.

The ALTER TABLE statement has been expanded to include the RENAME COLUMN clause. Now, you can rename a column without needing to drop and rebuild a table, which is less time consuming, reduces processing costs, and provides a higher degree of availability. The ability to rename a column in this manner also allows for better reuse of columns because you do not need to add additional columns.

The RENAME statement has been expanded to include the INDEX keyword, which lets you rename an index through a catalog operation. The ability to rename indexes in this manner also lets you consolidate index names that are used in different releases of SAP.

---

## Modify EARLY code without an IPL

The new DB2 command, REFRESH DB2,EARLY, lets you refresh EARLY code when DB2 is not active.

No IPL is required to complete the refresh operation. When you enter the command, the EARLY code modules are reloaded, and the EARLY control block is rebuilt. Previous copies of the EARLY modules are deleted the next time that you start DB2.

---

## ALTER TABLESPACE and index logging improvements

In this release of DB2 for z/OS, the ability to alter a table space to suppress logging is extended to base table spaces, XML table spaces, and the indexes that are associated with them.

Indexes inherit logging attributes from the base table with which they are associated. Suppression of logging is advantageous in a number of situations in which data is being duplicated. In those cases, if the data is lost, you can regenerate it from the original source rather than from an image copy and subsequent application of log records.

### Related concepts

“Expanded support for not logging table spaces” on page 7

---

## Support for using SMS storage classes with DB2-defined data sets

This release of DB2 for z/OS supports the ability to use SMS storage classes with DB2-defined data sets through the DB2 STOGROUP statement.

The CREATE STOGROUP and ALTER STOGROUP statements have been enhanced to include SMS data class, management class, and storage class as optional

parameters. Enabling DB2-defined data sets to use SMS storage classes increases the flexibility of handling these data sets while minimizing the time-consuming manual effort that is involved.

---

## | **DB2 support for extended address volumes (EAV)**

| z/OS Version 1 Release 11 introduces extended address volumes (EAV) support for  
| extended format sequential data sets and VSAM data sets.

| With EAV support, the maximum number of cylinders on a volume is 262,668. This  
| means that you can store more DB2 data on a single volume. However, the  
| maximum amount of data that you can store in a single data set of a DB2 table  
| space or index space is the same for EAV and non-extended address volumes,  
| which is 64 GB.

---

## Chapter 4. Performance in Version 9

This release of DB2 for z/OS provides improved performance through reductions in CPU processing time for a number of utilities, faster disk access, improved logging and insert operations, improved query optimization, and index improvements.

---

### Reduction in CPU processing time for utilities

CPU processing time is now reduced when you extract non-padded index keys from data rows by using certain utilities.

The following utilities extract non-padded index keys from data rows:

- COPY
- REORG
- LOAD
- REBUILD INDEX
- RECOVER
- RUNSTATS
- CHECK INDEX

---

### SQL optimization improvements

This release of DB2 for z/OS offers a number of SQL optimization enhancements, including support for histogram statistics, improved optimization techniques, and better management with the new DB2 Optimization Service Center.

#### Better data for the optimizer with histogram statistics

This release of DB2 for z/OS supports histogram statistics, which provide better data for the optimizer.

Data distribution statistics are important for query optimization. DB2 chooses the best access path based on cost. The basic foundation of cost is predicate selectivity estimation, which relies heavily on data distribution statistics. Prior versions of DB2 for z/OS rely on frequency statistics that are collected (through the RUNSTATS utility) on single values, either from a single column or from multiple columns. Now, with histogram statistics, DB2 can improve access path selection by estimating predicate selectivity from value-distribution statistics that are collected over the entire range of values in a data set, unlike frequency statistics.

#### Improved optimization techniques

This release of DB2 for z/OS adds or improves several techniques for optimizing query performance.

##### Global query optimization

In prior versions of DB2 for z/OS, query performance problems can result when DB2 breaks a query into multiple parts (for example, a correlated subquery and an outer query) and optimizes each of those parts independently. While each of the individual parts might be optimized to run efficiently, the overall result can be inefficient when these parts are combined. Version 9 supports global query optimization, which enables DB2 to optimize a query as a whole rather than as an

independent part. When a query is optimized as a whole entity, DB2 can consider the effect of one query block on another, and can consider reordering query blocks to determine the most efficient query path.

## **Sparse index and in-memory data caching enhancements**

In Version 8 of DB2 for z/OS, materialized work files for star join queries can be stored in the global storage pool above the 2-GB bar (an in-memory data cache) as more cost-effective alternative to the sparse index technique that is used in prior versions of DB2 for z/OS. A sparse index is used as a fallback at runtime if enough memory is not available in the global storage pool.

In Version 9, the use of in-memory data caching is generalized to support more types of queries. In addition, data is cached in a local storage pool above the 2-GB bar, which can reduce potential storage contention because data caching storage management is associated with each thread. You can allocate up to 20 MB of memory for data caching. If the required memory exceeds the specified limit, a sparse index is used instead. This expanded access method provides potential benefits for tables that lack an appropriate index or enough statistics, and competes with other options for the most efficient access path.

## **Complex query optimization through page-range screening enhancements**

The number of partitions that DB2 accesses to evaluate a query predicate can affect the performance of that query. A query that provides data retrieval through a data-partitioned secondary index (DPSI) might access some or all partitions of the DPSI. For a query that is based only on a DPSI key value or range, DB2 must examine all partitions. However, if the query also has predicates on the leading columns of the partitioning key, DB2 does not need to examine all of the partitions. Version 8 of DB2 for z/OS introduced page-range screening or, limited partition scan, as a technique for removing inapplicable partitions from consideration for access. Several techniques for writing or modifying queries to take advantage of page-range screen were also offered.

Now, Version 9 takes advantage of page-range screening by writing queries that create a non-matching predicate on the partitioning keys when literal values are used.

## **Autonomic reoptimization for dynamic queries**

The access path for static and dynamic queries that contain host variables is chosen by the optimizer during bind time, before the values of the host variables are available. Because the values of the host variables are not considered, the access path that is chosen might not always be optimal. In versions prior to Version 8 of DB2 for z/OS, this problem is partially solved by the REOPT(ALWAYS) bind option, which prepares the statement again at run time when the host input variables are available. As a result, the optimizer can reoptimize the access path based on known input variable values. However, frequently called SQL statements that take little time to execute are not good candidates for this method.

Version 8 of DB2 for z/OS introduced the bind option, REOPT(ONCE), which reoptimizes the access path of a dynamic query only once at run time, no matter how many times the same statement is executed. With this technique, DB2 chooses the access path based on the set of input variable values, and the access path is stored in the dynamic statement cache for use by subsequent executions of the

query. While this option offers additional flexibility, the impact of host variables on the selection of optimal access paths continues to be observed.

This release of DB2 for z/OS offers greater flexibility with the REOPT(AUTO) option, which enables DB2 to autonomously determine if a new access path is needed for a statement in the dynamic statement cache. When you specify REOPT(AUTO), DB2 optimizes the access path for dynamic SQL statements at the first execute or open operation. Each time a statement is executed, DB2 determines whether a new access path is needed to improve the performance of the statement. If a new access path will improve the performance, DB2 generates a new access path and replaces the previous access path in the dynamic statement cache.

### **FETCH FIRST *n* ROWS ONLY and ORDER BY in subselects and fullselects**

In prior versions of DB2 for z/OS, the ORDER BY and FETCH FIRST *n* ROWS ONLY clauses were supported only at the statement level as part of select-statement or a SELECT INTO statement. Version 9 delivers additional flexibility by allowing both FETCH FIRST *n* ROWS ONLY and ORDER BY clauses when they are specified as part of a subselect or a fullselect statement. With the new support for FETCH FIRST *n* ROWS ONLY, you can select the top *n* rows in the result table of a table expression, a leg of a union, or a subquery (for a fullselect that is a component of a predicate). If you specify the FETCH FIRST *n* ROWS ONLY clause with the ORDER BY clause, the ordering is performed on the entire result table before the first *n* rows are returned.

## **IBM Optimization Service Center for DB2 for z/OS**

IBM Optimization Service Center for DB2 for z/OS (Optimization Service Center) is a workstation tool that provides an innovative set of capabilities to improve the performance of individual DB2 SQL queries and entire SQL workloads that run on your DB2 for z/OS subsystem.

You can use Optimization Service Center to identify and analyze problem SQL statements and receive expert advice about statistics that you might gather to improve the performance of an individual statement.

For individual queries, Optimization Service Center can:

- Annotate the query
- Draw an access plan graph
- Generate query reports
- Facilitate generation of optimization hints
- Suggest statistics needed by the SQL optimizer

At your request, Optimization Service Center can send query-related information back to IBM service.

For SQL workloads, Optimization Service Center can gather workloads from various sources such as the dynamic statement cache, catalog tables, files and directories, QMF tables, and profile monitoring.

Additionally, Optimization Service Center can:

- Capture SQL queries immediately or at a scheduled time, either periodically or on a one-time basis.
- Suggest important statistics to be collected to improve performance for the entire workload.

- Manage and use profile monitoring. Profile monitoring provides a lightweight exception monitoring mechanism in which only exception statements are pushed out by DB2.

OSC is offered as part of the DB2 Accessories Suite for z/OS, an orderable, no-charge feature of DB2 Version 9.1 for z/OS.

## IBM DB2 Optimization Expert for z/OS

IBM DB2 Optimization Expert for z/OS (DB2 Optimization Expert) is a workstation tool that offers a comprehensive set of index and statistics advisors that you can use to improve system performance and lower your total cost of ownership.

In addition to a powerful set of expert advisor tools, DB2 Optimization Expert includes all of the Optimization Service Center functions.

With DB2 Optimization Expert, you can:

- View the query activity to find performance problems from a snapshot of the statement cache of your DB2 subsystem and a number of other sources (the DB2 catalog, for example)
- Get and implement expert tuning recommendations to improve the performance of an SQL query by using some or all of the following expert advisors:
  - Statistics advisor, which recommends statistics to update or collect to improve the performance of a query
  - Query advisor, which recommends ways to rewrite an SQL query to improve performance
  - Access path advisor, which alerts you to problematic access paths in the access plan for the query that might cause poor performance
  - Index advisor, which recommends new indexes to enhance the performance of an SQL query and generates CREATE INDEX statements that you can run from your workstation to implement the recommendations.
  - Optimization Service Center functions, which support advanced tuning with powerful optimization tools.

All the Optimization Service Center functions of the DB2 Accessories Suite are included with DB2 Optimization Expert. In summary, they enable experienced DBAs to understand, analyze, format, and optimize SQL statements.

- View the activity of groups of SQL statements to find performance problems: You can monitor the health of the SQL workloads. When the performance of SQL queries have been optimized, you can create monitor profiles for normal processing and exception processing that:
  - Monitor the health of SQL processing on the subsystem
  - Alert you when problems develop
  - Alert you when more tuning activities might be advised
- Get expert tuning advice from the following expert advisor functions to improve the performance of an entire SQL workload:
  - Workload statistics advisor, which recommends statistics to collect or update to improve the overall performance of the statements that make up an SQL workload.
  - Workload index advisor, which recommends indexes to create or modify to improve the overall performance of the statements that make up an SQL

workload. The workload index advisor also generates CREATE INDEX statements that you can run from your workstation to implement the recommendations.

- Workload query advisor, recommends ways to rewrite specific SQL queries to improve performance of a workload. Best-practice fixes to common query-writing mistakes are recommended.

DB2 Optimization Expert for z/OS is a separately orderable, priced DB2 Tool.

---

## Indexing improvements

Indexing improvements contribute to the overall improvements in query performance.

Specific improvements include index compression, index on expression, index key randomization, and larger index page sizes.

### Index compression

This release of DB2 for z/OS provides the ability to compress an index without using a dictionary. Compressing an index reduces the physical storage space that an index requires. By eliminating the need for a dictionary, index data can be compressed as soon as the first index entries are added to an index, and the space that would normally be used by a dictionary becomes available. You can choose whether you want to use index compression by specifying COMPRESS YES or COMPRESS NO on the CREATE INDEX or the ALTER INDEX statements.

### Index on expression

Support for a new index type, index on expression, lets you create an index on a general expression. Query performance can be enhanced if the optimizer chooses that index. When you use an index on an expression, the results of the expressions are evaluated during insertion time or during an index rebuild and are kept in the index. If the optimizer chooses to use that index, the predicate is evaluated against the values that are stored in the index. As a result, run-time performance overhead is eliminated.

### Larger index page sizes

Prior versions of DB2 for z/OS limit the size of an index page to 4 KB. The size of an index page limits the number of index keys that the index page can accommodate and can cause contention in indexes that split frequently. Version 9 lifts these restrictions by offering expanded index page sizes of 8 KB, 16 KB, and 32 KB. An index page size that is greater than 4 KB accommodates more index keys per page and can reduce the frequency of index page splits. You can use the INDEXBP option on both CREATE DATABASE and ALTER DATABASE statements to specify 4-KB, 8-KB, 16-KB, or 32-KB index buffer pools. You can also use the BUFFERPOOL keyword on the CREATE INDEX statement to specify 8-KB, 16-KB, and 32-KB buffer pools.

### Index key randomization

Currently, online transaction processing (OLTP) workloads in a data sharing environment can experience locking contention on the last page of an index, especially when an application program uses indexes on columns that contain current timestamps or ever-increasing sequential values create insertion hot spots.

As a result of this contention, an application must wait to acquire an index page. To offer relief from locking contention, Version 9 provides support to randomize index key columns through the new RANDOM option on both CREATE INDEX and ALTER INDEX statements.

When you specify the RANDOM option, values are stored at random places in the index tree. As a result, the number of consecutive insertions on a page decreases and contention is also decreased. You can use a randomly ordered index to perform equality lookups on a specified column. In addition, key columns that are in random order can be used in non-matching index scans, and index-only access on random key columns is possible. Even though values are stored in random order, you can retrieve the original value of the random key column.

#### **Related concepts**

“Capability to create an index on an expression” on page 9

“Relief for sequential key insert”

---

## **Improved performance for varying-length rows**

In Version 9, the format in which a row that contains varying-length columns is stored in the table has been changed to facilitate locating columns within the row for data retrieval and predicate evaluation.

As a result, you no longer need to run a sequential scan, and the performance improves for access to data in tables that store rows with varying-length columns.

In previous versions of DB2 for z/OS, if you store a value that is shorter than the length of a column in a varying-length column, the data is not padded to the full length of the column. As a result, columns that follow varying-length columns are at variable offsets in the row. When you need to locate and access such a column, you must scan the columns sequentially after the first varying-length column.

---

## **Relief for sequential key insert**

This release of DB2 for z/OS includes support for asymmetric splitting of index pages and increased index page sizes (larger than 4 KB).

In prior releases, an index page splits so that approximately half of the index keys on the splitting page remain on one page, and the remaining index keys move to a new page. This 50:50 split can cause frequent page splits on an index that has sequential insert patterns. As a result, half of the splitting pages are empty. Non-partitioned indexes that are updated by LOAD utility jobs that run in parallel against multiple partitions are especially susceptible to these problems because sequential inserts into multiple ranges in the non-partitioned index can occur.

Allowing index pages to split asymmetrically can improve space utilization and reduce contention that results from frequent page splits in an index with sequential insert patterns in the middle of the index. An index page size that is greater than 4 KB can also relieve contention by accommodating more index keys per page, which reduces the frequency of page splits in indexes. You can use the INDEXBP option on both CREATE DATABASE and ALTER DATABASE statements to specify 4-KB, 8-KB, 16-KB, or 32-KB index buffer pools, and the BUFFERPOOL keyword on the CREATE INDEX statement to specify 8-KB, 16-KB, and 32-KB buffer pools.

---

## Improved logging performance

This release of DB2 for z/OS provides several logging enhancements to improve performance.

These logging enhancements are as follows:

- In prior versions, data sharing workloads that perform a high volume of logging can experience significant delays because of the extra processing required to set unique LRSN (log record sequence number) values for each record. In Version 9, the process for setting and updating LRSN values has been streamlined to improve performance.
- Prior versions of DB2 for z/OS use the basic direct access method (BDAM) to read archive logs from disk, which facilitates rapid access to the VSAM control intervals, but which does not support striped data sets and extended format data sets. Version 9 replaces BDAM with the basic sequential access method (BSAM), which supports both striped and extended format data sets, and improves read and write performance.
- In prior versions of DB2 for z/OS, an active log can have up to 4 GB of tracks on a single disk volume, whereas the size limit for an archive log is only about 3 GB. As a result, installations that use 4-GB active logs are forced to put archive logs on tape. Version 1.7 of z/OS now supports large data sets and removes the 64-KB archive log limit. Version 9 takes advantage of the new limit by introducing the DSNTYPE=LARGE attribute of the PRIMARY QUANTITY field of installation panel DSNTIPA. The new attribute supports an archive log that has up to 4 GB of tracks per disk volume.

---

## Improved data insert performance

The APPEND option of the CREATE TABLE and ALTER TABLE statement offers increased performance for inserting data into the end of a table.

This option reduces the processing that is used to target the locations for new rows.

Like the LOAD RESUME utility, the rows are placed at the end of the table and are not clustered as they would be for a normal insert operation. The APPEND option provides a trade-off that favors inserts rather than retrievals and the need to reorganize.

Other Version 9 functions that can improve data insert performance include several index improvements, logging performances improvements, and not logging table spaces.

### Related concepts

“ALTER TABLESPACE and index logging improvements” on page 23

“Indexing improvements” on page 29

“Improved logging performance”



---

## Chapter 5. Security and regulatory compliance in Version 9

Regulatory compliance, security, and auditing are in the headlines and growing much more important.

This release of DB2 for z/OS helps you respond to increasing needs to protect security, to assure integrity, and to comply with regulations such as the Sarbanes-Oxley Act, the Health Insurance Portability and Accountability Act (HIPAA), and Payment Card Industry (PCI) Security Standards. Improved access control with network trusted context and roles allows more precise control of security. Improved filtering makes auditing more usable. Secure Sockets Layer (SSL) data encryption on networks is more secure.

---

### Roles and network trusted contexts

Support for trusted context addresses the problem of establishing a trusted relationship between DB2 and an external entity such as a database administrator or a middleware server.

With trusted context support, a series of trusted attributes are evaluated to determine whether a specific context can be trusted. After a trusted context is established, you can define a unique set of interactions between DB2 and the external entity, such as a middleware server, so that the existing database connection can be used by a different user without requiring authentication of the new connection user.

Support for trusted context also provides the ability, within a specific trusted context, for a DB2 authorization ID to acquire a special set of privileges that are not available outside that trusted context by defining roles. A role is a database entity that groups together one or more privileges and can be assigned to users. A role provides privileges, in addition to the current set of privileges, that are granted to the primary and secondary authorization identifiers. A role can own objects if the objects are created in a trusted context with the role defined as the owner. If a role is defined as an owner, then only the privileges that are granted to the role are considered for object ownership.

---

### Improved auditing

Improved trace filtering makes the jobs of auditing and performance management easier.

Many more options can be used to minimize the amount of data collected, so that the overhead is reduced and the extraneous data does not need to be processed.

---

### Support for Secure Socket Layer protocol

This release of DB2 for z/OS supports the Secure Socket Layer (SSL) protocol by implementing the z/OS Communications Server IP Application Transparent Transport Layer Security (AT-TLS) function.

The z/OS V1R7 Communications Server for TCP/IP introduces the AT-TLS function in the TCP/IP stack for applications that require secure TCP/IP connections. AT-TLS performs transport layer security on behalf of the application,

such as DB2, by invoking the z/OS system SSL in the TCP layer of the TCP/IP stack. The z/OS system SSL provides support for TLS V1.0, SSL V3.0, and SSL V2.0 protocols.

---

## More security options with INSTEAD OF triggers

INSTEAD OF triggers, which are defined on views, provide another way of ensuring security within the database.

