

IBM Db2 Analytics Accelerator for z/OS
Version 7.1.5

User's Guide



Note

Before you use this information and the product it supports, read the information in “Notices” on page 169.

Sixth Edition, December 2018

This edition applies to version 7.1.5 of IBM Db2 Analytics Accelerator for z/OS (product number 5697-DA7), and to all subsequent releases and modifications until otherwise indicated in new editions. This edition replaces SH12-7082-04. Changes to this edition are marked with a vertical bar.

© **Copyright IBM Corporation 2010, 2018.**

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

Contents

Figures	v
--------------------------	----------

About this book	vii
----------------------------------	------------

Who should read this book	ix
--	-----------

What's new in version 7?	xi
---	-----------

Accessibility features	xiii
---	-------------

Chapter 1. Introduction	1
--	----------

Connecting to a data server	1
Creating a database connection profile.	1
Connecting to a database	1
Adding accelerators	1
Defining the data to load into an accelerator.	2
Selecting the proper tables.	2
Checking the table size	2
Working with accelerator-shadow tables	2
Loading data into selected tables	2
Enabling tables for query acceleration	3
Disabling query acceleration for a table	3
Removing tables	3
Order in which to complete tasks	4

Chapter 2. Basic concepts	5
--	----------

Accelerator	5
Static SQL query support	5
Accelerator-shadow table	5
How to select tables for query acceleration	6
Distribution keys	7
Tables states	9
Accelerator-only tables and in-database transformation	11
Workload balancing	16
IBM Db2 Analytics Accelerator for z/OS system table entries	16

Chapter 3. Conditions for query routing to an accelerator	17
--	-----------

Configuring Db2 for z/OS for the acceleration of dynamic SQL queries	17
Configuring Db2 for z/OS for the acceleration of static SQL queries	18
Supported queries	21
Supported data types	22
Supported SQL functions and expressions	26
Supported functions of the database engine	35
Supported special registers	37
Conditions that prevent query routing to an accelerator.	39
Isolation levels	40

Incompatibilities between Db2 for z/OS and IBM Db2 Analytics Accelerator	41
Data types.	41
Division and multiplication of decimal values	41
Encoding issues	42
Exception handling.	43
Maximum row length	43
Predicate evaluation	43
SQL issues.	44
Time and date values	44

Incompatibilities related to IBM InfoSphere Data Replication for z/OS	47
Restrictions of accelerator-only tables.	47
Newer Db2 features	50
Differences in floating-point error tolerances	54

Chapter 4. IBM Db2 Analytics Accelerator Studio.	57
---	-----------

Chapter 5. Interface elements of IBM Db2 Analytics Accelerator Studio	59
--	-----------

Menu bar	59
Administration Explorer	59
Object List Editor	61
Accelerator view.	64

Chapter 6. Tasks	71
-----------------------------------	-----------

Creating a database connection profile	71
Connecting to a database server	73
Exploring a database server	74
Adding accelerators	75
Obtaining the pairing code for authentication	75
Completing the authentication using the Add Accelerator wizard	77
Displaying accelerator networking details	78
Defining tables on an accelerator	79
Determining the status of an accelerator.	80
Enabling an accelerator	81
Disabling an accelerator	82
Enabling or disabling an accelerator for members of a data sharing group	82
Loading tables	83
Enabling tables for query acceleration	88
Disabling query acceleration for tables	89
Specifying or changing a distribution key	90
Running an SQL script from IBM Db2 Analytics Accelerator Studio	93
EXPLAIN information.	94
Selecting and ordering query monitoring columns	99
Selectable columns in the Query Monitoring section	103
Canceling tasks.	109
Removing tables from an accelerator	110
Exporting a table specification.	110
Importing a table specification.	111

Removing accelerators	112
Tracing	113
Configuring the trace behavior	113
Saving trace information.	114
Setting a refresh interval or a skew threshold.	116
Updating accelerator tables continually.	117
Making queries wait for incremental updates	119
Starting or stopping incremental updates	123
Including or excluding tables from incremental updates	125
Monitoring incremental updates	128
Using the event viewer for incremental updates	130
Warnings related to incremental update processes	131
Collecting CDC trace information	132
What to do after changing a table or table space in Db2 for z/OS	133
Updating IBM Db2 Analytics Accelerator Studio	133
Installing updates	135
Updating IBM Db2 Analytics Accelerator software	135
Removing obsolete software packages from an accelerator	140
Opening a service request	140

Chapter 7. Status information about the system and the incremental update function 145

Chapter 8. Troubleshooting 151

Firewall blocks operations because crucial ports are closed	151
IBM Db2 Analytics Accelerator Studio does not start	152
IBM Db2 Analytics Accelerator Studio on Linux does not start	152
Package not found when running a stored procedure from IBM Db2 Analytics Accelerator Studio.	153
Long-running tasks are not completed	154
SQL code -430 from IBM Db2 Analytics Accelerator for z/OS stored procedures.	154
Database access problems after updating IBM Db2 Analytics Accelerator Studio or IBM Optim Query Tuner	155
DRDA connection does not work.	156
Different results for floating-point aggregates.	156
Stored procedure calls fail with message AQT10206I while temporary files are being created.	157
Query history is not displayed - error message returned	157
Db2 for z/OS returns SQLCODE -30040	158

Appendix A. Impact of special register, ZPARM, or bind option settings on accelerator-only tables . . . 159

Appendix B. Using telnet to log on to the IBM Db2 Analytics Accelerator Console. 165

Appendix C. Using ssh to log on to the IBM Db2 Analytics Accelerator Console. 167

Notices 169

Trademarks	171
Terms and conditions for product documentation	171

GDPR considerations 173

GDPR	173
Why is GDPR important?	173
Product Configuration for GDPR	174
How to configure the offering so that it can be used in a GDPR environment	174
Data Life Cycle.	175
What types of data?	175
Where in the process?	175
For what purpose?	175
Personal data used for online contact with IBM	175
Data Collection.	176
Data Storage	176
Data Access	177
Data Processing	177
Data Deletion	178
Data Monitoring	178
Responding to Data Subject Rights	178
Does the offering facilitate being able to meet data subject rights?	178

Glossary 181

A	181
C	181
D	181
F	182
I.	182
L	182
P	182
Q	182
R	182
S	183
T	183
V	184
W	184

Index 185

Figures

1. IBM Db2 Analytics Accelerator Studio tasks in the overall process flow 4
2. Access plan graph of the query in the example 7
3. Possible table state transitions 11
4. IBM Db2 Analytics Accelerator setup 57
5. Administration Explorer 60
6. Object List Editor 61
7. Accelerator view 64
8. Database server connection in the Administration Explorer 74
9. Wait time, execution time, fetch time, and elapsed time in comparison 106
10. **Replication status** and functional link in the header of the accelerator view 124
11. Timestamp columns and **Replication** button in the accelerator view 127
12. **Replication latency** in the accelerator view 130
13. All members in the same LPAR 148
14. Members in two different LPARs 149
15. SQL code -430 message window 155

About this book

This book provides information about the following subjects:

- General information on components and objects used by IBM® Db2® Analytics Accelerator for z/OS®
- Conditions for query acceleration with IBM Db2 Analytics Accelerator for z/OS (supported queries, data types, functions, limitations, and so on)
- Connecting to a data server
- Adding accelerators
- Adding and maintaining tables for query acceleration with IBM Db2 Analytics Accelerator for z/OS
- Configuring and activating system traces
- Adding, removing, activating, and deactivating individual accelerators
- Component updates

Who should read this book

This book is intended for database professionals and database administrators who need to complete the following tasks with IBM Db2 Analytics Accelerator for z/OS:

- Accelerate Db2 for z/OS queries with IBM Db2 Analytics Accelerator for z/OS
- Complete administration tasks with the help of IBM Db2 Analytics Accelerator Studio

What's new in version 7?

IBM Db2 Analytics Accelerator for z/OS Version 7.1.0 is a newly designed product rather than a major upgrade. It runs on different, more powerful hardware, which allows unprecedented acceleration rates. It also comes with a new deployment option, which does not require external hardware, but runs exclusively in a dedicated LPAR on your existing IBM Z hardware.

- Common Db2 in-memory, columnar SQL engine with BLU acceleration and SQL compatibility improvements, for example:
 - Native support for the EBCDIC MBCS and GRAPHIC data types (instead of a conversion to UTF-8 as in earlier versions)
 - Acceleration of all types of correlated sub-queries (only small subset was offloaded in version 5), including table expressions with sideways references
 - Support for timestamps of precision 12 (these were truncated to precision 6 in earlier versions)
 - Native support for the FOR BIT DATA subtype, for all types of table encoding (EBCDIC, UNICODE, ASCII). This was available for EBCDIC only in earlier versions).
 - Native support for the TIMESTAMP value 24:00:00 (this was mapped to 23:59:59 in earlier versions)
 - Improved support for scalar functions that were previously not offloaded when applied to certain data types
 - Support for MIN, MAX, DAY, LAST_DAY, BIT*, TIMESTAMP_ISO, VARIANCE, and STDDEV with the UNIQUE clause
 - Support for the HEX function
 - Improved support for tables in mixed encoding
 - EBCDIC tables can be added to accelerator even if UNICODE tables are already present
 - Better time synchronization through use of system heartbeat
 - Improved accuracy for CURRENT_TIME, CURRENT_TIMESTAMP, and CURRENT_DATE special registers
- Smart Load, meaning that the SYSPROC.ACCEL_LOAD_TABLES stored procedures optimize the workload and the degree of parallelism for load jobs.
- New software update approach: Updates for the accelerator are not part of the z/OS installation package, but can be downloaded individually from IBM FixCentral, which significantly reduces the installation effort.
- New and changed stored procedures and interfaces:
 - SYSPROC.ACCEL_UPDATE_SOFTWARE2 replaces SYSPROC.ACCEL_UPDATE_SOFTWARE
 - The GetActivationLog sub-function extends SYSPROC.ACCEL_CONTROL_SYSPROC.ACCELERATOR, which allows you to monitor software activations.
 - New monitoring interface extensions for query details
 - SYSPROC.ACCEL_GET_QUERIES2 extends the SYSPROC.ACCEL_GET_QUERIES interface by returning query statements in a result set rather than in an output parameter of limited size. Query statements in the result set are not truncated if they exceed 128 bytes in length.

- SYSPROC.ACCEL_GET_QUERY_DETAILS2 retrieves query plan information from the remote database management system in separate result sets. The information is not combined with the query text anymore.
- SYSPROC.ACCEL_GET_QUERY_EXPLAIN returns the Db2 EXPLAIN information in a compressed, base64-encoded format.
- The following stored procedures can write IFCID392 audit traces: SYSPROC.ACCEL_ADD_TABLES, SYSPROC.ACCEL_ALTER_TABLES, SYSPROC.ACCEL_REMOVE_TABLES

Enhancements in version 7.1.5

See the release notes.

Enhancements in version 7.1.4

See the release notes.

Enhancements in version 7.1.3

See the release notes.

Enhancements in version 7.1.2

See the release notes.

Enhancements in version 7.1.1

See the release notes.

Accessibility features

Accessibility features help users who have a disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The following list includes the major accessibility features in IBM Db2 Analytics Accelerator Studio:

- Keyboard-only operation
- Interfaces that are commonly used by screen readers

The IBM Db2 Analytics Accelerator for z/OS Information Center, and its related publications, are accessibility-enabled. The accessibility features of the information center are described in the topic Accessibility features in Eclipse.

Keyboard navigation

In addition to the standard Microsoft Windows navigation keys, this product uses keyboard shortcuts and controls of the Eclipse framework. For more information about these features, see the following website:

Navigating the user interface using the keyboard).

Interface information

The Eclipse framework supports the JAWS screen reader software. If you are facing difficulties in getting JAWS to read out table column headers, see the following websites:

- Configuring accessibility options for textual editors
- Tips for making user interfaces accessible

IBM and accessibility

See the IBM Human Ability and Accessibility Center for more information about the commitment that IBM has to accessibility.

Chapter 1. Introduction

IBM Db2 Analytics Accelerator for z/OS is a query accelerator capable of reducing response times for Db2 for z/OS queries by an order of magnitude.

IBM Db2 Analytics Accelerator for z/OS integrates deeply into your existing Db2 for z/OS environment and can process very large amounts of data. The addition of IBM Db2 Analytics Accelerator for z/OS makes Db2 for z/OS a truly universal database management system with optimum performance characteristics for transactional and analytical applications.

IBM Db2 Analytics Accelerator Studio is the graphical frontend of IBM Db2 Analytics Accelerator for z/OS. It offers functions for the configuration and administration, as well as analytical tools that collect performance metrics or troubleshooting information.

This chapter provides an overview of the basic tasks that can be completed with IBM Db2 Analytics Accelerator Studio.

Connecting to a data server

IBM Db2 Analytics Accelerator for z/OS needs access to your data servers to read the database catalogs and invoke the stored procedures for the accelerators. To connect to a database server, you must complete the tasks that are described in this section.

Creating a database connection profile

IBM Db2 Analytics Accelerator Studio is the user interface for administering accelerators. It connects to your Db2 for z/OS data server using Java™ Database Connectivity (JDBC), like other database clients. The configuration parameters for accessing a Db2 for z/OS data server are stored in database connection profiles. Create a database connection profile for each Db2 for z/OS data server with an accelerator.

Related tasks:

“Creating a database connection profile” on page 71

Connecting to a database

To connect IBM Db2 Analytics Accelerator Studio to a Db2 for z/OS database, double-click one of the database connection profiles in the Administration Explorer.

Adding accelerators

When a connection to a database has been established through one of your database connection profiles, you can see all currently deployed accelerators in the Object List Editor. You access these interface elements from the Accelerator Perspective in IBM Db2 Analytics Accelerator Studio. To connect a newly installed accelerator, use the Add Accelerator wizard in IBM Db2 Analytics Accelerator Studio.

Related tasks:

“Completing the authentication using the Add Accelerator wizard” on page 77

Defining the data to load into an accelerator

Defining the data to load into an accelerator basically means selecting the proper tables. However, you also need to complete a number of related sub-tasks. This section lists all required tasks in the correct order.

Selecting the proper tables

In choosing the proper tables, you determine the data to be queried. For each Db2 for z/OS table that you select, a corresponding table is created on one or more accelerators. These tables are called accelerator-shadow tables. Accelerator-shadow tables are not necessarily copies of their originals. Often, they contain fewer columns than the tables they stem from because columns with unsupported data types are excluded. Accelerator-shadow tables must be filled with data before they can be queried. This is done in a later step (see loading accelerator-shadow tables).

Related tasks:

“Defining tables on an accelerator” on page 79

Checking the table size

When you select a table for an accelerator in the Add Tables wizard, IBM Db2 Analytics Accelerator Studio displays the table size. This table size is calculated on the basis of the values in the CARDF and AVGROWLEN columns of the SYSIBM.SYSTABLES table. Thus the metrics come from Db2 for z/OS. Comparing this size with the available disk space on your accelerator, you can roughly assess how much space the table will occupy on your accelerator. However, Db2 for z/OS and the accelerator use completely different compression algorithms, so the size of a table in Db2 might differ from the size of the same table on the accelerator.

Working with accelerator-shadow tables

To populate, maintain, and control accelerator-shadow tables, you complete the tasks that are described here.

Loading data into selected tables

To enable users to run accelerated queries against selected database tables, you must load the empty tables that you previously added to the accelerator with data. In doing so, you create a snapshot copy of your existing Db2 for z/OS data on the accelerator.

The Db2 Optimizer calculates the expected response times for incoming queries. If an accelerator can process a query against the selected tables faster than the database management system, the query is routed to the accelerator and evaluated against the populated accelerator-shadow tables.

You can reload accelerator-shadow tables, and, under certain conditions, also reload selected partitions of such tables. Doing so replaces the current data with a more recent snapshot, resulting in an update. When you reload tables or partitions, the accelerator automatically detects the tables and partitions that changed since the last load. This way, you can avoid loading the same data twice. The load wizard shows the load status for each table or partition and recommends maintenance activities.

Related concepts:

“Updating accelerator tables continually” on page 117

Related tasks:

“Loading tables” on page 83

Enabling tables for query acceleration

Query acceleration can be switched on and off for each accelerator-shadow table. This allows you to do maintenance work on a particular table while query acceleration remains active for other tables on the same accelerator.

Disabling query acceleration for a table

Disabling a table, you can prevent accelerated queries against this table. This might be necessary when a query must be run against very recent data, which has not yet been copied to the accelerator.

Removing tables

If tables are no longer needed for queries, you can use the removal function in IBM Db2 Analytics Accelerator Studio to remove the accelerator-shadow tables from the accelerator.

Related tasks:

“Removing tables from an accelerator” on page 110

Order in which to complete tasks

See a diagram that shows the order of the most common tasks.

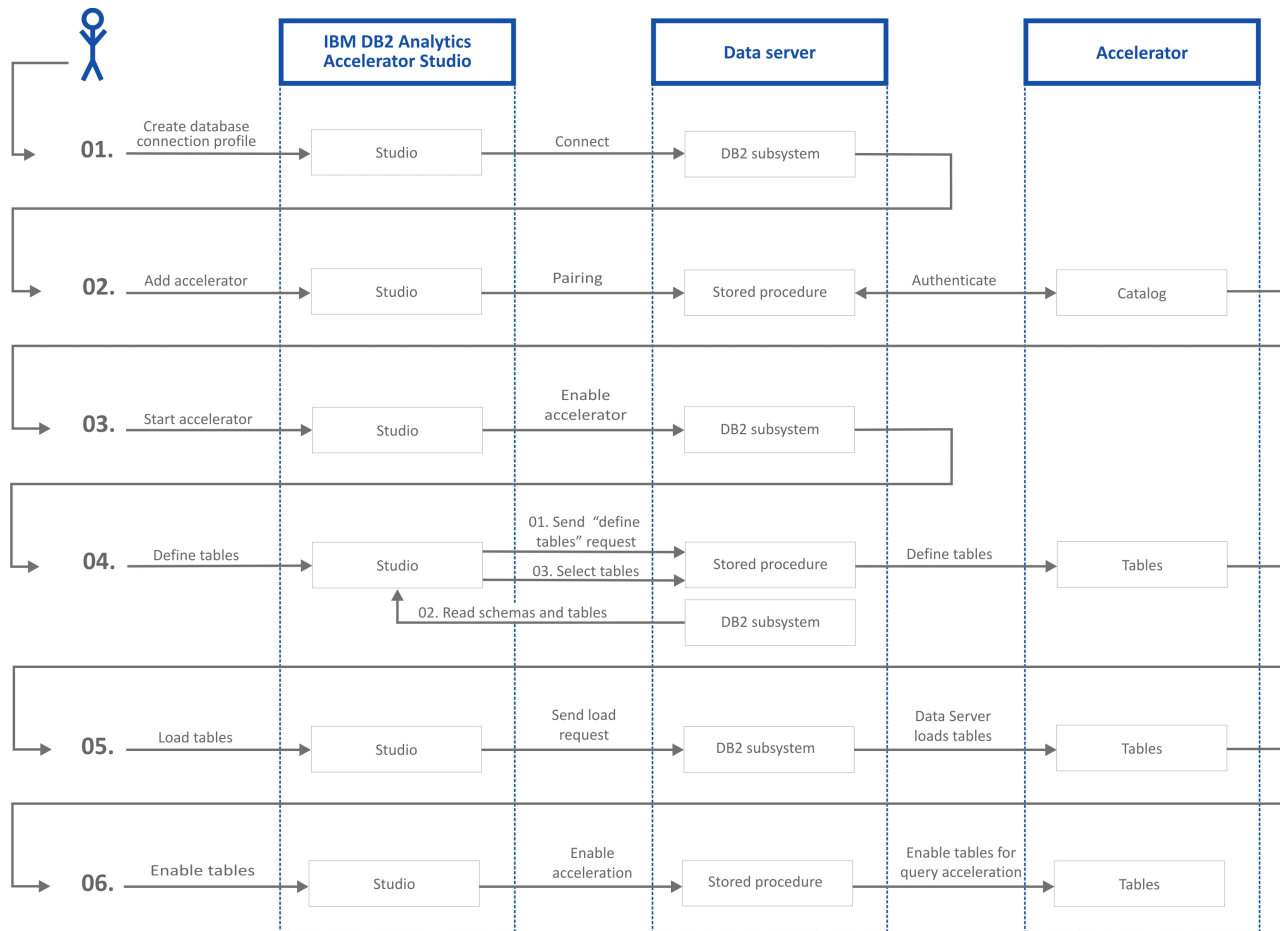


Figure 1. IBM Db2 Analytics Accelerator Studio tasks in the overall process flow

Chapter 2. Basic concepts

Get familiar with the basic concepts of IBM Db2 Analytics Accelerator for z/OS.

Accelerator

IBM Db2 Analytics Accelerator for z/OS consists of a high-performance hardware platform and an optimized database query engine. These components work together to support a variety of data analysis and business reporting tasks.

The high-performance query processing capabilities are embedded in a “shared nothing” architecture, which means that each worker node is able to process data independently, without network traffic or communication between the nodes. Hence there is hardly a contention for shared resources.

Each query is processed in parallel by all worker nodes, with each node processing a portion of each accelerator-shadow table. These portions are called *data slices*. The accelerator-shadow tables are thus horizontally partitioned because each worker node processes a data slice of each table.

Static SQL query support

Static SQL query support means that you can accelerate SQL queries that are embedded within Db2 for z/OS application programs.

Static SQL queries are prepared at the time the application program is bound as a package in Db2 for z/OS, that is, before the execution of the application in Db2. To accelerate static SQL queries within these application programs, you must BIND or REBIND the Db2 package for that application and specify a new bind option that requests static query acceleration.

Accelerator-shadow table

An accelerator-shadow table represents a table in your database. When you define accelerator-shadow tables in IBM Db2 Analytics Accelerator Studio, you instruct IBM Db2 Analytics Accelerator for z/OS to copy the definition of that table (an empty table) to an accelerator.

Queries including this table are then routed to the accelerator provided that the requirements for query redirection are met. A query can only be routed to an accelerator if the query refers to a subset or all of the tables on the accelerator.

Restriction: A table can be added to multiple accelerators, but an accelerated query using that table can run only on one accelerator at a time.

In addition, a query can only be accelerated if all tables that are referenced by the query reside on the same accelerator, if the accelerator-shadow tables are loaded, and if query acceleration is enabled for these tables. A query can reference tables indirectly if the table references a view or a table alias. For example, if the query references a view that joins two tables, both tables must be present on the accelerator.

When tables are added to an accelerator, Db2 for z/OS keeps track of these accelerator-shadow tables in its system catalog. The system table entries enable the optimizer of the database management system to complete the following tasks:

- Query matching, that is, comparing an incoming query with the information stored in the system tables to check whether the query can be handled by IBM Db2 Analytics Accelerator for z/OS.
- Query evaluation, that is, calculating the estimated query response time. If a performance gain can be predicted, the query is sent to IBM Db2 Analytics Accelerator for z/OS. If not, the query is handled by the database management system itself.

If a table contains columns with unsupported data types, these columns will not be included in the corresponding accelerator-shadow table. All other columns will be copied. So the job of defining a table will be completed even if some columns do not fulfill the criteria. However, queries that address columns with unsupported data types will not be accelerated.

Related tasks:

“Defining tables on an accelerator” on page 79

How to select tables for query acceleration

Selecting the proper tables for a query to be accelerated is basically simple: You need to define the tables that are referenced by the query. Slightly more thought needs to be spent on the choice of proper distribution keys and organizing keys.

Suppose that you want to accelerate the following query, which extracts the total number of items and the total amount of money paid by each customer from a set of sales tables. The query does not reach infinitely into the past; rather, it considers just the orders that were received after a certain date. The results will be grouped by customer number.

Note: The 0_TOTALPRICE column denotes the total price per order. It is not the total price that is calculated by the query.

Query:

```
SELECT SUM(L_QUANTITY), SUM(0_TOTALPRICE), 0_CUSTKEY
FROM TPCH.LINEITEM L JOIN TPCH.ORDERS 0 on L.L_ORDERKEY = 0.0_ORDERKEY
WHERE 0.0_ORDERDATE > '01.01.1994' GROUP BY 0.0_CUSTKEY;
```

The query references the following tables:

- LINEITEM (L)
- ORDERS (0)

You would thus define these tables on the accelerator.

The query contains the following join predicate (join condition):

```
L.L_ORDERKEY = 0.0_ORDERKEY
```

Suppose that 0.0_ORDERKEY is a primary key and that the L table uses this key as a foreign key (L.L_ORDERKEY). To enable colocated joins, which lead to shorter query response times, the columns of accelerator-shadow tables must be on the same processing node. In order to achieve this, you would thus use 0.0_ORDERKEY and L.L_ORDERKEY as distribution keys. Hence you would specify 0.0_ORDERKEY as the distribution key for 0 and L.L_ORDERKEY as the distribution key for L.

Access plan graph

If you are unsure whether the query refers to views or aliases rather than actual tables, you can display the access plan graph of the query.

In the access plan graph for this example, you can see that TPC.H.LINEITEM and TPC.H.ORDERS are tables because they appear in the access plan graph. Views and aliases are not represented in access plan graphs. See Figure 2.

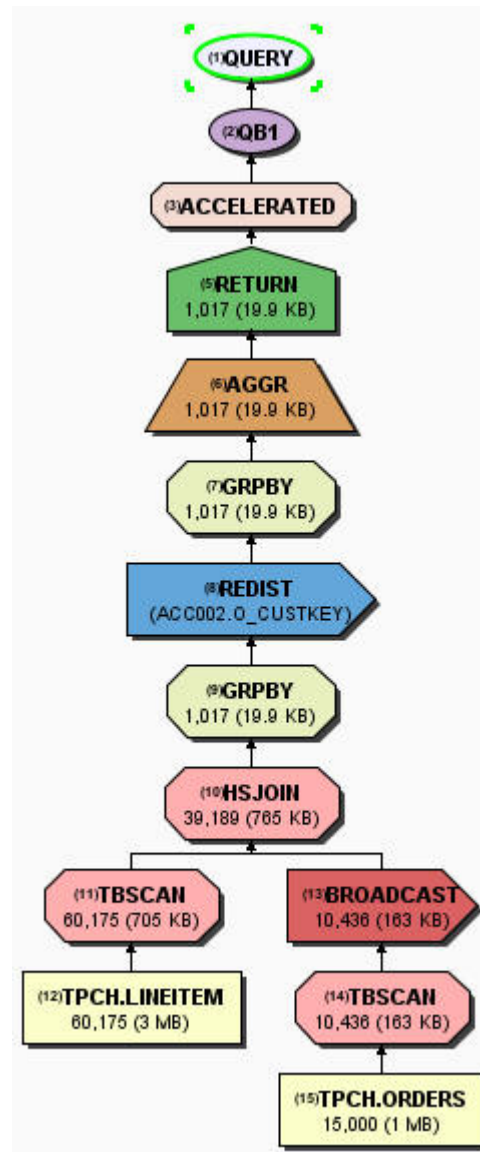


Figure 2. Access plan graph of the query in the example

Distribution keys

A distribution key determines how tables rows are distributed among the available working nodes. IBM Db2 Analytics Accelerator for z/OS selects a suitable distribution key automatically when tables are loaded. The distribution varies between the two extremes: an even distribution (called random distribution) without the use of a key and a maximally skewed distribution that uses a distribution key.

Random distribution

A distribution that is based on random numbers, which serve as input for a distribution algorithm. The output of this algorithm are numbers that identify data slices. Hence the rows to be processed can be assigned to certain data slices, which can be seen as chunks of data to be processed by the individual working nodes. The final result is a more or less even distribution across the working nodes.

Use of a distribution key (hash partitioning)

The accelerator uses a hash function on the key columns to determine the working node that receives a table row. Rows with equal hash values are distributed to the same working node.

Impact of the distribution key on the query performance

Parallel processing works best if the table rows to be processed are evenly distributed across the worker nodes. If the rows of a table that is referenced by a query are unevenly distributed, the worker nodes with fewer rows have to wait for the node with the biggest number of rows to finish processing. However, if your queries involve joins of big tables (100 million to billions of rows), and the rows of these accelerator-shadow tables are scattered across the working nodes even though they are evenly distributed in terms of quantity, you can easily spend much more time on data redistributions and broadcasts than you actually gain by the advantage of an even distribution.

Impact of distribution key on join performance

When two tables are joined, the query runs fastest if the matched rows of both accelerator-shadow tables reside on the same working node (collocated join). A collocated join is instigated by selecting the join columns as the distribution key for both tables.

However, if an accelerator-shadow table F1 is joined with two or more other tables D1, ... D_n using different join keys, you can only locate one of the pairs (F1-D1, F1-D2, ..., F1-D_n) on the same node because you can only select one of the join columns in table F1 as the distribution key for F1. In this case, the product will colocate the biggest tables.

Best practices for selecting distribution keys

Follow these guidelines when using distribution keys:

- For fact tables and the largest dimension tables, specify a distribution key. For all other tables, start with the default (random distribution).
- Prefer single-column distribution keys to multiple-column keys.
- Select a column as the distribution key that distributes the table rows as evenly as possible across the worker nodes, that is, select primary keys or columns with a unique index. Do not specify columns that only have a small number of distinct values. For example, Boolean values like *male* and *female* would place the table rows on two of the worker nodes only.
- When joining two large tables, try to colocate the tables by specifying the join columns as the distribution key for both tables. Note that to instigate a collocated join, the data types of the join columns in both tables must match.
- In general, do not specify a date, time, or timestamp column as the distribution key. Such keys would also distribute the table rows evenly across the worker nodes, but many queries restrict the number of result rows by a range predicate

on the date, time, or timestamp column. For example, a query retrieving revenue figures from last month only would be processed on a single worker node if a column MONTH was selected as the distribution key because rows with equal values in this column would all be assigned to the same data slice.

- Although adverse to the goal of faster query acceleration, a distribution key on the primary key or unique constraint of a table helps if you must update tables continually. Therefore, decide what is more important to you: query performance or the latency of incremental updates. The accelerator automatically picks a distribution key when you enable incremental updates for a table unless you have defined a distribution key by yourself.

Tables states

A table can go through various states while it is being processed by IBM Db2 Analytics Accelerator for z/OS. The meaning of these states is explained here. In addition, it is shown which operation you can run on a table in a particular state. Finally, a diagram shows the possible shifts from one state to another (state transitions).

Table states are displayed in IBM Db2 Analytics Accelerator Studio, in the Accelerator view for regular accelerators (**Tables** section, **Last Load** column). A table can be in one of the following states:

InitialLoadPending

State after the definition of an accelerator-shadow table on the accelerator, which indicates that the table must be loaded with data.

InitialLoadInProgress

State that indicates the loading of an empty accelerator-shadow table or the reloading of an accelerator-shadow table that was in the *Error* state.