INSTEAD OF triggers are used to process insert, update, and delete operations (through trigger logic) rather than relying on the INSERT, UPDATE, or DELETE statement to activate the trigger.

### Related concepts

“INSTEAD OF triggers” on page 5

---

## Support for AES encryption

This release of DB2 for z/OS supports Advanced Security Standard (AES) encryption of user IDs and passwords as a server and a requestor.

AES encryption is supported only if the server and requester are at the DRDA<sup>®</sup> security manager level 9 or later, and only for TCP/IP communications. Integrated Cryptographic Service Facility (ICSF) must be installed, configured, and active before servers can offer AES encryption and decryption services.

---

## Appendix. Planning for DB2 Version 9.1 for z/OS

Every new version of DB2 for z/OS introduces some technical changes that you need to be aware of as you plan for your migration. When you migrate from Version 8 to Version 9 of DB2 for z/OS, review the technical changes to ensure that your migration goes as smoothly as possible.

For example, Version 9 of DB2 for z/OS includes changes to commands, utilities, SQL statements, the DB2 catalog, performance monitoring, and instrumentation facility component identifiers (IFCIDs).

This information is intended for all users of DB2, including application programmers, database administrators, and system programmers. It assumes that you are familiar with Version 8 of DB2® Universal Database™ for z/OS.

This information assumes that your DB2 subsystem is running in Version 9 new-function mode. Generally, new functions that are described, including changes to existing functions, statements, and limits, are available only in new-function mode. Two exceptions to this general statement are new and changed utilities and optimization enhancements, which are also available in conversion mode unless stated otherwise.

---

### Command changes in Version 9

This release of DB2 for z/OS includes new and changed commands.

For complete information about all the changes, such as the syntax for new or changed commands, see DB2 commands (DB2 Command Reference).

#### New commands in Version 9

This release of DB2 for z/OS includes several new commands, which are listed in the following table.

*Table 1. New commands*

Command	Description
ACCESS DATABASE	Forces a physical open of a table space, index space, or partition, or removes the GBP-dependent status for a table space, index space, or partition.
DISPLAY PROFILE	Displays if profiling is active or inactive.
REFRESH DB2, EARLY	Reloads the EARLY code modules, and rebuilds the EARLY control block.
START PROFILE	Loads or reloads the profile table into a data structure in memory.
STOP PROFILE	Stops or disables the profile function.

#### Changed commands in Version 9

This release of DB2 for z/OS includes changed commands.

The following table shows that several existing commands have new and changed options.

Table 2. Changes to existing commands

Command	Description of enhancements and notes
-ALTER BUFFERPOOL (DB2)	<p>The ALTER BUFFERPOOL command now changes the values of certain parameters.</p> <p><b>New options:</b>            AUTOSIZE(NO)            AUTOSIZE(YES)</p> <p>The AUTOSIZE(NO) option is the default and specifies that the buffer pool does not use Workload Manager (WLM) services for automatic buffer pool sizing adjustment.</p> <p>The AUTOSIZE(YES) option specifies that the buffer pool uses WLM services, if available, to automatically adjust the size of the buffer pool based on dynamic monitoring of the workload goals and the available storage on the system.</p>
-ALTER UTILITY (DB2)	<p>The ALTER UTILITY command now changes the values of certain parameters of an execution of the REBUILD utility that uses SHRLEVEL CHANGE.</p> <p><b>New and changed options:</b>            REBUILD            REORG            DELAY (integer)</p> <p>The REBUILD option specifies that a REBUILD SHRLEVEL CHANGE utility is being altered.</p> <p>The REORG option specifies that a REORG SHRLEVEL REFERENCE or REORG SHRLEVEL change utility is being altered.</p> <p>The DELAY (integer) option specifies a lower bound for the interval between the time when the utility sends the LONGLOG message to the console and the time when the utility performs the action specified by the LONGLOG parameter. The integer is the delay in seconds.</p>
BIND PLAN (DSN) BIND PACKAGE (DSN) REBIND PLAN (DSN) REBIND PACKAGE (DSN)	<p><b>New and changed options:</b>            BIND PACKAGE DEPLOY            DBPROTOCOL            REOPT(AUTO)            ROUNDING            SET CURRENT PACKAGESET            SET CURRENT PACKAGE PATH</p> <p>The BIND PACKAGE DEPLOY bind option deploys a native SQL procedure.</p> <p>The DBPROTOCOL option now has DBPROTOCOL(DRDA) as the default value.</p> <p>The REOPT option with the keyword AUTO specified autonomically determines if a new access path needs to be generated to further optimize the performance for each execution.</p> <p>The ROUNDING option specifies the rounding mode at bind time. Rounding mode can be used to manipulate DECFLOAT data.</p> <p>You can use the BIND PACKAGE command with the SET CURRENT PACKAGESET option and SET CURRENT PACKAGE PATH option.</p>

Table 2. Changes to existing commands (continued)

Command	Description of enhancements and notes
-DISPLAY DATABASE (DB2)	<p>In Version 9.1, you can use the DISPLAY DATABASE to display the following objects:</p> <ul style="list-style-type: none"> <li>• XML table spaces</li> <li>• Clone table information</li> </ul> <p>The DISPLAY DATABASE command can display information about the status of XML table spaces, which are displayed with a type of 'XS'. The DISPLAY DATABASE command can display information about base table objects and their clones. The information is automatically displayed if a clone table exists.</p>
-DISPLAY DDF (DB2)	<p>The DISPLAY DDF command's output now includes IPv6 addresses that are in colon-hexidecimal format. When IPv4 addresses are used, IPv4 dotted decimal format is the only accepted format.</p>
-DISPLAY LOCATION (DB2)	<p>The output for the DISPLAY LOCATION command now includes IPv6 addresses that are in colon-hexidecimal format. When IPv4 addresses are used, IPv4 dotted decimal format is the only accepted format.</p>
-DISPLAY PROCEDURE (DB2)	<p>In Version 9.1, you can use the DISPLAY PROCEDURE to display native SQL procedures if you run in DEBUG mode. In Version 9, you need to run the procedure in DEBUG mode if the WLM environment column in the output contains the WLM ENVIRONMENT FOR DEBUG that you specified when you created the native SQL procedure. The DISPLAY PROCEDURE output shows the statistics of native SQL procedures as '0' if the native SQL procedures are under the effect of a STOP PROCEDURE command.</p>
-DISPLAY THREAD (DB2)	<p><b>New and changed options:</b> TYPE (PROC) LIMIT</p> <p>The TYPE keyword now has a PROC option which displays information about threads that are executing stored procedures and user-defined functions.</p> <p>The LIMIT option accepts numeric input that specifies the number of lines of output that is desired.</p> <p>Output for this command now includes IPv6 addressing.</p>

Table 2. Changes to existing commands (continued)

Command	Description of enhancements and notes
-DISPLAY TRACE (DB2)	<p data-bbox="553 275 699 296"><b>New options:</b></p> <ul style="list-style-type: none"> <li data-bbox="683 306 748 327">ROLE</li> <li data-bbox="683 333 764 354">XPLAN</li> <li data-bbox="683 361 797 382">XPKGLOC</li> <li data-bbox="683 388 797 409">XPKGCOL</li> <li data-bbox="683 415 813 436">XPKGPROG</li> <li data-bbox="683 443 797 464">XAUTHID</li> <li data-bbox="683 470 748 491">XLOC</li> <li data-bbox="683 497 781 518">XUSERID</li> <li data-bbox="683 525 813 546">XAPPNAME</li> <li data-bbox="683 552 797 573">XWRKSTN</li> <li data-bbox="683 579 797 600">XCONNID</li> <li data-bbox="683 606 797 627">XCORRID</li> <li data-bbox="683 634 764 655">XROLE</li> </ul> <p data-bbox="683 732 1198 753">The ROLE option specifies the connection roles.</p> <p data-bbox="683 768 1341 789">The XPLAN option provides exclude filtering for plan name.</p> <p data-bbox="683 804 1341 856">The XPKGLOC option provides exclude filtering for package location name.</p> <p data-bbox="683 871 1341 924">The XPKGCOL option provides exclude filtering for package collection ID.</p> <p data-bbox="683 938 1357 991">The XPKGPROG option provides exclude filtering for package program name.</p> <p data-bbox="683 1005 1395 1058">The XAUTHID option provides exclude filtering for authorization ID.</p> <p data-bbox="683 1073 1292 1094">The XLOC option provides exclude filtering for location.</p> <p data-bbox="683 1108 1325 1129">The XUSERID option provides exclude filtering for user ID.</p> <p data-bbox="683 1144 1317 1197">The XAPPNAME option provides IFC exclude filtering for application or transaction name.</p> <p data-bbox="683 1211 1292 1264">The XWRKSTN option provides IFC exclude filtering for workstation name.</p> <p data-bbox="683 1278 1411 1299">The XCONNID option provides exclude filtering for connection ID.</p> <p data-bbox="683 1314 1398 1335">The XCORRID option provides exclude filtering for correlation ID.</p> <p data-bbox="683 1350 1398 1371">The XROLE option provides exclude filtering for connection roles.</p> <p data-bbox="553 1402 1414 1486">The DISPLAY TRACE command now allows IP addresses to be specified in the LOCATION KEYWORD and the command accepts new colon-hexidecimal form for IP addresses.</p>
-DISPLAY UTILITY (DB2)	<p data-bbox="553 1524 691 1545"><b>New output:</b></p> <p data-bbox="683 1551 1341 1635">Output during UNLOAD phase of REORG with SHRLEVEL CHANGE or SHRLEVEL REFERENCE. Progress of the RECOVER utility.</p> <p data-bbox="553 1661 1395 1803">During the UNLOAD phase of REORG with SHRLEVEL CHANGE or SHRLEVEL REFERENCE, DSNU111I messages are issued for the following subtasks: unloading nonpartitioned indexes, building shadow nonpartitioned indexes, sort, build, and inline statistics. The phase for subtasks for unloading nonpartitioned indexes is UNLOADIX.</p> <p data-bbox="553 1829 1414 1887">During the LOGAPPLY phase, you can use the DISPLAY UTILITY command to check the progress status of the RECOVER utility.</p>

Table 2. Changes to existing commands (continued)

Command	Description of enhancements and notes
MODIFY irlmproc,SET (z/OS IRLM)	<p><b>New and changed options:</b>  DEADLOCK=<i>nnnn</i>  PVT=<i>nnnn</i></p> <p>DEADLOCK specifies the number, in milliseconds, of how often the local deadlock processing is scheduled.</p> <p>PVT specifies the upper limit of private storage that is used for locks. You can specify this value in megabytes or gigabytes by specifying M (for megabytes) or G (for gigabytes) after the value, as follows, <i>nnnnM</i> or <i>nnnnG</i>.</p>
-RESET INDOUBT DDF (DB2)	<p>When you issue the RESET INDOUBT command, you must specify IP addresses and ports with a double period (..) instead of a colon (:). When you are specifying the IPADDR attribute, you must now also specify an IP address and port.</p>
-START DATABASE (DB2)	<p>In Version 9.1, you can use the START DATABASE command on the following objects:</p> <ul style="list-style-type: none"> <li>• Databases</li> <li>• Table spaces</li> <li>• Index spaces</li> <li>• Physical partitions of partitioned table spaces or index spaces (including index spaces that contain data-partitioned secondary indexes)</li> <li>• Logical partitions of nonpartitioned secondary indexes</li> </ul>
-START DB2 (DB2)	<p><b>New and changed options:</b>  LIGHT(YES)  LIGHT(NOINDOUBTS)</p> <p>LIGHT(YES) specifies that a restart light is to be performed. DB2 starts with reduced storage and terminates normally after freeing retained locks. DB2 now waits for indoubt units of recovery to resolve before it terminates.</p> <p>LIGHT(NOINDOUBTS) specifies that DB2, during a restart light, does not wait for indoubt units of recovery to resolve before terminating.</p>
-START PROCEDURE (DB2)	<p>In Version 9.1, the START PROCEDURE command has new affects for native SQL procedures. The START PROCEDURE command affects current version of the native SQL procedures that you specify in the command.</p>
START irlmproc (z/OS IRLM)	<p><b>New and changed options:</b>  LTE=<i>nnnn</i>  MAXCSA=  PC=</p> <p>LTE specifies the number of lock table entries that are required in the coupling facility lock structure.</p> <p>MAXCSA is a required positional parameter but is currently unused.</p> <p>PC is a required positional parameter but is currently unused.</p> <p>MAXCSA and PC are currently unused because IRLM Version 2 Release 2 places locks only in private storage.</p>

Table 2. Changes to existing commands (continued)

Command	Description of enhancements and notes
-START TRACE (DB2)	<p data-bbox="553 279 699 300"><b>New options:</b></p> <ul style="list-style-type: none"> <li data-bbox="683 306 748 327">ROLE</li> <li data-bbox="683 333 764 354">XPLAN</li> <li data-bbox="683 361 797 382">XPKGLOC</li> <li data-bbox="683 388 797 409">XPKGCOL</li> <li data-bbox="683 415 813 436">XPKGPROG</li> <li data-bbox="683 443 797 464">XAUTHID</li> <li data-bbox="683 470 748 491">XLOC</li> <li data-bbox="683 497 781 518">XUSERID</li> <li data-bbox="683 525 813 546">XAPPNAME</li> <li data-bbox="683 552 797 573">XWRKSTN</li> <li data-bbox="683 579 797 600">XCONNID</li> <li data-bbox="683 606 797 627">XCORRID</li> <li data-bbox="683 634 764 655">XROLE</li> </ul> <p data-bbox="683 737 1198 758">The ROLE option specifies the connection roles.</p> <p data-bbox="683 772 1341 793">The XPLAN option provides exclude filtering for plan name.</p> <p data-bbox="683 808 1341 863">The XPKGLOC option provides exclude filtering for package location name.</p> <p data-bbox="683 877 1341 932">The XPKGCOL option provides exclude filtering for package collection ID.</p> <p data-bbox="683 947 1357 1001">The XPKGPROG option provides exclude filtering for package program name.</p> <p data-bbox="683 1016 1393 1071">The XAUTHID option provides exclude filtering for authorization ID.</p> <p data-bbox="683 1085 1292 1106">The XLOC option provides exclude filtering for location.</p> <p data-bbox="683 1121 1328 1142">The XUSERID option provides exclude filtering for user ID.</p> <p data-bbox="683 1157 1317 1211">The XAPPNAME option provides IFC exclude filtering for application or transaction name.</p> <p data-bbox="683 1226 1295 1281">The XWRKSTN option provides IFC exclude filtering for workstation name.</p> <p data-bbox="683 1295 1409 1316">The XCONNID option provides exclude filtering for connection ID.</p> <p data-bbox="683 1331 1398 1352">The XCORRID option provides exclude filtering for correlation ID.</p> <p data-bbox="683 1367 1398 1388">The XROLE option provides exclude filtering for connection roles.</p> <p data-bbox="553 1409 1414 1486">The START TRACE command now allows IP addresses to be specified in the LOCATION KEYWORD and the command accepts new colon-hexidecimal form for IP addresses.</p>
-STOP PROCEDURE (DB2)	<p data-bbox="553 1528 1414 1612">In Version 9.1, the STOP PROCEDURE command has new affects for native SQL procedures. The STOP PROCEDURE command affects current version of the native SQL procedures that you specify in the command.</p>

Table 2. Changes to existing commands (continued)

Command	Description of enhancements and notes
-STOP TRACE (DB2)	<p data-bbox="583 268 735 296"><b>New options:</b></p> <ul style="list-style-type: none"> <li data-bbox="712 302 781 329">ROLE</li> <li data-bbox="712 331 800 359">XPLAN</li> <li data-bbox="712 361 834 388">XPKGLOC</li> <li data-bbox="712 390 834 417">XPKGCOL</li> <li data-bbox="712 420 850 447">XPKGPROG</li> <li data-bbox="712 449 834 476">XAUTHID</li> <li data-bbox="712 478 786 506">XLOC</li> <li data-bbox="712 508 821 535">XUSERID</li> <li data-bbox="712 537 857 564">XAPPNAME</li> <li data-bbox="712 567 834 594">XWRKSTN</li> <li data-bbox="712 596 834 623">XCONNID</li> <li data-bbox="712 625 829 653">XCORRID</li> <li data-bbox="712 655 800 682">XROLE</li> </ul> <p data-bbox="712 730 1232 758">The ROLE option specifies the connection roles.</p> <p data-bbox="712 766 1373 793">The XPLAN option provides exclude filtering for plan name.</p> <p data-bbox="712 802 1373 863">The XPKGLOC option provides exclude filtering for package location name.</p> <p data-bbox="712 871 1373 932">The XPKGCOL option provides exclude filtering for package collection ID.</p> <p data-bbox="712 940 1390 1001">The XPKGPROG option provides exclude filtering for package program name.</p> <p data-bbox="712 1010 1430 1071">The XAUTHID option provides exclude filtering for authorization ID.</p> <p data-bbox="712 1079 1328 1106">The XLOC option provides exclude filtering for location.</p> <p data-bbox="712 1115 1360 1142">The XUSERID option provides exclude filtering for user ID.</p> <p data-bbox="712 1150 1349 1211">The XAPPNAME option provides IFC exclude filtering for application or transaction name.</p> <p data-bbox="712 1220 1328 1281">The XWRKSTN option provides IFC exclude filtering for workstation name.</p> <p data-bbox="712 1289 1446 1316">The XCONNID option provides exclude filtering for connection ID.</p> <p data-bbox="712 1325 1435 1352">The XCORRID option provides exclude filtering for correlation ID.</p> <p data-bbox="712 1360 1430 1388">The XROLE option provides exclude filtering for connection roles.</p> <p data-bbox="583 1409 1451 1486">The STOP TRACE command now allows IP addresses to be specified in the LOCATION KEYWORD and the command accepts new colon-hexidecimal form for IP addresses.</p>
-STOP DATABASE (DB2)	<p data-bbox="583 1528 1419 1589">In Version 9.1, you can use the STOP DATABASE command on the following objects:</p> <ul style="list-style-type: none"> <li data-bbox="583 1598 719 1625">• Databases</li> <li data-bbox="583 1633 745 1661">• Table spaces</li> <li data-bbox="583 1669 748 1696">• Index spaces</li> <li data-bbox="583 1705 1459 1766">• Physical partitions of partitioned table spaces or index spaces (including index spaces that contain data-partitioned secondary indexes)</li> <li data-bbox="583 1774 1200 1801">• Logical partitions of nonpartitioned secondary indexes</li> </ul>

---

## Changes to utilities in Version 9

This information summarizes the changes to utilities in this release of DB2 for z/OS. This release contains no new utilities.

### Utility changes in Version 9

This release of DB2 for z/OS includes changed utilities.

The following table lists and describes the new and changed options for many existing DB2 for z/OS utilities.

Table 3. New and changed utility options

Utility name	Description of enhancements and notes
BACKUP SYSTEM	<p><b>New option:</b> DUMP, DUMPONLY, DUMPCLASS, FORCE, TOKEN, ESTABLISH FCINCREMENTAL, and END FCINCREMENTAL</p> <p>DUMP, DUMPONLY, DUMPCLASS, FORCE, and TOKEN were added to support fast replication copies of the database copy pool or the log copy pool.</p> <p>ESTABLISH FCINCREMENTAL and END FCINCREMENTAL were added to specify a persistent or last incremental FlashCopy<sup>®</sup> relationship is to be established.</p>
CATMAINT	<p><b>New option:</b> SCHEMA SWITCH, OWNER FROM, and VCAT SWITCH</p> <p>SCHEMA SWITCH updates the owner, creator, or schema name, OWNER FROM changing the ownership of objects from a user to a role, and the VCAT SWITCH changes the catalog name used by the storage groups, user indexes, and table spaces.</p>
CHECK DATA	<p><b>New option:</b> CLONE, LOBERROR, and XMLERROR</p> <p><b>Changed option:</b> SHRLEVEL REFERENCE, SHRLEVEL CHANGE, PUNCHDDN, DRAIN_WAIT, RETRY, and RETRY_DELAY</p> <p>CLONE was added to support checking clone tables.</p> <p>LOBERROR and XMLERROR were added to specify what action to perform if a LOB or XML error is found.</p> <p>SHRLEVEL REFERENCE and SHRLEVEL CHANGE were added to support LOB table spaces.</p> <p>PUNCHDDN was added to specify a DD statement for a data set.</p> <p>DRAIN_WAIT, RETRY, and RETRY_DELAY were added to improve availability.</p>
CHECK INDEX	<p><b>New option:</b> CLONE</p> <p>CLONE was added to check only the specified indexes that are on clone tables.</p>

Table 3. New and changed utility options (continued)

Utility name	Description of enhancements and notes
CHECK LOB	<p><b>New option:</b> CLONE</p> <p><b>Changed option:</b> SHRLEVEL REFERENCE, SHRLEVEL CHANGE, PUNCHDDN, DRAIN_WAIT, RETRY, and RETRY_DELAY</p> <p>CLONE was added to check the LOB table space for only the clone table, not the LOB data for the base table.</p> <p>SHRLEVEL REFERENCE and SHRLEVEL CHANGE were added to support LOB table spaces.</p> <p>PUNCHDDN was added to specify a DD statement for a data set.</p> <p>DRAIN_WAIT, RETRY, and RETRY_DELAY were added to improve availability.</p>
COPY	<p><b>New option:</b> CLONE, SCOPE ALL, and SCOPE PENDING</p> <p><b>Changed option:</b> CHANGELIMIT</p> <p>CLONE was added to copy only clone data in a specified table space or index space.</p> <p>SCOPE ALL copies all of the specified objects and SCOPE PENDING allows you to only copy objects in COPY-pending or informational COPY-pending status.</p> <p>CHANGELIMIT ANY was added to take a full image copy if any pages have changed since the last image copy.</p>
COPYTOCOPY	<p><b>New option:</b> CLONE</p> <p>CLONE was added to support processing only image copy data sets that were taken against clone tables or indexes on clone tables.</p>
DIAGNOSE	<p><b>New option:</b> CLONE</p> <p>CLONE was added to support displaying information for only the specified objects that are clone tables, table spaces that contain clone tables, indexes on clone tables, or index spaces that contain clone tables.</p>
LISTDEF	<p><b>New option:</b> CLONED and XML</p> <p>CLONED indicates that the INCLUDE or EXCLUDE expression is to return only the names of clone tables, table spaces that contain clone tables, indexes on clone tables or index spaces that contain indexes on clone tables.</p> <p>XML was added to specify that only XML objects are to be included in this element of the list.</p>

Table 3. New and changed utility options (continued)

Utility name	Description of enhancements and notes
LOAD	<p><b>New options:</b>            BIGINT, BINARY, VARBINARY, DECFLOAT_ROUNDMODE, and            DECFLOAT</p> <p>BIGINT, BINARY, and VARBINARY data types were added to specify a binary string in a column.</p> <p>DECFLOAT_ROUNDMODE was added to support various rounding modes.</p> <p>DECFLOAT was added to support the Decimal Floating Point data type.</p>
MERGECOPY	<p><b>New option:</b>            CLONE</p> <p>CLONE was added to support processing only image copy data sets that were taken against clone objects.</p>
MODIFY RECOVERY	<p><b>New option:</b>            CLONE, LAST (<i>integer</i>), and LOGLIMIT</p> <p>CLONE was added to support deleting SYSCOPY records and any related SYSLGRNX records for only clone objects.</p> <p>LAST (<i>integer</i>) was added to specify the number of recent records to be retained in SYSIBM.SYSCOPY.</p> <p>LOGLIMIT was added to determine the oldest archive log timestamp.</p>
QUIESCE	<p><b>New option:</b>            CLONE</p> <p>CLONE was added to support creating a quiesce point for only the specified table spaces that contain clone tables.</p>
REBUILD INDEX	<p><b>New option:</b>            CLONE and SHRLEVEL CHANGE</p> <p>CLONE was added to support rebuilding only the specified indexes that are on clone tables.</p> <p>SHRLEVEL CHANGE was added to allow application to read from and write to the index, table space, or partition that is to be checked.</p>
RECOVER	<p><b>New option:</b>            CLONE, RESTOREBEFORE <i>X'byte-string'</i>, CURRENTCOPYONLY,            FROMDUMP, and DUMPCLASS</p> <p>CLONE was added to support recovering only clone table data in the specified table spaces or the specified index spaces that contain indexes on clone tables.</p> <p>RESTOREBEFORE was added to recover an image copy with an RBA or LRSN value earlier than the specified <i>X'byte-string'</i> value for use in the RESTORE phase.</p> <p>CURRENTCOPYONLY was added to specify that the primary copy is used for the restore.</p> <p>FROMDUMP and DUMPCLASS were added to support fast replication copies of the database copy pool or the log copy pool.</p>

Table 3. New and changed utility options (continued)

Utility name	Description of enhancements and notes
REORG INDEX	<p><b>New option:</b> CLONE</p> <p>CLONE was added to support reorganizing only the specified index spaces that contain indexes on clone tables.</p>
REORG TABLESPACE	<p><b>New option:</b> CLONE</p> <p><b>Changed options:</b> SHRLEVEL REFERENCE</p> <p>CLONE was added to support reorganizing only clone tables from the specified table spaces.</p> <p>SHRLEVEL REFERENCE was updated to support LOB table spaces.</p> <p>REORG was modified to reduce the amount of virtual storage that it uses to build compression dictionaries.</p>
REPAIR	<p><b>New options:</b> CLONE, VERSIONS, INDEXSPACE, NOAREOENDSTAR</p> <p><b>Changed options:</b> DBD REBUILD</p> <p>CLONE was added to support processing only the specified table spaces that contain clone tables.</p> <p>The VERSIONS option updates the version information for the named table space or index in the catalog and directory. Use this option when you are moving objects from one system to another or as a part of version number management.</p> <p>The INDEXSPACE option allows you to identify the index by specifying the qualified name of the index space, which you can obtain from the SYSIBM.SYSINDEXES table.</p> <p>You can use the NOAREOENDSTAR option to reset the advisory REORG-pending (AREO*) status of the specified table space or index.</p> <p>You no longer need to start a database for access by utilities only before REPAIR DBD REBUILD could be performed. DB2 now performs this step for you.</p>
REPORT	<p><b>New options:</b> SHOWDSNS</p> <p>SHOWDSNS was added to include VSAM data set names in the TABLESPACESET report.</p>
RESTORE SYSTEM	<p><b>New options:</b> FROMDUMP, DUMPCLASS, TAPEUNITS, and RSA</p> <p>FROMDUMP and DUMPONLY were added to support fast replication copies of the database copy pool or the log copy pool.</p> <p>TAPEUNITS was added to limit the number of tape drives that the utility should dynamically allocate.</p> <p>RSA was added to specify a key-label in the utility control statement.</p>

Table 3. New and changed utility options (continued)

Utility name	Description of enhancements and notes
RUNSTATS	<p><b>New options:</b> HISTOGRAM and NUMQUANTILES</p> <p><b>Changed options:</b> INDEX LIST</p> <p>HISTOGRAM and NUMQUANTILES were added to collect histogram statistics from columns.</p> <p>Support for the correlation stats-spec keywords was added, when specified along with the RUNSTATS INDEX LIST keywords.</p>
TEMPLATE	<p><b>New options:</b> LIMIT</p> <p>LIMIT added to support template switching.</p>
UNLOAD	<p><b>New options:</b> CLONE, BIGINT, BINARY, VARBINARY, SKIP LOCKED DATA, DECFLOAT_ROUNDMODE, DECFLOAT, and XML</p> <p><b>Changed options:</b> INTEGER EXTERNAL</p> <p>CLONE was added to support unloading data only from clone tables in the specified table spaces.</p> <p>BIGINT, BINARY, and VARBINARY data types were added to specify a binary string in a column.</p> <p>SKIP LOCKED DATA was added to skip rows on which incompatible locks are held by other transactions.</p> <p>DECFLOAT_ROUNDMODE was added to support various rounding modes.</p> <p>DECFLOAT was added to support the Decimal Floating Point data type.</p> <p>XML was added to specify that an XML column is being unloaded directly to the output record.</p> <p>For INTEGER EXTERNAL, The greater than and less than values of the original data where changed to have a broader range.</p>
DSNJU003 (change log inventory)	<p><b>New options:</b> IPV4, IPV6, GRPIPV4, GRPIPV6, NOIPV4, NOIPV6, NGRPIPV4, NGRPIPV6</p> <p>IPV4, IPV6, GRPIPV4, and GRPIPV6 were added to the DDF statement to identify an IP address.</p> <p><b>Changed options:</b> SYSPITR</p> <p>FFFFFFFFFFFF value added to cause system point-in-time recovery to occur without log truncation.</p>
DSNJU004 (print log map)	<p><b>New options:</b> IPV4, IPV6, GRPIPV4, GRPIPV6,</p> <p>IPV4, IPV6, GRPIPV4, and GRPIPV6 were added to the print log output.</p>

## Other utility changes in Version 9

This release of DB2 for z/OS includes general changes to utilities.

Other changes to utilities are:

- You can use the following utilities to interact with the new partition-by-growth table spaces: COPY, LOAD, REBUILD INDEX, RECOVER, REORG TABLESPACE, and DSN1COPY.
- XML data can now be loaded or unloaded.
- You can use the RUNSTATS utility to scan an index on expression.
- You cannot run concurrent **REORG TABLESPACE SHRLEVEL CHANGE PART *integer*** on the same table space. Instead of submitting multiple jobs, you can merge the jobs into one job and specify a range using **REORG TABLESPACE SHRLEVEL CHANGE PART *integer1:integer2***, or specify **REORG TABLESPACE SHRLEVEL CHANGE SCOPE PENDING** if multiple partitions are in a REORG-pending state.

---

## SQL statement changes in Version 9

This release of DB2 for z/OS features new and changed SQL statements.

For complete information about all changes, such as the syntax for new or changed SQL statements, comprehensive descriptions of keywords, and examples of usage, see DB2 SQL (DB2 SQL Reference).

## New SQL statements in Version 9

This release of DB2 for z/OS includes new SQL statements, which are listed in the following table.

Table 4. New SQL statements

SQL statement	Description
ALTER PROCEDURE (SQL - native)	Changes the description of or defines additional versions for a native SQL procedure
ALTER TRUSTED CONTEXT	Changes the description of a trusted context
CREATE PROCEDURE (SQL - native)	Defines a native SQL procedure
CREATE ROLE	Defines a role
CREATE TRUSTED CONTEXT	Defines a trusted context
EXCHANGE	Exchanges the data between a base table and the associated clone table
MERGE	Updates and/or inserts one or more rows of a table
RENAME	Renames an existing table or index
SET CURRENT DEBUG MODE	Assigns a value to the CURRENT DEBUG MODE special register
SET CURRENT DECFLOAT ROUNDING MODE	Assigns a value to the CURRENT DECFLOAT ROUNDING MODE special register
SET CURRENT ROUTINE VERSION	Assigns a value to the CURRENT ROUTINE VERSION special register
TRUNCATE	Deletes all rows from a table

## Changed SQL statements in Version 9

In this release of DB2 for z/OS, many existing SQL statements have new and changed clauses.

The following table shows the changes to existing SQL statements.

Table 5. Changes to existing SQL statements

SQL statement	Description of enhancements and notes
ALTER FUNCTION (external)	<b>New clauses:</b> NO PACKAGE PATH or PACKAGE PATH <i>package-path</i>
ALTER INDEX	<b>New clauses:</b>  REGENERATE
ALTER PROCEDURE (external)	<b>New clauses:</b> NO PACKAGE PATH or PACKAGE PATH <i>package-path</i>
ALTER STOGROUP	<b>New clauses:</b>  DATACLAS <i>dc-name</i> MGMTCLAS <i>mc-name</i> STORCLAS <i>sc-name</i>
ALTER TABLE	<b>New clauses:</b>  FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP HIDDEN ADD CLONE DROP CLONE APPEND NO or APPEND YES RENAME COLUMN <i>source-column-name</i> TO <i>target-column-name</i>  <b>Changed clauses:</b>  SESSION USER or USER
ALTER TABLESPACE	<b>New clauses:</b>  LOGGED or NOT LOGGED MAXPARTITIONS <i>integer</i> SEGSIZE <i>integer</i>
COMMENT	<b>New clauses:</b> ACTIVE VERSION or VERSION <i>routine-version-id</i> ROLE TRUSTED CONTEXT <i>context-name</i>
CREATE DATABASE	<b>New clauses:</b> AS WORKFILE ROLE TRUSTED CONTEXT <i>context-name</i>
CREATE FUNCTION (external scalar)	<b>New clauses:</b>  NO PACKAGE PATH or PACKAGE PATH <i>package-path</i>

Table 5. Changes to existing SQL statements (continued)

SQL statement	Description of enhancements and notes
CREATE FUNCTION (external table)	<p><b>New clauses:</b></p> <p>NO PACKAGE PATH or PACKAGE PATH <i>package-path</i></p>
CREATE INDEX	<p><b>New clauses:</b></p> <p>RANDOM  <i>key-expression</i>            GENERATE KEY USING XMLPATTERN            prolog            pattern-expression            AS SQL data-type</p>
CREATE PROCEDURE (external)	<p><b>New clauses:</b></p> <p>ALLOW DEBUG MODE, DISALLOW DEBUG MODE,            or DISABLE DEBUG MODE            NO PACKAGE PATH or PACKAGE PATH <i>package-path</i></p>
CREATE STOGROUP	<p><b>New clauses:</b></p> <p>DATACLAS <i>dc-name</i>            MGMTCLAS <i>mc-name</i>            STORCLAS <i>sc-name</i></p>
CREATE TABLE	<p><b>New clauses:</b></p> <p>FOR EACH ROW ON UPDATE AS ROW CHANGE TIMESTAMP            HIDDEN            PARTITION BY SIZE            APPEND NO or APPEND YES</p> <p><b>Changed clauses:</b></p> <p>SESSION_USER or USER</p> <p>If a database name or a table space name is not specified for the CREATE TABLE statement, and DB2 is in conversion mode, DB2 will create a segmented table space with a SEGSIZE of 4 and LOCKSIZE ROW. If DB2 is in new-function mode and a database name or a table space name is not specified, DB2 will implicitly create a database or table space.</p>
CREATE TABLESPACE	<p><b>New clauses:</b></p> <p>LOGGED or NOT LOGGED            MAXPARTITIONS <i>integer</i></p> <p>If the SEGSIZE, NUMPARTS, or MAXNUMPARTS clauses are not specified, a segmented table space is created with a default SEGSIZE of 4. If a database name is not specified, a segmented table space with the SEGSIZE of 4 is created in the default database, DSNDB04.</p>
CREATE TRIGGER	<p><b>New clauses:</b></p> <p>INSTEAD OF            ON <i>view-name</i></p>
DECLARE GLOBAL TEMPORARY TABLE	<p><b>New clauses:</b></p> <p>SESSION_USER or USER</p>

Table 5. Changes to existing SQL statements (continued)

SQL statement	Description of enhancements and notes
DELETE	<p><b>New clauses:</b></p> <p><i>include-column</i> SKIP LOCKED DATA</p>
DROP	<p><b>New clauses:</b></p> <p>ROLE TRUSTED CONTEXT <i>context-name</i></p>
EXECUTE	<p><b>New clauses:</b></p> <p><i>source-row-data</i></p>
EXPLAIN	<p><b>New clauses:</b></p> <ul style="list-style-type: none"> <li>• STMTS MONITORED</li> <li>• STMTS ALL</li> <li>• STMTS SCOPE</li> </ul>
FETCH	<p><b>New clauses:</b></p> <p>WITH CONTINUE CONTINUE</p>
GET DIAGNOSTICS	<p><b>New clauses:</b></p> <p>DB2_LINE_NUMBER</p>
GRANT	<p><b>New clauses:</b></p> <p>ROLE <i>role-name</i> DEBUGSESSION</p>
INSERT	<p><b>New clauses:</b></p> <p><i>include-column</i></p>
PREPARE	<p><b>New clauses:</b></p> <p>SKIP LOCKED DATA</p>
RENAME	<p><b>New clauses:</b></p> <p>INDEX</p>
REVOKE	<p><b>New clauses:</b></p> <p>ROLE <i>role-name</i> DEBUGSESSION</p>
SELECT INTO	<p><b>New clauses:</b></p> <p>SKIP LOCKED DATA</p>

Table 5. Changes to existing SQL statements (continued)

SQL statement	Description of enhancements and notes
SET PATH	<b>New clauses:</b> SESSION_USER or USER
SET SCHEMA	<b>New clauses:</b> SESSION_USER or USER
SET <i>transition-variable</i> assignment	<b>New clauses:</b> DEFAULT
UPDATE	<b>New clauses:</b>  <i>include-column</i> DEFAULT SKIP LOCKED DATA

## New functions in Version 9

This release of DB2 for z/OS includes new built-in functions that improve the power of the SQL language.

The following table shows the new built-in functions.

Table 6. New functions

Function name	Description
ADMIN_TASK_LIST	Returns a table with one row for each of the tasks that are defined in the task list of the administrative task scheduler.
ADMIN_TASK_STATUS	Returns a table with one row for each task in the task list of the administrative task scheduler that contains the status for the last time the task was run
ASCII CHR	Returns the character that corresponds to the ASCII code value that is specified by the argument
ASCII_STR	Returns the ASCII version of the character or graphic string argument
COLLATION_KEY	Returns a string that represents the collation key of the argument in the specified collation
COMPARE_DECFLOAT	Returns a SMALLINT value that indicates whether two arguments are equal or unordered, or whether one argument is greater than the other
CORRELATION	Returns the coefficient of the correlation of a set of number pairs
COVARIANCE or COVARIANCE_SAMP	Returns the covariance (population) of a set of number pairs
DECFLOAT	Returns a DECFLOAT representation of its argument
DECFLOAT_SORTKEY	Returns a binary value that can be used when sorting DECFLOAT values
DECRYPT_BINARY	Returns the decrypted value of an encrypted argument
DIFFERENCE	Returns a value that represents the difference between the sound of two strings based on applying the SOUNDEX function to the strings
EBCDIC CHR	Returns the character that corresponds to the EBCDIC code value that is specified by the argument
EBCDIC_STR	Returns an EBCDIC version of the string argument
EXTRACT	Returns a portion of a date or timestamp based on its arguments

Table 6. New functions (continued)

Function name	Description
GENERATE_UNIQUE	Returns a character string of bit data that is unique compared to any other execution of the function
GETVARIABLE	Returns a varying-length character string representation of the value of a session variable
LEFT	Returns a string that consists of the specified number of leftmost bytes or the specified string length units
LOCATE_IN_STRING	Returns the position at which the first occurrence of an argument starts within a specified string
LPAD	Returns a string that is padded on the left with blanks or a specified string
NORMALIZE_DECFLOAT	Returns a DECFLOAT value that is the result of normalizing the input argument
NORMALIZE_STRING	Returns a string value that is the result of normalizing the input Unicode value
OVERLAY	Returns a string that is composed of an argument inserted into another argument at the same position where some number of bytes have been deleted
QUANTIZE	Returns a DECFLOAT value that is equal in value (except for any rounding) and sign to one argument and which has an exponent set to be equal to the exponent of the second argument
RID	Returns the RID of a row
RIGHT	Returns a string that consists of the specified number of rightmost bytes or specified code units of a string
RPAD	Returns a string that is padded on the right with blanks or a specified string
SOUNDEX	Returns a value that represents the sound of the words in the argument
TIMESTAMPADD	Returns a timestamp derived from adding the specified interval to a timestamp
TIMESTAMP_ISO	Returns a timestamp derived from its arguments
TOTALORDER	Returns a SMALLINT value that indicates the comparison order of two arguments
UNICODE	Returns the Unicode (UTF-16) code value of the leftmost character of the argument as an integer
UNICODE_STR	Returns a string in Unicode (UTF-8 or UTF-16) that represents a Unicode encoding of the argument
VARCHAR_FORMAT	Returns a varying-length character string representation of a timestamp, with the string in a specified format
XMLATTRIBUTES	Returns an XML sequence that contains an XQuery attribute node for each non-null argument
XMLCOMMENT	Returns an XML value with a single comment node from a string expression
XMLDOCUMENT	Returns an XML value with a single document node and zero or more nodes as its children
XMLPARSE	Returns an XML value from parsing the argument as an XML document
XMLPI	Returns an XML value with a single processing instruction node
XMLQUERY	Returns an XML value from the evaluation of an XPath expression against a set of arguments
XMLSERIALIZE	Returns an SQL character string or a BLOB value from an XML value
XMLTEXT	Returns an XML value with a single text node that contains the value of the argument

## Reserved words

Certain words cannot be used as ordinary identifiers in some contexts because those words might be interpreted as SQL keywords. For example, ALL cannot be a column name in a SELECT statement. Each word, however, can be used as a delimited identifier in contexts where it otherwise cannot be used as an ordinary identifier. For example, if the quotation mark (") is the escape character that begins and ends delimited identifiers, "ALL" can appear as a column name in a SELECT statement.