Loaded

The load process has been finished.

SchemaModificationInProgress

The schema of a table is being modified. This happens when you change distribution key or organizing key columns, or if background jobs create foreign-key relationships with columns of the table.

UpdateInProgress

The data of the table is being updated. For example, an accelerator-shadow table is in this state if a reload has been enforced by various circumstances.

ReplicationInProgress

The table data is currently being synchronized by the incremental update function.

Error State that indicates a processing error, after which an accelerator-shadow table is reset to the *InitialLoadPending* state. That is, a reload is required.

Possible operations

Table 1 shows which operations you can run on a table in a particular state.

Table 1. Possible operations on a table depending on its state

Table state	Enable or disable query acceleration	Load or update	Schema modification	Remove
<i>InitialLoadPending</i>	No	Yes. The entire accelerator-shadow table is loaded or reloaded. Partition lists are ignored.	Yes	Yes

Table 1. Possible operations on a table depending on its state (continued)

Table state	Enable or disable query acceleration	Load or update	Schema modification	Remove
<i>InitialLoadInProgress</i>	No	No	No	Yes
<i>Loaded</i>	Yes	Yes	Yes	Yes
<i>SchemaModificationInProgress</i>	Yes	No	No	Yes
<i>UpdateInProgress</i>	Yes	No	No	Yes
<i>ReplicationInProgress</i>	Yes	Yes	No	Yes
<i>Error</i>	No	Yes. The entire accelerator-shadow table is loaded or reloaded. Partition lists are ignored.	Yes	Yes

State transitions

Figure 3 on page 11 shows the state transitions that are possible.

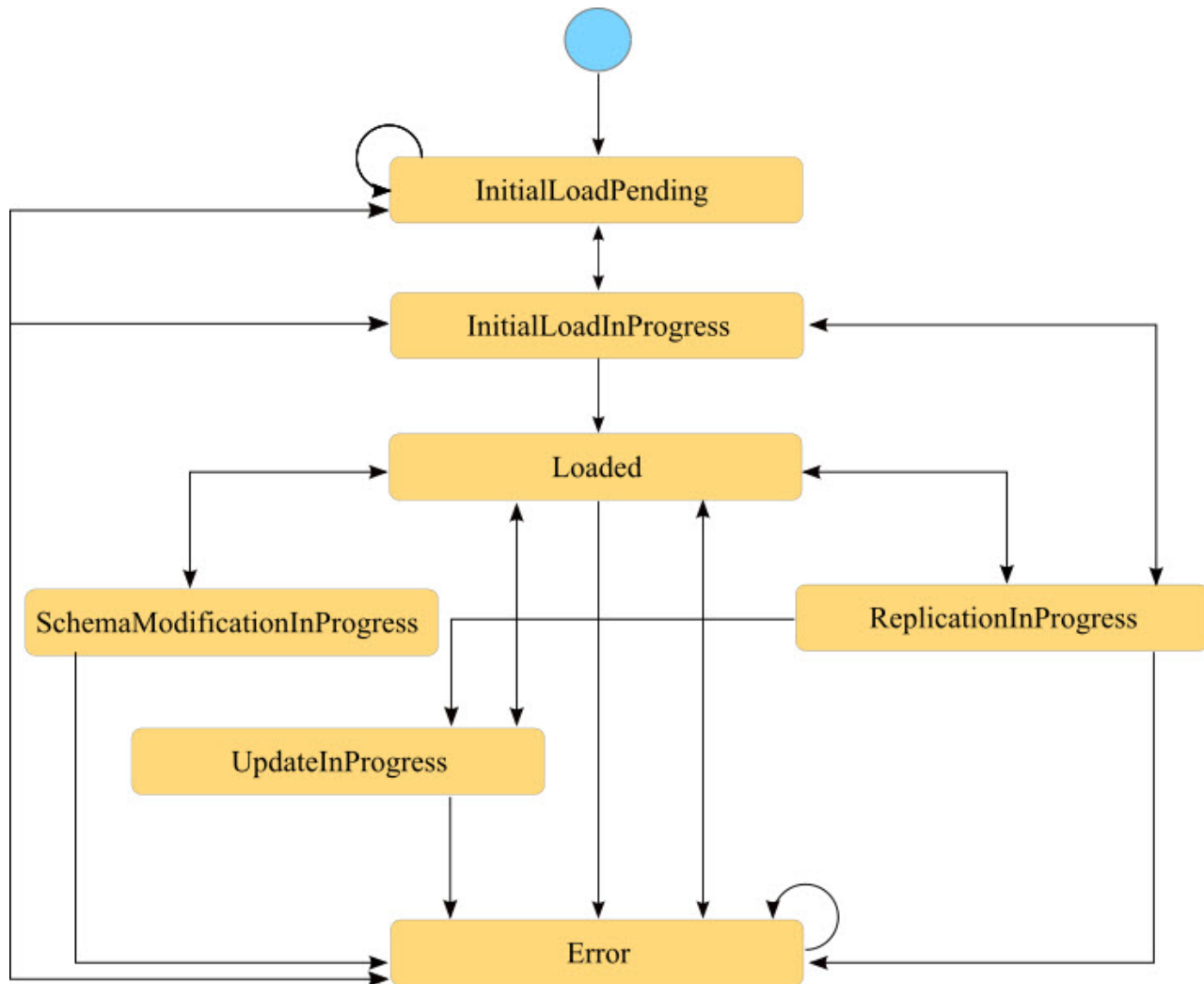


Figure 3. Possible table state transitions

Related concepts:

“Accelerator view” on page 64

Accelerator-only tables and in-database transformation

Accelerator-only tables are tables that do not originate from Db2 base tables. Their data exists only on an accelerator. Data in accelerator-only tables can only be modified by data-manipulation (DML) statements (INSERT, UPDATE, and DELETE). With accelerator-only tables, you can do in-database transformation (IDT) in a more efficient way.

Queries and data-manipulation statements against accelerator-only tables are always routed to an accelerator. If a SQL statement does not satisfy the conditions for acceleration, the statement fails. Furthermore, DML statements that are run on an accelerator can only manipulate data in accelerator-only tables.

Accelerator-only tables are useful for statistics and analytic tools that employ temporary data for reports because the high velocity of execution allows these tools to gather all required data quickly. Since the data in these tables can be

modified fast, they are also ideal for data-preparation tasks that must be completed before the data can be used for predictive modeling.

Use accelerator-only tables if you want to accelerate queries that access previously calculated, interim result sets. To take full advantage of the product's high-speed capabilities, store all your interim results in accelerator-only tables. This allows subsequent queries or data transformations to process all relevant data on the accelerator.

To create an accelerator-only table, you submit a CREATE TABLE statement in Db2 that uses the IN ACCELERATOR <acceleratorname> clause. Accelerator-only tables are registered in the Db2 catalog with a type of 'D'.

Accelerator-only tables cannot be loaded by the SYSPROC.ACCEL_LOAD_TABLES stored procedure or by the corresponding function in IBM Db2 Analytics Accelerator Studio. You have to use INSERT statements. These can be submitted from interactive client interfaces (SPUFI, IBM Data Studio), or from client application programs using ODBC, JDBC or static SQL. It is possible to transfer many rows in a single batch or transaction (multi-row insert). However, the multi-row insert support is limited to client application programs and cannot be used interactively.

Important:

The use of accelerator-only tables brings extra workloads to your accelerators that compete for the same resources, such as CPU, disk, and system memory. This might impact the stability of your production environment.

Conventional in-database transformation versus in-database transformation with accelerator-only tables

A conventional approach to in-database transformation might consist of the following steps:

1. CREATE TABLE T1 ...;
2. INSERT INTO T1 SELECT ... FROM T0 ...;
3. SELECT ... FROM T1 ...;
4. DROP TABLE T1;

Step 1 creates an interim table. Interim results that were calculated on the basis of the data in T0 are stored in table T1 (see step 2). In step 3, a second query reads the interim table T1 to calculate the final result. In step 4, the interim table T1 is dropped because it is no longer needed.

To accelerate the steps above, create table T1 in step 1 as an accelerator-only table. This accelerates the final (second) query and avoids the movement of data in Db2, which occupies considerable processing resources if T1 is large. If T1 is an accelerator-only table, the INSERT statement in step 2 is run on the accelerator rather than in Db2. You also omit the duplication of table T1 in Db2 because you do not need to run IBM Db2 Analytics Accelerator stored procedures to define, load, and enable table T1 on the accelerator. This is a big advantage because many extract-transform-and-load (ETL) tools that generate data transformation statements automatically do not support stored procedure calls. Using accelerator-only tables, you have to modify the statements generated by these tools only slightly to end up with accelerated in-database transformation steps that do not require stored procedure calls.

Before running the following modified steps, you must define, load and enable acceleration for table T0 on the accelerator by using the IBM Db2 Analytics Accelerator stored procedures or the corresponding functions in IBM Db2 Analytics Accelerator Studio. This is just a one-time preparation step. Then you continue with the following steps:

1. CREATE TABLE Q1.T1 ... IN ACCELERATOR ... IN DATABASE ...;
2. INSERT INTO Q1.T1 SELECT ... FROM T0 ...;
3. SELECT ... FROM Q1.T1 ...;
4. DROP TABLE Q1.T1;

Notes:

1. Depending on your situation, you might want to keep the accelerator-only table for later use and run step 4 at a later time.
2. In contrast to temporary tables, accelerator-only tables are much more like normal Db2 tables. It is therefore recommended that you create these tables in an authorized database by using a CREATE TABLE statement with the IN DATABASE clause. Otherwise, accelerator-only tables are created in an implicit database, just like normal Db2 tables. You might also want to add an explicit schema qualifier to the table name.

The following examples consist of extended data transformations that use more than one interim table. They show how to change conventional Db2 statements to arrive at an accelerated solution to in-database-transformation. The left column shows the conventional approach. The right column shows the in-database-transformation steps that include accelerator-only tables.

Example 1

The Db2 tables T1, T2 and T3 already exist. The conventional approach uses two declared global temporary tables to store the interim results.

Table 2. Example 1: Declared global temporary tables versus accelerator-only tables

	Conventional approach (Db2)	Accelerator-only tables
One-time preparation	N/A	Define and load the tables T1, T2 and T3 on accelerator ACCEL1 by using stored procedures or the corresponding functions in IBM Db2 Analytics Accelerator Studio.

Table 2. Example 1: Declared global temporary tables versus accelerator-only tables (continued)

	Conventional approach (Db2)	Accelerator-only tables
Step 2	<pre> DECLARE GLOBAL TEMPORARY TABLE SESSION.TEMP1 (C11 CHAR(2), ...) ON COMMIT PRESERVE ROWS; INSERT INTO SESSION.TEMP1 SELECT ... FROM T1 ...; DECLARE GLOBAL TEMPORARY TABLE SESSION.TEMP2 (C21 CHAR(2), ...) ON COMMIT PRESERVE ROWS; INSERT INTO SESSION.TEMP2 SELECT ... FROM T2 ...; SELECT ... FROM FROM SESSION.TEMP1 ... SESSION.TEMP2 ... T3 ...; DROP TABLE SESSION.TEMP1; DROP TABLE SESSION.TEMP2; </pre>	<pre> SET CURRENT QUERY ACCELERATION = ALL; CREATE DATABASE DBTEMP; CREATE TABLE TEMP1 (C11 CHAR(2), ...) IN ACCELERATOR ACCEL1 IN DATABASE DBTEMP; INSERT INTO TEMP1 SELECT ... FROM T1 ...; CREATE TABLE TEMP2 (C21 CHAR(2), ...) IN ACCELERATOR ACCEL1 IN DATABASE DBTEMP; INSERT INTO TEMP2 SELECT ... FROM T2 ...; SELECT ... FROM FROM TEMP1 ... TEMP2 ... T3 ...; DROP TABLE TEMP1; DROP TABLE TEMP2; DROP DATABASE DBTEMP; </pre>

Example 2

The Db2 tables T1 and T2 already exist. The final transformation result is stored in the existing Db2 table T3. The conventional approach uses two declared global temporary tables to store the interim transformation results.

Table 3. Example 2: Declared global temporary tables versus accelerator-only tables

	Conventional approach (Db2)	Accelerator-only tables
One-time preparation	N/A	Define and load the tables T1 and T2 on accelerator ACCEL1 by using stored procedures or the corresponding functions in IBM Db2 Analytics Accelerator Studio.

Table 3. Example 2: Declared global temporary tables versus accelerator-only tables (continued)

	Conventional approach (Db2)	Accelerator-only tables
Step 2	<pre> DECLARE GLOBAL TEMPORARY TABLE SESSION.TEMP1 (C11 CHAR(2), ...) ON COMMIT PRESERVE ROWS; INSERT INTO SESSION.TEMP1 SELECT ... FROM T1 ...; DECLARE GLOBAL TEMPORARY TABLE SESSION.TEMP2 (C21 CHAR(2), ...) ON COMMIT PRESERVE ROWS; INSERT INTO SESSION.TEMP2 SELECT ... FROM SESSION.TEMP1 ...; INSERT INTO T3 SELECT ... FROM FROM SESSION.TEMP2 ... T2 ...; DROP TABLE SESSION.TEMP1; DROP TABLE SESSION.TEMP2; </pre>	<pre> SET CURRENT QUERY ACCELERATION = ALL; CREATE DATABASE DBTEMP; CREATE TABLE TEMP1 (C11 CHAR(2), ...) IN ACCELERATOR ACCEL1 IN DATABASE DBTEMP; INSERT INTO TEMP1 SELECT ... FROM T1 ...; CREATE TABLE TEMP2 (C21 CHAR(2), ...) IN ACCELERATOR ACCEL1 IN DATABASE DBTEMP; INSERT INTO TEMP2 SELECT ... FROM TEMP1 ...; INSERT INTO T3 SELECT ... FROM FROM TEMP1 ... T2 ...; DROP TABLE TEMP1; DROP TABLE TEMP2; DROP DATABASE DBTEMP; </pre>

Notes:

- Bear in mind that alternative constructs, such as declared global temporary tables, which can exist in Db2, but not on an accelerator, cannot be accelerated by IBM Db2 Analytics Accelerator.
- It is not possible to create image copies of accelerator-only tables because these tables do not hold any data in Db2. You can only recover the data from their original sources, which is why accelerator-only tables should be used only for temporary data.
- An accelerator-only table can reside on just a single accelerator. Copies on other accelerators cannot exist (no high availability).
- For information on how to create an accelerator-only table using the CREATE TABLE command or the restrictions governing the use of these tables, follow the **Related information** link at the end of this topic.
- When you submit a CREATE TABLE statement to create an accelerator-only table, the statement is committed in Db2 for z/OS before the corresponding action is completed on the accelerator. That is, there is a brief period during which the table is listed in the Db2 catalog, but not in the accelerator catalog. If you run the SYSPROC.ACCEL_GET_TABLES_INFO stored procedure during that time, the integrity status ViolatedOnAccelerator is reported for that table in the result set. (In IBM Db2 Analytics Accelerator Studio, you see the error icon next to the table name in the **Tables** section of the Accelerator view. If you move the mouse pointer over the icon, the integrity status is displayed in a fly-over help window.)

Related concepts:

“Configuring Db2 for z/OS for the acceleration of dynamic SQL queries” on page 17

“Configuring Db2 for z/OS for the acceleration of static SQL queries” on page 18

Related reference:

“Restrictions of accelerator-only tables” on page 47

Appendix A, “Impact of special register, ZPARM, or bind option settings on accelerator-only tables,” on page 159

Related information:



http://www.ibm.com/support/knowledgecenter/SSEPEK_11.0.0/com.ibm.db2z11.doc.sqlref/src/tpc/db2z_sql_createtable.dita?lang=en

Workload balancing

Workload balancing can be described as follows: If more than one accelerator is connected to a Db2 subsystem, and the tables that are referenced in a query are defined on more than one of these accelerators, IBM Db2 Analytics Accelerator for z/OS automatically picks the accelerator with the lowest utilization for processing.

Product behavior

Workload balancing is possible between both types of version 7.1.0 accelerators: accelerators on an IBM Integrated Analytics System and Db2 Analytics Accelerator on Z accelerators. However, workload balancing does not occur if different product versions are used. For example, if a version 5.1.0 and a version 7.1.0 accelerator are attached to the same Db2 subsystem, IBM Db2 Analytics Accelerator will always favor the version 7.1.0 accelerator.

IBM Db2 Analytics Accelerator for z/OS system table entries

When you select tables for query acceleration, information about these tables is added to the IBM Db2 Analytics Accelerator for z/OS system tables in the Db2 for z/OS catalog of the connected subsystem. Each entry (system table row) serves as a link between Db2 and the table on the accelerator. An entry contains, among other information, the table name.

Chapter 3. Conditions for query routing to an accelerator

Queries can be routed to an accelerator only if certain conditions are met. There are also adverse conditions that prohibit query acceleration. Both types of conditions are discussed in this section.

A query can be routed to an accelerator if the following general conditions are met:

1. The accelerator is in an operational mode and has been started.
2. Db2 for z/OS has been configured so that the query will be routed to a connected accelerator provided that this query meets all other conditions.
3. The accelerator supports the SQL expressions that are used in the query.
4. All data that is referenced by the query, such as the contents of tables and views, is available on one and the same accelerator. To satisfy this condition, the accelerator must have been loaded with the data in question.

Related information:



DB2 11 for z/OS: SYSIBM.DSN_PROFILE_ATTRIBUTES

Configuring Db2 for z/OS for the acceleration of dynamic SQL queries

To make Db2 for z/OS route dynamic SQL queries to an accelerator, you must enable acceleration by using the CURRENT QUERY ACCELERATION special register. Alternatively, you can set the QUERY_ACCELERATION ZPARM in Db2 for z/OS. This also applies to queries against accelerator-only tables.

The value of the QUERY_ACCELERATION ZPARM provides the default setting for the CURRENT QUERY ACCELERATION special register. Both, the ZPARM and the special register accept the following values:

1 (NONE)

No routing of dynamic SQL queries to an accelerator. Queries will be processed by Db2 for z/OS only (inhouse query processing).

2 (ENABLE)

A dynamic SQL query will be routed to an accelerator if it fulfills all required conditions. An incoming query is tested against a set of heuristics, which include the table size and a response time estimate based on cost information from the SYSIBM.DSN_PROFILE_ATTRIBUTES table. Both tests ensure that a query will only be routed to an accelerator if the query can be expected to run faster than in Db2 for z/OS. However, if an error occurs while the query is being processed by the accelerator, Db2 for z/OS will return a negative SQLCODE to the application and query processing will stop.

3 (ENABLE WITH FAILBACK)

Dynamic queries are accelerated only if Db2 for z/OS determines that it is advantageous to do so. If an accelerator returns an error during the PREPARE phase or when first opening (OPEN) the query, the query is processed by Db2 for z/OS rather than sent to the accelerator. If the accelerator returns an error during a FETCH operation or a subsequent OPEN operation, Db2 for z/OS returns an error to the user and the query ends abnormally.

4 (ELIGIBLE)

Dynamic queries are accelerated if they are eligible for acceleration. Db2 for z/OS does not use cost information to determine whether to accelerate the

queries. Queries that are not eligible for acceleration are executed by Db2 for z/OS. If an accelerator fails while a query is running, or if the accelerator returns an error, Db2 for z/OS returns a negative SQL code to the application.

5 (ALL)

A dynamic query will always be routed to an accelerator, no matter if it fulfills the conditions or not. If processing cannot start or continue because an incoming query fails to fulfill all the conditions for accelerated query processing, Db2 for z/OS returns a negative SQLCODE to the application and query processing ends abruptly. That is, the query will not be processed at all.

A query against an accelerator-only table can only succeed if the CURRENT QUERY ACCELERATION special register is set to a value other than NONE. There is no difference to accelerator-shadow tables in this respect. However, this does not affect data-definition (DDL) statements, such as CREATE TABLE or DROP TABLE. DDL statements are executed regardless of the special register or ZPRAM setting.

To enable query routing to an accelerator on the Db2 side, you issue the following SQL statement:

```
SET CURRENT QUERY ACCELERATION = ENABLE
```

FAILBACK processing

If you specify SET CURRENT QUERY ACCELERATION = ENABLE WITH FAILBACK, the query is returned to Db2 for processing if an error occurs during the PREPARE phase. This is the time before the query is actually routed to the accelerator, and during which the heuristics are tested. A query is also returned if an error was caused by an accelerator failure, a network failure, a network connection timeout, or a similar error. Errors immediately following the initial OPEN request of a query usually fall into this category.

Restrictions:

FAILBACK processes are not started if one of the following conditions applies:

- Query results have already been returned. This holds true even if the results are not complete.
- Query routing to an accelerator fails although the same query could be routed successfully before.
- Queries do not qualify for routing to an accelerator. Such queries are prepared and executed in Db2, regardless of the setting of the CURRENT QUERY ACCELERATION special register.

Settings resulting from the preparation phase are stored in the Db2 Dynamic Statement Cache. This allows the query engine to skip the preparation phase when a query comes in for the second time. However, this is also often the reason for not returning a query to Db2 for z/OS in case of an error. If you assume that this has happened, invalidate the Dynamic Statement Cache and rerun the query.

Configuring Db2 for z/OS for the acceleration of static SQL queries

To accelerate static SQL queries, use the QUERYACCELERATION bind option for Db2 packages.

Important:

- The acceleration bind option QUERYACCELERATION applies to cursor queries and the SELECT portion of the SQL INSERT from SELECT statement. The SQL SELECT INTO statement can also be bound for acceleration.
- The functions and behavior of the acceleration bind option QUERYACCELERATION for static queries is the same as the comparable existing acceleration special register CURRENT QUERY ACCELERATION that is used for dynamic query acceleration, except that the Db2 decision to accelerate a static query and bind it for acceleration occurs at BIND or REBIND PACKAGE time, and not at query execution time when the application is run. The special registers do not affect the binding or the execution of static queries; instead, the new bind option is used to determine acceleration behavior for static queries.
- The accelerator need not be started at BIND or REBIND PACKAGE time for the static query to be bound for acceleration. However, you must start the accelerator before you run the application that executes the static query.

Settings of the QUERYACCELERATION bind option

1 (NONE)

No static query in the application is bound for acceleration or will be accelerated when the application is run.

2 (ENABLE)

A static query is bound for acceleration if it satisfies the acceleration criteria, including the cost and heuristics criteria. The query is routed to an accelerator when the application is run. If the static query does not satisfy the acceleration criteria, the query is bound for an execution in Db2.

In the following cases, Db2 returns a negative SQL code for the query to the application:

- If a failure occurs while executing the accelerated static query on the accelerator
- If the accelerator returns an error for the query
- If the accelerator is not started and Db2 cannot route the static query to the accelerator for execution

3 (ENABLEWITHFAILBACK)

Same as ENABLE, except for the following cases:

- The accelerator returns an error.
- The accelerator fails on the first OPEN request for the accelerated static query.
- The accelerator is not active or started when the first OPEN occurs.

In either of these cases, Db2 does not return a negative SQL code (accelerator failure) to the application, but executes a "failback" operation. That is, a temporary statement-level incremental bind of the query is performed, and the query is run in Db2. No "failback" to Db2 is possible after a successful OPEN request.

The bind option is supported for:

- BIND, BIND COPY, and REBIND PACKAGE (local and remote)
- BIND DEPLOY (for SQLPL procedures)
- ALTER PROCEDURE and ALTER FUNCTION for native SQLPL procedures and SQLPL scalar functions
- CREATE PROCEDURE and CREATE FUNCTION for native SQLPL procedures and SQLPL scalar functions

The bind option is not supported for:

- ALTER TRIGGER

- CREATE TRIGGER
- REBIND TRIGGER PACKAGE

Important: The default value of the bind option is *option not specified*. The default is not taken from the comparable Db2 installation system parameter (ZPARMs) QUERY_ACCELERATION.

- The default value for both new bind options is *option not specified*. The default for the bind options is not taken from the comparable DB2® installation system parameters (ZPARMs) QUERY_ACCELERATION and GET_ACCEL_ARCHIVE.
- For INSERT, UPDATE, DELETE, and SELECT statements that address accelerator-only tables, the QUERYACCELERATION bind option must be set to ENABLE or ELIGIBLE. If you specify ALL, the operation fails and an SQLCODE is returned.
- If you submit a CREATE, DROP, or COMMIT statement against an accelerator-only table together with an INSERT, UPDATE, DELETE, or SELECT statement (against the same table) in the same thread, the table does not come into existence during the bind phase. However, the current implementation requires that all referenced tables are defined and enabled for query acceleration during the bind phase.

As a workaround, you can use the VALIDATE(RUN) bind option, which binds the INSERT, UPDATE, DELETE, or SELECT statement incrementally at execution time. However, this usually does not give you the optimum performance. Therefore, consider running the static CREATE statement in a separate thread before running the INSERT, UPDATE, DELETE, or SELECT statement.

The bind option can also be used for the acceleration of dynamic queries. If specified, it sets the initial value in the same manner as the comparable special register CURRENT QUERY_ACCELERATION when the application is run, provided that the special register has not already been set explicitly before the invocation of that application. The value of the bind option is then used as the special register value for dynamic queries in that application, so that you do not have to add an explicit SET statement to the application for the acceleration special register.

Mind that you cannot use the bind options for dynamic queries under the following conditions:

- Dynamic and static queries are started from the same application or script.
- Both types of queries work on the same or share a number of tables.
- You want to accelerate the dynamic queries, but not the static queries.

In this case, you must use explicit SET statements to set the special registers in your application or script. This is because the bind options would also lead to an acceleration of the static queries. Special registers, on the other hand, have no effect on static queries.

1. Db2 installation system parameter QUERY_ACCELERATION
2. Bind option value if specified (QUERYACCELERATION)
3. Explicit SET statement for the special register

As with the acceleration of dynamic queries and the use of the acceleration special register, the use of the bind option requires you to set ACCEL (Db2 ZPARM) to a value other than NONE to enable the Db2 subsystem for query routing to an accelerator.

Supported queries

A set of conditions must be fulfilled before an accelerator can handle a query.

The following conditions must apply:

- The principal expression used by the query is a SELECT statement or an “INSERT FROM SELECT” statement. In this case, only the SELECT part of the query is accelerated.

Note: To enable support for “INSERT FROM SELECT” statements, set the QUERY_ACCEL_OPTIONS parameter (a ZPARM) to the value 2. For more information, see *Installing Db2 libraries with IBM Db2 Analytics Accelerator for z/OS support* in the *IBM Db2 Analytics Accelerator for z/OS: Installation Guide*.

- The query is defined as read-only. The criteria for read-only queries are documented in the subsection *Read-only cursors* of the topic *DECLARE CURSOR* in the Db2 10 for z/OS information center or the IBM Knowledge Center. A link is provided under **Related information** at the end of this topic.

Tip: If a query is not considered a read-only query by Db2 for z/OS, but you want the query to run on an accelerator, you can add the FOR READ ONLY clause to the SQL of the query.

- If the query is submitted by an application and introduces a cursor definition by means of the DECLARE CURSOR statement, then this cursor must be a read-only cursor that is neither a scrollable cursor, nor a rowset cursor that is declared by a remote application, such as an SQL statement submitted from the Db2 CLI for Windows. However, read-only rowset cursors are supported if they are declared by local applications (such as SPUFI or local stored procedure calls) in the following contexts:

- As part of a DELCARE CURSOR statement, for example:

```
EXEC SQL DECLARE DYN1 CURSOR WITH ROWSET POSITIONING FOR STMT1;
```

- As an attribute in a PREPARE statement, for example:

```
STMTSTR = 'SELECT COLI2,COLI8,COLI4,COLI2'
          | 'FROM NTZTB01'
          | 'WHERE 1=1'
          | 'ORDER BY COLI2'
          | 'FOR FETCH ONLY';
```

```
ATTRSTR = 'WITH ROWSET POSITIONING';
```

```
EXEC SQL PREPARE STMT1 ATTRIBUTES :ATTRSTR FROM :STMTSTR;
```

- In a rowset-positioned FETCH operation, for example:

```
DCL AINT2A(07) BIN FIXED(15);
DCL AINT2B(07) BIN FIXED(15);
DCL AINT4B(07) BIN FIXED(31);
DCL AINT8B(07) BIN FIXED(63);
```

```
EXEC SQL FETCH NEXT ROWSET FROM DYN1 FOR :NROWS ROWS INTO
       :AINT2A, :AINT8B, :AINT4B, :AINT2B;
```

- When #SET MULT_FETCH is used as an instruction for the DSNTEP4 program, for example:

```
//SYSIN DD *
--#SET MULT_FETCH 7
SET CURRENT_QUERY ACCELERATION = ALL;
SELECT COLI2, COLI8, COLI4, COLI2 FROM ADMF001.NTZTB01
WHERE 1=1 FOR FETCH ONLY;
```

- Plans for query processing must use packages rather than database request modules (DBRMs).
- Based on its own set of heuristics, Db2 classifies an acceleration of the query as favorable. That is, Db2 “expects” a much shorter query response time by routing the query to an accelerator.
- DRDA® is used for the transmission of the query from Db2 to the accelerator. Therefore, DRDA must be configured and started from any Db2 subsystem connected to the accelerator.

Related information:



DECLARE CURSOR

Supported data types

See this overview of supported Db2 data types, corresponding dashDB® types, ranges, and hints at restrictions. The section also lists the unsupported data types at the end.

Numeric data types

Table 4. Numeric data types

Db2 type	dashDB type	Range Db2	Range dashDB	Comment
SMALLINT	SMALLINT (INT2)	-32768 to +32767	-32768 to +32767	
INTEGER	INTEGER (INT, INT4)	-2147483648 to +2147483647	-2147483648 to +2147483647	
BIGINT	BIGINT (INT8)	-9223372036854775808 to +9223372036854775807	-9223372036854775808 to +9223372036854775807	
REAL	REAL	$\sim -7.2 * 10^{+75}$ to $-5.4 * 10^{-79}$ or $\sim +5.4 * 10^{-79}$ to $+7.2 * 10^{+75}$	$\sim -3.4 * 10^{+38}$ to $-1.18 * 10^{-38}$ or $\sim 1.18 * 10^{-38}$ to $+3.4 * 10^{+38}$	For details, see “Differences in floating-point error tolerances” on page 54
DECFLOAT	DECFLOAT	Precision of 34 or 16 digits (default is 34) 10^{-6143} to 10^{+6144} or 10^{-383} to 10^{+384}	Precision of 34 or 16 digits (default is 34) 10^{-6143} to 10^{+6144} or 10^{-383} to 10^{+384}	See Note under thus table.
DOUBLE	DOUBLE	$\sim -7.2 * 10^{+75}$ to $-5.4 * 10^{-79}$ or $\sim +5.4 * 10^{-79}$ to $+7.2 * 10^{+75}$	$\sim -1.79 * 10^{+308}$ to $-3.40 * 10^{-308}$ or $\sim +3.4 * 10^{-308}$ to $+1.79 * 10^{+308}$	For details, see “Differences in floating-point error tolerances” on page 54
DECIMAL(<i>precision</i> , <i>scale</i>)	NUMERIC(<i>precision</i> , <i>scale</i>)	Max. precision 31	Max. precision 31	

| **Note on DECFLOAT:** During tests, the extreme values at either end of the range could not be loaded into accelerator-only tables, which hints at a general problem with very low and very high values in this context. However, there was no problem loading the extreme values into regular accelerator-shadow tables-

| See also the entry for the ROUND function in Table 9 on page 27 .

Date, time, and timestamp data types

Table 5. Date, time, and timestamp data types

Db2 type	dashDB type	Range Db2	Range dashDB	Comment
DATE	DATE	0001-01-01 to 9999-12-31	0001-01-01 to 9999-12-31	The use of DATE = LOCAL is not supported.
TIME	TIME	00:00:00.000000000000 to 24:00:00.000000000000	00:00:00.000000000000 to 24:00:00.000000000000	The use of TIME = LOCAL is not supported. The dashDB database on the accelerator supports the same precision as Db2 for z/OS (up to 12 fractional digits). A truncation of values is therefore not necessary.
TIMESTAMP	TIMESTAMP	Date portion: 0001-01-01 to 9999-12-31 Time portion: 00:00:00.000000000000 to 24:00:00.000000000000	Date portion: 0001-01-01 to 9999-12-31 Time portion: 00:00:00.000000000000 to 24:00:00.000000000000	Some timestamp formats are supported by Db2 for z/OS, but not by dashDB: <ul style="list-style-type: none"> • 2004-02-15T11.34.56 • 20040215113456 • 2004-02-15 11.34.56 Important: If values are formatted in any of the formats shown above and stored as characters, Db2 for z/OS cannot prevent these values from being sent to an accelerator. Errors will be the result. Timestamps with timezone specifications are not supported, to the effect that: <ul style="list-style-type: none"> • SQL queries that require TIMESTAMP with a timezone specification as the result data type are not routed to an accelerator. • Timestamps with timezone constants are not accepted. • Functions and special registers that are related to timestamps with timezone specifications are not accepted.

String data types

Only the following combinations of coded character set identifiers (CCSIDs) and subtypes are allowed in string expressions:

Table 6. Supported combinations of CCSIDs and subtypes

CCSID	Allowed character subtypes	Comment
ASCII (for example 819; there are numerous others)	SBCS	An ASCII table cannot be added to an accelerator if it is part of a Db2 subsystem with a setting of mixed=yes. This holds true even if the table does not contain any multi-byte (or mixed) data.
EBCDIC (for example 37; there are numerous others)	SBCS MIXED (partially supported) DBCS (graphic, partially supported)	
UNICODE (367, 1208, 1200)	SBCS (367) MIXED (1208) DBCS (1200)	

Table 7. String data types

Db2 type	dashDB type	Supported CCSIDs	Comment
BINARY	BINARY	N/A	
VARBINARY	VARBINARY	N/A	
CHAR FOR SBCS DATA	CHAR	ASCII EBCDIC UNICODE (367)	
VARCHAR FOR SBCS DATA	VARCHAR	ASCII EBCDIC UNICODE (367)	The maximum length in Db2 for z/OS is 32704. In dashDB, it is 32592. A table load fails and message AQT20052E is displayed if a string exceeds the dashDB limit.
CHAR FOR MIXED DATA	CHAR	UNICODE (1208) EBCDIC (partially supported)	
VARCHAR FOR MIXED DATA	VARCHAR	UNICODE (1208) EBCDIC (partially supported)	
CHAR FOR BIT DATA	CHAR FOR BIT DATA	ASCII (single-byte only) EBCDIC UNICODE (1208)	
VARCHAR FOR BIT DATA	VARCHAR FOR BIT DATA	ASCII (single-byte only) EBCDIC UNICODE (1208)	
GRAPHIC	GRAPHIC	UNICODE (1200) EBCDIC	This type is fully supported for EBCDIC and UNICODE tables. Important: If your Db2 subsystem is set to MIXED=NO, and you use the GRAPHIC() function on a single-byte (SBCS) CHAR column, the query will fail when it is routed to an accelerator.
VARGRAPHIC	VARGRAPHIC	UNICODE (1200) EBCDIC	This type is fully supported for EBCDIC and UNICODE tables. The maximum length in Db2 for z/OS is 16352. In dashDB, it is 16296. An accelerator table load operation aborts when a string exceeds the dashDB limit. Important: If your Db2 subsystem is set to MIXED=NO, and you use the GRAPHIC() function on a single-byte (SBCS) VARCHAR column, the query will fail when it is routed to an accelerator.

ROWID data type

The ROWID data type is to uniquely and permanently identify rows in a table. A ROWID value can be used to access a table row directly. The ROWID data type is supported by IBM Db2 Analytics Accelerator. However, consider the following limitation:

Two formats of ROWIDs exist: an *external* format and an *internal* format. The external format consists of the internal format plus several hexadecimal strings. Db2 for z/OS can handle both formats, but stores just the internal format. Since the internal format is stored, this is also what is loaded into the corresponding accelerator-shadow tables.

ROWID columns can be used like any other column, as, for example, in queries like:

```
SELECT <RWID> FROM <table>;
```

, where <RWID> is supposed to be a column containing ROWIDs. In cases like this, Db2 for z/OS returns the longer, external format. The query would work in the same way if it were accelerated, but the accelerator would return the internal format because it cannot translate or otherwise infer the external format from the internal values.

In addition, the values in ROWID columns can be used in WHERE predicates for direct row access, as in queries like:

```
SELECT * FROM <table> WHERE RWID = ROWID(X'...')
```

Inside the parentheses of the ROWID(X'...') expression, you can use both, the external or the internal format, if the query is processed by Db2 for z/OS.

However, the same query can be accelerated only if the internal format is used, for the same reason as in the previous example: The accelerator "understands" just the internal format.

Notes:

- For some complex queries (including OUTER JOIN expressions and views, for example), Db2 for z/OS requires the external ROWID format. Such queries cannot be accelerated.
- ROWID expressions can also be used in INSERT INTO ... SELECT FROM statements, where just the SELECT portion of the query is accelerated.
- You cannot add columns with the ROWID data type to accelerator-only tables. If you try to include a column definition of that sort in a CREATE TABLE ... IN ACCELERATOR statement, Db2 for z/OS returns SQL Code -4747.
- To use ROWID expressions in accelerated static SQL queries, first rebind the static application or package with the Db2 bind option QUERYACCELERATION. Use a value other than NONE. For more details, see *Configuring Db2 for z/OS for the acceleration of static SQL queries* (link at the end of this topic) and the Db2 for z/OS documentation.

Unsupported Db2 data types

- BLOB
- CLOB
- DBCLOB
- XML

Related concepts:

"Conditions that prevent query routing to an accelerator" on page 39

"Configuring Db2 for z/OS for the acceleration of static SQL queries" on page 18

Supported SQL functions and expressions

See which aggregate functions and scalar functions are supported by IBM Db2 Analytics Accelerator.

Supported aggregate functions

IBM Db2 Analytics Accelerator supports all aggregate functions, except for the XMLAGG function.

The following aggregate functions are supported even if they are used in combination with the DISTINCT or UNIQUE keyword:

- STDDEV
- STDDEV_SAMP
- VARIANCE
- VARIANCE_SAMP

Note: Restrictions exist if you use the DISTINCT or UNIQUE keyword. See Table 9 on page 27.

Supported scalar functions

In addition, IBM Db2 Analytics Accelerator supports a variety of scalar functions and the CAST specification.

If restrictions exist for a function in Table 8, the function is marked with an asterisk(*). The restrictions are described in Table 9 on page 27.

Table 8. Supported scalar functions and specifications

Function name	Function name	Function name	Function name
• ABS	• DEGREES	• MOD	• SUBSTR
• ADD_MONTHS	• DIGITS	• MONTH	• SUBSTRING (see Table 9 on page 27)
• BINARY	• DOUBLE_PRECISION (also DOUBLE, FLOAT)	• MONTHS_BETWEEN	• TIME (see Table 9 on page 27)
• BIGINT	• EXP	• NEXT_DAY	• TIMESTAMP (see Table 9 on page 27)
• BITAND (see Table 9 on page 27)	• EXTRACT (see Table 9 on page 27)	• NEXT_WEEK	• TIMESTAMPDIF
• BITANDNOT	• FLOAT	• NEXT_MONTH	• TIMESTAMP_FORMAT (also TO_DATE; see Table 9 on page 27)
• BITNOT	• FLOOR	• NEXT_QUARTER	• TIMESTAMP_ISO (see Table 9 on page 27)
• BITOR	• GRAPHIC (see Table 9 on page 27)	• NORMALIZE_DECFLOAT	• TO_NUMBER (DECFLOAT_ FORMAT)
• BITXOR	• HOUR (see Table 9 on page 27)	• NULLIF	• TRANSLATE (see Table 9 on page 27)
• CEILING	• HEX (see Table 9 on page 27)	• NVL	• TRIM (also with two arguments; see Table 9 on page 27)
• CAST specification (see Table 9 on page 27)	• IFNULL	• POSITION	• TRUNCATE(see Table 9 on page 27)
• CHAR (see Table 9 on page 27)	• DOUBLE_PRECISION (also DOUBLE, FLOAT)	• POSSTR	• UPPER (UCASE) (see Table 9 on page 27)

Table 8. Supported scalar functions and specifications (continued)

Function name	Function name	Function name	Function name
• CHARACTER_LENGTH	• INTEGER (INT)	• POWER® (see Table 9)	• VALUE
• COALESCE	• JULIAN_DAY (see Table 9)	• QUARTER (see Table 9)	• VARBINARY
• COMPARE_DECFLOAT (see Table 9)	• LAST_DAY	• RADIANS	• VARCHAR
• CONCAT (see Table 9)	• LEFT (see Table 9)	• RAND (see Table 9)	• VARCHAR9 (see Table 9)
• CORR	• LENGTH	• REAL	• VARCHAR_FORMAT (TO_CHAR; see Table 9)
• COVAR	• LN	• REPEAT	• VARGRAPHIC (see Table 9)
• COVAR_SAMP	• LOCATE (see Table 9)	• REPLACE	• WEEK (see Table 9)
• DATE (see Table 9)	• LOCATE_IN_STRING (see Table 9)	• RIGHT (see Table 9)	• WEEK_ISO (see Table 9)
• DAY	• LOG10	• ROUND (see Table 9)	• YEAR
• DAYOFMONTH	• LOG	• ROWID (see Table 9)	
• DAYOFWEEK	• LOWER (LCASE) (see Table 9)	• RPAD	
• DAYOFWEEK_ISO (see Table 9)	• LPAD (see Table 9)	• RTRIM (also with two arguments)	
• DAYOFYEAR (see Table 9)	• LTRIM (also with two arguments)	• SECOND (see Table 9)	
• DAYS	• MAX (GREATEST)	• SIGN 4(see Table 9)	
• DECFLOAT (see Table 9)	• MEDIAN	• SMALLINT	
• DECFLOAT_FORMAT (TO_NUMBER)	• MICROSECOND	• SPACE	
• DECIMAL	• MIDNIGHT_SECONDS (see Table 9)	• SQRT	
• DECODE (also with two arguments)	• MIN (also LEAST)	• STRIP (see Table 9)	
	• MINUTE (see Table 9)	• SMALLINT	

Table 9. Restrictions

Function, keyword, or expression	Restriction
• BITAND	The lowest possible BIGINT value (-9223372036854775808) at any argument position returns a result data type of DECFLOAT(34) rather than BIGINT, as in Db2 for z/OS). For any other valid input value, the result data type is BIGINT.

Table 9. Restrictions (continued)

Function, keyword, or expression	Restriction
<ul style="list-style-type: none"> CAST specification 	<p>Restrictions exist if you want to use CAST expressions to convert string data types and, by specifying the string unit, determine the number of bytes to be used for the representation of characters in the resulting strings:</p> <p>CAST TO CHAR If the string unit is CODEUNITS16 or CODEUNITS32, IBM Db2 Analytics Accelerator might return a different number of trailing blanks than Db2 for z/OS.</p> <p>CAST TO GRAPHIC Allowed string units are CODEUNITS16 and CODEUNITS32. When compared with the output from Db2 for z/OS, IBM Db2 Analytics Accelerator might return a different number of trailing blanks.</p> <p>CAST TO VARGRAPHIC Allowed string units are CODEUNITS16 and CODEUNITS32.</p> <p>See also “Strings as time values and their conversion” on page 45.</p>
<ul style="list-style-type: none"> CHAR 	<ul style="list-style-type: none"> If the string unit is CODEUNITS16 or CODEUNITS32, IBM Db2 Analytics Accelerator might return a different number of trailing blanks than Db2 for z/OS. If a string of space characters is requested in a query, the CHAR function always returns strings with a length of 255, no matter which string length is specified in the query. Suppose, for example, that a table with the name TMP contains strings consisting of (subsequent) space characters. If you run a query like <pre>SELECT CHAR(SPACE(20)) FROM TMP;</pre> you will find blank strings with a length of 255 rather than a length of 20 in the results. If the function is used to retrieve a substring of a multibyte string, so that it will split a multibyte-character, Db2 for z/OS returns the correct byte-code of the split character, whereas IBM Db2 Analytics Accelerator replaces this portion of the byte-code with blanks. You can avoid this by using the <i>length</i> argument (OCTETS, CODEUNITS16, or CODEUNITS32). Suppose, for example, that you run the following query to retrieve the first five bytes of the string MAMA, consisting of full-width Latin capital letters. Each character is represented by three bytes. The mode of representation is UTF-8. <pre>SELECT HEX(CHAR('MAMA',5));</pre> To illustrate the differences in the outputs, the HEX function is used: <p>Result in Db2 for z/OS: EFBCADEFBC</p> <p>This value represents the first letter 'M' (in full) and the first two bytes of the first letter 'A'.</p> <p>Result returned by accelerator: EFBCAD2020</p> <p>This value represents the first letter 'M' (in full) and two blanks instead of the first two bytes of the letter 'A'.</p> <p>You could specify OCTETS as the length argument to avoid this problem: <pre>SELECT HEX(CHAR('MAMA',5,OCTETS));</pre> The result would then be EFBCADEFBC in both cases.</p>

Table 9. Restrictions (continued)

Function, keyword, or expression	Restriction
<ul style="list-style-type: none"> • CONCAT 	<p>If the result data type is VARCHAR, the maximum length of a result string is limited to 4000 characters.</p>
<ul style="list-style-type: none"> • DATE 	<ul style="list-style-type: none"> • Db2 accepts an argument in the format yyyynnn as input for the DATE function. IBM Db2 Analytics Accelerator does not accept this format and therefore issues an error. • If the DATE function is used on numeric expressions, it is converted to DATE(to_date('0000-12-31','YYYY-MM-DD')+trunc(numeric_exp)) during query routing to an accelerator. If the numeric_exp result is a negative number, IBM Db2 Analytics Accelerator adds BC to the date. An error is returned when the converted scalar function is evaluated in Db2.
<ul style="list-style-type: none"> • DAYOFWEEK_ISO • DAYOFYEAR • JULIAN_DAY • MIDNIGHT_SECONDS • QUARTER • TIMESTAMP_ISO • WEEK • WEEK_ISO 	<ul style="list-style-type: none"> • If Db2 for z/OS is configured to use a local DATE format (DATE=LOCAL on DSNTIP4 panel) and a query uses a DATE constant as an argument for any of these functions, the query is not accelerated, and SQLCODE -4742 is returned. • Queries containing the DAYOFWEEK_ISO, DAYOFYEAR, JULIAN_DAY, QUARTER, TIMESTAMP_ISO, or WEEK_ISO function fail with SQLCODE=-443 under the following conditions: <ol style="list-style-type: none"> 1. The query includes the clause FETCH FIRST n ROWS. 2. Any of the functions mentioned before uses string representations of U.S. or European date formats as arguments. <p>The same applies to queries containing MIDNIGHT_SECONDS, except that string representations of European date formats are allowed.</p> • Queries containing the TIMESTAMP_ISO function fail during the attempt to route them to an accelerator if string representations of TIME values are used as arguments.
<ul style="list-style-type: none"> • COMPARE_DECFLOAT 	<p>Since the database engine on the accelerator does not support DECFLOAT constants (see entry for DECFLOAT), the values to be compared are interpreted as floating-point numbers (DOUBLE), which results in a loss of precision for big numbers. This loss of precision might cause a value shift that leads to deviating comparison results. Consider the following cases:</p> <pre>COMPARE_DECFLOAT(0.123456789012345678901E-187, 1.234567890123456789010E-188), COMPARE_DECFLOAT(0.123456789012345678901E-185, 1.234567890123456789010E-188), COMPARE_DECFLOAT(0.123456789012345678901E-187, 1.234567890123456789010E-185)</pre> <p>The output of these comparisons is different, depending on where they are executed:</p> <p>Db2 for z/OS: '2','2','1'</p> <p>IBM Db2 Analytics Accelerator: '0','2','1'</p>

Table 9. Restrictions (continued)

Function, keyword, or expression	Restriction
<ul style="list-style-type: none"> • DECFLOAT 	<ul style="list-style-type: none"> • The database engine on the accelerator does not check, and therefore does not necessarily preserve the scale of the original Db2 for z/OS values. Db2 for z/OS defines the scale as the number of digits to the right of the decimal point. For example, 0.03E3 might be returned as 0.3E2. As long as the value is numerically accurate according to the IEEE754 specification, it is considered to be correct. • Different representations of the same numerical value (for example 1E2, 100, and 10E1) also have slightly different representations in memory and the database engine on the accelerator rarely normalizes these values. If the database is a row store, the values are not normalized at all. If it is a columnar store, normalization might occur more often, especially if a GROUP BY operation is performed on the values. However, you can never rely on the normalization of differently represented values. • The database engine on the accelerator does not support DECFLOAT constants. If the accelerator receives a DECFLOAT constant from Db2 for z/OS, and the value of this constant is within the range of DOUBLE values accepted by the database engine, the value is interpreted as DOUBLE, and the accelerator might return a value with a different precision. If a received value is not within the range of DOUBLE, SQL code -405 or -410 is returned. For example: <ul style="list-style-type: none"> – SQLCODE = -405, ERROR: THE NUMERIC CONSTANT 1.0E-308 CANNOT BE USED AS SPECIFIED BECAUSE IT IS OUT OF RANGE ... – SQLCODE = -410, ERROR: A NUMERIC VALUE 1234.5678901234567890123456E-55 IS TOO LONG, OR IT HAS A VALUE THAT IS NOT WITHIN THE RANGE OF ITS DATA TYPE ... • Db2 for z/OS and the database engine of IBM Db2 Analytics Accelerator compute different DECFLOAT values if one of the operands is Infinity (∞) and none of the other operands are of the type <i>not a number</i> (NaN). Signaling (sNaN) values are also counted as NaN. In the majority of these cases, Db2 for z/OS returns NaN as the result, whereas IBM Db2 Analytics Accelerator returns positive or negative Infinity.
<ul style="list-style-type: none"> • DISTINCT • UNIQUE 	<p>If the DISTINCT or UNIQUE keyword is applied to string values with trailing blanks, IBM Db2 Analytics Accelerator might cut off the trailing blanks from the resulting values. Example:</p> <pre>SELECT DISTINCT N_S_VCHR CONCAT ' ' FROM MATRIX.MBCS_EBCDIC;</pre> <p>The SQL code specifies that a space character or blank is to be added at the end of each value retrieved. Now see some of the rows returned by this query, once by Db2 for z/OS, and once by IBM Db2 Analytics Accelerator:</p> <p>Db2 for z/OS:</p> <ol style="list-style-type: none"> 1. [] (four blanks) 2. [HIJ] (two blanks at end of string) 3. [! "\$%&\ /] (one blank at end of string) <p>IBM Db2 Analytics Accelerator:</p> <ol style="list-style-type: none"> 1. [] (empty string) 2. [HIJ] (no blanks at end of string) 3. [! "\$%&\ /] (no blanks at end of string)
<ul style="list-style-type: none"> • GRAPHIC 	<p>If the string unit is CODEUNITS16 or CODEUNITS32, IBM Db2 Analytics Accelerator might return a different number of trailing blanks than Db2 for z/OS.</p>

Table 9. Restrictions (continued)

Function, keyword, or expression	Restriction
<ul style="list-style-type: none"> • HEX 	This function returns a CHAR value in Db2 for z/OS. If the query is accelerated by IBM Db2 Analytics Accelerator, the function returns a VARCHAR value.
<ul style="list-style-type: none"> • LEFT • LPAD • RIGHT 	Result deviations can occur if input strings contain multibyte or graphic EBCDIC characters.
<ul style="list-style-type: none"> • LOCATE • LOCATE_IN_STRING 	<p>Result deviations can occur under the following conditions:</p> <ul style="list-style-type: none"> • Input strings contain multibyte or graphic characters and the search begins in the middle of the string. • A string unit (OCTETS, CODEUNITS16, or CODEUNITS32) is not specified.

Table 9. Restrictions (continued)

Function, keyword, or expression	Restriction
<ul style="list-style-type: none"> LCASE LOWER UCASE UPPER 	<ul style="list-style-type: none"> If a string argument contains characters with diacritical marks, Db2 for z/OS does not deliver the expected result, but IBM Db2 Analytics Accelerator does. For example: <pre>CREATE TABLE XYZ.TMP (A GRAPHIC(64), B GRAPHIC(64) NOT NULL) CCSID UNICODE; INSERT INTO XYZ.TMP VALUES ('Ü','Ü'); SELECT LOWER(B) XYZ.TMP;</pre> <p>Db2 for z/OS: Returns Ü, same as before because characters with diacritical marks are not converted.</p> <p>IBM Db2 Analytics Accelerator: Returns ü (correct result). Reason: The string argument is converted before the function is executed.</p> Accelerated queries might return different results (including invalid values) or an error with SQLCODE=-171, SQLSTATE=42815 if any of these functions is applied to EBCDIC or ASCII data. For example, take the following query, which returns invalid values: <pre>SELECT UCASE('μ~stuvwxyz', 'De_CH') FROM SINGLE_EBCDIC;</pre> <p>The query is supposed to find the string 'μ~stuvwxyz' in a table containing single-byte EBCDIC data and convert the resulting strings to uppercase. The Swiss variant of the German character set is used.</p> <p>Depending on where the query is run, the following values are returned:</p> <p>Db2 for z/OS: μ△STUVWXYZ (correct result)</p> <p>Accelerator: <unprintable symbol>△STUVWXYZ (incorrect result)</p> Result deviations can occur if input strings contain multibyte or graphic EBCDIC characters. Db2 for z/OS and IBM Db2 Analytics Accelerator return different length values if any of these functions is applied to GRAPHIC data. For example: <pre>CREATE TABLE TMP (A GRAPHIC(64), B GRAPHIC(64) NOT NULL) CCSID UNICODE; INSERT INTO TMP VALUES ('a','b');</pre> <p>After that, you run the following query: <pre>SELECT LENGTH(UPPER(B)) FROM TMP;</pre> <p>Depending on where the query is run, the following values are returned:</p> <p>Db2 for z/OS: 64</p> <p>Accelerator: 192</p> <p>The reason is that the accelerator uses a different result data type, that is, CHAR(192) instead of GRAPHIC(64).</p></p>
<ul style="list-style-type: none"> POWER 	If the argument data type is BIGINT, the result data type will be cast to DOUBLE.
<ul style="list-style-type: none"> RAND 	Supported without a parameter only.

Table 9. Restrictions (continued)

Function, keyword, or expression	Restriction
• ROWID	In Db2 for z/OS, two types of ROWID values exist: the longer, external type, and the shorter internal type. If a query references ROWID values and these values are in the external format, the values are converted to the internal format when the query is routed to an accelerator because IBM Db2 Analytics Accelerator can handle the internal type only. The internal ROWID type is also what the accelerator returns in the query results. If you have an application that relies on the external ROWID type to be returned, do not accelerate the query.
• ROUND	If ROUND works on a numerical expression of type DECFLOAT, and the rounding would result in a range underflow or overflow, Db2 for z/OS returns positive or negative Infinity ($-\infty$ or $+\infty$), whereas IBM Db2 Analytics Accelerator returns <i>not a number</i> (NaN).
• SIGN	If the data type of an input argument is DECIMAL, the result must be cast to this type.
• STRIP	Result deviations can occur if input strings (in fields or columns) contain graphic EBCDIC characters.
• SUBSTRING	SUBSTRING is supported only if the expression refers to character or graphic data types.
• TIMESTAMP • TIMESTAMP_FORMAT • TO_DATE	<p>String literals in the following formats are not supported as arguments of the TIMESTAMP function:</p> <ul style="list-style-type: none"> An argument in the format <code>yyyyxxddhhmmss</code> or a 13-byte string. Queries that contain strings in these formats are routed to the accelerator none the less because Db2 bind time is unaware of this incompatibility. The string could be a valid TIMESTAMP string. String literals with a space character () or dash (-) as the separator (delimiter) between the date and the time portion and a period (.) or colon (:) to separate hours, minutes, and seconds: Examples: <ul style="list-style-type: none"> <code>yyyy-mm-dd[hh.mm.ss[.nnnnnn]]</code> (space character before hh; period to separate hours, minutes, and seconds) <code>yyyy-mm-dd[-hh:mm:ss[.nnnnnn]]</code> (dash before hh; colon to separate hours, minutes, and seconds) String literals with the capital letter T as the separator (delimiter) between the date and the time portion and a period (.) or colon (:) to separate hours, minutes, and seconds: <ul style="list-style-type: none"> <code>yyyy-mm-dd[T hh.mm.ss[.nnnnnn]]</code> (T before hh; period to separate hours, minutes, and seconds) <code>yyyy-mm-dd[T hh:mm:ss[.nnnnnn]]</code> (T before hh; colon to separate hours, minutes, and seconds) <p>See also “Dates in TIMESTAMP expressions” on page 45.</p>
• TRANSLATE	<ul style="list-style-type: none"> If the arguments are of data type CHAR or VARCHAR, the corresponding characters in the to-string expression and the from-string expression must have the same number of bytes (except in the case of a zero-length string). This means, for example, that you cannot convert a single-byte character to a multibyte character, or a multibyte character to a single-byte character. The padding character (pad-char-exp argument) cannot be the first byte of a valid multibyte character. If it is, error -171 is returned. Result deviations can occur if arguments contain multibyte characters.
• TRUNCATE	You cannot use the TRUNCATE function on accelerator-only tables.
• TRIM	Result deviations can occur if input strings contain multibyte or graphic EBCDIC characters.

Table 9. Restrictions (continued)

Function, keyword, or expression	Restriction
<ul style="list-style-type: none"> UPPER UCASE 	See LOWER/LCASE.
<ul style="list-style-type: none"> VARCHAR9 	<p>Result deviations can occur if the VARCHAR9 function is used to convert DECIMAL values to strings:</p> <ul style="list-style-type: none"> When compared with the results returned by Db2 for z/OS, an accelerated query might show a leading blank in the results. For example: <pre>SELECT VARCHAR9(13.1) FROM TABLE_1</pre> <p>gives you the following results:</p> <p>Db2 for z/OS: 13.1</p> <p>IBM Db2 Analytics Accelerator 13.1 (leading blank)</p> <p>In accordance with these results, the length function also returns different values: <pre>SELECT LENGTH(VARCHAR9(13.1)) FROM TABLE_1</pre> <p>Db2 for z/OS: 4</p> <p>IBM Db2 Analytics Accelerator: 5</p> </p> When compared with the results returned by Db2 for z/OS, an accelerated query might show leading zeros (0s) in places where the result value does not require all of the formatted digits. For example, if 1.0 is one of the values returned by the following query: <pre>SELECT VARCHAR9(DECIMAL(col1,7,5)) FROM TABLE_1</pre> <p>you get the following results:</p> <p>Db2 for z/OS: 1.00000</p> <p>IBM Db2 Analytics Accelerator 01.00000 (leading zero)</p>
<ul style="list-style-type: none"> VARCHAR_FORMAT 	<p>The following date and timestamp formats are not supported:</p> <ul style="list-style-type: none"> FF[n] - fractional seconds (000000-999999) The optional number <i>n</i> is used to specify the number of digits to include in the return value. Valid values for <i>n</i> are the integers from 1-12. The default is 6. ID - ISO day of the week (1-7) The value 1 represents Monday. The value 7 represents Sunday. IYYY - ISO year (0000-9999) The last four digits of the year based on the ISO week that is returned. NNNNNN - microseconds (000000-999999) This format is equivalent to FF6. RR - last two digits of the year (00-99) SSSSS - seconds since the previous midnight

Unsupported scalar functions

Table 10. Unsupported scalar functions

• ACOS	• CCSID_ENCODING	• DIFFERENCE	• MULTIPLY_ALT	• SCORE	• TOTALORDER
• ASCII	• CLOB	• EBCDIC_CHR	• MQ <xxx>	• SIN	• TRUNC_TIMESTAMP
• ASCII_CHR	• COLLATION_KEY	• EBCDIC_STR	• NORMALIZE_STRING	• SINH	• UNICODE
• ASCII_STR	• COS	• ENCRYPT_TDES	• OVERLAY	• SOA <xxx>	• UNICODE_STR
• ASIN	• COSH	• GENERATE_UNIQUE	• QUANTIZE	• SOUNDEX	• VERIFY_GROUP_FOR_USER
• ATAN	• CONTAINS	• GETHINT	• PACK	• TAN	• VERIFY_ROLE_FOR_USER
• ATAN2	• DBCLOB	• GETVARIABLE	• RAISE_ERROR	• TANH	• VERIFY_TRUSTED_CONTENT_FOR_USER
• ATANH	• DECFLOAT_SORTKEY	• IDENTITY_VAL_LOCVAl	• RID	• TIMESTAMP_ADD	• <xxx>XML<xxx>
• BLOB	• DECRYPT <xxx>	• INSERT	• ROUND_TIMESTAMP	• TIMESTAMP_TZ	

Supported functions of the database engine

The product's database engine offers a range of aggregate functions and online analytical processing (OLAP) functions. The latter are frequently called built-in functions. A subset of these functions is supported. That is, you can use these functions in your SQL queries, and they will be processed if the query can be accelerated. A list of the supported functions is provided here.

General information

- To use any of these functions, the QUERY_ACCELERATION special register must be set to ALL, ENABLE, or ELIGIBLE.
- All functions listed here can be referenced in a database view.
- None of the functions can be specified as the source function of a user-defined function (UDF). Trying to do so results in SQLCODE -120.
- Db2 for z/OS verifies just the validity of the parameter data types. All other processing, including the validation, is done by the database engine of the accelerator.
- Some of the functions listed here are available as OLAP functions and as aggregate functions, but most of them are OLAP only. If a function is also available as an aggregate function, you find a link to an external description. All OLAP functions are externally described in a single topic. A link is provided at the bottom.