New reserved words for this version of DB2 for z/OS are identified with notes in each section of this topic. In addition, some topics in this information might indicate words that cannot be used in the specific context that is being described.

IBM SQL has additional reserved words that DB2 for z/OS does not enforce. Therefore, you should not use these additional reserved words as ordinary identifiers in names that have a continuing use. See *IBM DB2 SQL Reference for Cross-Platform Development* for a list of the words.

### A

---

ADD	ALTER	ASSOCIATE
AFTER	AND	ASUTIME
ALL	ANY	AT <sup>1</sup>
ALLOCATE	AS	AUDIT
ALLOW	ASENSITIVE	AUX
		AUXILIARY

---

**Note:** 1. New reserved word for Version 9.1.

---

### B

---

BEFORE  
BEGIN  
BETWEEN  
BUFFERPOOL  
BY

---

### C

---

CALL	CLUSTER	CONTAINS
CAPTURE	COLLECTION	CONTENT
CASCADED	COLLID	CONTINUE
CASE	COLUMN	CREATE
CAST	COMMENT	CURRENT
CCSID	COMMIT	CURRENT_DATE
CHAR	CONCAT	CURRENT_LC_CTYPE
CHARACTER	CONDITION	CURRENT_PATH
CHECK	CONNECT	CURRENT_SCHEMA <sup>1</sup>
CLONE <sup>1</sup>	CONNECTION	CURRENT_TIME
CLOSE	CONSTRAINT	CURRENT_TIMESTAMP
		CURSOR

---

**Note:** 1. New reserved word for Version 9.1.

---

## D

---

DATA	DELETE	DO
DATABASE	DESCRIPTOR	DOCUMENT <sup>1</sup>
DAY	DETERMINISTIC	DOUBLE
DAYS	DISABLE	DROP
DBINFO	DISALLOW	DSSIZE
DECLARE	DISTINCT	DYNAMIC
DEFAULT		

---

**Note:** 1. New reserved word for Version 9.1.

---

## E

---

EDITPROC	ENDING	EXECUTE
ELSE	END-EXEC <sup>1</sup>	EXISTS
ELSEIF	ERASE	EXIT
ENCODING	ESCAPE	EXPLAIN
ENCRYPTIONEND	EXCEPT	EXTERNAL
	EXCEPTION	

---

**Note:** COBOL only

---

## F

---

FENCED	FREE
FETCH	FROM
FIELDPROC	FULL
FINAL	FUNCTION
FOR	

---

## G

---

GENERATED	GRANT
GET	GROUP
GLOBAL	
GO	
GOTO	

---

## H

---

HANDLER
HAVING
HOLD
HOUR
HOURS

---

## I

---

IF	INF <sup>1</sup>	INSERT
IMMEDIATE	INFINITY <sup>1</sup>	INTERSECT <sup>1</sup>
IN	INHERIT	INTO
INCLUSIVE	INNER	IS
INDEX	INOUT	ISOBID
	INSENSITIVE	ITERATE

---

**Note:** 1. New reserved word for Version 9.1.

---

## J

---

JAR  
JOIN

---

## K

---

KEEP<sup>1</sup>  
KEY

---

**Note:** 1. New reserved word for Version 9.1.

---

## L

---

LABEL	LIKE	LOCK
LANGUAGE	LOCAL	LOCKMAX
LC_CTYPE	LOCALE	LOCKSIZE
LEAVE	LOCATOR	LONG
LEFT	LOCATORS	LOOP

---

## M

---

MAINTAINED	MINUTES
MATERIALIZED	MODIFIES
MICROSECOND	MONTH
MICROSECONDS	MONTHS
MINUTE	

---

## N

---

NAN <sup>1</sup>	NULL
NEXTVAL	NULLS
NO	NUMPARTS
NONE	
NOT	

---

**Note:** 1. New reserved word for Version 9.1.

---

## O

---

OBID	OPTIMIZE
OF	OR
ON	ORDER
OPEN	OUT
OPTIMIZATION	OUTER

---

## P

---

PACKAGE	PATH	PRIQTY
PARAMETER	PIECESIZE	PRIVILEGES
PART	PLAN	PROCEDURE
PADDED	PRECISION	PROGRAM
PARTITION	PREPARE	PSID
PARTITIONED	PREVVAL	PUBLIC <sup>1</sup>
PARTITIONING		

---

**Note:** 1. New reserved word for Version 9.1.

---

## Q

---

QUERY
QUERYNO

---

## R

---

READS	RESULT_SET_LOCATOR	ROUND_DOWN <sup>1</sup>
REFERENCES	RETURN	ROUND_FLOOR <sup>1</sup>
REFRESH	RETURNS	ROUND_HALF_DOWN <sup>1</sup>
RESIGNAL	REVOKE	ROUND_HALF_EVEN <sup>1</sup>
RELEASE	RIGHT	ROUND_HALF_UP <sup>1</sup>
RENAME	ROLE <sup>1</sup>	ROUND_UP <sup>1</sup>
REPEAT	ROLLBACK	ROW <sup>1</sup>
RESTRICT	ROUND_CEILING <sup>1</sup>	ROWSET
RESULT		RUN

---

**Note:** 1. New reserved word for Version 9.1.

---

## S

---

SAVEPOINT	SET	STOGROUP
SCHEMA	SIGNAL	STORES
SCRATCHPAD	SIMPLE	STYLE
SECOND	SNAN <sup>1</sup>	SUMMARY
SECONDS	SOME	SYNONYM
SECQTY	SOURCE	SYSFUN
SECURITY	SPECIFIC	SYSIBM
SEQUENCE	STANDARD	SYSPROC
SELECT	STATIC	SYSTEM
SENSITIVE	STATEMENT <sup>1</sup>	
SESSION_USER <sup>1</sup>	STAY	

---

**Note:** 1. New reserved word for Version 9.1.

---

## T

---

TABLE	TRUNCATE <sup>1</sup>
TABLESPACE	TYPE <sup>1</sup>
THEN	
TO	
TRIGGER	

---

**Note:** 1. New reserved word for Version 9.1.

---

## U

---

UNDO	USER
UNION	USING
UNIQUE	
UNTIL	
UPDATE	

---

## V

---

VALIDPROC	VCAT
VALUE	VIEW
VALUES	VOLATILE
VARIABLE	VOLUMES
VARIANT	

---

## W

---

WHEN  
WHENEVER  
WHERE  
WHILE  
WITH  
WLM

---

## X

---

XMLELEMENT  
XMLEXISTS<sup>1</sup>  
XMLNAMESPACES<sup>1</sup>  
XMLCAST<sup>1</sup>

---

**Note:** 1. New reserved word for Version 9.1.

---

## Y

---

YEAR  
YEARS

---

## Other SQL language changes in Version 9

In addition to the many new SQL statements and functions, this release of DB2 for z/OS includes other enhancements to the SQL language.

The following table lists these additional changes to the SQL language.

Table 7. Other changes to the SQL language

Item	Description
Expressions	ROW CHANGE <i>expression</i> and OLAP specifications are provided as new expressions. A ROW CHANGE expression returns a token or a timestamp that represents the last change to a row. OLAP specifications (RANK, DENSE_RANK, and ROW_NUMBER) provide the ability to return ranking, and row numbering information as a scalar value in the result of a query. RANK, DENSE_RANK, and ROW_NUMBER are sometimes referred to as window functions.
Functions	The following functions now allow a source string that has a data type of CHAR FOR BIT DATA, VARCHAR FOR BIT DATA, or CLOB FOR BIT DATA: <ul style="list-style-type: none"><li>• CHAR</li><li>• INSERT</li><li>• LEFT</li><li>• LOCATE</li><li>• RIGHT</li><li>• SUBSTRING</li><li>• VARCHAR</li></ul>
Predicates	XMLEXISTS is provided as a new predicate. XMLEXISTS evaluates an XPath expression and returns a true or false value.
Special registers	This release of DB2 for z/OS introduces several new special registers. <p>The CURRENT DEBUG MODE specifies the default value for the DEBUG MODE option of a CREATE PROCEDURE statement for a native SQL procedure or a Java procedure.</p> <p>The CURRENT DECFLOAT ROUNDING MODE specifies the default rounding mode that is used for DECFLOAT values.</p> <p>The CURRENT ROUTINE VERSION specifies the version identifier that is to be used when invoking a native SQL routine.</p> <p>The ENCRYPTION PASSWORD special register specifies the encryption password and the password hint (if one exists) that are used by the ENCRYPTION and DECRYPTION built-in functions.</p> <p>The SESSION_USER special register replaces the USER special register (USER can be specified as a synonym for SESSION_USER).</p>

Table 7. Other changes to the SQL language (continued)

Item	Description
Session variables	<p>Similar to special registers, session variables are another way to provide information to applications. This release of DB2 for z/OS supports many new DB2-defined session variables that store information that can be referenced by SQL statements. You can use the GETVARIABLE built-in function to retrieve the values of session variables.</p> <p>The new DB2-defined session variables are:</p> <ul style="list-style-type: none"> <li>• SYSIBM.APPLICATION_ENCODING_SCHEME</li> <li>• SYSIBM.COBOL_STRING_DELIMITER</li> <li>• SYSIBM.DATE_FORMAT</li> <li>• SYSIBM.DATE_LENGTH</li> <li>• SYSIBM.DECIMAL_ARITHMETIC</li> <li>• SYSIBM.DECIMAL_POINT</li> <li>• SYSIBM.DEFAULT_DECFLOAT_ROUND_MODE</li> <li>• SYSIBM.DEFAULT_DEFAULT_SSID</li> <li>• SYSIBM.DEFAULT_LANGUAGE</li> <li>• SYSIBM.DEFAULT_LOCALE_LC_CTYPE</li> <li>• SYSIBM.DISTRIBUTED_SQL_STRING_DELIMITER</li> <li>• SYSIBM.DSNHDECP_NAME</li> <li>• SYSIBM.DYNAMIC_RULES</li> <li>• SYSIBM.ENCODING_SCHEME</li> <li>• SYSIBM.MIXED_DATA</li> <li>• SYSIBM.NEWFUN</li> <li>• SYSIBM.PAD_NUL_TERMINATED</li> <li>• SYSIBM.SQL_STRING_DELIMITER</li> <li>• SYSIBM.SSID</li> <li>• SYSIBM.STANDARD_SQL</li> <li>• SYSIBM.TIME_FORMAT</li> <li>• SYSIBM.TIME_LENGTH</li> </ul>
Support to change existing columns from LONG VARCHAR and LONG VARGRAPHIC to VARCHAR and VARGRAPHIC	<p>Although LONG VARCHAR and LONG VARGRAPHIC continue to be supported, you should define your columns as VARCHAR or VARGRAPHIC instead. To assist you in changing columns in existing tables from LONG VARCHAR and LONG VARGRAPHIC to VARCHAR and VARGRAPHIC, support is added to ALTER TABLE to change the data types of columns from LONG VARCHAR and LONG VARGRAPHIC to VARCHAR and VARGRAPHIC.</p>

## Catalog changes in Version 9

This release of DB2 for z/OS includes changed catalogs and some new catalogs and indexes.

For a complete description of the columns of the new and changed catalog tables, see DB2 SQL (DB2 SQL Reference).

For a summary of when the catalog changes are made, see Installation and migration (DB2 Installation Guide).

## New catalog tables in Version 9

This release of DB2 for z/OS includes new catalog tables, which are listed in the following table.

Table 8. New catalog tables

Catalog table name	Description
SYSIBM.SYSCONTEXT	Contains one row for each trusted context.
SYSIBM.SYSCONTEXTAUTHIDS	Contains one row for each authorization ID with which the trusted context can be used.
SYSIBM.SYSCTXTRUSTATTRS	Contains one row for each list of attributes for a given trusted context.
SYSIBM.SYSDEPENDENCIES	Records the dependencies between objects.
SYSIBM.SYSENVIRONMENT	Records the environment variables when an object is created.
SYSIBM.SYSINDEXSPACESTATS	Contains real time statistics for index spaces.
SYSIBM.SYSJAVAPATHS	Contains the complete JAR class resolution path, and records the dependencies that one JAR has on the JARs in its Java path.
SYSIBM.SYSKEYTARGETS	Contains one row for each key-target that is participating in an extended index definition.
SYSIBM.SYSKEYTARGETSTATS	Contains partition statistics for selected key-targets. For each key-target, a row exists for each partition in the table. Rows are inserted when RUNSTATS collects indexed key statistics or non-indexed key statistics for a partitioned table space. No row is inserted if the table space is nonpartitioned.
SYSIBM.SYSKEYTARGETS_HIST	Contains rows from the SYSKEYTARGETS table. Whenever rows are added or changed in SYSKEYTARGETS, the rows are also written to this table.
SYSIBM.SYSKEYTGTDIST	Contains one or more rows for the first key-target of an extended index key.
SYSIBM.SYSKEYTGTDISTSTATS	Contains zero or more rows per partition for the first key-target of a data-partitioned secondary index. Rows are inserted when RUNSTATS scans a data-partitioned secondary index. No row is inserted if the index is a secondary index.
SYSIBM.SYSKEYTGTDIST_HIST	Contains rows from the SYSKEYTGTDIST table. Whenever rows are added or changed in SYSKEYTGTDIST, the rows are also written to this table.
SYSIBM.SYSOBJROLEDEP	Lists the dependent objects for each role.
SYSIBM.SYSROLES	Contains one row for each role.
SYSIBM.SYSROUTINESTEXT	An auxiliary table for the TEXT column of SYSIBM.SYSROUTINES and is required to hold the LOB data.
SYSIBM.SYSTABLESPACESTATS	Contains real time statistics for table spaces.
SYSIBM.SYSXMLRELS	Contains one row for each XML table that is created for an XML column.
SYSIBM.SYSXMLSTRINGS	Each row contains a single string and its unique ID that are used to condense XML data. The string can be an element name, attribute name, name space prefix, or a namespace URI.

Table 8. New catalog tables (continued)

Catalog table name	Description
SYSIBM.XSRCOMPONENT	Auxiliary table for the BLOB column COMPONENT in SYSIBM.SYSXSROBJECTCOMPONENTS. It is in LOB table space SYSXSRA3.
SYSIBM.XSROBJECTS	Contains one row for each registered XML schema. Rows in this table can only be changed using static SQL statements issued by the DB2-supplied XSR stored procedures.
SYSIBM.XSROBJECTCOMPONENTS	Contains one row for each component (document) in an XML schema. Rows in this table can only be changed using static SQL statements issued by the DB2-supplied XSR stored procedures.
SYSIBM.XSROBJECTGRAMMAR	An auxiliary table for the BLOB column GRAMMAR in SYSIBM.SYSXSROBJECTS. It is in LOB table space SYSXSRA1.
SYSIBM.XSROBJECTHIERARCHIES	Contains one row for each component (document) in an XML schema to record the XML schema document hierarchy relationship. Rows in this table can only be changed using static SQL statements issued by the DB2-supplied XSR stored procedures.
SYSIBM.XSROBJECTPROPERTY	An auxiliary table for the BLOB column PROPERTIES in SYSIBM.SYSXSROBJECTS. It is in LOB table space SYSXSRA2.
SYSIBM.XSRPROPERTY	An auxiliary table for the BLOB column COMPONENT in SYSIBM.SYSXSROBJECTCOMPONENTS. It is in LOB table space SYSXSRA3.

## Changed catalog tables in Version 9

Many existing catalog tables are changed in this release of DB2 for z/OS.

The following table shows a list of the new columns and the existing columns that are revised. Revisions to columns include new column descriptions, new values for a column, changed data types, changed column lengths, or both changed data types and lengths.