Note: The following list makes frequent use of the term *OLAP window*. An online analytical processing window (OLAP window) can be seen as the scope or range of values preceding or succeeding the current value. The selected window serves as the basis for the calculation.

CUME_DIST

This is a distribution function that returns a cumulative distribution of a row within an OLAP window, expressed as a value between 0.0 and 1.0.

Result computation

The number of rows preceding or at the same level as the current row in the OLAP window is divided by the total number of rows in the OLAP window. The result cannot be NULL.

Result data type

DECFLOAT(34)

More information

CUME_DIST aggregate function

FIRST_VALUE

This function returns the expression value for the first row in an OLAP window.

Result computation

If IGNORE NULLS is specified, all rows where the expression value for the row is the null value are not considered in the calculation. If IGNORE NULLS is specified and all values in the OLAP window are null, FIRST_VALUE returns the null value.

LAG This function returns the expression value for the row at *offset* rows *before* the current row. The offset must be a positive integer constant.

Result computation

An offset value of 0 means the current row. If a *window-partition-clause* is specified, *offset* means offset rows before the current row and within the current partition. If *offset* is not specified, the value 1 is used.

If *default-value* (which can be an expression) is specified, it will be returned if the offset goes beyond the scope of the current partition. Otherwise, the null value is returned.

If IGNORE NULLS is specified, all rows where the expression value for the row is the null value are not considered in the calculation. If IGNORE NULLS is specified and all rows are null, *default-value* is returned. If *default-value* was not specified, the null value is returned.

LAST_VALUE

This function returns the expression value for the last row in an OLAP window.

Result computation

If IGNORE NULLS is specified, all rows where the expression value for the row is the null value are not considered in the calculation. If IGNORE NULLS is specified and all values in the OLAP window are null, FIRST_VALUE returns the null value.

LEAD This function returns the expression value for the row at *offset* rows *after* the current row.

Result computation

An offset value of 0 means the current row. If a *window-partition-clause* is specified, *offset* means offset rows before the current row and within the current partition. If *offset* is not specified, the value 1 is used.

If *default-value* (which can be an expression) is specified, it will be returned if the offset goes beyond the scope of the current partition. Otherwise, the null value is returned.

If IGNORE NULLS is specified, all rows where the expression value for the row is the null value are not considered in the calculation. If IGNORE NULLS is specified and all rows are null, *default-value* is returned. If *default-value* was not specified, the null value is returned.

NTH_VALUE

This function returns the expression value for the *n*th row in an OLAP window.

Result computation

Db2 for z/OS does not support the following keywords :

- FROM FIRST
- FROM LAST
- RESPECT NULLS

- IGNORE NULLS

If you specify any of these, SQLCODE -199 is returned. Hence you can only use this function without these keywords. The database engine of the accelerator executes the function with default settings, which equates to specifying FROM FIRST RESPECT NULLS.

NTILE

This function returns the quantile rank of a row. If the argument can be null, the result can be null. If the argument is null, the result is the null value. The result is the quantile rank, which is associated with the current row. Equivalent sort keys are not considered when rows are divided into quantiles. Rows with equivalent sort keys can be assigned to different quantiles based on the non-deterministic order of these sort keys. Therefore, NTILE is a non-deterministic function.

Result computation

The NTILE function computes the quantile rank of a row by dividing the ordered rows within the OLAP window into *num-tiles* quantiles and returns a value between 1 and $MIN(n, num-tiles)$, where *n* is the number of rows within the OLAP window. If *n* is evenly divisible by *num-tiles*, the rows in the OLAP window are grouped into *num-tiles* quantiles, each containing $n/num-tiles$ rows. Otherwise, each of the quantiles 1 through $MOD(n, num-tiles)$ is assigned $n/num-tiles + 1$ rows while each of the quantiles $MOD(n, num-tiles) + 1$ through *num-tiles* is assigned $n/num-tiles$ rows.

Result data type

DECFLOAT(34)

PERCENT_RANK

This is a distribution function that returns a relative percentile rank of a row within an OLAP window, expressed as a value between 0.0 and 1.0.

Result computation

If the number of rows in the OLAP window is greater than 1, the result is the *RANK of the current row in the OLAP window - 1* divided by the *total number of rows in the OLAP window - 1*.

If the number of rows in the OLAP window is less than or equal to 1, the result is 0.0.

More information

PERCENT_RANK aggregate function

RATIO_TO_REPORT

This function returns the ratio of an *argument* to the *sum of the arguments* in an OLAP partition.

Result computation

The division is always executed by using the DECFLOAT(34) data type. If the argument can be null, the result can be null; if the argument is null, the result is the null value.

Result data type

DECFLOAT(34)

Related information:



OLAP specification

Supported special registers

IBM Db2 Analytics Accelerator supports most Db2 special registers. During SQL routing to an accelerator, these special registers are replaced with the equivalent dashDB functions or with a literal string containing the register's value at the time of statement execution. In some cases, this leads to different query results.

The following special registers are supported by dashDB function substitution:

- CURRENT ACCELERATOR

- CURRENT APPLICATION COMPATIBILITY
- CURRENT APPLICATION ENCODING SCHEME
- CURRENT CLIENT_ACCTNG
- CURRENT CLIENT_APPLNAME
- CURRENT CLIENT_CORR_TOKEN
- CURRENT CLIENT_WRKSTNNAME

- CURRENT DATE

Replaced with the dashDB function DATE(CURRENT DATE) on the accelerator.

- CURRENT DEBUG MODE
- CURRENT DECFLOAT ROUNDING MODE
- CURRENT DEGREE
- CURRENT EXPLAIN MODE
- CURRENT GET_ACCEL_ARCHIVE
- CURRENT LOCALE LC_CTYPE
- CURRENT MAINTAINED TABLE TYPES FOR OPTIMIZATION
- CURRENT MEMBER
- CURRENT OPTIMIZATION HINT
- CURRENT PACKAGE PATH
- CURRENT PACKAGESET
- CURRENT PATH
- CURRENT PRECISION
- CURRENT QUERY ACCELERATION
- CURRENT QUERY ACCELERATION WAITFORDATA
- CURRENT REFRESH AGE
- CURRENT ROUTINE VERSION
- CURRENT RULES
- CURRENT SCHEMA
- CURRENT SERVER
- CURRENT SQLID
- CURRENT TEMPORAL BUSINESS_TIME
- CURRENT TEMPORAL SYSTEM_TIME
- CURRENT TIME

Replaced with dashDB function TIME(CURRENT TIME) on the accelerator.

Restriction: Microseconds are not returned in the query results. A zero is displayed in the positions where you would expect microseconds.

- CURRENT TIMESTAMP

Replaced with the dashDB function TIMESTAMP(CURRENT TIMESTAMP).

Restriction: Microseconds are not returned in the query results. A zero is displayed in the positions where you would expect microseconds.

- SESSION_USER
- SESSION TIME ZONE

Most of the remaining special registers are supported through literal string substitution in the SQL. For example, a reference to CURRENT SQLID will be replaced in the SQL with the literal string value from that register before the SQL is submitted to dashDB for processing.

Conditions that prevent query routing to an accelerator

Db2 for z/OS does not route a query to an accelerator if any of the following conditions applies.

- You use a distributed data facility (DDF) client, such as an Open Database Connectivity (ODBC) client, that does not support *level 8* of the IBM Distributed Relational Database Architecture™ (DRDA).
- The encoding scheme of an SQL statement is M because the tables use different encoding schemes, or the query contains an expression that explicitly relates to a coded character set identifier (CCSID), for example, a cast specification with a CCSID option.
- The FROM clause of the query specifies a data-change-table-reference, or, in other words, the query is selected from a FINAL TABLE or from an OLD TABLE.
- The query contains a string expression (including columns) with an unsupported subtype. See Table 6 on page 24.
- The query contains an expression with an unsupported result data type or subtype. See “Supported data types” on page 22.
- The query refers to a column that uses a field procedure (FIELDPROC).
- The query uses a special register other than:
 - CURRENT DATE
 - CURRENT TIME
 - CURRENT TIMESTAMP
- A LOCAL date format is used, that is, one or more of the following conditions apply:
 1. The query contains a CHAR function in which LOCAL is specified as the second argument.
 2. The query contains a date or time expression and the DATE FORMAT field of the DSNTIP4 installation panel specifies LOCAL.
 3. Application programs that process SQL on Db2 for z/OS have been precompiled with the DATE(LOCAL) option.

Exception: You can enable LOCAL date expressions in the format dd/mm/yyyy or yyyyymmdd (with two space characters at the end) by including the QUERY_ACCEL_OPTIONS ZPARM value 4 or 7. For more information, see *Installing Db2 libraries with IBM Db2 Analytics Accelerator for z/OS support* in the *IBM Db2 Analytics Accelerator for z/OS: Installation Guide*.

- The query contains a sequence expression (NEXTVAL or PREVVAL).
- The query contains a user-defined function (UDF)
- The query contains a ROW CHANGE expression
- The query contains a string constant that is longer than 16000 characters.
- A new column name is referenced in a sort-key expression, for example:
SELECT C1+1 AS X, C1+2 AS Y FROM T WHERE ... ORDER BY X+Y;
- The query uses a DAY function where the argument of the function specifies a duration.
- The query uses a TRANSLATE function with only two arguments.

- The query uses one of the following scalar functions and one or more of its arguments contain a parameter marker or a scalar fullselect:
 - LOCATE
 - MICROSECOND
 - MIDNIGHT_SECONDS
 - TRANSLATE
- The query uses a table function, such as ADMIN_TASK_LIST or ADMIN_TASK_STATUS.
- The query uses an unsupported scalar function. See Table 10 on page 35
- The instruction to process “SELECT FROM ... INSERT INTO ... ” statements (ZPARM QUERY_ACCEL_OPTIONS includes the value 2) is ignored if the target table uses an encoding scheme that is different from the encoding scheme of the tables in the SELECT statement.
- A *static* SQL query that includes a FOR SYSTEM_TIME ... or FOR BUSINESS_TIME ... is not routed to an accelerator if one of the following bind options (value YES, default) is used.

Table 11. Unsupported bind options

Clause	Bind option (value YES, which is the default)
<ul style="list-style-type: none"> • FOR BUSINESS_TIME AS OF • FOR BUSINESS_TIME FROM ... TO ... • FOR BUSINESS_TIME BETWEEN ... AND... 	<ul style="list-style-type: none"> • BUSTIMESENSITIVE • BUSINESS_TIME SENSITIVE
<ul style="list-style-type: none"> • FOR SYSTEM_TIME AS OF • FOR SYSTEM_TIME FROM ... TO ... • FOR SYSTEM_TIME BETWEEN ... AND ... 	<ul style="list-style-type: none"> • SYSTIMESENSITIVE • SYSTEM_TIME SENSITIVE

In cases like this, you receive error message -4742 with reason code 18.

Related concepts:

“EXPLAIN information” on page 94

Isolation levels

Db2 isolation levels are ignored by IBM Db2 Analytics Accelerator.

Every table on an accelerator has its counterpart. That is, the table has been derived from a table that resides in Db2 for z/OS. An accelerator-shadow table is thus a copy of a projection of a Db2 table. In most cases, the projection is the Db2 table itself. Columns of the original Db2 table are excluded from the projection only if a column uses an unsupported data type.

You can change a Db2 table practically at any time by means of INSERT, UPDATE, or DELETE operations, mass imports (LOAD), and schema modifications (selected DDL). In this version of IBM Db2 Analytics Accelerator for z/OS, changes to a DB2 table are propagated automatically to the associated accelerator-shadow table only if the incremental update feature is used. If incremental updates are not enabled, you must update the accelerator-shadow tables manually in order to synchronize the table data. You can start the update from IBM Db2 Analytics Accelerator Studio or run the SYSPROC.ACCEL_LOAD_TABLES stored procedure for this purpose.

The update of accelerator-shadow tables is an asynchronous process, that is, there is no guarantee that the data in the Db2 tables and the data in the accelerator-shadow tables are always in sync. Even with incremental updates, a

certain degree of latency is involved, which means that tables on an accelerator are updated somewhat later than the original Db2 tables. Therefore, the result set that a query returns when it is processed by Db2 might differ from the result set that the same query produces when it is processed by IBM Db2 Analytics Accelerator. This implies that isolation levels set for Db2 data are ignored or must be ignored by applications that send queries to IBM Db2 Analytics Accelerator. Result sets that are returned by IBM Db2 Analytics Accelerator do not necessarily comply with any isolation level setting.

To ensure data consistency on a relatively safe level, update accelerator-shadow tables immediately before you run the query and choose an appropriate lock scope so that the affected Db2 tables cannot be altered during the update. This, however, still does not guarantee that the results of an accelerated query will show the latest changes because the lock might prevent or postpone an update of the original Db2 table data.

If you use the incremental update function, you can ensure that committed Db2 for z/OS changes are propagated to the accelerator by calling the `ACCEL_CONTROL_ACCELERATOR` stored procedure with the `<waitForReplication>` command.

If the most recent table changes must be reflected in the result set, set the `CURRENT QUERY ACCELERATION` special register to the value `NONE`. This means that the query is executed by Db2 and will not be accelerated.

Related concepts:

“Updating accelerator tables continually” on page 117

Related tasks:

“Loading tables” on page 83

Incompatibilities between Db2 for z/OS and IBM Db2 Analytics Accelerator

In the list of incompatibilities, you find explanations for execution errors or different query results that Db2 for z/OS and IBM Db2 Analytics Accelerator might return for the same query.

Data types

See how IBM Db2 Analytics Accelerator handles certain Db2 data types:

FLOAT

The `FLOAT` data type is an approximate data type. The result can be different due to platform differences (hexadecimal floating-point numbers (HFP) versus IEEE754 floating-point numbers) or other internal handling differences. This applies to expressions that return `FLOAT` data and to expressions that contain a `FLOAT` data-type argument.

Related concepts:

“Differences in floating-point error tolerances” on page 54

Division and multiplication of decimal values

During the division or multiplication of decimal values, Db2 for z/OS might truncate operand values if at least one of the operands is defined with a precision

greater than 15. If that happens, the result of the calculation will be imprecise or wrong. IBM Db2 Analytics Accelerator does not truncate operand values internally, so the results will be more precise.

Example: Consider the following query:

```
CREATE TABLE TEST_TABLE(
D_DECIMAL_4_2 DECIMAL(4,2),
D_DECIMAL_17_2 DECIMAL(17,2),
D_DECIMAL_30_6 DECIMAL(30,6)
);
INSERT INTO TEST_TABLE VALUES(3.5, 3.5, 0.5);
INSERT INTO TEST_TABLE VALUES(1.5, 1.5, 1.5);

SELECT D_DECIMAL_17_2/D_DECIMAL_30_6, D_DECIMAL_17_2*D_DECIMAL_30_6
FROM TEST_TABLE;

SELECT D_DECIMAL_4_2/D_DECIMAL_30_6, D_DECIMAL_4_2*D_DECIMAL_30_6
FROM TEST_TABLE;
```

Db2 for z/OS returns results as follows:

- For the first calculation in the query:

```
+-----+
| ? | 1.5 |
| 1 | 1.5 |
+-----+
```

- For the second calculation in the query:

```
+-----+
| ? | 1.75 |
| 1.5 | 2.25 |
+-----+
```

The question mark (?) means that an overflow has occurred (division by zero) because Db2 for z/OS has internally reduced the value 0.5 to 0.

IBM Db2 Analytics Accelerator would have returned the correct results, which are the same for both calculations in the query:

```
+-----+
| 7 | 1.75 |
| 1 | 2.25 |
+-----+
```

The truncation problem is discussed in subsections of the *Arithmetic with two decimal operands* topic of the SQL reference in the Db2 11 for z/OS Knowledge Center. A link to this topic is provided under **Related information**.

Related information:



Arithmetic with two decimal operands

Encoding issues

Consider the following points when you have to deal with Unicode SQL statements, multibyte (mixed) data in connection with parameter markers, and multibyte or double-byte EBCDIC columns.

Unicode SQL statements

Generally, when a UNICODE-encoded SQL statement contains characters or strings (as a literal, host variable, or parameter marker) that refer to a single-byte-encoded EBCDIC column, then Db2 tries to convert these strings to EBCDIC. Sometimes, such a string cannot be converted properly because it contains invalid UNICODE

characters or characters that do not exist in the target EBCDIC code page. In such cases, the results of the SQL operation and the SQLCODE returned by the accelerator might be different from those returned by Db2 for z/OS with regard to the following aspects:

- Queries might return a different number of rows.
- SQLCODE 0 might be returned instead of the SQL warning codes +331 or +335.
- Different substitution characters might be put into returned or inserted strings.
- Returned or inserted strings or substitution characters might be different if multiple encoding is enabled on the accelerator.

Mixed data and parameter markers

If a Db2 subsystem is set up to contain mixed data (mixed=yes), that is, if differently encoded data is allowed to coexist, and the coded character set identifier (CCSID) is not specified for parameter markers in a query, the query might return incorrect results.

The LIKE operator in connection with binary wildcard characters

Depending on the system executing a query (Db2 for z/OS or IBM Db2 Analytics Accelerator), the query might not return any results or wrong results if the LIKE operator is used in combination with a *binary* wildcard character (%) or (_). The reason for that is that both systems expect a different binary encoding of the wildcard characters.

Db2 for z/OS

Valid binary encodings of the wildcard characters are bx'6c' for the percent sign (%) and bx'6d' for the underscore character (_).

IBM Db2 Analytics Accelerator

Valid binary encodings of the wildcard characters are X'25' for the percent sign (%) and X'5F' for the underscore character (_).

This means, for example, that if a query uses bx'6c' to refer to the percent sign as a wildcard character, Db2 for z/OS will return the expected result, but IBM Db2 Analytics Accelerator will not.

On the contrary, if an expression like SELECT COL FROM TABLE WHERE COL LIKE BINARY('A%') is encountered, IBM Db2 Analytics Accelerator will give you the expected result, but Db2 for z/OS will not.

Exception handling

When an exception, such as a buffer overflow, a division by zero, or an out-of-range value occurs, Db2 returns the value NULL combined with a +802 warning, provided that the faulty expression is in the outermost SELECT statement and affects only one row. This is also called *friendly arithmetic*. IBM Db2 Analytics Accelerator will always issue an error in cases like these.

Maximum row length

The maximum row length for a table on an accelerator is 1048319 bytes. If the combined length of all column fields in a Db2 table exceeds this limit, you cannot define that table on the accelerator.

Predicate evaluation

Db2 tolerates SQL errors during predicate evaluation as long as the error has no impact on the result of the evaluation. IBM Db2 Analytics Accelerator, on the contrary, always returns an error. This rule applies to predicates consisting of multiple terms (an SQL error in a predicate consisting of only one term cannot be

tolerated). An SQL error in a predicate does not influence the outcome of the predicate evaluation if it does not matter whether the error makes the predicate logically *true* or *false*. For example, if the predicate is a conjunction of terms as in P1 AND P2, and the evaluation of P1 results in an SQL error, then this error does not influence the outcome if P2 evaluates to *false* because both, *true AND P2* and *false AND P2*, evaluate to *false*. Table 12 shows all relevant permutations for two-term predicates (E = *error*, T = *true*, F = *false*). Values in parentheses indicate that the P2 term is not evaluated if an evaluation of P1 already yields the result of the entire predicate evaluation.

Table 12. Relevant combinations of two-term predicates

P1	P2	P1 AND P2	P1 OR P2
E	E	E	E
E	T	E	T
E	F	F	E
T	E	E	(T)
F	E	(F)	E

The reason for choosing this evaluation method is that Db2 sometimes changes the order of term evaluation internally. The change is not announced or reported, and therefore goes unnoticed. Depending on the order of term evaluation, Db2 returns an error in one case, but not the other.

The behavior is very similar for the CASE and DECODE functions. These are basically SQL implementations of IF ... THEN ... ELSE constructs and exist in Db2 and IBM Db2 Analytics Accelerator. Db2 will ignore a wrongly defined option as long as it can select a correct one that evaluates to true. IBM Db2 Analytics Accelerator, on the other hand, will first check all available options, and return an error if one of these is not correctly defined. In the latter case, it does not matter whether selecting one of the correctly defined options would have led to a correct result.

SQL issues

In a grouping select, the HAVING clause can only reference expressions that are single-valued within a group. That is, you can only reference group fields, aggregates, or single-valued expressions that are derived from group fields or aggregates (which must include constants).

Time and date values

Time and date constructs in the dashDB database often differ from the constructs used by Db2 for z/OS, so that values and expressions must be converted. The conversion is often accompanied by limitations.

24:00:00 conversion

This product supports the Db2 for z/OS time representation 24.00.00 for midnight (end of day) in TIME and TIMESTAMP columns or SQL expressions. There is no need to set up a conversion.

However, although the time 24:00:00 is accepted as a valid time, it is never returned by the accelerator as the result of a time addition or subtraction. This holds true even if the time added or subtracted is 0 (seconds, minutes, or hours) or 24 hours: The result will always be 00:00:00. Db2 for z/OS will return 24:00:00 in these cases.

Suppose you have a table TEST with a timestamp column C1. You insert just the following value into the column: 9999-12-01-24.00.00.000000000000. Now you run the following query, which includes a simple calculation:

```
SELECT C1 - 1 MONTH,C1 FROM TEST
```

That is, you want to retrieve the values from C1, but one month is to be deducted from each value. Since there is just one value, you will see only one record in the query results. But note the difference depending on where the query was executed:

Db2 for z/OS


```
9999-11-01-24.00.00.000000000000
```

IBM Db2 Analytics Accelerator

```
9999-11-02-00.00.00.000000000000
```

As you can see, Db2 for z/OS gives you 24:00, end of day, on the 1st of November, but IBM Db2 Analytics Accelerator gives you 00:00, beginning of the next day (2nd of November). Both values mean exactly the same, but there is a difference in the representation.

Related information:

 https://www.ibm.com/support/knowledgecenter/SSEPGG_11.1.0/com.ibm.db2.luw.sql.ref.doc/doc/r0323457.html

Dates in TIMESTAMP expressions

If a date is supplied as the only argument to a TIMESTAMP expression, for example `TIMESTAMP(DATE('2015-12-31'))`, Db2 for z/OS returns a timestamp with a time portion that shows no fractional seconds, whereas IBM Db2 Analytics Accelerator returns fractional seconds with a precision of six digits:

Db2 for z/OS	2015-12-31 00:00:00
IBM Db2 Analytics Accelerator	2015-12-31 00:00:00.000000

You can circumvent this problem by supplying the date as a string value, for example `TIMESTAMP('2015-12-31')` because this gives the same return value in both systems, that is:

```
2015-12-31 00:00:00.000000
```

Strings as time values and their conversion

For some scalar functions, such as `TIME`, Db2 accepts a time value, a timestamp, and also a string representation of times and timestamps. When Db2 evaluates the scalar function, it first checks whether a value can be interpreted as a time. If this fails, it tries to interpret the value as a timestamp. The corresponding accelerator functions do not accept string values. During query conversion, the content of a string is not evaluated. Instead, it is always cast to a time value. If the string represents a valid time format on the accelerator, IBM Db2 Analytics Accelerator processes the query. Otherwise, if the string represents a timestamp, it returns an error. A workaround is to cast the string explicitly to a timestamp in the query.

Affected functions:

- `CAST` (if the operation converts the string to a Db2 `TIME` value)
- `EXTRACT(HOUR FROM <arg>)`
- `EXTRACT(MINUTE FROM <arg>)`
- `HOUR`
- `MIDNIGHT_SECONDS`

- MINUTE
- SECOND
- TIME

Trailing blanks

Trailing blanks cause incorrect results under the following conditions:

- The result data type of an SQL expression is VARCHAR or VARGRAPHIC (as with LEFT, LTRIM, REPLACE, RIGHT, or TRIM for example).
- The original values returned as VARCHAR or VARGRAPHIC contain trailing blanks.
- A GROUP BY clause is used to sort the results on the VARCHAR or VARGRAPHIC column.

Queries of this type return incorrect results because the accelerator uses column-organized tables for performance reasons. The columnar engine used for these tables removes trailing blanks from VARCHAR or VARGRAPHIC values that are sorted by a GROUP BY clause. If the original values are of varying length and contain trailing blanks, the blanks are removed when the rewritten query is executed. Hence the grouped values in the column-organized table do not contain trailing blanks, and the length calculation will inevitably lead to deviating results.

Example

```
SELECT <varchar_col> || 'a' FROM <table> GROUP BY <varchar_field>
```

In the first part of the query, the letter a is added to each value of a VARCHAR column at the end. The result data type is also VARCHAR. If the original values contain trailing blanks, and the first part of the query is run on its own, you see the following, different outputs depending on where the query is run:

Db2 for z/OS:

```
<varchar_value>    a
```

That is, the trailing blanks are preserved for each value in the column.

IBM Db2 Analytics Accelerator:

```
<varchar_value>a
```

The trailing blanks are removed by the accelerator.

The second part of the query adds a GROUP BY clause to the first part. The outputs of the entire query differ as follows:

Db2 for z/OS:

- <varchar_value> a
- <varchar_value>a

Db2 for z/OS returns both possibilities for each grouped value.

IBM Db2 Analytics Accelerator:

```
<varchar_value>a
```

IBM Db2 Analytics Accelerator returns just the values without the trailing blanks.

Important: Sometimes, queries are internally rewritten, so that they contain a GROUP BY clause even though no such clause was given in the original query. In this case, you might run into the very same problem.

Incompatibilities related to IBM InfoSphere Data Replication for z/OS

You might encounter execution errors or deviating query results when you use the incremental update function. Some of these errors or result deviations are due to incompatibilities between Db2 for z/OS, IBM Db2 Analytics Accelerator for z/OS, and IBM InfoSphere® Change Data Capture for z/OS. The following section deals with these incompatibilities.

IBM InfoSphere Data Replication for z/OS is the main software component behind the incremental update function of IBM Db2 Analytics Accelerator for z/OS.

All incompatibilities that are listed in the section *Incompatibilities between Db2 for z/OS and IBM Db2 Analytics Accelerator for z/OS* also apply to IBM InfoSphere Data Replication for z/OS.

In addition, incompatibilities that affect the regular load function of IBM Db2 Analytics Accelerator for z/OS (the SYSPROC.ACCEL_LOAD_TABLES stored procedure) also affect IBM InfoSphere Data Replication for z/OS. However, the following exceptions exist when IBM InfoSphere Data Replication for z/OS transfers data to IBM Db2 Analytics Accelerator for z/OS:

- Values of VARCHAR FOR MIXED and VARGRAPHIC columns that exceed 16000 characters are cut off when they are received by IBM Db2 Analytics Accelerator for z/OS. The difference to Db2 for z/OS is that in the case of IBM InfoSphere Data Replication for z/OS, *no warning or error message* is issued.

Restrictions of accelerator-only tables

Here is a summary of the restrictions that exist for accelerator-only tables.

Current limitations

The use of accelerator-only tables brings extra workloads to your accelerators that compete for the same resources, such as CPU, disk, and system memory. This might impact the stability of your production environment.

Data-definition (DDL) statements

The following restrictions exist for data-definition statements and accelerator-only tables:

- The IN ACCELERATOR clause can be used only once in a statement.
- A CREATE TABLE must not contain a materialized query definition if the table is an accelerator-only table.
- A CREATE MASK statement must not define a mask for an accelerator-only table.
- A CREATE PERMISSION statement must not define a column permission for an accelerator-only table.
- A DECLARE GLOBAL TEMPORARY TABLE statement must not reference an accelerator-only table.
- A CREATE TRIGGER or ALTER TRIGGER statement must not reference an accelerator-only table in the body of the trigger.
- A CREATE TRIGGER or ALTER TRIGGER statement with an INSTEAD OF clause must not reference a view that references an accelerator-only table in the body of the trigger.
- An ALTER TABLE statement is not allowed if the table is an accelerator-only table.

- An ALTER TABLESPACE statement is not allowed if the table space contains accelerator-only tables.
- A DROP TABLESPACE statement fails if the table space contains accelerator-only tables. You must first drop the accelerator-only tables by using a DROP TABLE statement.
- The DROP DATABASE statement does not behave as expected:
 - If you create an accelerator-only table without specifying a database, the table is created in the default database as usual, but a DROP DATABASE statement does not remove that database.
 - A DROP DATABASE statement does not remove a database and the tables therein if some of the tables are accelerator-only tables. To remove such a database, first remove all accelerator-only tables by using separate DROP TABLE statements, and then the database.
- A RENAME TABLE statement fails if the table is an accelerator-only table.
- A LOCK TABLE statement must not refer to an accelerator-only table.
- A CREATE INDEX statement fails if the table to be indexed is an accelerator-only table.
- A CREATE SYNONYM statement fails if the synonym is for an accelerator-only table.
- You cannot create more than one accelerator-only table in a single transaction. The following example does not work:

```
CREATE TABLE AOT1 ... IN ACCELERATOR ...
CREATE TABLE AOT2 ... IN ACCELERATOR ...
COMMIT;
```

This example, on the other hand, does work:

```
CREATE TABLE AOT1 ... IN ACCELERATOR ...
COMMIT;
CREATE TABLE AOT2 ... IN ACCELERATOR ...
COMMIT;
```

In addition, the following clauses or keywords cannot be used in DDL statements that refer to accelerator-only tables:

Table 13. Unsupported clauses and keywords

APPEND	AS FULLSELECT	AUDIT	BUFFERPOOL
CHECK	COMPRESS	DATA CAPTURE	DSSIZE
EDITPROC	FIELDPROC	FOREIGN KEY	GENERATED
HASH	IMPLICITLY HIDDEN	IN TABLESPACE	INLINE LENGTH
LIKE	LOGGED	MEMBER CLUSTER	NOT LOGGED
PARTITION BY	PERIOD	PRIMARY KEY	TRACKMOD
UNIQUE INDEX	VALIDPROC	VOLATILE	WITH DEFAULT

Other SQL statements

The following statements are not allowed:

- You cannot run an INSERT INTO <accelerator-only table> SELECT ... FROM ... statement if the SELECT part of the statement references a normal Db2 table that has no loaded counterpart on the accelerator. The statements succeeds only if the SELECT part references one or more tables on the same accelerator as the accelerator-only table.

- You cannot submit an UPDATE statement against table columns that serve as distribution keys.
- You cannot insert values directly into a decimal column of an accelerator-only table if these values are the result of a division where the dividend, the divisor, or both are decimals. However, the operation works if the division is part of the SELECT portion in an INSERT INTO ... SELECT FROM ... statement.