Table 9. Summary of new and revised catalog table columns

Catalog table name	New column	Revised column
IPLIST		IPADDR
IPNAMES		IPADDR USERNAMES SECURITY_OUT
LOCATIONS	TRUSTED SECURE	
SYSAUXRELS	RELCREATED	TBOWNER AUXTBOWNER
SYSCHECKDEP		TBOWNER
SYSCHECKS	RELCREATED	TBOWNER
SYSCHECKS2	RELCREATED	TBOWNER

Table 9. Summary of new and revised catalog table columns (continued)

Catalog table name	New column	Revised column
SYSCOLAUTH	GRANTORTYPE	GRANTEETYPE CREATOR
SYSCOLDIST	QUANTILENO LOWVALUE HIGHVALUE	TYPE CARDF FREQUENCYF TBOWNER
SYSCOLDIST_HIST	QUANTILENO LOWVALUE HIGHVALUE	TYPE CARDF FREQUENCYF TBOWNER
SYSCOLDISTSTATS	QUANTILENO LOWVALUE HIGHVALUE	TYPE CARDF FREQUENCYF TBOWNER
SYSCOLSTATS		TBOWNER HIGHKEY HIGH2KEY LOWKEY LOW2KEY
SYSCOLUMNS	RELCREATED	COLTYPE LENGTH HIGH2KEY LOW2KEY LENGTH2 DEFAULT DEFAULTVALUE TBOWNER
SYSCOLUMNS_HIST		TBCREATOR COLTYPE LENGTH LENGTH2 TBOWNER
SYSCONSTDEP	DTBOWNER OWNERTYPE	DTBCREATOR
SYSCOPY	LOGGED TTYPE INSTANCE RELCREATED	STYPE ICTYPE RELCREATED TIMESTAMP PIT_RBA
SYSDATABASE	IMPLICIT CREATOR RELCREATED	
SYSDATATYPES	OWNERTYPE RELCREATED	CREATEDBY
SYSDBAUTH	GRANTEETYPE GRANTORTYPE	
SYSDBRM	PLCREATOR RELCREATED	

Table 9. Summary of new and revised catalog table columns (continued)

Catalog table name	New column	Revised column
SYSINDEXES	KEYTARGET_COUNT UNIQUE_COUNT IX_EXTENSION_TYPE COMPRESS OWNER OWNERTYPE DATAREPEATFACTORF ENVID	PGSIZE TBCREATOR UNIQUERULE CLUSTERRATIOF CREATOR
SYSINDEXES_HIST	DATAREPEATFACTORF	TBCREATOR CREATOR
SYSINDEXPART		IXCREATOR LEAFDIST FAROFFPOSF NEAROFFPOSF
SYSINDEXPART_HIST		IXCREATOR
SYSINDEXSTATS	DATAREPEATFACTORF	OWNER
SYSINDEXSTATS_HIST	DATAREPEATFACTORF	OWNER
SYSJAROBJECTS	OWNERTYPE	
SYSKEYS		COLSEQ ORDERING
SYSPACKAGE	OWNERTYPE ROUNDING	TYPE REOPTVAR
SYSPACKAUTH	GRANTORTYPE	GRANTEETYPE
SYSPACKDEP	DOWNERTYPE	BTYPE DTYPE BQUALIFIER
SYSPACKSTMT		SEQNO
SYSPARMS	VERSION OWNERTYPE	ROWTYPE PARMNAME CCSID
SYSPLAN	CREATOR ROUNDING	REOPTVAR
SYSPLANAUTH	GRANTEETYPE GRANTORTYPE	
SYSPLANDEP		BTYPE BCREATOR
SYSRELS	RELCREATED	CREATOR REFTBCREATOR IXOWNER
SYSRESAUTH	GRANTEETYPE GRANTORTYPE	
SYSROUTINEAUTH	GRANTORTYPE	GRANTEETYPE

Table 9. Summary of new and revised catalog table columns (continued)

Catalog table name	New column	Revised column
SYSROUTINES	VERSION	ORIGIN
	CONTOKEN	FENCED
	ACTIVE	CREATEDBY
	DEBUG_MODE	WLMENVIRONMENT
	TEXT_ENVID	EXTERNAL_SECURITY
	TEXT_ROWID	
	TEXT	
	OWNERTYPE	
	PARAMETER_VARCHARFORM	
	RELCREATED	
SYSSCHEMAAUTH	GRANTEETYPE	
	GRANTORTYPE	
SYSSEQUENCEAUTH	GRANTORTYPE	GRANTEETYPE
SYSSEQUENCES	OWNERTYPE	SEQTYPE
	RELCREATED	CREATEDBY
SYSSEQUENCESDEP	DOWNER	DTYPE
	DOWNERTYPE	
SYSSTMT	PLCREATORTYPE	
SYSSTOGROUP	CREATORTYPE	
	DATACLAS	
	MGMTCLAS	
	STORCLAS	
	RELCREATED	
SYSSYNONYMS	CREATORTYPE	TBCREATOR
	RELCREATED	
SYSTABAUTH	GRANTORTYPE	GRANTEETYPE
		SCREATOR
		TCREATOR
SYSTABCONST	CREATORTYPE	TBCREATOR
	RELCREATED	IXOWNER
SYSTABLEPART	FORMAT	IXCREATOR
	REORG_LR_TS	
	RELCREATED	
SYSTABLES	APPEND	DBID
	OWNER	OBID
	OWNERTYPE	TYPE
	RELCREATED	CREATOR
		TBCREATOR
SYSTABLES_HIST		CREATOR
SYSTABLESPACE	MAXPARTITIONS	LOG
	CREATORTYPE	TYPE
	INSTANCE	
	CLONE	
	RELCREATED	
SYSTABSTATS		OWNER
SYSTABSTATS_HIST		OWNER

Table 9. Summary of new and revised catalog table columns (continued)

Catalog table name	New column	Revised column
SYSTRIGGERS	OWNERTYPE	OWNER
	ENVID	CREATEDBY
	RELCREATED	TRIGTIME
		TBOWNER TBNAME
SYSUSERAUTH	GRANTEETYPE	
	GRANTORTYPE	
	DEBUGSESSIONAUTH	
SYSVIEWDEP	DOWNER	DCREATOR
	OWNERTYPE	
SYSVIEWS	OWNER	CREATOR
	OWNERTYPE	
	RELCREATED	
SYSVOLUMES	RELCREATED	
USERNAMES		TYPE

## New and changed indexes in Version 9

This release of DB2 for z/OS includes new indexes and changes to existing indexes.

The following table shows the new and changed indexes.

Table 10. New and changed indexes

Table space DSNDB06. ...	Catalog table SYSIBM. ...	INDEX SYSIBM. ...	INDEX FIELDS
SYSCTX	SYSCONTEXT	DSNCTX01	NAME
		DSNCTX02	SYSTEMAUTHID
		DSNCTX03	CONTEXID
		DSNCTX04	DEFAULTROLE
	SYSCONTEXTAUTHIDS	DSNCDX01	CONTEXTID.AUTHID
		DSNCDX02	ROLE
	SYSCTXTRUSTATTRS	DSNCAX01	CONTEXTID.NAME.VALUE
SYSDBASE	SYSRELS	DSNDLX03	IXOWNER.IXNAME
	SYSTABAUTH	DSNATX01	GRANTOR.GRANTORTYPE
	SYSTABLEPART	DSNDPX04	IXCREATOR.IXNAME
SYSDBAUT	SYSDBAUTH	DSNADH01	GRANTEE.NAME.GRANTEETYPE
		DSNADX01	GRANTOR.NAME.GRANTORTYPE
SYSGPAUT	SYSRESAUTH	DSNAGH01	GRANTEE.QUALIFIER.NAME.OBTYPE. GRANTEETYPE
		DSNAGX01	GRANTOR.QUALIFIER.NAME.OBTYPE. GRANTORTYPE

Table 10. New and changed indexes (continued)

Table space DSNDB06. ...	Catalog table SYSIBM. ...	INDEX SYSIBM. ...	INDEX FIELDS
SYSHIST	SYSKEYTARGETS_HIST	DSNHKX01	IXSCHEMA.IXNAME.KEYSEQ.STATSTIME
	SYSKEYTGTDIST_HIST	DSNTDX02	IXSCHEMA.IXNAME.KEYSEQ.STATSTIME
SYSJAVA	SYSJAVAPATHS	DSNJIPX01	JARSCHEMA.JAR_ID.ORDINAL
		DSNJIPX02	PE_JARSCHEMA.PE_JAR_ID
SYSOBJ	SYSDEPENDENCIES	DSNONX01	BSHEMA.BNAME.BCOLNAME.BTYPE. DSHEMA.DNAME.DCOLNAME.DTYPE
		DSNONX02	DSHEMA.DNAME.DCOLNAME.DTYPE. BSHEMA.BNAME.BCOLNAME.BTYPE
	SYSENVIRONMENT	DSNOEX01	ENVID
	SYSPARMS	DSNOPX04	SCHEMA.SPECIFICNAME.ROUTINETYPE.VERSION
	SYSROUTINEEAUTH	DSNOAX01	GRANTOR.SCHEMA.SPECIFICNAME. ROUTINETYPE.GRANTEETYPE.EXECUTEAUTH. GRANTORTYPE
	SYSROUTINES	DSNOFX01	NAME.PARM_COUNT. ROUTINETYPE.PARM_SIGNATURE. SCHEMA.PARM1.PARM2.PARM3. PARM4.PARM5.PARM6.PARM7. PARM8.PARM9.PARM10.PARM11. PARM12.PARM13.PARM14.PARM15. PARM16.PARM17.PARM18.PARM19. PARM20.PARM21.PARM22.PARM23. PARM24.PARM25.PARM26.PARM27. PARM28.PARM29.PARM30. VERSION
			DSNOFX02
DSNOFX07			NAME.PARM_COUNT. ROUTINETYPE. SCHEMA. PARM_SIGNATURE. PARM1.PARM2.PARM3. PARM4.PARM5.PARM6.PARM7. PARM8.PARM9.PARM10.PARM11. PARM12.PARM13.PARM14.PARM15. PARM16.PARM17.PARM18.PARM19. PARM20.PARM21.PARM22.PARM23. PARM24.PARM25.PARM26.PARM27. PARM28.PARM29.PARM30. VERSION
SYSSCHEMAAUTH	DSNSKX01	GRANTEE.SCHEMANAME.GRANTEETYPE	
		DSNSKX02	GRANTOR.GRANTORTYPE

Table 10. New and changed indexes (continued)

Table space DSNDB06. ...	Catalog table SYSIBM. ...	INDEX SYSIBM. ...	INDEX FIELDS
SYSPKAGE	SYSPACKAUTH	DSNKAX01	GRANTOR.LOCATION.COLLID.NAME. GRANTORTYPE
		DSNKAX02	GRANTEE.LOCATION.COLLID.NAME.BINDAUTH. COPYAUTH.EXECUTEAUTH.GRANTEETYPE
	SYSPACKSTMT	DSNKSX01	LOCATION.COLLID.NAME.CONTOKEN.STMTNOI. SECTNOI.SEQNO
SYSPLAN	SYSPLANAUTH	DSNAPH01	GRANTEE.NAME.EXECUTEAUTH.GRANTEETYPE
		DSNAPX01	GRANTOR.GRANTORTYPE
SYSROLES	SYSOBJROLESEP	DSNRDX01	DSHEMA.DNAME.DTYPE.ROLENAME
		DSNRDX02	ROLENAME
	SYSROLES	DSNRLX01	NAMES
SYSRTSTS	SYSTABLESPACESTATS	DSNRTX01	DBID.PSID.PARTITION.INSTANCE
	SYSINDEXSPACESTAT	DSNRTX02	DBID.ISOBID.PARTITION.INSTANCE
SYSSEQ2	SYSSEQUENCEAUTH	DSNWCX02	GRANTOR.SCHEMA.NAME.GRANTORTYPE
		DSNWCX03	GRANTEE.SCHEMA.NAME.GRANTEETYPE
SYSSTATS	SYSKEYTARGETSTATS	DSNTKX01	IXSCHEMA.IXNAME.KEYSEQ.PARTITION
	SYSKEYTGDIST	DSNTDX01	IXSCHEMA.IXNAME.KEYSEQ
	SYSKEYTGTDISTSTATS	DSNTSX01	IXSCHEMA.IXNAME.KEYSEQ.PARTITION
SYSTARG	SYSKEYTARGETS	DSNRKX01	IXSCHEMA.IXNAME.KEYSEQ
		DSNRKX02	DATATYPEID.KEYSPEC_INTERNAL
SYSUSER	SYSUSERAUTH	DSNAUH01	GRANTEE.GRANTEDTS.GRANTEETYPE
		DSNAUX01	GRANTOR.GRANTORTYPE
SYSXML	SYSXMLRELS	DSNXRX01	TBOWNER.TBNAME
		DSNXRX02	XMLTBOWNER.XMLTBNAME
	SYSXMLSTRINGS	DSNXSX01	STRINGID
		DSNXSX02	STRING

Table 10. New and changed indexes (continued)

Table space DSNDB06. ...	Catalog table SYSIBM. ...	INDEX SYSIBM. ...	INDEX FIELDS
SYSXSR	XSROBJECTS	XSROBJ01	XSROBJECTID
		XSROBJ02	XSROBJECTSCHEMA.XSROBJECTNAME
		XSROBJ03	TARGETNAMESPACE.SCHEMALOCATION
		XSROBJ04	SCHEMALOCATION
	XSROBJECT- COMPONENTS	XSRCOMP01	XSRCOMPONENTID
		XSRCOMP02	TARGETNAMESPACE.SCHEMALOCATION
	XSROBJECT- HIERARCHIES	XSRHIER01	XSROBJECTID.TARGETNAMESPACE. SCHEMALOCATION
		XSRHIER02	XSROBJECTID.TARGETNAMESPACE
SYSXSRA1	XSROBJECTGRAMMAR	XSRXOG01	GRAMMAR
SYSXSRA2	XSROBJECTPROPERTY	XSRXOP01	PROPERTIES
SYSXSRA3	XSRCOMPONENT	XSRXCC01	COMPONENT
SYSXSRA4	XSRPROPERTY	XSRXCP01	PROPERTIES

## Performance monitoring and tuning changes in Version 9

This release of DB2 for z/OS includes changes that effect performance and changes to monitoring tools such as EXPLAIN, PLAN\_TABLE, and DSN\_STATEMNT\_TABLE.

### Performance changes in Version 9

In this release of DB2 for z/OS, changes to performance include improvements to parallel processing, new page-range screening methods, a new DB2 system monitor, and changes to how DB2 selects access paths.

#### Parallelism changes

In previous releases of DB2, if the access path for a query used a data-partitioned secondary index that provided data ordering, DB2 could not use parallelism when it executed the query. In this release of DB2 for z/OS, parallelism is considered when DB2 executes this type of query.

For example, suppose that table T1 has two columns, C1 and C2, and that a data-partitioned secondary index is defined on C1. In previous releases of DB2, no parallelism was considered for the following query:

```
SELECT * FROM TABLE1 ORDER BY C1;
```

Now, DB2 considers parallelism for this query.

## Page-range screening for tables with data-partitioned secondary indexes

You can now increase performance by writing queries so that page-range screening will reduce the number of partitions that DB2 accesses. You can take advantage of page-range screening by:

- Joining the tables on the partitioning column
- Creating a non-matching predicate on the partitioning keys

## DB2 system monitor

The DB2 system monitor looks for CPU stalls that result in latch contention. When it detects a CPU stall, DB2 system monitor attempts to clear the latch contention by temporarily boosting WLM priority. In addition, the system monitor issues message DSNV508I, to report when DBM1 storage below the 2GB bar reaches critical storage thresholds, and a series of DSNV512I messages to identify the agents that consume the most storage.

Set the xxxxMSTR address space to SYSSTC dispatching priority in WLM in order for the monitor to work effectively.

## RUNSTATS and randomized key columns

The values of HIGH2KEY and LOW2KEY columns in the SYSCOLSTATS and SYSCOLUMNS catalog tables are updated with decoded values if the column is a randomized key column.

## Improved access path selection for duplicated and reverse-clustered indexes

For indexes that contain either many duplicate key values or key values that are highly clustered in reverse order, cost estimation that is based purely on CLUSTERRATIOF can lead to repetitive index scans. In the worst case, an entire page could be scanned one time for each row in the page. DB2 access path selection can avoid this performance problem by using a new cost estimation formula based on the DATAREPEATFACTORF statistic to choose indexes. Whether DB2 will use this statistic depends on the value of the STATCLUS subsystem parameter. To take advantage of the new formula, set the STATCLUS parameter to ENHANCED. Otherwise, set the value to STANDARD.

## EXPLAIN table changes in Version 9

This information includes the complete definitions for key DB2 EXPLAIN tables, and provides a description of the columns in EXPLAIN tables that are new and changed in this release of DB2 for z/OS.

Before you can use EXPLAIN, you must create a table called PLAN\_TABLE to hold the results of EXPLAIN. If you have an existing PLAN\_TABLE from a subsystem that ran on a previous version of DB2 you can alter it to add the new columns. "Format of PLAN\_TABLE in Version 9" on page 70 shows the format of the PLAN\_TABLE.

You can also create EXPLAIN tables called DSN\_STATEMNT\_TABLE, DSN\_FUNCTION\_TABLE, and DSN\_STATEMENT\_CACHE\_TABLE to capture different kinds of information about SQL statements. Related topics provide additional information about the changes to these EXPLAIN tables.

## Format of PLAN\_TABLE in Version 9

In this release of DB2 for z/OS, PLAN\_TABLE has one new column, PARENT\_PLANNO.

The PARENT\_PLANNO column of PLAN\_TABLE corresponds to the plan number in the parent query block where a correlated subquery is invoked. For non-correlated subqueries it corresponds to the plan number in the parent query block that represents the work file for the subquery. The following figure shows the format of PLAN\_TABLE.

Figure 1. Format of the PLAN\_TABLE

```
CREATE TABLE userid.PLAN_TABLE
  (QUERYNO          INTEGER          NOT NULL,
   QBLOCKNO        SMALLINT         NOT NULL,
   APPLNAME        VARCHAR(24)      NOT NULL,
   PROGNAME        VARCHAR(128)     NOT NULL,
   PLANNO          SMALLINT         NOT NULL,
   METHOD          SMALLINT         NOT NULL,
   CREATOR         VARCHAR(128)     NOT NULL,
   TNAME          VARCHAR(128)     NOT NULL,
   TABNO          SMALLINT         NOT NULL,
   ACESSTYPE      CHAR(2)          NOT NULL,
   MATCHCOLS      SMALLINT         NOT NULL,
   ACCESSCREATOR  VARCHAR(128)     NOT NULL,
   ACCESSNAME     VARCHAR(128)     NOT NULL,
   INDEXONLY      CHAR(1)         NOT NULL,
   SORTN_UNIQ     CHAR(1)         NOT NULL,
   SORTN_JOIN     CHAR(1)         NOT NULL,
   SORTN_ORDERBY  CHAR(1)         NOT NULL,
   SORTN_GROUPBY  CHAR(1)         NOT NULL,
   SORTC_UNIQ     CHAR(1)         NOT NULL,
   SORTC_JOIN     CHAR(1)         NOT NULL,
   SORTC_ORDERBY  CHAR(1)         NOT NULL,
   SORTC_GROUPBY  CHAR(1)         NOT NULL,
   TSLOCKMODE     CHAR(3)         NOT NULL,
   TIMESTAMP      CHAR(16)        NOT NULL,
   REMARKS        VARCHAR(762)     NOT NULL,
   PREFETCH       CHAR(1)         NOT NULL WITH DEFAULT,
   COLUMN_FN_EVAL CHAR(1)         NOT NULL WITH DEFAULT,
   MIXOPSEQ       SMALLINT         NOT NULL WITH DEFAULT,
   VERSION        VARCHAR(122)     NOT NULL WITH DEFAULT,
   COLLID        VARCHAR(128)     NOT NULL WITH DEFAULT,
   ACCESS_DEGREE  SMALLINT         ,
   ACCESS_PGROUP_ID SMALLINT       ,
   JOIN_DEGREE    SMALLINT         ,
   JOIN_PGROUP_ID SMALLINT         ,
   SORTC_PGROUP_ID SMALLINT       ,
   SORTN_PGROUP_ID SMALLINT       ,
   PARALLELISM_MODE CHAR(1)       ,
   MERGE_JOIN_COLS SMALLINT       ,
   CORRELATION_NAME VARCHAR(128)  ,
   PAGE_RANGE     CHAR(1)         NOT NULL WITH DEFAULT,
   JOIN_TYPE      CHAR(1)         NOT NULL WITH DEFAULT,
   GROUP_MEMBER   VARCHAR(24)     NOT NULL WITH DEFAULT,
   IBM_SERVICE_DATA VARCHAR(254)  FOR BIT DATA NOT NULL WITH DEFAULT,
   WHEN_OPTIMIZE  CHAR(1)         NOT NULL WITH DEFAULT,
   QBLOCK_TYPE    CHAR(6)         NOT NULL WITH DEFAULT,
   BIND_TIME      TIMESTAMP       NOT NULL WITH DEFAULT,
   OPTHINT        VARCHAR(128)    NOT NULL WITH DEFAULT,
   HINT_USED      VARCHAR(128)    NOT NULL WITH DEFAULT,
   PRIMARY_ACESSTYPE CHAR(1)     NOT NULL WITH DEFAULT,
   PARENT_QBLOCKNO SMALLINT       NOT NULL WITH DEFAULT,
   TABLE_TYPE    CHAR(1)         ,
```

```

TABLE_ENCODE          CHAR(1)          NOT NULL WITH DEFAULT,
TABLE_SCCSID          SMALLINT         NOT NULL WITH DEFAULT,
TABLE_MCCSID          SMALLINT         NOT NULL WITH DEFAULT,
TABLE_DCCSID          SMALLINT         NOT NULL WITH DEFAULT,
ROUTINE_ID            INTEGER           NOT NULL WITH DEFAULT,
CTEREF                SMALLINT         NOT NULL WITH DEFAULT,
STMTOKEN              VARCHAR(240)...,
PARENT_PLANNO         SMALLINT         NOT NULL WITH DEFAULT.)
IN database-name.table-space-name
CCSID UNICODE;

```

Your PLAN\_TABLE can use many other formats with fewer columns. However, the 59-column format provides the most information.

To alter an existing plan table with fewer than 59 columns to the 59-column format:

1. Determine whether your PLAN\_TABLE has the following columns:
  - PROGNAME
  - CREATOR
  - TNAME
  - ACESSTYPE
  - ACCESSNAME
  - REMARKS
  - COLLID
  - CORRELATION\_NAME
  - IBM\_SERVICE\_DATA
  - OPTHINT
  - HINT\_USED
2. For any columns that exist, use the values in the preceding figure to change the data types of these columns to the appropriate data types.
3. For any columns that are not in PLAN\_TABLE, add these columns to the table, using the column definitions in the preceding figure.

#### Descriptions of new and changed columns in PLAN\_TABLE in Version 9:

The PLAN\_TABLE contains new and changed columns.

The following table shows the contents of each of the new or changed columns.

Table 11. Descriptions of new and changed columns in PLAN\_TABLE

Column Name	Data Type	Description	New or changed
QUERYNO	INTEGER NOT NULL	A number intended to identify the statement being explained. For a row produced by an EXPLAIN statement, specify the number in the QUERYNO clause. For a row produced by non-EXPLAIN statements, specify the number using the QUERYNO clause, which is an optional part of the SELECT, INSERT, UPDATE, MERGE and DELETE statement syntax. Otherwise, DB2 assigns a number based on the line number of the SQL statement in the source program.  When the values of QUERYNO are based on the statement number in the source program, values greater than 32767 are reported as 0. However, in a very long program, the value is not guaranteed to be unique. If QUERYNO is not unique, the value of TIMESTAMP is unique.	Changed

Table 11. Descriptions of new and changed columns in PLAN\_TABLE (continued)

Column Name	Data Type	Description	New or changed
APPLNAME	VARCHAR(24) NOT NULL	The name of the application plan for the row. Applies only to embedded EXPLAIN statements executed from a plan or to statements explained when binding a plan. Blank if not applicable.	Changed
ACCESSTYPE	CHAR(2) NOT NULL	The method of accessing the new table:  <b>DI</b> By an intersection of multiple DOCID lists to return the final DOCID list  <b>DU</b> By a union of multiple DOCID lists to return the final DOCID list  <b>DX</b> By an XML index scan on the index named in ACCESSNAME to return a DOCID list  <b>I</b> By an index (identified in ACCESSCREATOR and ACCESSNAME)  <b>I1</b> By a one-fetch index scan  <b>M</b> By a multiple index scan (followed by MX, MI, or MU)  <b>MI</b> By an intersection of multiple indexes  <b>MU</b> By a union of multiple indexes  <b>MX</b> By an index scan on the index named in ACCESSNAME. When the access method MX follows the access method DX, DI, or DU, the table is accessed by the DOCID index using the DOCID list returned by DX, DI, or DU.  <b>N</b> By an index scan when the matching predicate contains the IN keyword	Changed
MATCHCOLS	SMALLINT NOT NULL	For ACCESSTYPE I, I1, N, MX, or DX, the number of index keys used in an index scan; otherwise, 0.	Changed
ACCESSCREATOR	VARCHAR(128) NOT NULL	For ACCESSTYPE I, I1, N, MX, or DX, the creator of the index; otherwise, blank.	Changed
ACCESSNAME	VARCHAR(128) NOT NULL	For ACCESSTYPE I, I1, N, MX, or DX, the name of the index; otherwise, blank.	Changed

Table 11. Descriptions of new and changed columns in PLAN\_TABLE (continued)

Column Name	Data Type	Description	New or changed
TSLOCKMODE	CHAR(3) NOT NULL	<p>An indication of the mode of lock to be acquired on either the new table, or its table space or table space partitions. If the isolation can be dedtined at bind time, the values are:</p> <p><b>IS</b> Intent share lock</p> <p><b>IX</b> Intent exclusive lock</p> <p><b>S</b> Share lock</p> <p><b>U</b> Update lock</p> <p><b>X</b> Exclusive lock</p> <p><b>SIX</b> Share with intent exclusive lock</p> <p><b>N</b> UR isolation; no lock</p> <p>If the isolation cannot be determined at bind time, then the lock mode determined by the isolation at run time is shown by the following values.</p> <p><b>NS</b> For UR isolation, no lock; for CS, RS, or RR, an S lock.</p> <p><b>NIS</b> For UR isolation, no lock; for CS, RS, or RR, an IS lock.</p> <p><b>NSS</b> For UR isolation, no lock; for CS or RS, an IS lock; for RR, an S lock.</p> <p><b>SS</b> For UR, CS, or RS isolation, an IS lock; for RR, an S lock.</p> <p>The data in this column is right justified. For example, IX appears as a blank followed by I followed by X. If the column contains a blank, then no lock is acquired.</p> <p>If the access method in the ACCESTYPE column is DX, DI, or DU, no latches are acquired on the XML value index page and no lock is acquired on the new base table data page or row, nor on the XML table and the corresponding table spaces. TSLOCKMODE is a blank in this case.</p>	Changed
MIXOPSEQ	SMALLINT NOT NULL WITH DEFAULT	<p>The sequence number of a step in a multiple index operation.</p> <p><b>1, 2, ... n</b> For the steps of the multiple index procedure (ACCESTYPE is MX, MI, MU, DX, DI, or DU.)</p> <p><b>0</b> For any other rows (ACCESTYPE is I, I1, M, N, R, or blank.)</p>	Changed
VERSION	VARCHAR(122) NOT NULL	The version identifier for the package. Applies only to an embedded EXPLAIN statement executed from a package or to a statement that is explained when binding a package. A blank indicates that the column is not applicable.	Changed

Table 11. Descriptions of new and changed columns in PLAN\_TABLE (continued)

Column Name	Data Type	Description	New or changed
JOIN_TYPE	CHAR(1)	The type of join: <b>F</b> FULL OUTER JOIN <b>L</b> LEFT OUTER JOIN <b>P</b> Pair-wise join <b>S</b> Star join <b>blank</b> INNER JOIN or no join	Changed
QBLOCK_TYPE	CHAR(6) NOT NULL WITH DEFAULT	For each query block, an indication of the type of SQL operation performed. For the outermost query, this column identifies the statement type. Possible values: <b>SELECT</b> SELECT <b>INSERT</b> INSERT <b>UPDATE</b> UPDATE <b>MERGE</b> MERGE <b>DELETE</b> DELETE <b>SELUPD</b> SELECT with FOR UPDATE OF <b>DELCUR</b> DELETE WHERE CURRENT OF CURSOR <b>TRUNCA</b> TRUNCATE <b>UPDCUR</b> UPDATE WHERE CURRENT OF CURSOR <b>CORSUB</b> Correlated subselect or fullselect <b>NCOSUB</b> Non-correlated subselect or fullselect <b>TABLEX</b> Table expression	Changed

Table 11. Descriptions of new and changed columns in PLAN\_TABLE (continued)

Column Name	Data Type	Description	New or changed
PRIMARY_ACCESTYPE	CHAR(1) NOT NULL WITH DEFAULT	<p>Indicates whether direct row access is attempted first:</p> <p><b>D</b> DB2 tries to use direct row access. If DB2 cannot use direct row access at run time, it uses the access path described in the ACCESTYPE column of PLAN_TABLE.</p> <p><b>T</b> The base table or result file will be materialized into a workfile, and the workfile will be accessed via sparse index access. If a base table is involved, then ACCESTYPE indicates how the base table is accessed.</p> <p><b>blank</b> DB2 does not try to use direct row access using a rowid column, or sparse index access for a work file. The value of the ACCESTYPE column of PLAN_TABLE provides information on the method of accessing the table.</p>	Changed
TABLE_TYPE	CHAR(1)	<p>The type of new table:</p> <p><b>B</b> Buffers for SELECT from INSERT, SELECT from UPDATE, SELECT from MERGE, or SELECT from DELETE statement.</p> <p><b>C</b> Common table expression</p> <p><b>F</b> Table function</p> <p><b>M</b> Materialized query table</p> <p><b>Q</b> Temporary intermediate result table (not materialized). For the name of a view or nested table expression, a value of Q indicates that the materialization was virtual and not actual. Materialization can be virtual when the view or nested table expression definition contains a UNION ALL that is not distributed.</p> <p><b>R</b> Recursive common table expression</p> <p><b>T</b> Table</p> <p><b>W</b> Work file</p>	Changed
PARENT_PLANNO	SMALLINT NOT NULL WITH DEFAULT	Corresponds to the plan number in the parent query block where a correlated subquery is invoked. For non-correlated subqueries, it corresponds to the plan number in the parent query block that represents the workfile for the subquery.	New

Your PLAN\_TABLE can use many other formats with fewer columns. However, you should use the 59-column format, because it gives you the most information.

To alter an existing plan table with fewer than 59 columns to the 59-column format:

- Determine whether your PLAN\_TABLE has the following columns:
  - PROGNAME
  - CREATOR
  - TNAME
  - ACCESTYPE
  - ACCESSNAME

- REMARKS
- COLLID
- CORRELATION\_NAME
- IBM\_SERVICE\_DATA
- OPTHINT
- HINT\_USED

2. For any columns that exist, use the values in Figure 1 on page 70 to change the data types of these columns to the appropriate Version 8 data types.
3. For any columns that are not in PLAN\_TABLE, add these columns to the table, using the column definitions in Figure 1 on page 70.

### Format of DSN\_STATEMNT\_TABLE in Version 9

In this release of DB2 for z/OS, DSN\_STATEMNT\_TABLE has one new column, TOTAL\_COST.

The statement table contains EXPLAIN information about the estimated cost of a statement.

```
CREATE TABLE DSN_STATEMNT_TABLE (
  QUERYNO          INTEGER          NOT NULL WITH DEFAULT,
  APPLNAME         VARCHAR(24)      NOT NULL WITH DEFAULT,
  PROGNAME        VARCHAR(128)     NOT NULL WITH DEFAULT,
  COLLID          VARCHAR(128)     NOT NULL WITH DEFAULT,
  GROUP_MEMBER    VARCHAR(24)      NOT NULL WITH DEFAULT,
  EXPLAIN_TIME    TIMESTAMP        NOT NULL WITH DEFAULT,
  STMT_TYPE       CHAR(6)          NOT NULL WITH DEFAULT,
  COST_CATEGORY   CHAR(1)         NOT NULL WITH DEFAULT,
  PROCMS         INTEGER          NOT NULL WITH DEFAULT,
  PROCSU         INTEGER          NOT NULL WITH DEFAULT,
  REASON         VARCHAR(254)     NOT NULL WITH DEFAULT,
  STMT_ENCODE     CHAR(1)         NOT NULL WITH DEFAULT,
  TOTAL_COST      FLOAT           NOT NULL WITH DEFAULT
);
IN database-name.table-space-name
CCSID UNICODE;
```

Figure 2. The CREATE TABLE statement for DSN\_STATEMNT\_TABLE.

### Descriptions of new and changed columns in DSN\_STATEMNT\_TABLE in Version 9:

The statement table, DSN\_STATEMNT\_TABLE, contains information about the estimated cost of specified SQL statements.

The table below describes new and changed columns in DSN\_STATEMNT\_TABLE.

Table 12. Descriptions of new and changed columns in DSN\_STATEMNT\_TABLE

Column Name	Data Type	Description	New or changed
STMT_TYPE	CHAR(6)	The type of statement being explained. Possible values are:  <b>SELECT</b> SELECT  <b>INSERT</b> INSERT  <b>UPDATE</b> UPDATE  <b>MERGE</b> MERGE  <b>DELETE</b> DELETE  <b>SELUPD</b> SELECT with FOR UPDATE OF  <b>DELCUR</b> DELETE WHERE CURRENT OF CURSOR  <b>UPDCUR</b> UPDATE WHERE CURRENT OF CURSOR	Changed

Table 12. Descriptions of new and changed columns in DSN\_STATEMNT\_TABLE (continued)

Column Name	Data Type	Description	New or changed
REASON	VARCHAR(254)	<p>A string that indicates the reasons for putting an estimate into cost category B.</p> <p><b>HAVING CLAUSE</b> A subselect in the SQL statement contains a HAVING clause.</p> <p><b>HOST VARIABLES</b> The statement uses host variables, parameter markers, or special registers.</p> <p><b>REFERENTIAL CONSTRAINTS</b> Referential constraints of the type CASCADE or SET NULL exist on the target table of a DELETE statement.</p> <p><b>TABLE CARDINALITY</b> The cardinality statistics are missing for one or more of the tables that are used in the statement.</p> <p><b>TRIGGERS</b> Triggers are defined on the target table of an insert, update, or delete operation.</p> <p><b>UDF</b> The statement uses user-defined functions.</p> <p><b>MATERIALIZATION</b> Statistics are missing because the statement uses materialized views or nested table expresses.</p>	Changed
TOTAL_COST	FLOAT	The overall estimated cost of the statement. This cost should be used only for reference purposes.	New

## Format of DSN\_FUNCTION\_TABLE in Version 9

In this release of DB2 for z/OS, DSN\_FUNCTION\_TABLE has two columns with changed data types.

```

CREATE TABLE DSN_FUNCTION_TABLE
  (QUERYNO          INTEGER          NOT NULL WITH DEFAULT,
   QBLOCKNO        INTEGER          NOT NULL WITH DEFAULT,
   APPLNAME        VARCHAR(24)     NOT NULL WITH DEFAULT,
   PROGNAME        VARCHAR(128)    NOT NULL WITH DEFAULT,
   COLLID          VARCHAR(128)    NOT NULL WITH DEFAULT,
   GROUP_MEMBER    VARCHAR(24)     NOT NULL WITH DEFAULT,
   EXPLAIN_TIME    TIMESTAMP       NOT NULL WITH DEFAULT,
   SCHEMA_NAME     VARCHAR(128)    NOT NULL WITH DEFAULT,
   FUNCTION_NAME   VARCHAR(128)    NOT NULL WITH DEFAULT,
   SPEC_FUNC_NAME  VARCHAR(128)    NOT NULL WITH DEFAULT,
   FUNCTION_TYPE   CHAR(2)         NOT NULL WITH DEFAULT,
   VIEW_CREATOR    VARCHAR(128)    NOT NULL WITH DEFAULT,
   VIEW_NAME       VARCHAR(128)    NOT NULL WITH DEFAULT,
   PATH            VARCHAR(2048)    NOT NULL WITH DEFAULT,
   FUNCTION_TEXT   VARCHAR(1500)   NOT NULL WITH DEFAULT)
  IN database-name.table-space-name CCSID UNICODE;

```

Figure 3. The CREATE TABLE statement for the DSN\_STATEMNT\_TABLE.

### Descriptions of new and changed columns in DSN\_FUNCTION\_TABLE in Version 9:

The function table, DSN\_FUNCTION\_TABLE, contains information about user-defined functions that are referenced in SQL statements.

The following table describes the new and changed columns in DSN\_FUNCTION\_TABLE.

Table 13. Descriptions of new and changed columns in DSN\_FUNCTION\_TABLE

Column name	Data Type	Description	New or changed
APPLNAME	VARCHAR(24)	The name of the application plan for the row, or blank.	Changed
GROUP_MEMBER	VARCHAR(24)	The member name of the DB2 subsystem that executed EXPLAIN, or blank.	Changed

### New statement cache table in Version 9

The statement cache table, DSN\_STATEMENT\_CACHE\_TABLE, contains information about the SQL statements that are in the statement cache.

This statement cache information is captured when you issue an EXPLAIN STATEMENT CACHE ALL statement.

## Related tasks

 Finding information about statements in the statement cache (Application Programming and SQL Guide)

---

## New and changed IFCIDs in Version 9

This release of DB2 for z/OS contains a number of trace enhancements.

### PSPI


Trace enhancements include:

- More ways to filter trace records  
DB2 adds these new filter types in the constraint block of a START TRACE, STOP TRACE, or DISPLAY TRACE command:
  - PKGPROG: Package
  - PKGLOC: Location
  - PKGCOL: Collection ID
  - USERID: User ID
  - APPNAME: Application name
  - WKRSTN: Workstation name
  - CONNID: Connection ID
  - CORRID: Correlation ID
  - ROLE: Role

In addition, you can exclude trace records as well as include trace records by any filter type. For example, the following statement excludes trace records for plans named A, B, or C:

```
-START TRACE(ACCTG) XPLAN(A,B,C)
```

- The record length of the IFCID flat file (DSNWMSGs) has been increased from 87 to 92, to accommodate larger trace field names.
- As of DB2 UDB for z/OS V8, the IFCID flat file has moved from the *prefix.SDSNSAMP* data set to the *prefix.SDSNIVPD* data set.

This information briefly describes the new IFCIDs and the changes to the existing IFCIDs for each new function. For a detailed description of the fields in each IFCID record, refer to the mapping macros data set library DSN910.SDSNMACS. 

## New IFCIDs in Version 9

This release of DB2 for z/OS includes new IFCIDs, which are listed in the following table.

### PSPI

Table 14. New IFCIDs

IFCID	Trace	Class	Mapping macro	Description
<b>Trusted connections</b>				
0269	AUDIT	10	DSNDQW04	Records information about establishment and reuse of a trusted connection.
<b>Excessive temporary storage usage</b>				
0343	PERFORMANCE	3	DSNDQW04	Records information about an agent when the MAXTEMPS subsystem parameter value is exceeded.
	STATISTICS	4		

Table 14. New IFCIDs (continued)

IFCID	Trace	Class	Mapping macro	Description
<b>READS package detail</b>				
0346			DSNDQW04	Records package detail information.



## Changed IFCIDs in Version 9

This release of DB2 for z/OS introduces changes to a number of trace records.



*Changes to selected trace records:* The following table gives an overview of changes to specific IFCIDs. Changes to IFCID 0106, the system parameters record, are not included.

Table 15. Changed IFCIDs

IFCID	Description of changes
<b>Clone table support</b>	
0006, 0007, 0008, 0010, 0011, 0012, 0013, 0015, 0016, 0017, 0019, 0020, 0021, 0023, 0024, 0025, 0044, 0105, 0107, 0108, 0124, 0125, 0127, 0128, 0143, 0144, 0150, 0172, 0196, 0198, 0199, 0211, 0212, 0213, 0215, 0216, 0218, 0221, 0223, 0226, 0227, 0251, 0252, 0255, 0258, 0259, 0263, 0337, 0342	In the page set ID (PSID) field of each of these records, if the high-order bit is on, the instance number that is associated with the object is instance number 2. For PSID fields that are represented as FIXED(15) fields, if the instance number is 2, the PSID value in the trace record is a negative value.
<b>Decimal floating-point data type</b>	
0184, 0247, 0248	The decimal floating-point type is added to data type fields.
<b>FETCH statement with CONTINUE</b>	
0002, 0003, 0148	A counter is added to record the number of FETCH CURRENT CONTINUE statements that have occurred.
0059	A field is added to indicate whether the WITH CONTINUE or CURRENT CONTINUE clause is specified on a FETCH statement.
<b>Global query optimization</b>	
0022	Fields are added and changed to synchronize this trace record with the Version 9.1 PLAN_TABLE.
<b>Native SQL procedures</b>	
0003	Plan-level accounting information for SQL procedures includes native SQL procedure data as well as external SQL procedure data.
0148, 0239	The field that specifies the package type is updated to include a package for a native SQL procedure.
<b>Recover to a point in time with consistency</b>	
0023, 0024	The new LOGCSR and LOGUNDO phases or RECOVER are added.
<b>RENAME INDEX statement</b>	

| Table 15. Changed IFCIDs (continued)

IFCID	Description of changes
0062	RENAME INDEX is added to the start of execution record.
0140	RENAME INDEX is added to the authorization failure record.
REORG enhancements	
0023, 0024, 0025	<ul style="list-style-type: none"> <li>Trace records are written for new subtasks for unloading nonpartitioned indexes.</li> <li>Due to the elimination of the BUILD2 phase, IFCID 0024 records with the BUILD2 phase are no longer written.</li> </ul>
0140	RENAME INDEX is added to the authorization failure record.
Skip locked rows	
0018	A field is added to record the number of rows that were skipped because another transaction had a lock on those rows.
XML support	
0020, 0021, 0044, 0107, 0150, 0172, 0196, 0337.	Fields are added and modified for XML locks.
Miscellaneous changes	
0001, 0225	Fields are added to record stack storage statistics.
0034, 0035	<ul style="list-style-type: none"> <li>These trace records now include active log prefetch read wait time as well as read wait time. IFCID 0034 includes a new flag that indicates the type of wait time.</li> <li>These trace records are now included in accounting class 3 and 8, and monitor class 3 and 8.</li> </ul>
0104	IFCID 0104 records are now available through the IFI READS interface.
0142	IFCID 0142 records now trace ALTER operations, regardless of whether the AUDIT attribute of the table has changed.
0191	<ul style="list-style-type: none"> <li>A new section is added.</li> <li>When a DRDA error occurs that results in reason code 00D3444E or 00D3444F, and a trace for IFCID 0191 is active, IFCID 0274 trace records are also generated.</li> </ul>
0199	A field is added to indicate whether the data set on which statistics are reported is a shadow data set.
0201, 0202	Fields are added to record whether automatic buffer pool sizing is in effect.
0217, 0225	Fields are added to record the amount of virtual shared storage that is used by the <i>ssnmDBM1</i> address space.
0225	These trace records are changed from SMF type 102 to SMF type 100 and subtype 4.
0239	A field is added to record the number of times that a package is allocated.