Restrictions on data types, column types, and code pages

Note that restrictions that apply to certain data types in general also apply to the use of these types in accelerator-only tables. In addition to that, the following restrictions exist for data types, column types, and code pages (CCSIDs), or are especially noteworthy:

- The column names in an accelerator-only table must not start with a tilde (~) followed by an underscore (_) character.
- The CHAR FOR BIT DATA and VARCHAR FOR BIT DATA data types are not supported in accelerator-only tables if the environment has been set up for the use of mixed data (ZPARM MIXED=YES).
- You cannot define a large-object (LOB) column for an accelerator-only table.
- You cannot define a security label column for an accelerator-only table.

Stored procedures

Some of the IBM Db2 Analytics Accelerator stored procedures do not work on accelerator-only tables:

SYSPROC.ACCEL_ADD_TABLES

Submit a CREATE TABLE ... IN ACCELERATOR statement instead.

SYSPROC.ACCEL_REMOVE_TABLES

Submit a suitable DROP TABLE statement instead.

SYSPROC.ACCEL_SET_TABLES_REPLICATION

Accelerator-only tables cannot be enabled for incremental updates.

SYSPROC.ACCEL_ARCHIVE_TABLES

You cannot archive partitions of accelerator-only tables because accelerator-only tables are not partitioned.

SYSPROC.ACCEL_RESTORE_ARCHIVE_TABLES

Since you cannot archive partitions of accelerator-only tables, you can neither restore archived partitions.

SYSPROC.ACCEL_SET_TABLES_ACCELERATION

You cannot disable acceleration on accelerator-only tables because they exist on an accelerator only, and thus must be enabled for acceleration.

Database limitations

In general, all limitations of the database on the IBM Integrated Analytics System also apply to accelerator-only tables.

Different results

Results obtained by running the same query against a normal Db2 table and an accelerator-only table might differ under the following conditions:

Cross-loading of tables

When a LOAD DATA INCURSOR ... INTO statement is used, the query inherits the setting of the CURRENT QUERY ACCELERATION special register. Hence queries are automatically routed to an accelerator if this has been enabled by the CURRENT QUERY ACCELERATION special register.

CURRENT EXPLAIN MODE special register

Query routing to an accelerator works for queries that use the CURRENT EXPLAIN MODE special register if the special register is set to the value YES or EXPLAIN. Meaning of these settings:

CURRENT EXPLAIN MODE = YES

Enables the Db2 EXPLAIN facility and causes EXPLAIN information to be inserted into the EXPLAIN tables for eligible dynamic SQL statements after preparing and executing the statement. Dynamic SQL statements are hence compiled and executed normally.

CURRENT EXPLAIN MODE = EXPLAIN

Enables the Db2 EXPLAIN facility and captures EXPLAIN information for eligible dynamic SQL statement after preparing the statement. Note that with setting, a dynamic statement or query will not be executed, except for the SET clauses that it might contain.

New OLAP functions

IBM Db2 Analytics Accelerator supports the following moving aggregate functions:

- AVG
- CORRELATION (see **Notes**[®])
- COUNT
- COUNT_BIG
- COVARIANCE (see **Notes**)
- COVARIANCE_SAMP (see **Notes**)
- MAX
- MIN
- STDDEV
- SUM
- VARIANCE (VARIANCE_SAMP)

Notes:

- The result data type that IBM Db2 Analytics Accelerator returns for the CORRELATION, COVARIANCE, and COVARIANCE_SAMP functions is DOUBLE, and not DECFLOAT, as in Db2 for z/OS. This results in a loss of precision.
- Queries that use the previously mentioned moving aggregate functions are not accelerated if the window-aggregation-group clause specifies just a group-end clause, for example:

```
SELECT C1, C2, C3, SUM(C2)
OVER (ORDER BY C3 ROWS UNBOUNDED FOLLOWING)
FROM <table>
ORDER BY 1,2,3,4;
```

Temporal tables

. Queries referencing temporal tables usually contain the following expressions or clauses:

- FOR SYSTEM_TIME AS OF
- FOR SYSTEM_TIME FROM ... TO ...
- FOR SYSTEM_TIME BETWEEN ... AND ...
- FOR BUSINESS_TIME AS OF
- FOR BUSINESS_TIME FROM ... TO ...
- FOR BUSINESS_TIME BETWEEN ... AND ...

Restriction: A *static* SQL query that includes a FOR SYSTEM_TIME ... or FOR BUSINESS_TIME ... is not routed to an accelerator if one of the following bind options (value YES, default) is used.

Table 14. Unsupported bind options

Clause	Bind option (value YES, which is the default)
<ul style="list-style-type: none">• FOR BUSINESS_TIME AS OF• FOR BUSINESS_TIME FROM ... TO ...• FOR BUSINESS_TIME BETWEEN ... AND...	<ul style="list-style-type: none">• BUSTIMESENSITIVE• BUSINESS_TIME SENSITIVE
<ul style="list-style-type: none">• FOR SYSTEM_TIME AS OF• FOR SYSTEM_TIME FROM ... TO ...• FOR SYSTEM_TIME BETWEEN ... AND ...	<ul style="list-style-type: none">• SYSTIMESENSITIVE• SYSTEM_TIME SENSITIVE

In cases like this, you receive error message -4742 with reason code 18.

Timestamps

The portion that denotes fractional seconds in timestamps can have up to twelve digits in Db2 for z/OS and also in dashDB. Hence IBM Db2 Analytics Accelerator fully supports this format. That is, values are not truncated as in earlier versions of IBM Db2 Analytics Accelerator for z/OS.

Restrictions: Timestamps with timezone specifications are not supported, to the effect that:

- SQL queries that require TIMESTAMP with a timezone specification as the result data type are not routed to an accelerator.
- Timestamps with timezone constants are not accepted.
- Functions and special registers that are related to timestamps with timezone specifications are not accepted.

Note: If a string constant or expression is used in connection with timestamps, such as string_col = timestamp_col, the query is routed to IBM Db2 Analytics Accelerator regardless of the content of the string expression because the string constant or expression is not validated. This might cause a run-time error during query execution if the string column contains unsupported timestamp values or formats.

Global variables

Db2 11 for z/OS introduces the concept of global variables, which can be created, instantiated, accessed, and modified by applications. Global variables are named

memory variables that you can access and modify through SQL statements. Global variables enable you to share relational data between SQL statements without the need for an application to transfer data. A global variable is associated with a specific application context, and contains a value that is unique to the scope of that context.

Global variables are not supported by IBM Db2 Analytics Accelerator. A query that contains a global variable statement is hence not routed to an accelerator.

Furthermore, the `CURRENT_QUERY_ACCELERATION` special register cannot be set to the value of a global variable.

Support for the array data type in SQL PL, CALL, and Java

An array consists of an ordered set of elements of a single built-in data type. Starting with Db2 11 for z/OS, Db2 offers limited support for arrays within SQL to provide more options for routines and data access code.

Arrays can be defined as parameters and variables for SQL routines. In addition, arrays can be passed from one procedure to another as arguments for the **IN** and **OUT** parameters, function parameters, or as a function return value. Db2 11 for z/OS and later versions support both, ordinary arrays and associative arrays. Ordinary arrays have a defined upper limit, while associative array do not have such a limit. When the array data type is used in a query, the query is not routed to an accelerator.

Aliases and synonyms

An alias is an alternative name for an object, such as a Db2 table, a Db2 view, or another Db2 alias. It can be used to reference an object wherever that object can be referenced directly. Synonyms are similar to aliases, but are supported only for compatibility with previous releases. Synonyms will be deprecated in a future release. Db2 11 for z/OS extends the support for aliases, so that these can be defined for sequence objects, and also provides support for *public aliases*. IBM Db2 Analytics Accelerator does not support aliases if these are used in a sequence object.

SQL grouping sets and supergroups

Grouping sets are supported by this product.

A grouping-sets specification allows multiple grouping clauses to be specified in a single statement. This can be thought of as the union-all of two or more groups of rows into a single result set. It is logically equivalent to the union-all of multiple subselects with the `GROUP BY` clause in each subselect corresponding to one grouping set. A grouping set can be a single element or can be a list of elements delimited by parentheses and commas, where an element is either a grouping-expression or a supergroup. Grouping sets allow the computing of groups with just a single pass over the base table.

A supergroup stands for a `ROLLUP` clause, a `CUBE` clause, or a grand-total clause. Db2 11 for z/OS also introduces a new aggregate function named `GROUPING`. This function is used in conjunction with grouping sets and supergroups to return a value that indicates whether a row that is returned in a `GROUP BY` answer set is a null value generated by a grouping set.

Differences in floating-point error tolerances

For queries that involve floating-point aggregates, you can expect IBM Db2 Analytics Accelerator for z/OS to produce results that are different from those delivered by inhouse Db2 for z/OS processing.

Reason

You know that the same query processed twice by Db2 for z/OS can give you different results. This is due to the fact that the results depend on the order in which the rows or tuples are processed. It is not possible to determine the order that Db2 will choose beforehand.

In the case of IBM Db2 Analytics Accelerator for z/OS, another sort of result deviation occurs, which is caused by the different methods of representation for floating-point numbers. Db2 for z/OS uses hexadecimal floating-point (HFP) numbers, whereas IBM Db2 Analytics Accelerator for z/OS uses IEEE754 floating-point numbers.

When a query is routed from Db2 for z/OS to IBM Db2 Analytics Accelerator for z/OS, all processing takes place on the accelerator.

The main difference between these two methods is the base or radix of the exponential part. HFP uses a base of 16. IEEE754 uses a base of 2. The base of 16 requires the exponent of HFP numbers to be increased in steps of 4 during the calculation of an aggregate. IEEE754 numbers, on the other hand, need to be increased just in steps of 1. The multiplier in front of the exponent (mantissa) of an HFP number loses 4 bits at the right end each time the exponent is increased. The mantissa of IEEE754 numbers loses just 1 bit. The difference in error thus lies in the difference of three bits, and the amount per step depends on the values of these. If the values of the three bits are all zero, the error difference is also zero. If all values are 1, the difference in error is maximized. Because of the lower exponential increase per step, IEEE754 is the more precise method of representation.

Whether the difference in precision matters much mainly depends on the number of rows or tuples (n) used for floating-point aggregates. Naturally, if n gets higher, the result deviation or error increases, too. How high the total deviation will be is hard to quantify as it further depends on the type of the mathematical function that is used (compare, for example, a simple addition with an operation that takes the square of each number involved).

Important: In contrast to HFP numbers of the type REAL in Db2 for z/OS, which might range from $-7.2E+75$ to $+7.2E+75$, IEEE754 is only capable of processing REAL floating point numbers between $-3.4E+38$ and $3.4E+38$. This means that it is not possible to load an accelerator-shadow table with REAL values outside the range of $-3.4E+38$ to $3.4E+38$.

Also bear in mind that HFP and IEEE754 have different minimum and maximum values for REAL and DOUBLE numbers:

Table 15. Minimum and maximum REAL and DOUBLE values (HFP and IEEE754)

HFP (Db2 for z/OS)		IEEE754 (IBM Db2 Analytics Accelerator for z/OS)	
REAL	DOUBLE	REAL	DOUBLE

Table 15. Minimum and maximum REAL and DOUBLE values (HFP and IEEE754) (continued)

	HFP (Db2 for z/OS)		IEEE754 (IBM Db2 Analytics Accelerator for z/OS)	
Min.	$\sim -7.2 * 10^{+75}$	$\sim -7.2 * 10^{+75}$	$\sim -3.4 * 10^{+38}$	$\sim -1.79 * 10^{+308}$
Max.	$\sim +7.2 * 10^{+75}$	$\sim +7.2 * 10^{+75}$	$\sim +3.4 * 10^{+38}$	$\sim +1.79 * 10^{+308}$
Min. positive	$\sim +5.4 * 10^{-79}$	$\sim +5.4 * 10^{-79}$	$\sim +1.18 * 10^{-38}$	$\sim +3.4 * 10^{-308}$
Max. negative	$\sim -5.4 * 10^{-79}$	$\sim -5.4 * 10^{-79}$	$\sim -1.18 * 10^{-38}$	$\sim -3.4 * 10^{-308}$

Chapter 4. IBM Db2 Analytics Accelerator Studio

You control the product from IBM Db2 Analytics Accelerator Studio, which is the administration client for IBM Db2 Analytics Accelerator.

Instead of using IBM Db2 Analytics Accelerator Studio, you can also control and operate the product by using Db2 commands and the stored procedures that come with the product.

Launching an IBM Db2 Analytics Accelerator Studio function causes the administration client to connect to your data server, which in turn calls a stored procedure that executes the corresponding command on the accelerator. The accelerator is thus always controlled and maintained indirectly via the data server.

In most cases, you do not have to change existing database applications to use IBM Db2 Analytics Accelerator. If an accelerator has been properly deployed and tables have been added to the accelerator, queries that come from database applications are usually routed automatically from your data server to the accelerator provided that this results in shorter response times.

See the following diagram:

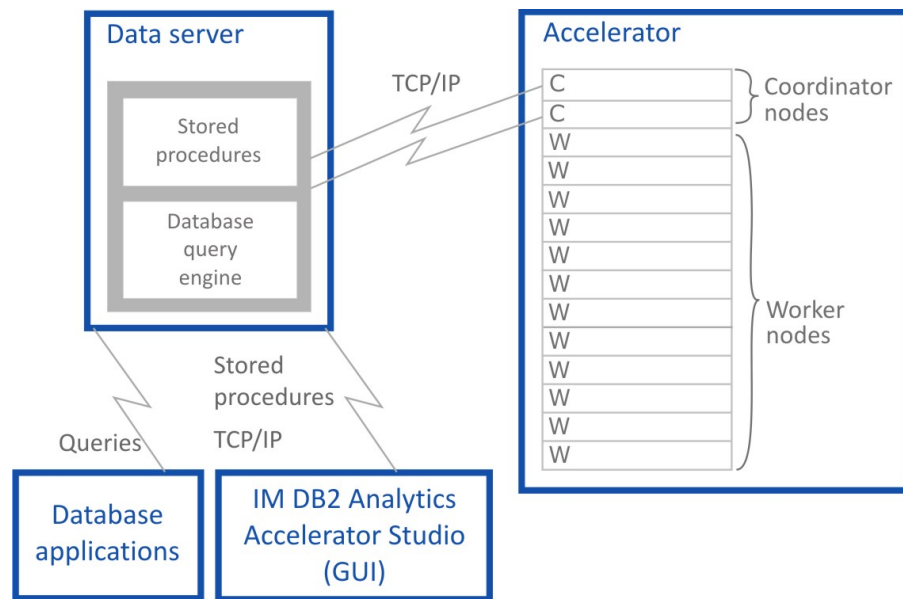


Figure 4. IBM Db2 Analytics Accelerator setup

Chapter 5. Interface elements of IBM Db2 Analytics Accelerator Studio

Here is a brief description of the interface elements that show IBM Db2 Analytics Accelerator Studio functionality.

Note: When you start IBM Db2 Analytics Accelerator Studio for the first time, you see the Welcome screen. To see the main window of IBM Db2 Analytics Accelerator Studio with the principal functions and controls, just close the Welcome screen by clicking the cross-shaped icon on the tab of the Welcome screen (usually positioned on the upper left).

Menu bar

When IBM Db2 Analytics Accelerator Studio has been opened, specific menus or menu choices are added to the menu bar of the Accelerator perspective. In addition, some standard menu choices of the Data perspective might have a different effect when IBM Db2 Analytics Accelerator Studio is loaded.

Edit > Select All

Selects all elements in the currently active view.

Window > Open Perspective > Other > Accelerator

Opens the Accelerator perspective, which shows IBM Db2 Analytics Accelerator Studio (if it has been installed).

Window > Reset Perspective

Restores all default views of IBM Db2 Analytics Accelerator Studio to their default size and position in the IBM Data Studio main window.

Window > Show View

Opens a submenu that allows you to reopen a currently closed IBM Db2 Analytics Accelerator Studio view.

Help > Welcome > Overview > IBM Db2 Analytics Accelerator Studio

Opens the *Introduction* topic in the online help system.

Help > Welcome > First Steps > Demonstration Videos

Opens the *Demonstration videos* topic in the online help system, which lists the available videos. Clicking a link starts the playback of a video.

Help > Welcome > Workbench

Closes the Welcome screen and displays the main view of IBM Db2 Analytics Accelerator Studio.

Administration Explorer

The Administration Explorer shows your database connection profiles and various objects like schemas, tables, and stored procedures that are defined in the database catalogs of your data servers.

The Administration Explorer is the navigation tree that you can see in Figure 5 on page 60. It is usually located on the left-hand side.

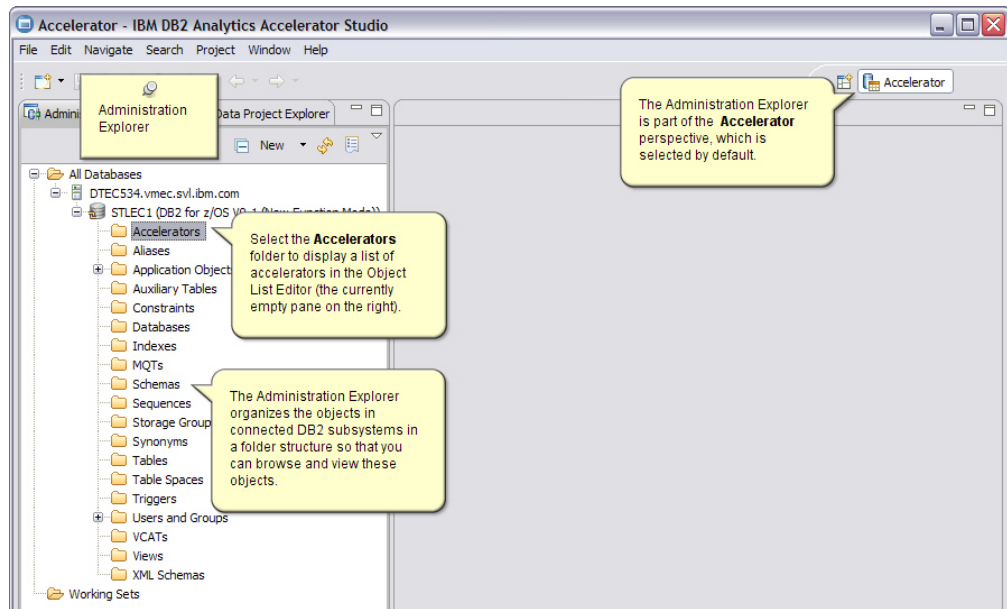


Figure 5. Administration Explorer

Menus

Depending on your selection in the Administration Explorer, the menu (right-click action) offers the following menu choices specific to IBM Db2 Analytics Accelerator for z/OS:

An existing database connection profile

When you right-click an existing database connection profile, you can select the following items from the menu:

Connect

Connects to the database that is referred to in the profile. You are asked for the password of the logon user if the option **Save password** is not selected in the profile.

Disconnect

Disconnects an active database connection.

Ping

Issues a **ping** command to check whether the network connection to the database server works.

Manage Connection

Gives you access to a submenu from which you can rename, delete, or copy (duplicate) the selected profile.

Properties

Opens the Properties sheet of the existing profile so that you can edit it.

Folders

When a database server has been connected to, you can see the various objects on that server, which are organized in folders. Most folders pertain to database objects that are created or configured in Db2 for z/OS. An exception is the **Accelerators** folder, which pertains to IBM Db2 Analytics Accelerator for z/OS only.

Accelerators folder

Selecting the **Accelerators** folder opens the Object List Editor in the main view (pane on the right).

The context menu of the Accelerators folder offers the following choices:

Add Accelerator

Starts the Add Accelerator wizard, which allows you to add an accelerator to the system configuration.

Add Virtual Accelerator

Starts the Add Virtual Accelerator wizard, which allows you to add a virtual accelerator to the system configuration.

Save Trace

Opens the Save Trace window, in which you can select the trace information that you want to save and either save it to a file or submit it to IBM.

Refresh

Refreshes the list of accelerators in the Object List Editor (right pane).

Object List Editor

Apart from various other functions, the Object List Editor contains a table with an entry for each accelerator that has been configured. The Object List Editor opens when you select the Accelerators folder in the Administration Explorer.

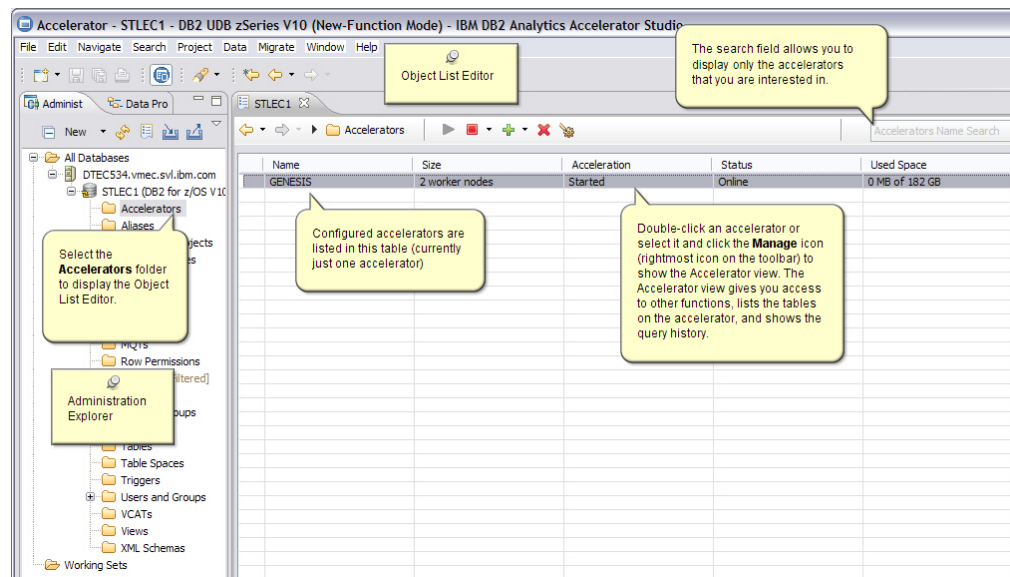


Figure 6. Object List Editor

Buttons in the Object List Editor

On top of the table in the Object List Editor, you find the following buttons:



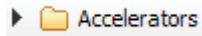
The arrow pointing to the left returns to the previous display in the Object List Editor.

The downward-pointing arrow allows you to select a particular earlier display from a list.



To switch forward to a display in the Object List Editor that was opened later than the current one.

The downward-pointing arrow allows you to select a particular later display from a list.



To change the type of object that is currently displayed in the Object List Editor. The currently selected type is displayed to the right of the twistie (here: accelerators).



To enable the selected accelerator in Db2 for z/OS.

Notes:

- Multiple selections are not possible here. You can enable only one accelerator at a time.
- If the accelerator is attached to a Db2 data sharing group, acceleration is enabled for all members of the group.



To disable the selected accelerator in Db2 for z/OS.

The downward-pointing arrow on the right opens a menu from which you can select the following choices:

Stop Same as clicking the button on the toolbar

Stop Force

A function that disables the selected accelerator immediately, that is, without waiting for the completion of not yet finished queries.

Notes:

- Multiple selections are not possible here. You can disable only one accelerator at a time.
- If the accelerator is attached to a Db2 data sharing group, acceleration is disabled for all members of the group.



To remove the selected accelerator from the system configuration.

Note: Multiple selections are not possible here. You can remove only one accelerator at a time.



To open the Accelerator view for the selected accelerator, in which you can define, remove, enable, or disable accelerator-shadow tables and invoke various other functions.

Clicking this button has the same effect as double-clicking an accelerator name in the table of the Object List Editor.

Note: Multiple selections are not possible here. You can only open the Accelerator view for one accelerator at a time.

The search field makes it easier to find particular accelerators if the list is long. Type the names of accelerators in this field, either fully or partially and press Enter. The table will display just those accelerators bearing or starting with that name.

Column information

The table in the Object List Editor provides various information about the accelerators, such as:

Name Name of an accelerator

Size Size of an accelerator, that is, its number of worker nodes (3, 6, or 12).

Acceleration

Whether acceleration has been *started* or *stopped* for the Db2 subsystem that is associated with an accelerator.

Note: The status of an accelerator cannot not been determined if acceleration has been stopped for a Db2 subsystem.

Status Current[®] status or state that an accelerator is in. The following states are possible:

Cluster Is Initializing

Interim state that is shown when a new software version is applied. When this process has finished, the accelerator goes to *Fully Operational*.

Fully Operational

Ready to be enabled in Db2 for z/OS.

Online

Ready for processing.

Running

The accelerator is processing queries.

Stopping

Interim state that is shown when you click the **Stop** button or issue the **STOP -ACCEL** command from Db2 for z/OS. During the *Stopping* phase, already started queries are completed.

Stopped

The accelerator is currently disabled in Db2 for z/OS.

Unknown

The current status of the accelerator cannot be determined.

Used Space

Overall size of the accelerator-shadow tables on an accelerator

Active Queries

The number of queries that is currently being processed on an accelerator

Replication

Indicates whether the incremental update function is enabled for this accelerator in the DB2 subsystem. Possible values:

Disabled

The incremental update function has not been configured for this accelerator and is therefore not available.

Error

The incremental update function has been configured, but an error occurred. In most cases, this means that a component cannot be reached. Therefore, it cannot be started or controlled. The event log might provide useful information about the root cause of the error.

Started

The incremental update function is running and updates are applied to accelerator-shadow tables that have been included in the process.

Stopped

The incremental update function is available and accelerator-shadow tables have been included in the process. However, the function is not running right now.

Accelerator view

To show the tables on an accelerator, view the query history, or access various other functions, double-click an accelerator name in the Object List Editor, which opens the related Accelerator view.

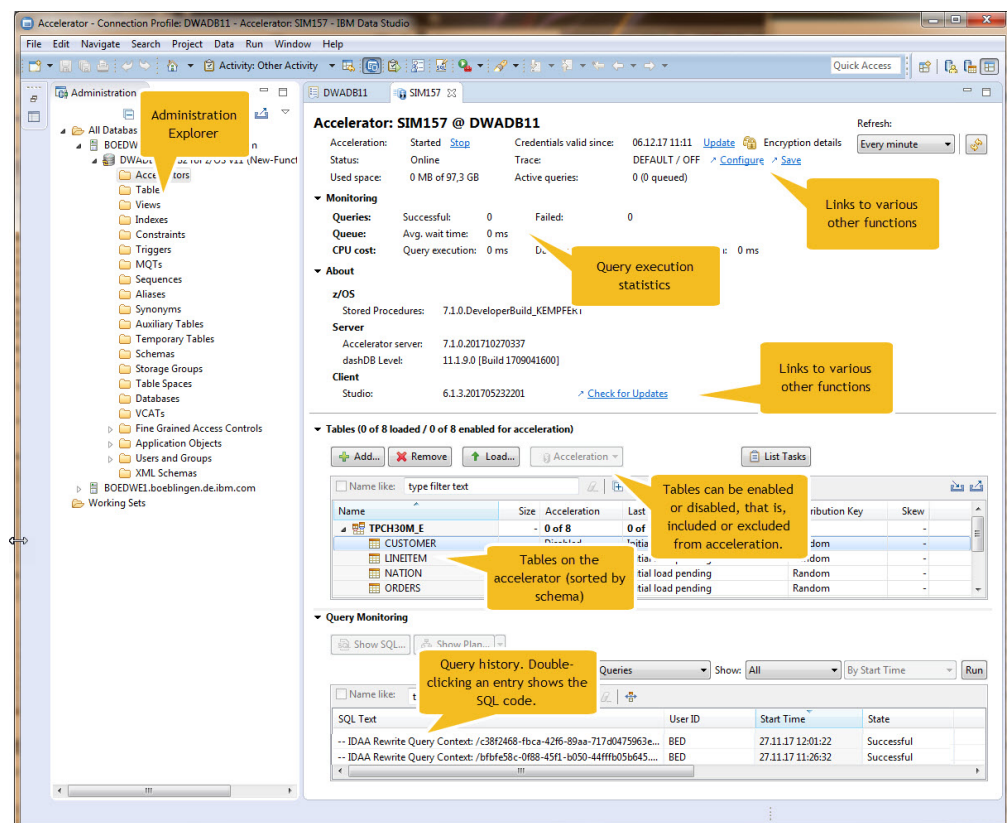


Figure 7. Accelerator view

Status information and functional links

The header of the Accelerator view contains status information and links to related functions.

Table 16. Header information

Label	Description
Acceleration	<p>Whether the accelerator is currently enabled in Db2 for z/OS. To change the current status, click Start or Stop.</p> <p>If the accelerator is attached to a Db2 data sharing group, the behavior is different. The status <i>Started</i> is displayed if acceleration has been started for at least one member of the data sharing group. Other statuses are shown only if the status is the same for all members of the data sharing group. If the status of individual members is different, a warning icon is displayed.</p> <p>If you move the mouse pointer over the displayed status or the warning icon, a pop-up window shows the acceleration status for all members of the data sharing group.</p> <p>An additional functional link labeled Change is displayed for data sharing groups. Clicking it opens a window in which you can change the acceleration status of each member in the data sharing group.</p>
Status	<p>The status of the accelerator hardware.</p> <p>If the accelerator is attached to a Db2 data sharing group, you can move the mouse pointer over the status information to display the status of each member in the data sharing group.</p>
Used space	The amount of disk space that is taken up by the tables, in proportion to the available space on the accelerator.
Replication	Only visible if incremental updates are configured. A status of Started indicates that the incremental update process is running for tables included in the process. Stopped indicates that the incremental update process has been suspended. To change the current status, click Start or Stop .
Credentials valid since	Shows the date on which the last authentication of the accelerator took place. To renew the authentication, click Update .
Encryption details	If you click the Encryption details , the Encryption Details window opens, which shows information about the use of encryption for the selected accelerator. This includes the status of hard-disk encryption, the status of encryption of data in motion, a list of encrypted IPsec tunnels (IP connections), and a list of the certificates that are used.
Trace	Shows the trace profiles for the accelerator and the stored procedures that are currently in use. To change a profile, click Configure . To change the settings for the saving of trace information to a file, click Save .
Active queries	The number of accelerated queries that are currently being processed. In parentheses, you see the number of queued queries.