---

## Information resources for DB2 for z/OS and related products

Many information resources are available to help you use DB2 for z/OS and many related products. A large amount of technical information about IBM products is now available online in information centers or on library Web sites.

**Disclaimer:** Any Web addresses that are included here are accurate at the time this information is being published. However, Web addresses sometimes change. If you visit a Web address that is listed here but that is no longer valid, you can try to find the current Web address for the product information that you are looking for at either of the following sites:

- <http://www.ibm.com/support/publications/us/library/index.shtml>, which lists the IBM information centers that are available for various IBM products
- <http://www.elink.ibm.com/public/applications/publications/cgibin/pbi.cgi>, which is the IBM Publications Center, where you can download online PDF books or order printed books for various IBM products

### DB2 for z/OS product information

The primary place to find and use information about DB2 for z/OS is the Information Management Software for z/OS Solutions Information Center (<http://publib.boulder.ibm.com/infocenter/imzic>), which also contains information about IMS, QMF, and many DB2 and IMS Tools products. This information center is also available as an installable information center that can run on a local system or on an intranet server. You can order the Information Management for z/OS Solutions Information Center DVD (SK5T-7377) for a low cost from the IBM Publications Center ([www.ibm.com/shop/publications/order](http://www.ibm.com/shop/publications/order)).

The majority of the DB2 for z/OS information in this information center is also available in the books that are identified in the following table. You can access these books at the DB2 for z/OS library Web site (<http://www.ibm.com/software/data/db2/zos/library.html>) or at the IBM Publications Center (<http://www.ibm.com/shop/publications/order>).

Table 16. DB2 Version 9.1 for z/OS book titles

Title	Publication number	Available in information center	Available in PDF	Available in BookManager <sup>®</sup> format	Available in printed book
<i>DB2 Version 9.1 for z/OS Administration Guide</i>	SC18-9840	X	X	X	X
<i>DB2 Version 9.1 for z/OS Application Programming &amp; SQL Guide</i>	SC18-9841	X	X	X	X
<i>DB2 Version 9.1 for z/OS Application Programming Guide and Reference for Java</i>	SC18-9842	X	X	X	X
<i>DB2 Version 9.1 for z/OS Codes</i>	GC18-9843	X	X	X	X
<i>DB2 Version 9.1 for z/OS Command Reference</i>	SC18-9844	X	X	X	X
<i>DB2 Version 9.1 for z/OS Data Sharing: Planning and Administration</i>	SC18-9845	X	X	X	X

Table 16. DB2 Version 9.1 for z/OS book titles (continued)

Title	Publication number	Available in information center	Available in PDF	Available in BookManager® format	Available in printed book
<i>DB2 Version 9.1 for z/OS Diagnosis Guide and Reference</i> <sup>1</sup>	LY37-3218		X	X	X
<i>DB2 Version 9.1 for z/OS Diagnostic Quick Reference</i>	LY37-3219				X
<i>DB2 Version 9.1 for z/OS Installation Guide</i>	GC18-9846	X	X	X	X
<i>DB2 Version 9.1 for z/OS Introduction to DB2</i>	SC18-9847	X	X	X	X
<i>DB2 Version 9.1 for z/OS Licensed Program Specifications</i>	GC18-9848		X		X
<i>DB2 Version 9.1 for z/OS Messages</i>	GC18-9849	X	X	X	X
<i>DB2 Version 9.1 for z/OS ODBC Guide and Reference</i>	SC18-9850	X	X	X	X
<i>DB2 Version 9.1 for z/OS Performance Monitoring and Tuning Guide</i>	SC18-9851	X	X	X	X
<i>DB2 Version 9.1 for z/OS Optimization Service Center</i>		X			
<i>DB2 Version 9.1 for z/OS Program Directory</i>	GI10-8737		X		X
<i>DB2 Version 9.1 for z/OS RACF Access Control Module Guide</i>	SC18-9852	X	X	X	
<i>DB2 Version 9.1 for z/OS Reference for Remote DRDA Requesters and Servers</i>	SC18-9853	X	X	X	
<i>DB2 Version 9.1 for z/OS Reference Summary</i>	SX26-3854		X		
<i>DB2 Version 9.1 for z/OS SQL Reference</i>	SC18-9854	X	X	X	X
<i>DB2 Version 9.1 for z/OS Utility Guide and Reference</i>	SC18-9855	X	X	X	X
<i>DB2 Version 9.1 for z/OS What's New?</i>	GC18-9856	X	X		X
<i>DB2 Version 9.1 for z/OS XML Extender Administration and Programming</i>	SC18-9857	X	X	X	X
<i>DB2 Version 9.1 for z/OS XML Guide</i>	SC18-9858	X	X	X	X
<i>IRLM Messages and Codes for IMS and DB2 for z/OS</i>	GC19-2666	X	X	X	

**Note:**

1. *DB2 Version 9.1 for z/OS Diagnosis Guide and Reference* is available in PDF and BookManager formats on the DB2 Version 9.1 for z/OS Licensed Collection kit, LK3T-7195. You can order this License Collection kit on the IBM Publications Center site (<http://www.elink.ibm.com/public/applications/publications/cgibin/pbi.cgi>). This book is also available in online format in DB2 data set DSN910.SDSNIVPD(DSNDR).

## Information resources for related products

In the following table, related product names are listed in alphabetic order, and the associated Web addresses of product information centers or library Web pages are indicated.

Table 17. Related product information resource locations

Related product	Information resources
C/C++ for z/OS	Library Web site: <a href="http://www.ibm.com/software/awdtools/czos/library/">http://www.ibm.com/software/awdtools/czos/library/</a> This product is now called z/OS XL C/C++.
CICS Transaction Server for z/OS	Information center: <a href="http://publib.boulder.ibm.com/infocenter/cicsts/v3r1/index.jsp">http://publib.boulder.ibm.com/infocenter/cicsts/v3r1/index.jsp</a>
COBOL	Information center: <a href="http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp">http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp</a> This product is now called Enterprise COBOL for z/OS.
DB2 <sup>®</sup> Connect <sup>™</sup>	Information center: <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp">http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp</a> This resource is for DB2 Connect 9.
DB2 Database for Linux, UNIX, and Windows	Information center: <a href="http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp">http://publib.boulder.ibm.com/infocenter/db2luw/v9/index.jsp</a> This resource is for DB2 9 for Linux, UNIX, and Windows.
DB2 Query Management Facility	Information center: <a href="http://publib.boulder.ibm.com/infocenter/imzic">http://publib.boulder.ibm.com/infocenter/imzic</a>
DB2 Server for VSE & VM	One of the following locations: <ul style="list-style-type: none"> <li>• For VSE: <a href="http://www.ibm.com/support/docview.wss?rs=66&amp;uid=swg27003758">http://www.ibm.com/support/docview.wss?rs=66&amp;uid=swg27003758</a></li> <li>• For VM: <a href="http://www.ibm.com/support/docview.wss?rs=66&amp;uid=swg27003759">http://www.ibm.com/support/docview.wss?rs=66&amp;uid=swg27003759</a></li> </ul>
DB2 Tools	One of the following locations: <ul style="list-style-type: none"> <li>• Information center: <a href="http://publib.boulder.ibm.com/infocenter/imzic">http://publib.boulder.ibm.com/infocenter/imzic</a></li> <li>• Library Web site: <a href="http://www.ibm.com/software/data/db2imstools/library.html">http://www.ibm.com/software/data/db2imstools/library.html</a></li> </ul> <p>These resources include information about the following products and others:</p> <ul style="list-style-type: none"> <li>• DB2 Administration Tool</li> <li>• DB2 Automation Tool</li> <li>• DB2 Log Analysis Tool</li> <li>• DB2 Object Restore Tool</li> <li>• DB2 Query Management Facility</li> <li>• DB2 SQL Performance Analyzer</li> </ul>
DB2 Universal Database for iSeries <sup>®</sup>	Information center: <a href="http://www.ibm.com/systems/i/infocenter/">http://www.ibm.com/systems/i/infocenter/</a>
Debug Tool for z/OS	Information center: <a href="http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp">http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp</a>
Enterprise COBOL for z/OS	Information center: <a href="http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp">http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp</a>
Enterprise PL/I for z/OS	Information center: <a href="http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp">http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp</a>
IMS	Information center: <a href="http://publib.boulder.ibm.com/infocenter/imzic">http://publib.boulder.ibm.com/infocenter/imzic</a>

Table 17. Related product information resource locations (continued)

Related product	Information resources
IMS Tools	<p>One of the following locations:</p> <ul style="list-style-type: none"> <li>• Information center: <a href="http://publib.boulder.ibm.com/infocenter/imzic">http://publib.boulder.ibm.com/infocenter/imzic</a></li> <li>• Library Web site: <a href="http://www.ibm.com/software/data/db2imstools/library.html">http://www.ibm.com/software/data/db2imstools/library.html</a></li> </ul> <p>These resources have information about the following products and others:</p> <ul style="list-style-type: none"> <li>• IMS Batch Terminal Simulator for z/OS</li> <li>• IMS Connect</li> <li>• IMS HALDB Conversion and Maintenance Aid</li> <li>• IMS High Performance Utility products</li> <li>• IMS DataPropagator™</li> <li>• IMS Online Reorganization Facility</li> <li>• IMS Performance Analyzer</li> </ul>
Integrated Data Management products	<p>Information center: <a href="http://publib.boulder.ibm.com/infocenter/idm/v2r2/index.jsp">http://publib.boulder.ibm.com/infocenter/idm/v2r2/index.jsp</a></p> <p>This information center has information about the following products and others:</p> <ul style="list-style-type: none"> <li>• IBM Data Studio</li> <li>• InfoSphere™ Data Architect</li> <li>• InfoSphere Warehouse</li> <li>• Optim™ Database Administrator</li> <li>• Optim Development Studio</li> <li>• Optim Query Tuner</li> </ul>
PL/I	<p>Information center: <a href="http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp">http://publib.boulder.ibm.com/infocenter/pdthelp/v1r1/index.jsp</a></p> <p>This product is now called Enterprise PL/I for z/OS.</p>
System z	<a href="http://publib.boulder.ibm.com/infocenter/eserver/v1r2/index.jsp">http://publib.boulder.ibm.com/infocenter/eserver/v1r2/index.jsp</a>
Tivoli OMEGAMONXE for DB2 Performance Expert on z/OS	<p>Information center: <a href="http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/index.jsp?topic=/com.ibm.ko2pe.doc/ko2welcome.htm">http://publib.boulder.ibm.com/infocenter/tivihelp/v15r1/index.jsp?topic=/com.ibm.ko2pe.doc/ko2welcome.htm</a></p> <p>In earlier releases, this product was called DB2 Performance Expert for z/OS.</p>
WebSphere Application Server	Information center: <a href="http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp">http://publib.boulder.ibm.com/infocenter/wasinfo/v6r0/index.jsp</a>
WebSphere Message Broker with Rules and Formatter Extension	<p>Information center: <a href="http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp">http://publib.boulder.ibm.com/infocenter/wmbhelp/v6r0m0/index.jsp</a></p> <p>The product is also known as WebSphere MQ Integrator Broker.</p>
WebSphere MQ	<p>Information center: <a href="http://publib.boulder.ibm.com/infocenter/wmqv6/v6r0/index.jsp">http://publib.boulder.ibm.com/infocenter/wmqv6/v6r0/index.jsp</a></p> <p>The resource includes information about MQSeries®.</p>
WebSphere Replication Server for z/OS	<p>Either of the following locations:</p> <ul style="list-style-type: none"> <li>• Information center: <a href="http://publib.boulder.ibm.com/infocenter/imzic">http://publib.boulder.ibm.com/infocenter/imzic</a></li> <li>• Library Web site: <a href="http://www.ibm.com/software/data/db2imstools/library.html">http://www.ibm.com/software/data/db2imstools/library.html</a></li> </ul> <p>This product is also known as DB2 DataPropagator.</p>
z/Architecture®	Library Center site: <a href="http://www.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www.ibm.com/servers/eserver/zseries/zos/bkserv/</a>

Table 17. Related product information resource locations (continued)

Related product	Information resources
z/OS	<p data-bbox="505 258 1365 281">Library Center site: <a href="http://www.ibm.com/servers/eserver/zseries/zos/bkserv/">http://www.ibm.com/servers/eserver/zseries/zos/bkserv/</a></p> <p data-bbox="505 310 1453 333">This resource includes information about the following z/OS elements and components:</p> <ul data-bbox="505 348 963 1444" style="list-style-type: none"> <li>• Character Data Representation Architecture</li> <li>• Device Support Facilities</li> <li>• DFSORT™</li> <li>• Fortran</li> <li>• High Level Assembler</li> <li>• NetView®</li> <li>• SMP/E for z/OS</li> <li>• SNA</li> <li>• TCP/IP</li> <li>• TotalStorage® Enterprise Storage Server®</li> <li>• VTAM®</li> <li>• z/OS C/C++</li> <li>• z/OS Communications Server</li> <li>• z/OS DCE</li> <li>• z/OS DFSMS™</li> <li>• z/OS DFSMS Access Method Services</li> <li>• z/OS DFSMSdss™</li> <li>• z/OS DFSMSHsm™</li> <li>• z/OS DFSMSdftp™</li> <li>• z/OS ICSF</li> <li>• z/OS ISPF</li> <li>• z/OS JES3</li> <li>• z/OS Language Environment®</li> <li>• z/OS Managed System Infrastructure</li> <li>• z/OS MVS</li> <li>• z/OS MVS JCL</li> <li>• z/OS Parallel Sysplex®</li> <li>• z/OS RMF™</li> <li>• z/OS Security Server</li> <li>• z/OS UNIX System Services</li> </ul>
z/OS XL C/C++	<p data-bbox="505 1465 1114 1488"><a href="http://www.ibm.com/software/awdtools/czos/library/">http://www.ibm.com/software/awdtools/czos/library/</a></p>

The following information resources from IBM are not necessarily specific to a single product:

- The DB2 for z/OS Information Roadmap; available at: <http://www.ibm.com/software/data/db2/zos/roadmap.html>
- DB2 Redbooks® and Redbooks about related products; available at: <http://www.ibm.com/redbooks>
- IBM Educational resources:
  - Information about IBM educational offerings is available on the Web at: <http://www.ibm.com/software/sw-training/>

- A collection of glossaries of IBM terms in multiple languages is available on the IBM Terminology Web site at: <http://www.ibm.com/software/globalization/terminology/index.jsp>
- National Language Support information; available at the IBM Publications Center at: <http://www.elink.ibm.com/public/applications/publications/cgi-bin/pbi.cgi>
- *SQL Reference for Cross-Platform Development*; available at the following developerWorks® site: <http://www.ibm.com/developerworks/db2/library/techarticle/0206sqlref/0206sqlref.html>

The following information resources are not published by IBM but can be useful to users of DB2 for z/OS and related products:

- Database design topics:
  - *DB2 for z/OS and OS/390® Development for Performance Volume I*, by Gabrielle Wiorkowski, Gabrielle & Associates, ISBN 0-96684-605-2
  - *DB2 for z/OS and OS/390 Development for Performance Volume II*, by Gabrielle Wiorkowski, Gabrielle & Associates, ISBN 0-96684-606-0
  - *Handbook of Relational Database Design*, by C. Fleming and B. Von Halle, Addison Wesley, ISBN 0-20111-434-8
- Distributed Relational Database Architecture™ (DRDA) specifications; <http://www.opengroup.org>
- Domain Name System: *DNS and BIND*, Third Edition, Paul Albitz and Cricket Liu, O'Reilly, ISBN 0-59600-158-4
- Microsoft Open Database Connectivity (ODBC) information; <http://msdn.microsoft.com/library/>
- Unicode information; <http://www.unicode.org>

---

## How to obtain DB2 information

You can access the official information about the DB2 product in a number of ways.

- “DB2 on the Web”
- “DB2 product information”
- “DB2 education” on page 90
- “How to order the DB2 library” on page 90

### DB2 on the Web

Stay current with the latest information about DB2 by visiting the DB2 home page on the Web:

[www.ibm.com/software/db2zos](http://www.ibm.com/software/db2zos)

On the DB2 home page, you can find links to a wide variety of information resources about DB2. You can read news items that keep you informed about the latest enhancements to the product. Product announcements, press releases, fact sheets, and technical articles help you plan and implement your database management strategy.

### DB2 product information

The official DB2 for z/OS information is available in various formats and delivery methods. IBM provides mid-version updates to the information in the information center and in softcopy updates that are available on the Web and on CD-ROM.

#### Information Management Software for z/OS Solutions Information Center

DB2 product information is viewable in the information center, which is the primary delivery vehicle for information about DB2 for z/OS, IMS, QMF, and related tools. This information center enables you to search across related product information in multiple languages for data management solutions for the z/OS environment and print individual topics or sets of related topics. You can also access, download, and print PDFs of the publications that are associated with the information center topics. Product technical information is provided in a format that offers more options and tools for accessing, integrating, and customizing information resources. The information center is based on Eclipse open source technology.

The Information Management Software for z/OS Solutions Information Center is viewable at the following Web site:

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

#### CD-ROMs and DVD

Books for DB2 are available on a CD-ROM that is included with your product shipment:

- DB2 V9.1 for z/OS Licensed Library Collection, LK3T-7195, in English

The CD-ROM contains the collection of books for DB2 V9.1 for z/OS in PDF and BookManager formats. Periodically, IBM refreshes the books on subsequent editions of this CD-ROM.

The books for DB2 for z/OS are also available on the following CD-ROM and DVD collection kits, which contain online books for many IBM products:

- IBM z/OS Software Products Collection , SK3T-4270, in English
- IBM z/OS Software Products DVD Collection , SK3T-4271, in English

#### **PDF format**

Many of the DB2 books are available in PDF (Portable Document Format) for viewing or printing from CD-ROM or the DB2 home page on the Web or from the information center. Download the PDF books to your intranet for distribution throughout your enterprise.

#### **BookManager format**

You can use online books on CD-ROM to read, search across books, print portions of the text, and make notes in these BookManager books. Using the IBM Softcopy Reader, appropriate IBM Library Readers, or the BookManager Read product, you can view these books in the z/OS, Windows, and VM environments. You can also view and search many of the DB2 BookManager books on the Web.

### **DB2 education**

IBM Education and Training offers a wide variety of classroom courses to help you quickly and efficiently gain DB2 expertise. IBM schedules classes in cities all over the world. You can find class information, by country, at the IBM Learning Services Web site:

[www.ibm.com/services/learning](http://www.ibm.com/services/learning)

IBM also offers classes at your location, at a time that suits your needs. IBM can customize courses to meet your exact requirements. For more information, including the current local schedule, contact your IBM representative.

### **How to order the DB2 library**

To order books, visit the IBM Publication Center on the Web:

[www.elink.ibm.com/public/applications/publications/cgibin/pbi.cgi](http://www.elink.ibm.com/public/applications/publications/cgibin/pbi.cgi)

From the IBM Publication Center, you can go to the Publication Notification System (PNS). PNS users receive electronic notifications of updated publications in their profiles. You have the option of ordering the updates by using the publications direct ordering application or any other IBM publication ordering channel. The PNS application does not send automatic shipments of publications. You will receive updated publications and a bill for them if you respond to the electronic notification.

You can also order DB2 publications and CD-ROMs from your IBM representative or the IBM branch office that serves your locality. If your location is within the United States or Canada, you can place your order by calling one of the toll-free numbers:

- In the U.S., call 1-800-879-2755.
- In Canada, call 1-800-426-4968.

To order additional copies of licensed publications, specify the SOFTWARE option. To order additional publications or CD-ROMs, specify the PUBLICATIONS option.

Be prepared to give your customer number, the product number, and either the feature codes or order numbers that you want.



---

## How to use the DB2 library

Titles of books in the library begin with DB2 Version 9.1 for z/OS. However, references from one book in the library to another are shortened and do not include the product name, version, and release. Instead, they point directly to the section that holds the information. The primary place to find and use information about DB2 for z/OS is the Information Management Software for z/OS Solutions Information Center (<http://publib.boulder.ibm.com/infocenter/imzic>).

If you are new to DB2 for z/OS, *Introduction to DB2 for z/OS* provides a comprehensive introduction to DB2 Version 9.1 for z/OS. Topics included in this book explain the basic concepts that are associated with relational database management systems in general, and with DB2 for z/OS in particular.

The most rewarding task associated with a database management system is asking questions of it and getting answers, the task called *end use*. Other tasks are also necessary—defining the parameters of the system, putting the data in place, and so on. The tasks that are associated with DB2 are grouped into the following major categories.

### Installation

If you are involved with installing DB2, you will need to use a variety of resources, such as:

- *DB2 Program Directory*
- *DB2 Installation Guide*
- *DB2 Administration Guide*
- *DB2 Application Programming Guide and Reference for Java*
- *DB2 Codes*
- *DB2 Internationalization Guide*
- *DB2 Messages*
- *DB2 Performance Monitoring and Tuning Guide*
- *DB2 RACF Access Control Module Guide*
- *DB2 Utility Guide and Reference*

If you will be using data sharing capabilities you also need *DB2 Data Sharing: Planning and Administration*, which describes installation considerations for data sharing.

If you will be installing and configuring DB2 ODBC, you will need *DB2 ODBC Guide and Reference*.

If you are installing IBM Spatial Support for DB2 for z/OS, you will need *IBM Spatial Support for DB2 for z/OS User's Guide and Reference*.

If you are installing IBM OmniFind® Text Search Server for DB2 for z/OS, you will need *IBM OmniFind Text Search Server for DB2 for z/OS Installation, Administration, and Reference*.

## End use

End users issue SQL statements to retrieve data. They can also insert, update, or delete data, with SQL statements. They might need an introduction to SQL, detailed instructions for using SPUFI, and an alphabetized reference to the types of SQL statements. This information is found in *DB2 Application Programming and SQL Guide*, and *DB2 SQL Reference*.

End users can also issue SQL statements through the DB2 Query Management Facility (QMF) or some other program, and the library for that licensed program might provide all the instruction or reference material they need. For a list of the titles in the DB2 QMF library, see the bibliography at the end of this book.

## Application programming

Some users access DB2 without knowing it, using programs that contain SQL statements. DB2 application programmers write those programs. Because they write SQL statements, they need the same resources that end users do.

Application programmers also need instructions for many other topics:

- How to transfer data between DB2 and a host program—written in Java, C, or COBOL, for example
- How to prepare to compile a program that embeds SQL statements
- How to process data from two systems simultaneously, for example, DB2 and IMS or DB2 and CICS
- How to write distributed applications across operating systems
- How to write applications that use Open Database Connectivity (ODBC) to access DB2 servers
- How to write applications that use JDBC and SQLJ with the Java programming language to access DB2 servers
- How to write applications to store XML data on DB2 servers and retrieve XML data from DB2 servers.

The material needed for writing a host program containing SQL is in *DB2 Application Programming and SQL Guide*.

The material needed for writing applications that use JDBC and SQLJ to access DB2 servers is in *DB2 Application Programming Guide and Reference for Java*. The material needed for writing applications that use DB2 CLI or ODBC to access DB2 servers is in *DB2 ODBC Guide and Reference*. The material needed for working with XML data in DB2 is in *DB2 XML Guide*. For handling errors, see *DB2 Messages* and *DB2 Codes*.

If you are a software vendor implementing DRDA clients and servers, you will need *DB2 Reference for Remote DRDA Requesters and Servers*.

Information about writing applications across operating systems can be found in *IBM DB2 SQL Reference for Cross-Platform Development*.

## System and database administration

*Administration* covers almost everything else. *DB2 Administration Guide* divides some of those tasks among the following sections:

- **DB2 concepts:** Introduces DB2 structures, the DB2 environment, and high availability.
- **Designing a database:** Discusses the decisions that must be made when designing a database and tells how to implement the design by creating and altering DB2 objects, loading data, and adjusting to changes.
- **Security and auditing:** Describes ways of controlling access to the DB2 system and to data within DB2, to audit aspects of DB2 usage, and to answer other security and auditing concerns.
- **Operation and recovery:** Describes the steps in normal day-to-day operation and discusses the steps one should take to prepare for recovery in the event of some failure.

*DB2 Performance Monitoring and Tuning Guide* explains how to monitor the performance of the DB2 system and its parts. It also lists things that can be done to make some parts run faster.

If you will be using the RACF access control module for DB2 authorization checking, you will need *DB2 RACF Access Control Module Guide*.

If you are involved with DB2 only to design the database, or plan operational procedures, you need *DB2 Administration Guide*. If you also want to carry out your own plans by creating DB2 objects, granting privileges, running utility jobs, and so on, you also need:

- *DB2 SQL Reference*, which describes the SQL statements you use to create, alter, and drop objects and grant and revoke privileges
- *DB2 Utility Guide and Reference*, which explains how to run utilities
- *DB2 Command Reference*, which explains how to run commands

If you will be using data sharing, you need *DB2 Data Sharing: Planning and Administration*, which describes how to plan for and implement data sharing.

Additional information about system and database administration can be found in *DB2 Messages* and *DB2 Codes*, which list messages and codes issued by DB2, with explanations and suggested responses.

## **Diagnosis**

Diagnosticians detect and describe errors in the DB2 program. They might also recommend or apply a remedy. The documentation for this task is in *DB2 Diagnosis Guide and Reference*, *DB2 Messages*, and *DB2 Codes*.



---

## Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd.  
3-2-12, Roppongi, Minato-ku  
Tokyo 106-8711  
Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
J46A/G4  
555 Bailey Avenue  
San Jose, CA 95141-1003  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

---

## Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)<sup>®</sup> are trademarks or registered marks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT<sup>®</sup>, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

---

## Glossary

The glossary is available in several locations.

- The Glossary topic in the Information Management Software for z/OS Solutions Information Center.
- In most printed product manuals and the corresponding PDFs.
- In the BookManager edition of the *Administration Guide*.



---

# Index

## A

- Advanced Security Standard encryption 34
- ALTER BUFFERPOOL command 36
- ALTER FUNCTION statement 48
- ALTER INDEX statement 48
- ALTER PROCEDURE statement 48
- ALTER TABLE statement
  - description of changes 48
- ALTER TABLESPACE statement 48
- ALTER UTILITY command 36
- application programming skills 10
- auditing 33
- autonomic reoptimization 25
- availability enhancements
  - universal table space
    - partition-by-growth table space 21
    - range-partitioned table space 21
- Availability enhancements
  - ALTER STOGROUP support for SMS storage classes 23
  - ALTER TABLESPACE 23
  - CLONE tables 21
  - column renaming 23
  - CREATE STOGROUP support for SMS storage classes 23
  - EARLY code, modify without IPL 23
  - faster replacement of tables 21
  - index logging 23
  - index renaming 23
  - online REBUILD INDEX 23
  - online REORG with no BUILD2 phase 21
  - SMS storage classes, support for 23

## B

- BACKUP SYSTEM utility 42
- BIGINT data type and function 5
- BINARY data type and function 6
- BIND PACKAGE command 36
- BIND PLAN command 36

## C

- catalog tables
  - columns
    - changed 61
    - new 61
  - indexes
    - changed 65
    - new 65
  - new 60
- catalogs
  - changes 59
- CATMAINT utility 42
- CD-ROM, books on 89
- CHECK DATA utility 42
- CHECK INDEX utility 42
- CHECK LOB utility 43
- CLONE tables 21
- COLLATION\_KEY function 9
- column, renaming 23

- commands
  - changes 35
  - new 35
- COMMENT statement 48
- consistency across DB2 family 10
  - large object support 11
- COPY utility 43
- COPYTOCOPY utility 43
- CREATE DATABASE statement 48
- CREATE FUNCTION statement 48
- CREATE INDEX statement 49
- CREATE PROCEDURE statement 49
- CREATE TABLE statement 49
- CREATE TABLESPACE statement 49
- CREATE TRIGGER statement 49

## D

- data caching, in-memory 25
- data insert
  - performance improvements 31
- database, automatic creation of 9
- DB2 books online 89
- DB2 Information Center for z/OS solutions 89
- DB2 Optimization Expert for z/OS 28
- DB2 QMF
  - DB2 QMF for TSO/CICS 15
  - DB2 QMF for WebSphere 15
  - DB2 QMF for Workstation 15
  - DB2 QMF High Performance Option 15
  - packaging options 15
- DB2 Tools 15, 18
  - DB2 Optimization Expert for z/OS 28
  - Optimization Service Center for DB2 for z/OS 27
- DB2 V9.1 for z/OS
  - overview 1
- DECFLOAT built-in data type 14
- DECLARE GLOBAL TEMPORARY TABLE statement 49
- DELETE statement
  - description of changes 50
- DIAGNOSE utility 43
- DISPLAY DATABASE command 37
- DISPLAY DDF command 37
- DISPLAY LOCATION command 37
- DISPLAY PROCEDURE command 37
- DISPLAY THREAD command 37
- DISPLAY TRACE command 38
- DISPLAY UTILITY command 38
- DROP statement 50
- DSN\_FUNCTION\_TABLE
  - columns 79
  - format 79
- DSN\_STATEMENT\_CACHE\_TABLE 79
- DSN\_STATEMENT\_TABLE
  - format 76
- DSN\_STATEMENT\_TABLE table
  - columns
    - changed 76
    - new 76
- DSNJU003 utility 46
- DSNJU004 utility 46

## E

EARLY code, modify without IPL 23  
encryption  
  passwords 34  
  user IDs 34  
EXCEPT keyword in a subselect 6  
EXECUTE statement 50  
EXPLAIN statement 50  
EXPLAIN tables  
  changes 69  
  DSN\_STATEMENT\_TABLE 69  
  PLAN\_TABLE 69  
extended address volumes 24

## F

FETCH FIRST n ROWS ONLY 25  
FETCH statement  
  description of changes 50  
  use for LOB data 11  
  use for XML data 11  
file reference variables 6  
functions  
  column 51  
  new 51  
  scalar 51

## G

GET DIAGNOSTICS statement  
  description of changes 50  
global query optimization 25

## H

histogram statistics 25

## I

IFCID (instrumentation facility component identifier)  
  changed 80  
  changed IFCIDs 81  
  new 80  
index compression 29  
index key randomization 29  
index on expression 29  
  creating 9  
index page size, larger 29  
index, renaming 23  
indexes  
  changed 65  
  new 65  
information on demand 3  
INSERT statement  
  description of changes 50  
INSTEAD OF triggers 5  
INTERSECT keyword in a fullselect 6

## K

keywords, reserved 53

## L

large object enhancements  
  FETCH statement for LOB and XML data 11  
  LOB file reference variables 11  
large object support  
  enhancements 11  
large objects  
  data retrieval optimization 13  
  elimination of LOB locks 13  
  performance enhancements 13  
large objects, utilities enhancements for  
  CHECK DATA utility 12  
  CHECK LOB utility 12  
  LOB handling for LOAD and UNLOAD utilities 12  
  REORG of LOB table spaces 12  
library 89  
LISTDEF utility 43  
LOAD utility 44  
LOB data, FETCH statement for 11  
LOB file reference variables 11  
logging  
  performance improvements 31

## M

MERGE statement 15  
MERGECOPY utility 44  
MODIFY irImproc command 39  
MODIFY RECOVERY utility 44

## N

network trusted contexts 33  
not logging table spaces, expanded support 7

## O

OLAP specifications  
  DENSE\_RANK 8  
  RANK 8  
  ROW\_NUMBER 8  
online 89  
online books 89  
online REBUILD INDEX 23  
online REORG with no BUILD2 phase 21  
optimistic currency control 14  
Optimization Service Center for DB2 for z/OS 27  
ORDER BY in subselects and fullselects 25

## P

page-range screening 25  
partition-by-growth table space 21  
performance improvements 25  
  CPU processing time for utilities, reduction in 25  
  CPU processing time reduction for utilities 25  
  data insert 31  
  DB2 Optimization Expert for z/OS 28  
  histogram statistics 25  
  logging 31  
  Optimization Service Center for DB2 for z/OS 27  
  query optimization management 27  
  query optimization techniques 25  
  query performance  
    index compression 29

- performance improvements (*continued*)
  - query performance (*continued*)
    - index key randomization 29
    - index on expression 29
    - index page size, larger 29
  - sequential key insert, relief for 30
  - SQL optimization improvements 25
  - varying-length rows 30
- PLAN\_TABLE
  - columns 71
  - format 70
- PREPARE statement 50

## Q

- query optimization management 27
- query optimization techniques
  - autonomic reoptimization 25
  - FETCH FIRST n ROWS ONLY 25
  - global query optimization 25
  - in-memory data caching 25
  - ORDER BY in subselects and fullselects 25
  - page-range screening 25
- QUIESCE utility 44

## R

- range-partitioned table space 21
- REBIND PACKAGE command 36
- REBIND PLAN command 36
- REBUILD INDEX utility 44
- REBUILD INDEX, online 23
- RECOVER utility
  - description of changes 44
- regulatory compliance 33
  - auditing, improved 33
  - INSTEAD OF triggers 34
  - network trusted contexts 33
  - roles 33
  - Secure Socket Layer protocol 33
  - SSL protocol 33
  - trace filtering 33
- RENAME COLUMN clause 23
- REORG INDEX utility 45
- REORG TABLESPACE utility 45
- REORG, online with no BUILD2 phase 21
- REPAIR utility 45
- REPORT utility 45
- reserved keywords 53
- RESET INDOUBT command 39
- RESTORE SYSTEM utility 45
- roles 33
- RUNSTATS utility
  - description of changes 46

## S

- Secure Socket Layer protocol 33
- SELECT FROM DELETE function 5
- SELECT FROM MERGE statement 15
- SELECT FROM UPDATE function
  - SQL consistency 5
- SELECT INTO statement
  - description of changes 50
- sequential key insert, relief for 30
- session variables 59

- softcopy publications 89
- Spatial Support for DB2 for z/OS 9
- SQL
  - ALTER FUNCTION statement 48
  - ALTER INDEX statement 48
  - ALTER PROCEDURE statement 48
  - ALTER TABLE statement 48
  - ALTER TABLESPACE statement 48
  - COMMENT statement 48
  - CREATE DATABASE statement 48
  - CREATE FUNCTION statement 48
  - CREATE INDEX statement 49
  - CREATE PROCEDURE statement 49
  - CREATE TABLE statement 49
  - CREATE TABLESPACE statement 49
  - CREATE TRIGGER statement 49
  - DECLARE GLOBAL TEMPORARY TABLE statement 49
  - DELETE statement
    - description of changes 50
  - DROP statement 50
  - EXECUTE statement 50
  - EXPLAIN statement 50
  - FETCH statement
    - description of changes 50
  - GET DIAGNOSTICS statement
    - description of changes 50
  - INSERT statement
    - description of changes 50
  - PREPARE statement 50
  - SELECT INTO statement
    - description of changes 50
  - UPDATE statement
    - description of changes 51
- SQL (Structured Query Language)
  - keywords, reserved 53
- SQL built-in data types
  - DECFLOAT 14
  - VARBINARY 14
- SQL compatibility and leadership 4
- SQL consistency
  - automatic creation of a database 9
  - automatic creation of a table space 9
  - automatic creation of all system-required objects 9
  - BIGINT data type and function 5
  - BINARY data type and function 6
  - COLLATION\_KEY function 9
  - EXCEPT keyword in a subselect 6
  - expanded support for not logging table spaces 7
  - file reference variables 6
  - index on expression 9
  - INSTEAD OF triggers 5
  - INTERSECT keyword in a fullselect 6
  - native support for SQL procedures 7
  - OLAP specifications
    - DENSE\_RANK 8
    - RANK 8
    - ROW\_NUMBER 8
  - SQL procedures, nested compound statements in 7
- SQL consistency improvements 5
- SQL language
  - changes 58
- SQL leadership, DB2 family firsts
  - DECFLOAT built-in data type 14
  - MERGE statement 15
  - optimistic currency control 14
  - SELECT FROM MERGE statement 15
  - update detection 14

- SQL leadership, DB2 family firsts (*continued*)
  - VARBINARY data type 14
- SQL leadership, family firsts
  - TRUNCATE TABLE statement 13
- SQL procedures
  - native support 7
- SQL procedures, nested compound statements in 7
- SQL statements
  - changes 47
  - clauses
    - changed 48
    - new 48
  - new 47
- SSL protocol 33
- START DATABASE command 39
- START DB2 command 39
- START irlmproc command 39
- START PROCEDURE command 39
- START TRACE command 40
- STOP DATABASE command 41
- STOP PROCEDURE command 40
- STOP TRACE command 41
- system-required objects, automatic creation of 9

## T

- table space
  - partition-by-growth 21
  - range-partitioned 21
  - universal 21
- table space, automatic creation of 9
- TEMPLATE utility 46
- trace enhancements 80
- trace filtering 33
- TRUNCATE TABLE statement 13

## U

- universal table space 21
- UNLOAD utility 46
- update detection 14
- UPDATE statement
  - description of changes 51
- utilities
  - BACKUP SYSTEM 42
  - CATMAINT 42
  - changes 42, 47
  - CHECK DATA 42
  - CHECK INDEX 42
  - CHECK LOB 43
  - COPY 43
  - COPYTOCOPY 43
  - DIAGNOSE 43
  - DSNJU003 46
  - DSNJU004 46
  - LISTDEF 43
  - LOAD 44
  - MERGECOPY 44
  - MODIFY RECOVERY 44
  - QUIESCE 44
  - REBUILD INDEX 44
  - RECOVER
    - description of changes 44
  - REORG INDEX 45
  - REORG TABLESPACE 45
  - REPAIR 45

- utilities (*continued*)
  - REPORT 45
  - RESTORE SYSTEM 45
  - RUNSTATS
    - description of changes 46
  - TEMPLATE 46
  - UNLOAD 46
- utilities enhancements for large objects
  - CHECK DATA utility 12
  - CHECK LOB utility 12
  - LOAD and UNLOAD utilities, LOB handling for 12
  - REORG of LOB table spaces 12
- utilities, CPU reduction time in 25

## V

- VARBINARY data type 14
- varying-length rows, improved performance for 30

## X

- XML
  - application development 3
  - database administration support 3
  - integration of XML data and relational data 3
  - performance benefits 3
  - XML document retrieval 3
  - XML document storage 3
- XML data, FETCH statement for 11





Program Number: 5635-DB2

Printed in USA

GC18-9856-05



Spine information:

DB2 Version 9.1 for z/OS

What's New?