Table 16. Header information (continued)

Label	Description
Replication latency	<p>Only visible if incremental updates are configured. This is not the case in Figure 7 on page 64. The value indicates how much the incremental update function is lagging behind. Possible values:</p> <p>Stopped Incremental updates have been stopped for the entire accelerator.</p> <p>Low It will take less than three minutes to propagate recent changes to the corresponding accelerator-shadow tables.</p> <p>Medium Recent changes to the original DB2 tables will be propagated to the corresponding accelerator-shadow tables within three and six minutes.</p> <p>High It will take six minutes or longer to apply recent changes to DB2 tables to the corresponding accelerator-shadow tables.</p> <p>When you move the mouse pointer over the latency value, a pop-up window shows the following information:</p> <p>Target Agent Operations completed by the incremental update function.</p> <p>Inserts Number of row inserts in accelerator-shadow tables that have been included in the incremental update process.</p> <p>Deletes Number of row deletions from accelerator-shadow tables that have been included in the incremental update process.</p> <p>Remember:</p> <ul style="list-style-type: none"> • An UPDATE is a DELETE followed by an INSERT, so these two figures account for all incremental updates that have been applied. • Since an UPDATE in DB2 is a DELETE followed by an INSERT on the accelerator side, the values can be different from those in Db2 for z/OS (see Source Agent). Furthermore, multiple operations on a single row within a short time might be reported as just one single operation. • If a load operation is in progress and lock mode Row has been selected, the original DB2 tables are not locked during the load. Hence it is possible to INSERT new rows into the DB2 tables even though a load is taking place. If incremental updates are enabled for the same tables, the incremental update function stores these new rows in the spill queue until the load has finished, and applies these changes thereafter (draining the spill queue). In the course of a drain, a DELETE and an INSERT operation is carried out for each row in the affected accelerator-shadow tables. The DELETE deletes the row that was inserted by the previous load operation, and the INSERT adds this row again (applies the latest change) . This has the effect that the number Deletes increases in cases like this. <p>Clicking Show events opens the event viewer, which keeps a record of all transactions that are related to incremental updates.</p>

Table 17. Header information (continued)

Label	Description
Replication latency	<p>Source Agent Number of data changes to relevant Db2 for z/OS tables (tables for which accelerator-shadow tables have been defined and that are included in the incremental update process).</p> <p>Inserts Number of row inserts in the relevant Db2 for z/OS tables.</p> <p>Updates Number of updates to rows in the relevant Db2 for z/OS tables.</p> <p>Deletes Number of row deletions from the relevant Db2 for z/OS tables.</p> <p>Latency Repetition of the Replication latency value.</p> <p>Clicking Show events opens the event viewer, which keeps a record of all transactions that are related to incremental updates.</p>

Monitoring

The **Monitoring** section provides the following information:

Table 18. Information in the **Monitoring** section

Label	Description
Queries	<p>Metrics related to queries since the start of the currently viewed accelerator, divided into the following categories:</p> <p>Successful Number of queries that ran successfully</p> <p>Failed Number of queries that failed</p>
Queue	<p>Metrics related to queuing since the start of the currently viewed accelerator:</p> <p>Avg. wait time Average time that queries had to stay in the queue. The basis for the calculation is all queries that were processed during the last minute.</p>
CPU Cost	<p>CPU time that was used since the start of the currently viewed accelerator, with regard to the connected Db2 subsystem. The total CPU time is divided into the following categories:</p> <p>Query execution CPU time needed for query processing</p> <p>Data maintenance CPU time needed for load operations</p> <p>Replication CPU time needed for incremental updates</p>

About

The following information is displayed in the **About** section:

Table 19. Information in the **About** section

Label	Description	Information blocks	Description
z/OS	Information about components on the z/OS data server	Stored procedures	The currently installed version of the IBM Db2 Analytics Accelerator stored procedures
Server	Information about software components that are installed on the IBM Integrated Analytics System or as part of your Db2 Analytics Accelerator on Z	Accelerator server	The version of the accelerator software that is currently in use.
		Level	The current version of the database run by your accelerator.
Access Server	Only visible if incremental updates are configured. This is the version of the CDC Access Server, the central component in the incremental update process, which manages the list of subscriptions (incremental update requests) and controls the agents involved in the process.		
Replication engine	Only visible if incremental updates are configured. This is the version of the CDC Apply Agent, which propagates data changes to the IBM Integrated Analytics System, according to the instructions received from the CDC Access Server.	Unicode sort order	Only visible if the connected DB2 subsystem or data sharing group is a DB2 11 for z/OS database. The value reflects the sort order that was selected for tables on the accelerator. Possible values are UTF-8 binary and UTF-16 binary . The choice has an impact on how supplementary Unicode characters are sorted in query results.
Client	Information about your IBM Db2 Analytics Accelerator Studio installation	Studio	The currently installed version of IBM Db2 Analytics Accelerator Studio. Click Check for Updates to look for a newer version.

Tables

The **Tables** section in the Accelerator view for contains buttons and a listing of the tables on the accelerator.

Table 20. Buttons in the **Tables** section

Button	Description
Add	Starts the Add Tables wizard.
Remove	Removes the selected tables from the accelerator.
Load	Opens the Load Tables window, in which you can select accelerator-shadow tables for loading and set the lock scope for the load process.
Acceleration	Opens a pull-down menu, from which you can enable or disable the selected accelerator-shadow tables so that these are included or excluded from query acceleration.
Replication	Only visible if incremental updates are configured. This is not the case in Figure 7 on page 64. Clicking the button opens a pull-down menu from which you can enable or disable incremental updates for selected accelerator-shadow tables.
List Tasks	Opens the List Tasks window, which displays the running tasks on the currently viewed accelerator, such as table load processes. To cancel a running task, select it in the list and click Cancel Selected Task .

Table 21. Icons and columns in the table of tables




Item	Description
	Icon representing the schema of a table
	Icon representing an accelerator-only table
	Icon representing an accelerator-shadow table
Name	Column containing the table names
Size	Column containing the table sizes
Acceleration	Column that shows whether acceleration is enabled for a table
Last Load	Column that shows the timestamp of the last successful load for each table or the current table state. Because accelerator-only tables are not loaded by IBM Db2 Analytics Accelerator functions, the column always shows the table state (for example <i>AOT ready</i>) for this type of table.
Storage Saver Partition	<p>Column that provides information about the archived partitions of a table. The column is visible only if tables with archived partitions exist.</p> <p>At the schema level, the column shows how many partitions out of the total number of partitions were archived. At the table level, it shows the compressed data size of the archived partitions and the number of rows in these partitions. If archiving is in progress for a partition, the columns shows <i>In Progress</i>.</p>
Replication Since	<p>Column that shows the last incremental-update capture-point (a timestamp) for incrementally updated accelerator-shadow tables or <i>Disabled</i> for the tables that are not updated in that way.</p> <p>The column shows <i>Suspended</i> when a table load has started and incremental updates are therefore on halt, and <i>Refreshing</i> when table changes are being applied, be this through the ongoing load process or the draining of incremental updates from the spill queue.</p>

Table 21. Icons and columns in the table of tables (continued)

Item	Description
Distribution Key	Column that shows <i>Random</i> if random distribution is used or the names of distribution-key-columns if a distribution key is used.
Skew	Column that shows the skew value for each table on the accelerator. The skew value is a metric for the distribution imbalance of table rows across the snippet-processing units (SPUs) or worker nodes. The value is the difference between the smallest and the largest data slice for a table in megabytes (MB).

Query Monitoring

The **Query Monitoring** section shows an extensive number of additional buttons, information columns, and other controls. For a description, see “Selecting and ordering query monitoring columns” on page 99.

Stored procedure call

To display or refresh the list of schemas and tables, IBM Db2 Analytics Accelerator Studio invokes the **SYSPROC.ACCEL_GET_TABLES_INFO** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Related concepts:

“Tables states” on page 9

“How to select tables for query acceleration” on page 6

Related tasks:

“Defining tables on an accelerator” on page 79

“Loading tables” on page 83

“Enabling tables for query acceleration” on page 88

“Specifying or changing a distribution key” on page 90

“Including or excluding tables from incremental updates” on page 125

“Selecting and ordering query monitoring columns” on page 99

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Chapter 6. Tasks

See an overview of the tasks that you can complete by using IBM Db2 Analytics Accelerator Studio.

Prerequisites

Before you go about a task, check whether you meet the requirements for running IBM Db2 Analytics Accelerator Studio:

- Network connections must exist between the database server and the machine on which IBM Db2 Analytics Accelerator Studio is installed. In addition, network connections must exist between the database server and the accelerators.
- At least one accelerator (hardware) must be installed and connected to your database server.
- The appropriate Program Temporary Fixes (PTFs) must be installed so that IBM Db2 Analytics Accelerator for z/OS can connect to your database management system.
- IBM Db2 Analytics Accelerator for z/OS stored procedures must exist on the database server.
- IBM Db2 Analytics Accelerator for z/OS jobs have been run to configure and integrate the accelerator hardware and software into your database environment.

Creating a database connection profile

Create a database connection profile to gain access to a Db2 subsystem on a database server. A Db2 subsystem houses one or more databases, in which the source data for query acceleration (schemas and tables) is kept.

About this task

In IBM Db2 Analytics Accelerator Studio, the connection information is stored in profiles for reuse. Having created a profile, you can reconnect to a database by double-clicking the icon representing the profile in the Administration Explorer.

Procedure

1. On the header of the Administration Explorer on the left, click the downward-pointing arrow next to **New** and select **New Connection to a Database**.
2. In the New Connection window, decide how to name the database connection profile:
 - To use the name of the database server that you want to connect to, leave **Use default naming convention** selected.
 - To choose a different name, clear **Use default naming convention**, and type the name in the **Connection Name** field.
3. From the **Select a database manager** list, select **Db2 for z/OS**.
4. Make sure that in the **JDBC driver** drop-down list, **IBM Data Server Driver for JDBC and SQLJ (JDBC 4.0) Default** is selected.
5. In the **Location** field, type the name of the database server that you want to connect to.

Tip: To determine the **Location**, **Host** name, and **Port number**, a Db2 for z/OS systems programmer or database administrator can issue a DIS DDF command.

6. In the **Host** field, type the host name or IP address of the data server on which the database server is located.
7. In the **Port number** field, you see that port number 446 is selected by default. Leave this setting unless the database server uses another port.
8. Select **Retrieve objects created by this user only** if you want to restrict database access to the databases, schemas, tables, and other objects that were created by the logon user. If you do not select this option (default), IBM Db2 Analytics Accelerator Studio will show and make selectable all databases, schemas, and tables that the logon user has access to, including those to which this user might have only read access.
9. In the **User name** field, type the user ID that you want to use to log on to the database server. Note that you can only use IBM Db2 Analytics Accelerator Studio successfully if this user has sufficient rights to run the stored procedures behind the IBM Db2 Analytics Accelerator Studio functions. The section *Appendix C. Required access rights* in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference* lists the privileges that are required to run a particular stored procedure. If you are uncertain, use an ID with SYSADMIN authority.

In many organizations, it is a common practice to have personal user IDs with restricted authority and special-purpose user IDs (groups in most cases) with extensive privileges in a certain field. IBM Db2 Analytics Accelerator Studio supports this practice in that you can specify a secondary user ID, which might have the privileges that your logon user ID lacks, such as the privilege to run stored procedures. If the secondary ID is a group user ID, the logon user must of course be a member of that group. To specify a secondary user ID, follow these steps:

- a. On the Connection Parameters page, click the **Optional** tab.
- b. In the **Property** field, type the following statement:
currentSQLID
- c. In the **Value** field, type the secondary user ID.
- d. Click **Add**.
- e. Click the **General** tab to return to that page and complete the logon.
10. In the **Password** field, type the password belonging to the logon user ID.
11. Leave the **Save password** check box deselected.
Attention: You can select **Save password** to avoid having to enter the password each time that you want to work with the database server. This, however, is not recommended because only a lightweight encryption is applied when the password is stored on your local hard disk.
12. Leave the **Default schema** field blank.
13. Click **Test Connection** to check if you can log on to the database server.
14. Click **Finish**.

Results

After creating the profile, IBM Db2 Analytics Accelerator Studio automatically connects to the Db2 subsystem.

What to do next

If it takes too long to load all objects of the Db2 subsystem into the Administration Explorer (more than one minute), you can set a filter to limit the number of schemas to be loaded:

1. In the Administration Explorer, right-click the icon representing the Db2 subsystem (database symbol).
2. Select **Properties** from the menu.
3. In the Properties for ... window, select **Default Schema Filter**.
4. Clear the **Disable filter** check box. This activates the filter controls.
5. From the **Name** drop-down list, select a suitable filter mask. In the adjacent text field, type the filter string. For example, to exclude all schemas whose names starts with the characters BLU:
 - a. From the **Name** drop-down list, select **Does not start with the characters**.
 - b. In the text field, type BLU.
6. Click **Apply**.

Connecting to a database server


To connect to a Db2 subsystem, you just double-click the icon representing the subsystem. The icon is the result of successfully creating a database connection profile.

Before you begin

If it takes too long to load all objects of the Db2 subsystem into the Administration Explorer (more than one minute), you can set a filter to limit the number of schemas to be loaded:

1. In the Administration Explorer, right-click the icon representing the Db2 subsystem (database symbol).
2. Select **Properties** from the menu.
3. In the Properties for ... window, select **Default Schema Filter**.
4. Clear the **Disable filter** check box. This activates the filter controls.
5. From the **Name** drop-down list, select a suitable filter mask. In the adjacent text field, type the filter string. For example, to exclude all schemas whose names starts with the characters BLU:
 - a. From the **Name** drop-down list, select **Does not start with the characters**.
 - b. In the text field, type BLU.
6. Click **Apply**.

Procedure

In the Administration Explorer, double-click the icon representing the Db2 subsystem. For example:  STLEC1

Results

When the connection was successful, the icon representing the subsystem changes and the database object types, such as table spaces or tables, are displayed in a folder hierarchy in the Administration Explorer.

Example

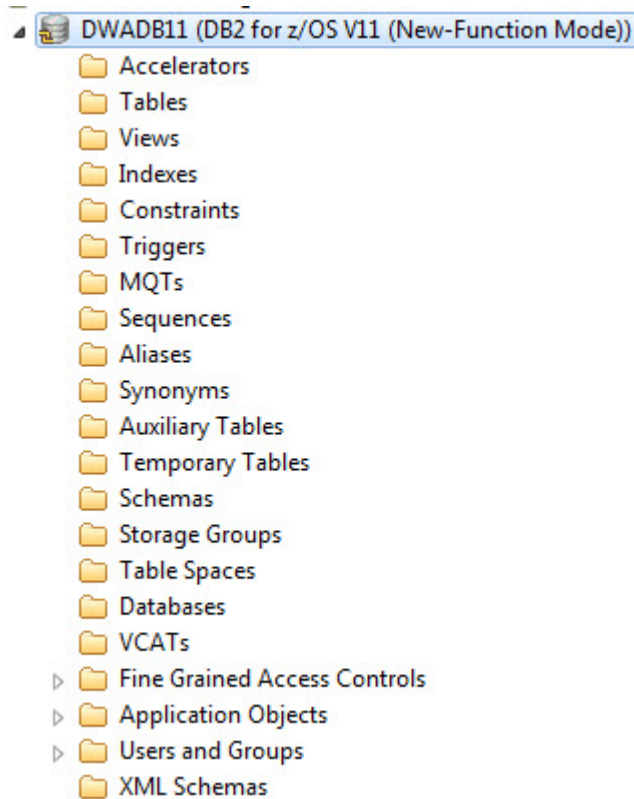


Figure 8. Database server connection in the Administration Explorer

Exploring a database server

When a connection to a database server has been established successfully by double-clicking the corresponding profile icon, the Administration Explorer shows the database object types in a folder hierarchy. You can navigate this folder hierarchy to obtain information about the objects, for example, to find out which schemas and tables the connection gives you access to. Selecting a folder shows the corresponding objects in the Object List Editor.

Procedure

1. If necessary, expand the node representing the database server connection.
2. Expand the single node underneath, which represents the database server.
3. Navigate downwards in the folder hierarchy to see the objects on the database server.

Results

When IBM Db2 Analytics Accelerator for z/OS has been set up correctly, the Administration Explorer shows a folder called **Accelerators**. You can open this folder to display the list of configured accelerators. Double-clicking an accelerator name opens a view that shows the details of this accelerator.

Adding accelerators

Adding an accelerator to your configuration is a two-step process. First, you must obtain a pairing code from the IBM Db2 Analytics Accelerator Console. Second, you enter this pairing code along with the IP address and the name of the accelerator in the Add Accelerator wizard of IBM Db2 Analytics Accelerator Studio.

Obtaining the pairing code for authentication

Communication between an accelerator and a Db2 subsystem requires both components to share credentials. These credentials are generated after you submit a temporarily valid pairing code. This step is required each time you add a new accelerator. The following steps describe how to obtain the pairing code.

About this task

Note: You can renew the authentication for an existing accelerator without having to use a new pairing code. To do so, click the **Update** link in the Accelerator view.

The steps *Obtaining the pairing code for accelerator authentication* and *Adding Accelerators* (next topic) belong together, but are seldom carried out by the same person. Since the pairing code obtained from the IBM Db2 Analytics Accelerator Console is only valid for a limited time (30 minutes by default), the persons operating the console and IBM Db2 Analytics Accelerator Studio must coordinate the steps.

To obtain the pairing code, the **SYSPROC.ACCEL_UPDATE_CREDENTIALS** stored procedure is run on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. Ask the network administrator or the person who did the TCP/IP setup for the IP address of the accelerator. Make a note of this information. You need to enter it as you complete the steps that follow.
For IBM Db2 Analytics Accelerator on an IBM Integrated Analytics System, this is the virtual IP or wall IP address.
For Db2 Analytics Accelerator on Z, this is the IP address of the network that you labeled DB2 in the Appliance Installer.
2. Log on to the IBM Db2 Analytics Accelerator Console by using **telnet** or **ssh**. The preferred method is **ssh**. For more information:
 - Appendix B, “Using telnet to log on to the IBM Db2 Analytics Accelerator Console,” on page 165
 - Appendix C, “Using ssh to log on to the IBM Db2 Analytics Accelerator Console,” on page 167
3. Press the Pause key, then Enter to display the following screen:

```

*****
*      Welcome to the IBM Db2 Analytics Accelerator Console
*****

You have the following options:
(0) - (Menu) Manage Configuration Console Users
(1) - (Menu) Run Accelerator Functions
(2) - (Menu) Manage Incremental Updates
(3) - (Menu) Manage Encryption of Data in Motion

-----
(x) - Exit the Configuration Console

```

4. Type 2 and press Enter to display the submenu:

```

main -> 3
-----
You have the following options:

(0) - Go back one level
(1) - Obtain pairing code, IP address, and port
(2) - List paired Db2 subsystems
(3) - Clear query history
(4) - Restart accelerator process
(5) - Stop backend database
(6) - Start backend database
(7) - Set the Db2 subsystem for time synchronization

```

5. Type 1 and press Enter:
6. When the message Specify for how long you want the pairing code to be valid. is displayed, enter an appropriate integer to specify the validity period in minutes. The time that you choose must be sufficient for you or a coworker to go to the workstation that runs IBM Db2 Analytics Accelerator Studio, start the Add New Accelerator wizard, and enter the information that is returned by the console. Values from 5 to 1440 are allowed. If you just press Enter, you accept the default of 30 minutes.

```

Press <return> to accept the default of 30 minutes.
Cancel the process by entering 0.

Accelerator pairing information:
Pairing code   : 6048
IP address    : 9.152.85.192
Port          : 1400
Port (AT-TLS) : 11400
Valid for     : 30 minutes

Press <return> to continue

```

Important: A pairing code is valid for a single try only. Furthermore, the code is bound to the IP address that is displayed on the console.

7. Make a note of the following information on the console:
 - Pairing code
 - IP address
 - Port (for unencrypted network communication). Use this port if you are not sure.
 - Port (AT-TLS) (for encrypted network communication).

The use of encryption requires extra configuration steps on the accelerator and on the participating z/OS L Pars.

8. Press Enter to return to the main menu of the console.
9. Type x and press Enter to exit the console and close the telnet session.

Related tasks:

“Completing the authentication using the Add Accelerator wizard”

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Completing the authentication using the Add Accelerator wizard

To complete the authentication, you specify the IP address, the port number, and the pairing code in the Add Accelerator wizard.

Before you begin

Make sure that the following conditions apply:

- You need privileges to run Db2 administration commands and stored procedures on z/OS. If you created a power user as suggested in the *Installation Guide*, the power user will have the required privileges.
- You have a valid pairing code. The pairing code, which is of temporary validity, can be obtained by using the IBM Db2 Analytics Accelerator Console.

Attention: Do not give ordinary users SELECT authorization on the SYSIBM.USERNAMES table because this allows the users to see the authentication information in the SYSIBM.USERNAMES.NEWAUTHID column.

About this task

You can renew the authentication for an existing accelerator without having to use a new pairing code. To do so, click the **Update** link in the Accelerator view.

Attention: Making a new backup of your Db2 catalog tables is strongly recommended after each authentication update because restoration processes in your Db2 subsystem can make an accelerator unusable. This happens if you must restore your Db2 catalog and the backup of the catalog was made before the last update of the accelerator credentials. In this case, the latest authentication information will not be in the catalog tables of the backup, and so the accelerator can no longer be used.

For the completion of this task, the following stored procedures are run on your data server:

- SYSPROC.ACCEL_TEST_CONNECTION
- SYSPROC.ACCEL_ADD_ACCELERATOR

For information about the privileges that are required to run these procedures and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related information** at the end of this section.

Procedure

1. Select the **Accelerators** folder in the Administration Explorer.
2. On the menu bar of the Object List Editor, click the downward-pointing arrow next to the green plus sign.

3. From the drop-down menu, select **Add Accelerator**.
4. In the **Name** field, type a name for the accelerator. This name is automatically copied to the **Location** field.

The location name is the unique name of the accelerator in the SYSIBM.LOCATIONS table. Mostly, this is the same name as the accelerator name.

Restriction: An accelerator cannot be shared between two or more Db2 subsystems if the subsystems use the same location name. If you copy an entire subsystem, make sure to change the location name of the copy afterwards.

5. In the **Pairing code** field, type the pairing code.
6. In the **IP address** field, type the IP address of the accelerator.
7. In the **Port** field, type 1400. This is the fixed port for network communication between the z/OS data server and the accelerator.
8. Click **Test Connection** to check whether the accelerator with the given address can be connected to.
9. Click **OK**. A connection test is carried out.

Related tasks:

“Obtaining the pairing code for authentication” on page 75

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

“DRDA connection does not work” on page 156

Displaying accelerator networking details

To display the networking details of connected accelerators, such as the host names, IP addresses, and ports used, you can issue an SQL query.


About this task

The information can be extracted from certain Db2 catalog tables and system tables:

- SYSACCEL.SYSACCELERATORS
- SYSIBM.IPNAMES
- SYSIBM.LOCATIONS

Procedure

1. In the header of the Administration Explorer in IBM Db2 Analytics Accelerator Studio, click the downward-pointing arrow and select **New SQL script**.
2. Copy and paste the following query in the script window on the right:


```
SELECT A.ACCELERATORNAME, A.LOCATION, I.IPADDR, L.PORT
FROM SYSACCEL.SYSACCELERATORS AS A, SYSIBM.IPNAMES AS I, SYSIBM.LOCATIONS AS L
WHERE A.LOCATION = L.LOCATION AND I.LINKNAME = L.LINKNAME;
```
3. Click .
4. If no connection profile is selected in the Administration Explorer, the Select Connection Profile window comes up. If so, select the proper database connection profile (database server) in the Connections list.
5. Click **Finish**.
6. Click the **Result1** tab in the lower right corner. The requested information is displayed in a table.

Example

ACCELERATORNAME	LOCATION	IPADDR	PORT
BC6	BC6	9.116.85.193	1277
TF12	TF12	10.101.6.2	1400
TF3	TF3	10.101.6.6	1400

Defining tables on an accelerator

Using the Add Tables to Accelerator wizard, you can define accelerator-shadow tables on an accelerator.

Before you begin

- Make sure that at least one accelerator has been configured.
- Make sure that a database connection is associated with this accelerator.
- Make sure that your resource database contains tables.

About this task

- The maximum number of tables that you can define on a single accelerator is 27132.
- Table columns of the type CHAR and VARCHAR can be encoded in UNICODE, EBCDIC, or ASCII.

You can add tables with differently encoded columns (EBCDIC, ASCII, and Unicode) to a single accelerator. However, a single query can only reference columns in tables that all have the same encoding.

- IBM Db2 Analytics Accelerator for z/OS does not support EDITPROC encryption. SQL encryption and decryption functions, such as DECRYPT_CHAR(), require binary table columns, which cannot be used on an accelerator. Furthermore, data that is stored in Db2 with the EDITPROC encryption tool will not be encrypted on the hard disk of the accelerator. However, the data on an accelerator is encoded with a special algorithm for efficient compression and therefore cannot be easily extracted from the media.

This task is carried out by the **SYSPROC.ACCEL_ADD_TABLES** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure




1. In IBM Db2 Analytics Accelerator Studio, in the Administration Explorer, go to the **Accelerators** folder.
2. In the Object List Editor, double-click the accelerator that you want to define tables on.
3. In the Accelerator view, click **Add** on the toolbar, which is located above the list of tables.
4. In the Add Tables to Accelerator window, you see a list of all available schemas and tables under **Available tables**. If the list is long, first narrow the choice by using the **Name like** filter field. Type the names of schemas or tables in this field, either fully or partially, to display just schemas and tables bearing or starting with that name. The names of tables that have already been selected are greyed out.

5. Specify the tables. The list of selectable items is organized hierarchically. At first, you might only see the available schemas. If you click the twistie in front of a schema name, the tables belonging to the schema come into view.


You cannot select accelerator-only tables because these tables already exist on an accelerator. Entries for these tables are listed, but are grayed out. Apart from the different icon, you find the information *already added* in parentheses to identify the table as an accelerator-only table. Remember that accelerator-only tables are created by a CREATE TABLE ... IN ACCELERATOR SQL statement.

The **Unsupported Columns** column lists the unsupported columns in each Db2 table (if any). If possible, do not select tables with unsupported columns because such columns are not loaded on the accelerator. Queries referencing such columns cannot be accelerated. In most cases, a column is not supported because its data type cannot be handled by IBM Db2 Analytics Accelerator for z/OS.


Note: Detecting unsupported columns takes a long time if IBM Db2 Analytics Accelerator Studio has to connect to a database with numerous tables. To save this time, you can suppress the display of unsupported columns. To do so, go to **Window > Preferences > IBM Db2 Analytics Accelerator > Accelerator View** and clear the **Check for unsupported columns** check box.

- To define a single table on the selected accelerator, select its name under **Available tables**, in the **Name** column, and click  to move the column to the **Tables to be added** list. To select multiple tables, hold down the CTRL key and click the names of the tables that you want to add. To move all visible tables to the **Tables to be added** list, click .
- To correct your choice and remove a table from the **Tables to be added** list, select it in the **Name** column of that list and click  to move the table back to the **Available tables** list.

To remove all tables from the **Tables to be added** list, click .

Note: If a Db2 source table was created after IBM Db2 Analytics Accelerator Studio connected to a Db2 subsystem on the data server, you might be unable to locate this table in the list of available tables. In such a case, click .

The problem can also occur if you have defined a filter in the Administration Explorer or in the Properties section of a database connection profile. If so, proceed as follows:

- a. Close the Add Tables window.
 - b. Remove the filter.
 - c. Reopen the Add Tables window.
 - d. Click .
6. Click **Finish**.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Determining the status of an accelerator

See how to check the status of an accelerator.

Before you begin

- A database connection profile must exist for the database that is associated with the accelerator.
- You (your user ID) must be connected to the database.

About this task

This task is carried out by the **SYSPROC.ACCEL_CONTROL_ACCELERATOR** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click an accelerator to open the Accelerator view.

Results

The status is displayed on top of the page that is displayed.

Related tasks:

“Enabling an accelerator”

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference


Enabling an accelerator

Queries can only be routed to an accelerator if the accelerator has been enabled in Db2 for z/OS.

Before you begin

- You need a connection to a database that has an accelerator attached to it.
- You (your user ID) must be connected to the database and have the rights to control the accelerator.
- The **Acceleration** status of the accelerator must be *Stopped*.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click an accelerator to open the Accelerator view.
3. Click  on the toolbar. This invokes the Db2 command **-START ACCEL** on your data server.

Results

When an accelerator has been enabled, its **Acceleration** status changes to *Started*.

Related tasks:

“Disabling an accelerator” on page 82




Disabling an accelerator


You can disable an accelerator to prevent the routing of queries to this accelerator. Disabling an accelerator is recommended, for example, before you update software on the data server or on the accelerator. You can also disable an accelerator temporarily to compare response time. During the disablement, all queries are processed by Db2. You might also want to disable an accelerator for maintenance tasks. It is not necessary to disable an entire accelerator for maintaining accelerator-shadow tables; each table can be disabled individually.

Before you begin

- You need a connection to a database that has an accelerator attached to it.
- You (your user ID) must be connected to the database and have the rights to control the accelerator.
- The status of the accelerator must be *Online*.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click an accelerator to open the Accelerator view.
3. Complete one of the following steps:
 - Click  on the accelerator to disable the selected accelerator without canceling running queries. The accelerator status first changes to *Stopping*, then to *Stopped*. During the *Stopping* phase, running queries are completed.
 - Click the  button on the right of the  button and select **Force** from the menu to disable the selected accelerator and cancel all running queries. The status of the accelerator changes to *Stopped* immediately.

Note: Disabling does not make an accelerator unusable or remove it from the configuration. It just deactivates it. To re-enable an accelerator, click the  button on the toolbar.

Enabling or disabling an accelerator for members of a data sharing group

If an accelerator is attached to a Db2 data sharing group, you can enable or disable acceleration individually for each member of the group.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click an accelerator to open the Accelerator view.
3. In the header of the Accelerator view, next to the **Acceleration** status, click the **Change** link.
4. The Start / Stop Acceleration window lists all members of the data sharing group and shows their acceleration status. To enable or disable the accelerator for members of the data sharing group, select the appropriate check boxes in front of the member names. If the group contains many members, you can speed up the process by using the **Select All** and **Deselect All** buttons.

5. Click one of the following buttons on top of the list:
 - Start** To enable the accelerator for the selected members
 - Stop** To disable the accelerator for the selected members
 - Stop Force**
To disable the accelerator immediately, that is, without waiting for the completion of not yet finished queries.
6. Click **Close** when finished or **Cancel** to close the window without making changes.

Loading tables

Successful queries against accelerator-shadow tables are possible only if the tables contain data. Therefore, you must load the tables after their definition (empty structure) has been copied to the accelerator.

Before you begin

- A database connection profile must exist for the database that is associated with the accelerator.
- You (your user ID) must be connected to the database. Your user ID requires privileges to:
 - Control the accelerator
 - Read the data to be loaded into the accelerator
- The accelerator-shadow tables that you want to load must exist on the accelerator.

About this task

- Loading accelerator-shadow tables can take a long time (for example minutes or hours) depending on the amount of data in the original Db2 tables.
- An external product can be used to speed up and facilitate the load process: the Db2 Analytics Accelerator Loader, which can also load external, non-Db2 for z/OS data into accelerator-shadow tables and comes with additional load options.
- The disk throughput can be quite volatile, depending on the disk configuration, the placement of data sets, the number of parallel I/O operations, and so on. To a large extent, the throughput depends on the following factors:
 - Partitioning of tables in your Db2 subsystem. In Db2 for z/OS, this determines the degree of parallelism that can be used internally by the Db2 Unload Utilities.
 - Number of DATE, TIME, and TIMESTAMP columns in your original table. The conversion of the values in such columns is CPU-intensive.
 - Compression of data in Db2 for z/OS.
 - Number of available processors.
 - Workload Manager (WLM) configuration.
 - Workload on the IBM Z server.
 - Workload on the accelerators.
- During a load, IBM Db2 Analytics Accelerator automatically selects a suitable distribution key.

Restrictions:

- IBM Db2 Analytics Accelerator for z/OS does not support EDITPROC encryption. SQL encryption and decryption functions, such as DECRYPT_CHAR(), require binary table columns, which cannot be used on an

accelerator. Furthermore, data that is stored in Db2 with the EDITPROC encryption tool will not be encrypted on the hard disk of the accelerator. However, the data on an accelerator is encoded with a special algorithm for efficient compression and therefore cannot be easily extracted from the media.

- During a load process, you cannot close IBM Db2 Analytics Accelerator Studio or the database connection before the loading has finished.
- The maximum number of tables on a single accelerator is 27132. Naturally, that also means that a load job cannot load more than this number of tables.
- Incremental updates are not restarted automatically if you enable these for just one table and complete the enablement steps in this order:
 1. Define the accelerator-shadow table.
 2. Enable incremental updates for this table.
 3. Load the accelerator-shadow table.
 4. Start incremental updates (**Replication > Start**).
 5. Reload the accelerator-shadow table.

Normally, when this is done for more than one table, incremental updates continue or are restarted for all tables that are not being loaded or where loading has been finished, which finally results in an automatic restart for all tables. In this special case (just one table), however, you must restart incremental updates manually after the reload.

- If you change the partitioning type of the table space of an accelerator-shadow table in Db2 for z/OS (for example, from *segmented* to *partitioned by growth*), you can no longer load the accelerator-shadow table. The operation will fail. Therefore, you must remove the table from the accelerator and redefine it after changing the partitioning type.

You can also integrate the loading and updating of accelerator-shadow tables into your scheduled ETL processes by invoking the `SYSPROC.ACCEL_LOAD_TABLES` stored procedure directly from a job that is run by your preferred scheduler (for example, Tivoli® workload scheduler).

Table loading is carried out by the `SYSPROC.ACCEL_LOAD_TABLES` stored procedure on your data server. In addition, the `SYSPROC.ACCEL_SET_TABLES_ACCELERATION` stored procedure is invoked to enable acceleration for the loaded table.

For information about the privileges that are required to run these procedures and further details, see *Appendix C. Required access rights* in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related information** at the end of this section.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click the accelerator containing the tables that you want to load.
3. In the Accelerator view, you see a list of the tables on the accelerator. Select the accelerator-shadow tables that you want to load. To see the tables, you might have to expand the schema nodes first, by clicking the plus sign in front of a schema name. Selecting an entire schema will select all accelerator-shadow tables belonging to that schema for loading.
4. Click **Load**. The list in the Load Tables window contains a column labeled **Load Recommended**. The information in this column is supposed to help you with

the selection. Because table loads are long-running processes, you should only load or reload an accelerator-shadow table or partition when necessary.

Important:

- You can only select table partitions if the tables are partitioned by ranged or partitioned by growth, and if they were already loaded in the past. In all other cases, you can only load entire tables.
- The information in the **Load Recommended** column might be inaccurate if Db2 realtime statistics are not up-to-date (RUNSTATS). or if incremental updates have been applied to the accelerator-shadow table since the last load.

The **Load Recommended** column contains one of the following statuses for each table or partition:

No - data unchanged

Table or partition data has not changed in Db2 for z/OS. Therefore, you need not load or reload the accelerator-shadow table or partition.

Yes - changed data

A load or reload of the accelerator-shadow table or partition is recommended because data changes have been applied to the original Db2 for z/OS tables.

Unknown - no statistics

A load or reload of the accelerator-shadow table or partition is recommended because the current status of the table or partition cannot be determined. It must therefore be assumed that the table is out of sync. The reason might be that access to Db2 statistics is not available or that the last successful invocation of RUNSTATS was some time ago.

Unknown - replication enabled

Data changes have been detected since the last load of the accelerator-shadow table, but the table is currently enabled for incremental updates. The data changes are propagated asynchronously by the incremental update function, but it can not be determined whether the table is already in sync. A reload is usually not required under this condition.

Yes - enforced

A partition with a value of **Yes - enforced** in the **Load Recommended** column will always be loaded if the entire accelerator-shadow table is selected for loading. A possible reason is that the partition is in *InitialLoadPending* state because it has never been loaded before.

If the list of tables and partitions is long, you can apply a filter to hide certain items from the display. To do so, select a different value from the **Show recommended** drop-down list. The following choices are available:

All (default)

Displays all accelerator-shadow tables and partitions.

All recommended

Displays only the accelerator-shadow tables and partitions for which a load or reload is recommended.

No - data unchanged

Displays only the accelerator-shadow tables and partitions that need not be loaded or reloaded.

Yes - changed data

Displays only the accelerator-shadow tables and partitions for which a load or reload is recommended because data in the original tables has changed.

Unknown - no statistics

Displays only the accelerator-shadow tables and partitions for which a load or reload is recommended because the status cannot be determined.

Yes - enforced

Displays only the accelerator-shadow tables and partitions for which a load or reload will be enforced.

5. Select the appropriate tables or partitions. Tick the check boxes in front of the table or partition names, or use the buttons on the right:

Select All

Selects all tables and partitions that are shown in the Load Tables window.

Deselect All

Deselects all tables and partitions.

Select Recommended

Selects all accelerator-shadow tables and partitions for which a load or reload is recommended in the **Load Recommended** column.

Expand All

Expands all table nodes so that the partitions come into view.

Collapse All

Collapses all table nodes so that the table partitions are hidden from view.

Next Unknown

Jumps to and selects the next accelerator-shadow table or partition with a status of **Unknown - no statistics** in the **Load Recommended** column. This might prove useful if the list is crowded.

6. Select **Lock Db2 tables while loading** if you want to protect (lock) the original tables or table partitions in the database against changes during the load operation:

All Tables

Protects the Db2 source tables of to-be-loaded accelerator-shadow tables against changes during the load operation.

Data availability

Newly loaded data becomes available for queries when the load job has been finished for the entire table set.

Error behavior

If an error occurs while loading one of the tables in the set, the load job for that table is aborted, and the job proceeds with the next table in the set. Message AQT20121E is issued in case of a load error.

Current Table

Protects just the Db2 source table of the accelerator-shadow tables that is currently being loaded.

Data availability

Newly loaded data in a table becomes available for queries when

the load job for that table has been finished and when the entire operations has moved on to the next table in the set.

Error behavior

If an error occurs while loading one of the tables in the set, the load job for that table is aborted, and the job proceeds with the next table in the set. Messages:

- Message AQT20121E is returned if the load of all tables in the set failed.
- Message AQT10420W is issued for each single table after a failure to load that table

Current Partitions

Protects the original Db2 table-space partition of the accelerator-shadow table partition to be loaded against changes during the load. With this setting, a non-partitioned table is always locked completely.

Data availability

Newly loaded data in a table partition becomes available for queries when the load job for that partition has been finished and when the entire operations has moved on to the next partition or table.

Error behavior

If an error occurs while loading a table partition, the load job for that partition is aborted, and the job proceeds with the next partition to be loaded. An error has no effect on other partitions or tables to be loaded; the job continues. Messages:

- Message AQT20121E is returned if the load of all tables or all partitions of a partitioned table in the set failed.
- Message AQT10421W is issued for each partition after a failure to load that partition.

Special cases

To avoid partially loaded tables, lock mode TABLE is applied during an initial load, even if you specified lock mode PARTITIONS. For details, see the description of lock mode TABLE.

If the reload of a partition that is enabled for incremental updates fails, incremental updates are disabled for the entire table even if the reload of the other table partitions succeeded.

NONE

No locking at all. However, only committed data is loaded into the accelerator-shadow tables because the Db2 data is unloaded with isolation level CS and SKIP LOCKED DATA.

Data availability

Data availability is the same as for lock mode PARTITIONS.

Error behavior

The error behavior is the same as for lock mode PARTITIONS.

7. The option **After the load, enable acceleration for disabled tables** ensures that you can run accelerated queries against all accelerator-shadow tables immediately after the completion of the load operation. It is selected by default. Deselect it only if there is a good reason to do so. The following scenario is thinkable: the table is big and one or more columns could not be loaded because their data type is not supported. These columns, however, would have held the crucial data. In this case, it makes perfect sense to switch the option off to prevent a waste of processing resources.

8. Click **OK**.

Note: If the load fails, first check whether DSNUTIL was started in Db2 for z/OS.

Related concepts:

“Updating accelerator tables continually” on page 117

“Distribution keys” on page 7

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference



IDLE THREAD TIMEOUT field (IDTHTOIN subsystem parameter)

Enabling tables for query acceleration

You can permit or prevent the sending of queries to an accelerator by enabling or disabling the corresponding accelerator-shadow tables.

Before you begin

- You need a connection to a database that has an accelerator attached to it.
- You (your user ID) must be connected to the database and have the rights to control the accelerator.
- The tables must have been defined on the accelerator.

About this task

After its creation, an accelerator-shadow table is in *load pending* state. Query acceleration is disabled for tables in this state. The *load pending* state is also reached when an error has led to an operation failure. Also, when data has been loaded into a table, the table is not automatically enabled. Therefore, you must enable the table so that it becomes *active* and will thus be used in accelerated queries. You can disable or re-enable a loaded accelerator-shadow table by switching query acceleration for this table on or off. To enable a table, the table must be in *loaded* or *update in progress* state.

Tip: If you want to disable all tables, it is better to stop the entire accelerator. For more information, see the link under **Related tasks** at the end.

The task of enabling tables for query acceleration is carried out by the **SYSPROC.ACCEL_SET_TABLES_ACCELERATION** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

When an accelerator-shadow table has been enabled, the corresponding entry in the ENABLED column of the SYSACCEL.SYSACCELERATEDTABLES is set to ON. Although an accelerator-shadow table can exist on more than one accelerator, it can be enabled on only one accelerator. If a table is in a state that does not permit enabling, the operation ends abnormally and the enablement is canceled.

Important:

- Query acceleration can be always disabled regardless of the state an accelerator-shadow table is in. This is useful if you need to recover from an error situation in which an accelerator is down or unreachable.
- You might have to reload or even remove tables from an accelerator after the execution of an ALTER TABLE or ALTER TABLESPACE statement in Db2 for z/OS. For more information, see *What to do after changing a table or table space in Db2 for z/OS*. You find a **Related reference** link at the end of this topic.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click the accelerator containing the tables that you want to enable.
3. In the list on the lower part of the view, select the tables that you want to enable.
4. Click **Acceleration > Enable** on the toolbar.

Results

Queries against these tables are accelerated if the associated accelerator has been started.

Related tasks:

“Disabling query acceleration for tables”

“Loading tables” on page 83

“Disabling an accelerator” on page 82

Related reference:

“What to do after changing a table or table space in Db2 for z/OS” on page 133

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Disabling query acceleration for tables

You can disable accelerated queries against selected tables without entirely disabling the associated accelerator.

Before you begin

- A database connection profile must exist for the database that is associated with the accelerator.
- You (your user ID) must be connected to the database and have the rights to control the accelerator.
- The tables that you want to disable must have been added to the accelerator.

About this task

The task of disabling tables is carried out by the **SYSPROC.ACCEL_SET_TABLES_ACCELERATION** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click the accelerator containing the tables that you want to disable.
3. In the list of tables in the Accelerator view, select the tables that you want to disable.
4. Click **Acceleration > Disable** on the toolbar.

Results

This disables query acceleration for the selected tables only. Other tables that are associated with the same accelerator are not affected.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Specifying or changing a distribution key

Distribution keys have a considerable impact on the query response time.

About this task

Distribution keys can influence the query response time positively as well as negatively. Therefore, make sure that you only select such keys when necessary, and that you select a proper key.

The default is not to use a distribution key, but to distribute the data (table rows) evenly among the processing nodes. This is what the **Random distribution without a distribution key** check box signifies. If **Random distribution without a distribution key** is selected, all table rows are distributed sequentially to the processing nodes (random distribution) so that each node receives an equal share of the workload. To distribute the data evenly is usually the best choice for small and medium-sized tables, so do not change the setting unless incoming queries demand big tables to be joined.

To execute a join, data might have to be transferred over the network, which slows down the process. In such a case, it might be better to use a distribution key. If you select a join column as the distribution key for all tables involved, you assign the rows of these tables to the same processing node. This has the effect that no data needs to be transferred to execute the join.

In general, unique key columns, such as primary keys or columns with a unique index, are a good choice for a distribution key because all values are distinct. Columns that allow NULL as a value are not a good choice if many NULL values can be expected. The reason is that all rows with a NULL in the specified column will end up on the same processing node.

Tip: If you cannot assess the distribution of values in table columns, start with random distribution. If you witness a slow performance, try using a few of the recommended columns as the distribution key. Always use as few columns as possible in a distribution key.

This task of altering or changing distribution keys is carried out by the **SYSPROC.ACCEL_ALTER_TABLES** stored procedure on your data server. For information about the privileges that are required to run this procedure and

further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. Double-click the accelerator containing the tables for which you want to specify distribution keys.
3. In the list of schemas and tables in the Accelerator view, select a table that contains the columns to be used as a distribution key.
4. Click **Alter Keys** on the toolbar.
5. Specify a distribution key: To use a distribution key instead of even distribution:
 - a. Clear the **Random distribution without a distribution key** check box. This enables the controls for the definition of a distribution key.
 - b. In the Alter Distribution Keys window, you see a list of the columns in the selected table. To specify a column as the distribution key or as part of it, select the column in the list and click the right-arrow button.

The list on the left has a **Recommendation** column, which suggests certain columns to be used in a distribution key. Suggestions are made on the basis of a previous run of Db2 RUNSTATS. So if no recommendations can be seen, use RUNSTATS and then return to the Alter Distribution Keys window.

RUNSTATS also calculates the column cardinality (**Cardinality** in the selection list). This is the number of distinct values in a column. The higher this number, the better a column is suited as a distribution key.

Important: The RUNSTATS utility cannot collect statistics for accelerator-only tables because the data of this type of table does not reside in Db2. Hence key recommendations that rely on Db2 statistics cannot be given. You only find key recommendations for accelerator-only tables if these can be derived from the column data type.

The **Name like** filter field makes it easier to find particular columns if the list is long. Type a column name in this field, either fully or partially, to display just the columns bearing or starting with that name. The names of columns that are currently selected as keys are grayed out.

Selected columns appear in the upper list box on the right. You can remove a selected column by first selecting it in this box, and then clicking the left-arrow button.

Using the buttons with the upward-pointing and downward-pointing arrows, you can change the order of the columns in the key. The order of the columns in the list has an influence on the hash value that is calculated to determine the target processing node. To place the rows of joined tables on the same processing node, the distribution keys of all tables must yield the same hash value. It is therefore important to specify the distribution key columns for all tables in the same order.

- c. Repeat this step to add further columns to the key. A maximum of four columns is allowed.

Remember: It is a best practice to use as few columns as possible in a distribution key.

- To use a distribution key instead of even distribution:

- a. Clear the **Random distribution without a distribution key** check box. This enables the controls for the definition of a distribution key.
- b. In the Alter Distribution or Organizing Keys window, you see a list of the columns in the selected table. To specify a column as the distribution key or as part of it, select the column in the list and click the right-arrow button.

The list on the left has a **Recommendation** column, which suggests certain columns to be used in a distribution key or as organizing keys. Suggestions are made on the basis of a previous run of Db2 RUNSTATS. So if no recommendations can be seen, use RUNSTATS and then return to the Alter Distribution or Organizing Keys window.

RUNSTATS also calculates the column cardinality (**Cardinality** in the selection list). This is the number of distinct values in a column. The higher this number, the better a column is suited as a distribution key.

Important: The RUNSTATS utility cannot collect statistics for accelerator-only tables because the data of this type of table does not reside in Db2. Hence key recommendations that rely on Db2 statistics cannot be given. You only find key recommendations for accelerator-only tables if these can be derived from the column data type.

The **Name like** filter field makes it easier to find particular columns if the list is long. Type a column name in this field, either fully or partially, to display just the columns bearing or starting with that name. The names of columns that are currently selected as keys are grayed out.

Selected columns appear in the upper list box on the right. You can remove a selected column by first selecting it in this box, and then clicking the left-arrow button.

Using the buttons with the upward-pointing and downward-pointing arrows, you can change the order of the columns in the key. The order of the columns in the list has an influence on the hash value that is calculated to determine the target processing node. To place the rows of joined tables on the same processing node, the distribution keys of all tables must yield the same hash value. It is therefore important to specify the distribution key columns for all tables in the same order.

- c. Repeat this step to add further columns to the key. A maximum of four columns is allowed.

Remember: It is a best practice to use as few columns as possible in a distribution key.

- To specify organizing keys:
 - a. Select a column to be used as an organizing key in the list on the left and click the right-arrow button next to the list box at the bottom.

The list on the left has a **Recommendation** column, which suggests certain columns to be used in a distribution key or as organizing keys. Suggestions are made on the basis of a previous run of Db2 RUNSTATS. So if no recommendations can be seen, use RUNSTATS and then return to the Alter Distribution or Organizing Keys window.

Important: The RUNSTATS utility cannot collect statistics for accelerator-only tables because the data of this type of table does not reside in Db2. Hence key recommendations that rely on Db2 statistics cannot be given. You only find key recommendations for accelerator-only tables if these can be derived from the column data type.

The **Name like** filter field makes it easier to find particular columns if the list is long. Type a column name in this field, either fully or partially, to display just the columns bearing or starting with that name. The names of columns that are currently selected as keys are grayed out.

Selected columns appear in the lower list box on the right. You can remove a selected column by first selecting it in this box, and then clicking the left-arrow button.

- b. Repeat this step to add further keys. A maximum of four keys is allowed.

6. Click **OK**.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Running an SQL script from IBM Db2 Analytics Accelerator Studio

To modify connected databases or create objects in these databases, you can run SQL scripts directly from IBM Db2 Analytics Accelerator Studio. This is useful because you need not change the system or console to execute SQL statements.

Before you begin

Make sure that the user running IBM Db2 Analytics Accelerator Studio has sufficient privileges on the selected databases to execute the SQL code.

About this task

If you run an accelerated query from the SQL Script Editor of IBM Data Studio, the SQL Script Editor invokes the JDBC method `Statement.setNumRows()`, which causes the JDBC driver to convert the query into a read-only query (FOR READ ONLY). This happens even if the FOR READ ONLY clause is not part of the query statement. If the clause is not part of the statement, the query might end with an error message and reason code 4 (The query is not read-only). The error occurs with actual queries in a production environment and EXPLAIN-only queries alike. With EXPLAIN-only queries, the error does not matter. Actual queries, however, will not return results when the error occurs. Therefore, add the FOR READ ONLY clause explicitly to the SQL statement text if you submit actual queries from the SQL Script Editor.

Procedure

1. In the Administration Explorer, double-click a database connection profile to connect to a Db2 subsystem on your Db2 for z/OS data server.
2. On the toolbar of the Administration Explorer, click the downward-pointing arrow next to the **New** button.
3. Select **New SQL Script** from the menu.
4. In the blank space of the Script<x>.sql workspace that opens on the upper right, type or paste the SQL statement.

Note: <x> stands for a counting number (integer). This means that your first SQL script is named Script1.sql, the second Script2.sql, and so on.

5. Select **Script > Run SQL** from the main menu.

What to do next

SQL scripts can be saved in your local workspace if you created an appropriate project before, for example a *Data Development* project. For information on how to create and work with Data Development projects, refer to *Creating a data development project* in the online help of IBM Data Studio.

EXPLAIN information

After creating the necessary EXPLAIN tables, you can analyze queries by invoking the Db2 EXPLAIN function. The analysis shows whether a query can be accelerated, indicates the reason for a failure, and gives a response time estimate. The outcome of the analysis can also be visualized in an access plan graph.

Preparations

To be able to use the EXPLAIN function in order to check whether a query can be accelerated, you need the following EXPLAIN tables:

- PLAN_TABLE
- DSN_QUERYINFO_TABLE

For basic EXPLAIN information, you need just the PLAN_TABLE. To create EXPLAIN tables (including DSN_QUERYINFO_TABLE) for a specific user, customize and run the DSNTESC member of the SDSNSAMP library. Modify and run the DSNTIJOS sample job to create EXPLAIN tables that are qualified by SYSIBM and Db2OSC, as these tables are used by Db2 optimization tools like IBM Query Tuner.

Description

The EXPLAIN tables provide the following information:

PLAN_TABLE

For each accelerated query, the query blocks are pruned and a single row is inserted in the PLAN_TABLE with an ACCESTYPE of 'A'. The query is thus represented by a single row with the value 1 in the QBLOCKNO column. For queries that are not accelerated, the PLAN_TABLE can contain multiple rows representing the tables accessed by each query block, with an ACCESTYPE other than 'A'. If there are multiple rows belonging to the same query, these are numbered consecutively in the QBLOCKNO column.

DSN_QUERYINFO_TABLE

Additional EXPLAIN information can be found in the DSN_QUERYINFO_TABLE. See Table 22.

Table 22. Important columns in DSN_QUERYINFO_TABLE

Column	Description
QINAME1	When the ACCESTYPE of a query in the PLAN_TABLE is A, and the REASON_CODE in DSN_QUERYINFO_TABLE is 0, this row contains the name of the accelerator that will process the query.

Table 22. Important columns in DSN_QUERYINFO_TABLE (continued)

Column	Description
QINAME2	When the ACESSTYPE of a query in the PLAN_TABLE is A, and the REASON_CODE in DSN_QUERYINFO_TABLE is 0, this row contains the location name of the accelerator that will process the query. The location name is the unique name of the accelerator in the SYSIBM.LOCATIONS table. Mostly, this is the same name as the accelerator name.
TYPE	The type of the output for this row. A type of A means that Db2 for z/OS routes the query to an accelerator.
REASON_CODE	The reason code for a query. When the value of TYPE is A and the reason code is 0, the query can be accelerated. If the REASON_CODE column contains a value other than 0, something prevents the acceleration of the query. To look up the other reason codes, refer to Table 23.
QI_DATA	When the value of TYPE is A and the reason code is 0, this row contains the SQL code of the converted query that was routed to the accelerator. If a query has a reason code other than 0, this column contains a description of the reason code.

Reason codes

Table 23 lists and describes the various reason codes. A reason code greater than 0 indicates that a query cannot be accelerated.

Table 23. Possible reason codes in DSN_QUERYINFO_TABLE

REASON_CODE	Description	Suggested solution
0	The query qualifies for routing to an accelerator.	N/A
1	An active accelerator or tables enabled for query acceleration could not be found when the statement was executed. If an accelerator is available, make sure that it is started, and that enabled tables exist on that accelerator.	Start the accelerator.
2	The query could not be routed to an accelerator because the CURRENT QUERY ACCELERATION special register is set to the value NONE.	Set the CURRENT QUERY ACCELERATION special register to one of the following values: <ul style="list-style-type: none"> • ALL • ELIGIBLE • ENABLE • ENABLE WITH FAILBACK
3	The query is a short-running Db2 query or query acceleration is not advantageous.	If you must accelerate the query for some reason, set the CURRENT QUERY ACCELERATION special register to one of the following values: <ul style="list-style-type: none"> • ALL • ELIGIBLE
4	The query is not read-only.	Ensure that the query meets the criteria for a read-only query.

Table 23. Possible reason codes in DSN_QUERYINFO_TABLE (continued)

REASON_CODE	Description	Suggested solution
6	The query uses a scrollable cursor or a remotely defined rowset-positioned cursor.	Remove the SCROLL clause or WITH ROWSET POSITIONING clause from the cursor declaration.
7	The query uses multiple encoding schemes.	Ensure that all objects to which the query refers have the same encoding scheme.
8	The FROM clause of the query specifies a data-change-table-reference.	Rewrite the FROM clause of the query so that it does not include a data-change-table-reference clause.
10	The query contains a recursive reference to a common table expression.	Ensure that the FROM clause of the query does not reference a common table expression.
11	The query contains an unsupported expression. The QI_DATA column contains this expression.	Remove the unsupported expression from the query.
12	The query references a table with one of the following characteristics: <ul style="list-style-type: none"> • The table is not defined on the accelerator. • The table is defined on a different accelerator. • The table is not enabled for query acceleration. 	Ensure that all tables that are referenced by the query are defined on the same accelerator, and that all tables are enabled for query acceleration.
13	The accelerator server that contains the tables that are referenced in the query is not started.	Start the accelerator that contains the tables that are referenced in the query.
14	A column that is referenced in the query has been altered after loading the column data into the corresponding accelerator-shadow table.	Reload the accelerator-shadow table.
15	The query uses functionality that is available only in newer Db2 for z/OS versions, but not in the versions supported by IBM Db2 Analytics Accelerator.	Ensure that the query does not use any functionality that is available only in not yet supported Db2 versions.
16	The query is not in a package.	Bind the application that contains the query into a package.
17	The query is an INSERT from SELECT statement. Subsystem parameter QUERY_ACCEL_OPTIONS does not specify option 2 to enable the acceleration of INSERT from SELECT statements.	Enable the acceleration of INSERT from SELECT statements by updating subsystem parameter QUERY_ACCEL_OPTIONS to include option 2.
18	The query uses new Db2 11 for z/OS functionality.	Try to rewrite the query so that it does not use the new (unsupported) Db2 11 for z/OS functionality.

Table 23. Possible reason codes in DSN_QUERYINFO_TABLE (continued)

REASON_CODE	Description	Suggested solution
19	The accelerator is not at the correct level (that is, you are using an older version) and therefore does not support all the functions in the SQL statement. The QI_DATA column in DSN_QUERYINFO_TABLE contains the function text or expression text that has caused the failure. Update the accelerator software to enable query acceleration for the SQL statement that contains the function text or expression text.	Update the accelerator software to a service level that supports all functions in the query.
20	The query is a rowset cursor that is declared WITH RETURN, executes remotely, or executes under an SQL PL routine.	<p>If the query executes remotely, turn off multi-row-fetch (use MULTI_FETCH=1).</p> <p>In the other cases, set the CURRENT QUERY ACCELERATION special register or QUERY_ACCELERATION ZPARM to the value 1 to turn off query acceleration.</p>
22	<p>The statement references an accelerator-only table, but cannot be routed to an accelerator. For example:</p> <ul style="list-style-type: none"> A MERGE statement references an accelerator-only table. An UPDATE or DELETE statement references an accelerator-only table, but the target table of the UPDATE or DELETE is a normal DB2 table, for example: <pre>UPDATE db2_table SET ... (SELECT ... FROM accelerator_only_table)</pre> <ul style="list-style-type: none"> An INSERT, UPDATE, or DELETE refers to a row expression with a sub-select, for example: <pre>UPDATE accelerator_only_table SET (C1,C2) = (SELECT C3,C4 FROM table_2)</pre>	<ul style="list-style-type: none"> Remove the reference to the accelerator-only table If possible, rewrite the statement so that it references a regular DB2 table instead of an accelerator-only table.
23	The SELECT INTO statement is bound for acceleration but is run as a remote SELECT INTO statement, which is not supported for acceleration.	Check if you can import (copy) the table to select from into the local Db2 subsystem (the one that is paired with the accelerator), or if you can pair the remote Db2 subsystem with the accelerator.

Table 23. Possible reason codes in DSN_QUERYINFO_TABLE (continued)

REASON_CODE	Description	Suggested solution
24	The DDL or DML statement cannot be offloaded because the connection to the accelerator server does not allow updates. This can occur when a two-phase commit requester (for example, a Db2 for z/OS requester) connects to a Db2 for z/OS server to offload a DDL or DML statement."	Issue the DDL or DML statement directly from the Db2 for z/OS server, or from a requester which does not require a two-phase commit, for example, the IBM Data Server Driver for JDBC.
25	The statement contains a reference to a column with an unsupported data type.	See if you can remove the column reference from the query. If not, consider changing the data type to a supported type. If none of this is possible, run the query in IBM Db2 Analytics Accelerator (not accelerated).
33	The query could not be routed to an accelerator because different product versions are used, and each version would have returned a different reason code.	Request EXPLAIN output and check the REASON_CODE column in the DSN_QUERYINFO_TABLE table. For situations that result in reason code 33, two rows are returned: one that explains why the query cannot be offloaded to IBM Db2 Analytics Accelerator versions earlier than V7, and another that explains why the query cannot be offloaded to IBM Db2 Analytics Accelerator or later. In the latter case, the error description might be prefixed by (Vx) to indicate the specific accelerator version. The (Vx) prefix is displayed only when reason code 33 is returned by the original query.
34	Your query was returned to Db2 for z/OS for FAILBACK processing. However, the query contains a pass-through-only expression, which can only be handled by an accelerator. Therefore, FAILBACK processing fails.	Set the CURRENT QUERY ACCELERATION special register to ENABLE , ELIGIBLE, or ALL to run queries that contain pass-through-only expressions.
35	The query uses new Db2 12 for z/OS functionality.	Try to rewrite the query so that it does not use the new (unsupported) Db2 12 for z/OS functionality.

For example, reason code 7 resolves to The query uses multiple encoding schemes.. If you receive this reason code, check the encoding schemes that are used by the tables. If different schemes are used, you might have to curtail the query so that it only refers to columns that use the same coded character set identifier (CCSID) or code page.

Sample EXPLAIN query

```
-- clean up explain tables
DELETE FROM PLAN_TABLE WHERE QUERYNO = 3825;
DELETE FROM DSN_QUERYINFO_TABLE WHERE QUERYNO = 3825;

-- enable accelerator for this session
```



```

SET CURRENT QUERY ACCELERATION=ENABLE;

-- explain query
EXPLAIN ALL SET QUERYNO = 3825 FOR
SELECT SUM(L_QUANTITY), SUM(O_TOTALPRICE), O_CUSTKEY
FROM TPCH.LINEITEM L JOIN TPCH.ORDERS O ON L.L_ORDERKEY = O.O_ORDERKEY
WHERE O.O_ORDERDATE > '01.01.1994' GROUP BY O.O_CUSTKEY;

-- check if query is eligible for acceleration
SELECT QBLOCKNO, ACCESTYPE FROM PLAN_TABLE AS P WHERE QUERYNO = 3825;

SELECT QINAME1 AS ACCELERATOR, REASON_CODE, QI_DATA FROM DSN_QUERYINFO_TABLE
WHERE QUERYNO = 3825;

```

The DELETE statements at the beginning remove old information from the EXPLAIN tables.

The SET CURRENT QUERY ACCELERATION=ENABLE special register is required, even for virtual accelerators. Leaving it out would mean to run an inhouse Db2 query that is not sent to any accelerator.

Then comes the actual test query. It starts with the mandatory EXPLAIN statement. A number is assigned to the query so that the EXPLAIN information pertaining to that query can be easily extracted from the EXPLAIN tables by means of a SELECT statement.

The first SELECT statement extracts the QBLOCKNO and the ACCESTYPE column values of the query from the PLAN_TABLE. The second SELECT statement extracts the QINAME1, REASON_CODE, and QI_DATA values of the query from DSN_QUERYINFO_TABLE.


Selecting and ordering query monitoring columns

You can select the columns that are displayed in the **Query Monitoring** section of the Accelerator view and also change their order of appearance.

Procedure

1. In IBM Db2 Analytics Accelerator Studio, open the Accelerators view.
2. Scroll down until you see the heading **Query Monitoring**.
3. Click the twistie next to the heading to reveal the contents of the section. The contents basically consists of a table that lists the recent query history (SQL statements and other information about recent queries). You can select various information columns, some of which are displayed by default. For a list of the available columns, click the appropriate **Related reference** link at the end of this topic.

Important: The reference for all times that are displayed is the accelerator system time. The accelerator system time is determined by the first Db2 subsystem that was added to the configuration. A warning is issued if the accelerator works with data from a Db2 subsystem that was not the first to be added. It is very important to keep this in mind if the Db2 subsystems (data servers) are situated in different time zones.


4. To add columns, remove columns, or change their order, click .
5. In the Adjust Query Monitoring Table window, you see the following lists:


Available columns


The names of available columns that are currently not selected for display

Shown columns



The names of the columns that are currently displayed. The order from top to bottom indicates their appearance in the Accelerator view from left to right.

- To add a single column for display in the Accelerator view, select it in the **Available columns** list and click  to move the column to the **Shown columns** list.

To move all available columns to the **Shown columns** list, click .

- To hide a single column from the display in the Accelerator view, select it in the **Shown columns** list and click  to move the column to the **Available columns** list.

To remove all columns from the display, click .

- To change the order of appearance in the Accelerator view, select a column in the **Shown columns** list and click  or  to move the column up or down.

The order of the columns from top to bottom represents the order in which these columns appear from left to right in the Accelerator view.

- To restore the default settings for the **Query Monitoring** section in the Accelerator view, click **Restore Defaults**.
6. Click **OK**. Your settings are applied to the table in the **Query Monitoring** section, that is, columns are added, removed or their order of appearance is changed.
 7. To limit the number of queries that are displayed in the **Query Monitoring** table:
 - a. Show all recent queries, just the queries that are currently being processed, or just the completed queries, by selecting the appropriate value from the **View** drop-down list:

All Queries

To show all recent queries

Active Queries

To show just the queries that are currently being processed

Query History

To show just the completed queries

- b. To limit the displayed queries on the basis of their values in one of the columns of the **Query Monitoring** table:
 - 1) Change the default (**All**) in the first drop-down list to the right of the **Show** label to one of the restricting values (**Highest 5**, **Highest 10** and so on).
 - 2) From the other drop-down list further to the right, select the column in the **Query Monitoring** table that the setting in the previous step refers to.
 - 3) Click **Run**.

This limits the number of displayed queries to those with the highest value in the selected column. If you select a column containing time values, this

means that the display will be restricted to the slowest queries. The designers of the product have opted for this order because the slowest queries usually have the highest potential for query optimizing, and are most probably the ones that a database administrator wants to touch again after they have been run.

8. The **Query Monitoring** section offers a few additional functions:
- To show further details about a selected query or to rerun the query, click:

Show SQL

To view the SQL of the query

Show Plan

To view the access plan graph of the query

Note: For accelerator-only tables, the **Show SQL** and **Show Plan** functions show information about data manipulation (DML) jobs, such as INSERT, UPDATE, or DELETE, but not data definition (DDL) jobs, such as CREATE or DROP.



A drop-down button that leads to the following functions:

Show Plan

Same as the button described before

Show Explain Details

Opens a window with a textual representation of a query's access plan and other information about the query execution. As an example, see the following extract:

```
...
FROM
  KLMDWEDB14."CUSTOMER-UID_01140003-V0" AS Q1
WHERE
  (Q1."DWA_Partition_ID (hidden)" <= 100001)
```

Access Plan:

```
-----
Total Cost:  105.849
Query Degree: 8
```

```

      Rows
      RETURN
      (   1)
      Cost
      I/O
      |
      333.333
      LTQ
      (   2)
      105.849
      15
      |
      333.333
      CTQ
      (   3)
      105.778
      15
      |
      333.333
      TBSCAN
      (   4)
      105.766
```

```

      15
      |
      1000
CO-TABLE: KLMDWEDB14
CUSTOMER-UID_01140003-V0
      Q1

```

Operator Symbols :

Symbol	Description
-----	-----
ATQ	: Asynchrony

...

Cancel

To cancel a running query or an INSERT, UPDATE, or DELETE operation on an accelerator-only table.

- To limit the number of queries that are displayed in the **Query Monitoring** table.
 - a. Show all recent queries, just the queries that are currently being processed, or just the completed queries, by selecting the appropriate value from the **View** drop-down list:

All Queries

To show all recent queries

Active Queries

To show just the queries that are currently being processed

Query History

To show just the completed queries

- b. To limit the displayed queries on the basis of their values in one of the columns of the **Query Monitoring** table:
 - 1) Change the default (**All**) in the first drop-down list to the right of the **Show** label to one of the restricting values (**Top 5**, **Top 10** and so on).
 - 2) From the other drop-down list further to the right, select the column in the **Query Monitoring** table that the setting in the previous step refers to.
 - 3) Click **Run**.

This limits the number of displayed queries to those with the highest value in the selected column. If you select a column containing time values, this means that the display will be restricted to the slowest queries. The designers of the product have opted for this order because the slowest queries usually have the highest potential for query optimizing, and are most probably the ones that a database administrator wants to touch again after they have been run.

Related reference:

"Selectable columns in the **Query Monitoring** section" on page 103

Related information:

"Query history is not displayed - error message returned" on page 157

Selectable columns in the Query Monitoring section

The list in this section briefly describes all columns that can be selected for display in the **Query Monitoring** section of an Accelerator view.

Accounting String

Accounting information that can be provided by an application programming interface (API) on the client side, such as the `setDb2ClientAccountingInformation` API of a connecting JDBC client.

For more information, see Providing extended client information to the data source with IBM Data Server Driver for JDBC and SQLJ-only methods

Client User ID

The ID of a connecting client user. This ID can be provided by an application programming interface (API) on the client side, such as the `setDb2ClientUser` API of a connecting JDBC client.

For more information, see Providing extended client information to the data source with IBM Data Server Driver for JDBC and SQLJ-only methods

Client Workstation Name

The name or ID of a connecting client workstation. This ID can be provided by an application programming interface (API) on the client side, such as the `setDb2ClientWorkstation` API of a connecting JDBC client.

For more information, see Providing extended client information to the data source with IBM Data Server Driver for JDBC and SQLJ-only methods

Collection ID

The ID or name that identifies a group of static SQL packages. If a collection ID is displayed, this means that the static SQL package containing the query belongs to that group.

Connection Name

The connection name. Possible values are:

For batch:

BATCH

For TSO:

TSO

For QMF™:

Db2CALL

For Db2 utilities:

UTILITY

For Db2 private protocol:

The Db2 subsystem ID

For IMS:

The IMS ID

For CICS®:

The CICS ID

For DRDA connections from non-Db2 requesters:

SERVER

For more information, see IFCID 147 - Thread Correlation Data.

Connection Type

The type of the Db2 connection. Valid values are:

APPLCTN
Application directed access

BATCH
Batch job

CICS CICS attach

Db2CAF
Db2 call attach

DLIBATCH
DL/I batch

IMS IMS thread

IMS BMP
IMS attach BMP

IMS TBMP
IMS transaction BMP

IMS CTL
IMS control region

IMS MPP
IMS attach MPP

SYSTEM
System directed access

TSO/E TSO/E foreground and background

UNKNOWN
Unrecognizable connection type

UTILITY
Db2 utility

For more information, see Thread SQL Counts Attributes.

Correlation ID

The correlation ID value. Possible values are:

For BATCH:

Job name.

For TSO:

Logon ID.

For IMS/VS:

PST#.PSBNAME

For CICS:

CONNECTION_TYPE.THREAD_TYPE.THREAD_#.TRAN-ID

For RRSAF:

CORRELATION-ID value from SIGNON function.

For threads using the Db2 private protocol or DRDA from a Db2 requester:

This field contains the correlation-ID name of the thread at the requesting location.

For threads using DRDA from a non-Db2 requester:

This field contains the first 12 characters in the DDM external

name (EXTNAM) parameter of the DDM EXCSAT command received as part of the SQL connect.

For more information, see IFCID 147 - Thread Correlation Data

CPU Time

The overall CPU time that was needed by an accelerator to process a query or data manipulation (DML) statement. This value reflects parallel processing such that the CPU value might exceed the accumulated elapsed time. This is an indicator for the amount of used resources.

For completed statements, this is the CPU time from the initial request to the last row being returned to Db2. In case of DML statements, this is measured until the final result (SQLCA) is sent to Db2.

For in-process statements, this is the time from the initial request to the current point in time. The counter includes the CPU time that was used by the product and also the CPU time spent on the accelerator (on the coordinator node and all worker nodes). Because parallel processing is the standard, the CPU time cannot be related to the elapsed time. If query processing has already finished on the accelerator, the CPU time spent inside this system is included. If the SQL statement is still running on the accelerator, only the CPU time that was used by IBM Db2 Analytics Accelerator is available.

Data Sharing Group Name

The name of the Db2 data sharing group (if the accelerator is connected to such a group rather than a Db2 subsystem)

Data Sharing Member Name

If the accelerator is connected to a Db2 data sharing group, the name of the group member that sent the query.

Db2 Subsystem ID

The Db2 subsystem ID.

Elapsed Time

The total time that was needed to process a query (from the state QUEUED to FAILED, SUCCESSFUL, or CANCELED).

The following diagram shows the beginning, duration, and the end of each time in the **Query Monitoring** section on a time scale. This way, you can better relate the values to each other. It starts with the **Start Time** (point in time, where the diagram starts and when the accelerator receives the query). The following interval, the **Wait Time** overlaps with the **Execution Time** because the **Execution Time** includes the period that a query stays in the accelerator's execution queue. The **Execution Time** overlaps with the following interval, the **Fetch Time**, because Db2 starts fetching results when the first result rows have been returned by the accelerator, that is, at a time when the entire query has not necessarily finished processing. Finally, the diagram shows the **Elapsed Time**, which includes all intervals mentioned before because it lasts from the beginning of the **Start Time** to the time when the query controller stops processing the query.

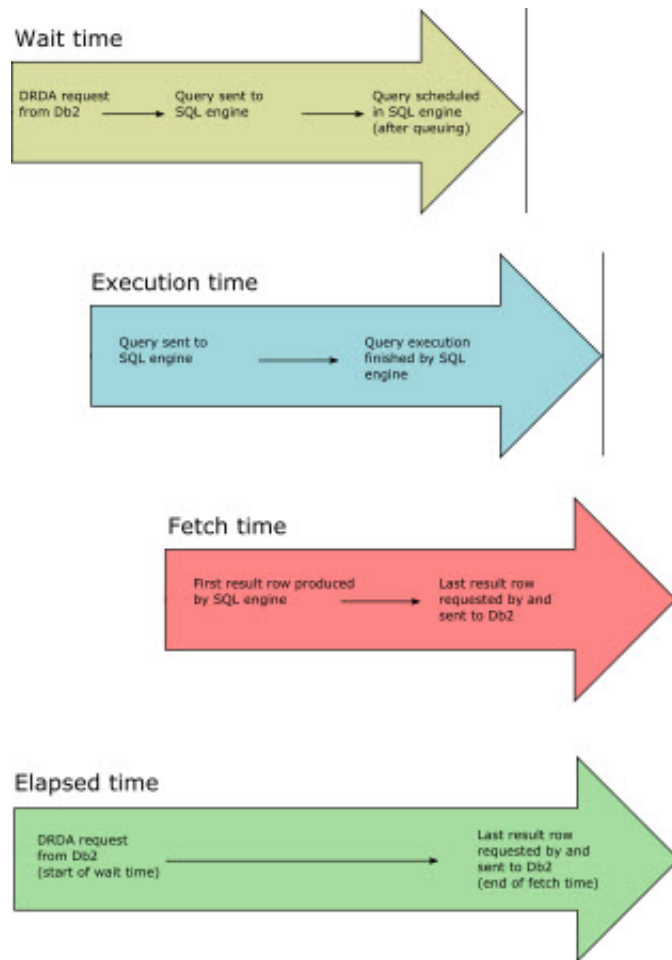


Figure 9. Wait time, execution time, fetch time, and elapsed time in comparison

Error Description

The error description or error token that is returned with the **SQL Code**. The description might be truncated.

Execution time

The time that a query spent in the queue before it was sent to the accelerator plus the time needed to process the query on the IBM Integrated Analytics System.

Fetch Time

The time interval starting with the first retrieval of a result row by Db2 and ending with the last. In most cases, query execution has already finished when the result retrieval process is still ongoing.

Location Name

Location name of the Db2 subsystem sending the query.

LU Name

Network LU (logical unit) name of the requester sending the query. This is used in the end-to-end monitoring of accelerated queries in multi-tier architectures where Db2 for z/OS handles requests passed on by an application server on behalf of a client application. This allows you to identify queries that were issued by a certain LU Name.

Network ID

Network ID of the requester sending the query. This is used in the end-to-end monitoring of accelerated queries in multi-tier architectures where Db2 for z/OS handles requests passed on by an application server on behalf of a client application. This allows you to identify queries that were issued by a certain Network ID.

Original Authorization ID

The original authorization ID. Possible values are:

For TSO:

The logon ID.

For batch:

The user ID on the job statement.

For IMS (message-driven regions):

The sign-on ID, LTERM, ASXBUSR, or PSB name.

For IMS (control regions):

The user ID on the job statement, or the RACF® started procedure entry if RACF is used.

For CICS:

The user ID, TERM ID, TRAN ID, or as specified in the resource control table.

For MVS operator commands and Db2 system internal agents:

SYSOPR

For a distributed application server (AS):

- If the application requester (AR) is a Db2 system, then this is the same value that was assigned at the AR.
- If the application requester is not a Db2 system, then this is the user ID used to make the initial connection with the application server.

For more information, see IFCID 147 - Thread Correlation Data

Package Name

The name that identifies a static SQL package or application package. If such a name is displayed, this means that the query was submitted by an application program in the form of a bound package for static SQL processing.

Plan Name

The plan name. It is blank for a Db2 command thread; otherwise:

DSNESP RR

For SPUFI with repeatable read

DSNESP CS

For SPUFI with cursor stability

DSNUT IL

For Db2 utilities

DSNTE P2

For DSNTEP2

DSNB IN D

For binding

The application plan name

For IMS or CICS

A blank plan name

For IMS and CICS commands

DSQPLAN

For QMF

The first 8 bytes of the application name

For DRDA connections to the common servers

For more information, see IFCID 147 - Thread Correlation Data.

Priority

The Workload Manager (WLM) importance level that the query or job holds in Db2.

Product ID

ID of the product (application requester) that routed the query, such as a JDBC driver.

Result Size

The data size of the rows that were returned in query results (in MB or GB) from the accelerator to Db2 for z/OS.

Rows Returned

The number of rows that were returned by the accelerator as query results.

SQL Code

The SQL code that is returned to the user after the query or job execution.

SQL Text

The SQL statements of recently submitted queries.

Start Time

Times when queries were received by IBM Db2 Analytics Accelerator.

State Processing states that queries are in. The following states are possible:

UNKNOWN

The state cannot be determined.

QUEUED

A query has been passed to the accelerator for processing.

RUNNING

The query is being processed by the accelerator.

FETCHING

The accelerator has finished processing the query, but Db2 for z/OS has not yet finished collecting the results.

FAILED

The query did not run to completion because one or more errors occurred.

SUCCESSFUL

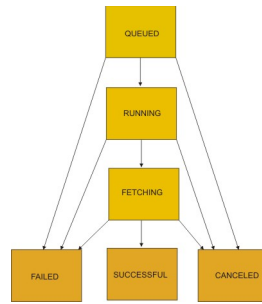
Query execution has been completed successfully.

CANCELED

The query did not run to completion because it was canceled by a user.

With the exception of the UNKNOWN state, each query passes through all the states, unless an error leads to an abnormal ending or a user cancels

the query. This is shown in the following diagram:



User ID

The names of users who submitted past queries.

Wait Time

The times that queries had to “spend” in the queue before they were processed by the IBM Integrated Analytics System.

Canceling tasks

See how to cancel tasks that are in progress on an accelerator, such as load operations.

About this task

Using the **List Tasks** function, you cannot cancel queries. To cancel a running query or an INSERT, UPDATE, or DELETE operation on an accelerator-only table, use the **Cancel** button in the **Query Monitoring** section of the Accelerator view.

Important: When a load job fails, and you run the command to check the status of the Db2 UNLOAD utility in Db2 for z/OS, you might find numerous UNLOAD threads in the STOPPED state. The stopped threads are ended automatically when you start another load job.

Procedure

1. Open the accelerator view of the accelerator that the task is running on.
2. Click **List Tasks** on the toolbar.
3. The table in the List Tasks window shows all tasks that are currently running on the accelerator. To cancel a task, select it in the table and click **Cancel Selected Task**. The table that lists the tasks contains the following columns, which give you further details about a task:

Name The type of the task, for example **Load Tables**.

Execution Time

The time that passed since the start of the task.

Progress

The degree of completeness since the start of the task.

If individual tables were selected for processing, the number denotes the number of records (table rows) that were processed so far.

If an entire schema was selected for processing, the number represents the percentage of completion. The maximum value is 1, denoting 100 percent. Therefore, a value of 0.25 means 25 percent complete.

Progress Message

Additional information about running tasks.

User The IDs of the users who started the tasks.

4. Click **Close** to close the List Tasks window.

Removing tables from an accelerator

Follow the steps in this topic to remove accelerator-shadow tables from an accelerator.

About this task

The table removal task is carried out by the **SYSPROC.ACCEL_REMOVE_TABLES** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. Double-click the accelerator containing the tables that you want to remove.
3. Select the tables that you want to remove.
4. Click **Remove** on the toolbar.

Related tasks:

"Including or excluding tables from incremental updates" on page 125

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Exporting a table specification

You can save a list of the accelerator-shadow tables on an accelerator to an XML file and in a later step import that list for use with another accelerator. This way, tables can be added more quickly than by adding schemas and tables individually.

About this task

A typical use case is an administrator who saves a valid table specification that was created in a test environment in order to use this specification in a production environment. To make the definition work in a different environment, you can edit the XML file with a text editor, and, for example, use the find-and-replace function to replace the names of schemas and tables.

Procedure

1. Click **File > Export**.
2. In the Export window, expand the **IBM Db2 Analytics Accelerator** folder to see its contents.
3. Select **Table Specifications**.
4. Click **Next**.
5. In the Export Table Specification window, select an appropriate database connection profile. The profile gives you access to the accelerators that a Db2

subsystem is connected to. Tables are exported from these accelerators. You thus have to select the proper profile, that is, one that gives you, by way of the connected Db2 subsystem, access to the accelerators that contain the right tables. If the proper profile does not exist yet, you can click **New** to create it.

6. Click **Next**.
7. From the **From accelerator** drop-down list, select the accelerator containing the tables to be exported.
8. In the table below, you can see the schemas and tables on the selected accelerator. From this table, you can select the tables or entire schemas whose specification you want to save to an XML file. If the list is long, first narrow the choice by using the **Name like** filter field. Type the names of schemas or tables in this field, either fully or partially, to display just schemas and tables bearing or starting with that name. The names of tables that have already been selected are greyed out.
9. Click **Browse**.
10. In the Export to File window, navigate to the folder in which you want to save your table specification XML file.
11. Type a file name in the **File name** field. By default, the file is given an extension of `idaa.xml`. Files of this type are encoded in UTF-8.
12. Click **Save**. You return to the Export Table Specification window. The path and the file name of your table specification XML file are displayed in the **To XML file** field.
13. Click **Finish**.

Importing a table specification

You can import a list of accelerator-shadow tables for query acceleration that was previously saved to an XML file.

About this task

A typical use case is an administrator who saves a valid table specification that was created in a test environment in order to use in a production environment. To make the definition work in a different environment, you can edit the XML file with a text editor, and, for example, use the find-and-replace function to replace the names of schemas and tables.

Procedure

1. Click **File > Import**.
2. In the Import window, expand the **IBM Db2 Analytics Accelerator** folder to see its contents.
3. Select **Table Specifications**.
4. Click **Next**.
5. In the In the Import Table Specification window, select an appropriate database connection profile. The profile gives you access to the accelerators that a Db2 subsystem is connected to. Tables are exported from these accelerators. You thus have to select the proper profile, that is, one that gives you, by way of the connected Db2 subsystem, access to the accelerators that contain the right tables. If the proper profile does not exist yet, you can click **New** to create it.
6. Click **Next**.

7. From the **To accelerator** drop-down list, select the accelerator on which you want to define the tables.
8. Click **Browse** to navigate to the folder that contains the XML file with the list of accelerator-shadow tables.
9. In the Import from File window, navigate to the proper XML file, select it, and click **Open**.

Important: XML files in UTF-8 format are recommended. Such files start with the following header:

```
<?xml version="1.0" encoding="UTF-8"?>
```

You return to the Import Table Specification window. The path and the file name of the selected XML file is displayed in the **From XML** file field.

10. In the table below, you can see the schemas and tables that are listed in the selected XML file. From this list, select entire schemas or individual tables that you want to define on the accelerator. If the list is long, first narrow the choice by using the **Name like** filter field. Type the names of schemas or tables in this field, either fully or partially, to display just schemas and tables bearing or starting with that name. The names of tables that have already been selected are greyed out.
11. Click **Finish**.

Removing accelerators

Follow the steps in this topic to remove accelerators from the system configuration. Bear in mind that this action will also remove the tables from an accelerator.


About this task

The table removal task is carried out by the following stored procedures on your data server:

- **SYSPROC.ACCEL_REMOVE_TABLES**
- **SYSPROC.ACCEL_REMOVE_ACCELERATOR**

For information about the privileges that are required to run these procedures and further details, see *Appendix C. Required access rights* in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related information** at the end of this section.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor on the right, select the accelerator that you want to remove.
3. Click  on the toolbar.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Tracing

IBM Db2 Analytics Accelerator offers a variety of predefined trace profiles in case you need diagnostic information. These profiles determine the trace detail level and the components or events that will generate trace information. After collecting the information, you can save it to a file. If you have access to the Internet, you can directly send the information to IBM from IBM Db2 Analytics Accelerator Studio using the built-in FTP function. This method automatically adds the information to an existing IBM problem management record (PMR).

Configuring the trace behavior

IBM Db2 Analytics Accelerator provides trace profiles. These profiles consist of lists of components and events that result in the collection of trace information. In addition, they determine the verbosity level of the information, that is, how detailed the trace information will be. Profiles are available for accelerators as well as for stored procedures. You can also use custom profiles that have been saved to an XML file.

About this task

When a traceable event was caused by a component that is covered by the selected profile, trace information about this event is collected and stored in a temporary file.

You can save the temporary trace information to a permanent file or even add the information to a problem management record (PMR). Where to save the collected information is configured in another dialog, the Save Trace window.

Recommendation: Tracing can have a considerable impact on the system performance. The more verbose the information, the bigger the (negative) impact. Therefore, use the default profiles unless you are instructed otherwise by IBM support.

If none of the available profiles covers your problem scenario, you can ask IBM support for an appropriate profile. If you open a PMR, IBM support will give you detailed instructions on how to configure the trace behavior.

Two sets of profiles are available. The first set captures accelerator events; the second events that were caused by stored procedures. The scope of these profile sets is different. Accelerator trace profiles are saved on the selected accelerator. When active, they capture any event on this accelerator, no matter if incidents were caused by an (external) application program or by a function of IBM Db2 Analytics Accelerator Studio. In contrast, stored procedure trace profiles are a part of IBM Db2 Analytics Accelerator Studio. This means that you can only capture events with the help of these profiles if a stored procedure is called by a function of IBM Db2 Analytics Accelerator Studio. Stored procedures that are called from external applications can thus not be traced. Furthermore, the temporary file, which holds the collected trace information about stored procedures, is deleted when you close IBM Db2 Analytics Accelerator Studio. Therefore, to retain access to this information, specify a permanent trace file in the Save Trace window.

Using the Configure Trace function invokes the **SYSPROC.ACCEL_CONTROL_ACCELERATOR** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator*

for *z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. Unless you are already connected to a database, double-click the database-connection-profile icon for the Db2 subsystem that is associated with the accelerator that you want to trace.
2. In the Administration Explorer, select the **Accelerators** folder.
3. In the Object List Editor, double-click an accelerator to open the Accelerator view.
4. In the header section at the top of the Accelerator view, click **Configure**.
5. Under the heading Accelerator, from the **Profile** drop-down list, select a profile for accelerator tracing. By default, the currently active trace profile is selected. You can change this setting by selecting one of the other standard profiles that are delivered with IBM Db2 Analytics Accelerator or by selecting a custom profile. A custom profile must be imported before you can select it from the **Profile** drop-down list. See step 6.
About the standard profiles, you find additional information in the **Description** field. The text in this field gives you details about the events that are covered by the profile.
6. To select a custom profile that has been saved to an XML file:
 - a. Click **Import**
 - b. Navigate to the XML file in your file system.
 - c. Select the file and click **OK**.
 - d. Select the custom profile from the **Profile** drop-down list.
7. Under the heading Stored Procedures, from the second **Profile** drop-down list, select a profile for stored-procedure tracing. By default, the currently active trace profile is selected. You can change this setting by selecting one of the other standard profiles that are delivered with IBM Db2 Analytics Accelerator or a custom profile. For importing a custom profile, see step 6.
8. To discard your individual settings and return to the default profiles for both, accelerator tracing and stored-procedure tracing (**DEFAULT** and **Off**), click **Restore Defaults**.
9. Click **OK** to save your settings.

Results

The selected profiles remain active until other profiles are selected or until they are reset to the default by the **Reset configuration profiles to default after collecting traces** option, which is selectable in the Save Trace window.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Saving trace information

The Save Trace function saves the collected trace information. In the Save Trace window, which opens before the action is completed, you can specify or change the settings for the save operation.

About this task

You can specify or change the location of the trace file on your local system. The Save Trace window also allows you to include or exclude certain information from the bulk of information that was collected as a result of using a particular profile. You can also choose to add the Eclipse error log to the trace file, since the information in the Eclipse error log is not generated by means of an IBM Db2 Analytics Accelerator trace profile. Moreover, you can send trace information to IBM using the file transfer protocol (FTP) to the effect that the trace information is added to an existing problem management record (PMR).

Using the Save Trace function invokes the **SYSPROC.ACCEL_CONTROL_ACCELERATOR** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor, double-click an accelerator to open the Accelerator view.
3. In the header section at the top of the Accelerator view, click **Save**.
4. Select the information that you want to store in the trace file.

Accelerator

Log and trace information produced by components running inside Docker container (mainly IBM Db2 Analytics Accelerator and SQL engine). This includes, for example: accelerator logs, SQL engine logs, system logs of the Docker container.

Note: When you open the Save Trace window from the context menu of the Accelerators folder rather than from the header of the Accelerator view, the **Accelerator** check box is inactive. The reason is that selecting the entire Accelerators folder does not allow you to specify or select the particular accelerator in whose trace information you are interested. The **Save Trace** menu choice on the context menu of the Accelerators folder is a kind of “last rescue” option. Without it, there would be no way to submit trace information to IBM in case an error prevented the proper listing of the accelerators in the right pane (accelerators view). If such an error occurs, you have no access to the **Save Trace** function for individual accelerators.

Appliance

Trace information from the physical system in which IBM Db2 Analytics Accelerator and the SQL engine run (IBM Integrated Analytics System or the Secure Service Container LPAR for Db2 Analytics Accelerator on Z)

Stored procedure trace information

Trace information that is collected while running stored procedures of IBM Db2 Analytics Accelerator for z/OS, as the result of using a particular profile for stored-procedure tracing in the Configure Trace window. Using the check box, you can include or exclude this information from the trace file to be saved.

Eclipse error log

Error information that is collected by the Eclipse framework, on which the foundation for IBM Db2 Analytics Accelerator Studio, IBM Data Studio, is built. Hence this information does not pertain to errors in IBM Db2 Analytics Accelerator for z/OS, but to its software environment.

5. To revoke all individual settings and in a future run save just the information that is selected by using the default trace profiles, select **Reset configuration profiles to default after collecting traces**. This prevents that verbose trace information consumes too much space on your hard disks. If you want to retain your individual settings after saving them once, clear the **Reset configuration profiles to default after collecting traces** check box or leave it deselected (default).
6. Under the heading **Save to local file system**, in the **Folder** field, type the location of the trace file (full directory path) on your local system. Alternatively, you can click **Browse** to select a folder.
7. Select **Add trace information to IBM problem management record (PMR) using FTP** to add the selected information to an existing PMR.
8. In the fields next to **PMR number**, type the identification number of the PMR.
9. In the **FTP password** field, type the password for logging on to the IBM RETAIN server on which the PMR was opened.

Note: Most IBM FTP servers for PMRs accept the registered email address as the password.

The currently selected server is displayed in square brackets in the text above the **Password** field. To change the server or the proxy settings required to access the server, select **Window > Preferences > IBM Db2 Analytics Accelerator for z/OS > Save Trace Dialog** from the IBM Data Studio main menu.

10. If you want to reset your choices in the Save Trace window to make corrections or start over, click **Restore Defaults**.

Note: This button just undoes all of your recent actions in the Save Trace window. It does not reset profiles and affects your current settings rather than future settings. Compare this with the previous description of the **Reset configuration profiles to default after collecting traces** check box.

11. Click **OK** to save your settings.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Setting a refresh interval or a skew threshold

Select **Preferences > IBM Db2 Analytics Accelerator > Accelerator View** from the main menu to set the refresh interval for the Accelerator view or the warning threshold for table skew.

Procedure

1. Set values:
 - To set or change the interval after which the information in the Accelerator view is automatically refreshed, select the appropriate value from the **Automatically refresh information in the Accelerator view** drop-down list.

The default is **Every minute**, meaning that the information in the Accelerator view is refreshed each time after the passing of one minute.

- To set or change a table skew value that will result in a warning message when it has been exceeded by a very big accelerator-shadow table in the range of 100 million rows, type a decimal number between 0 and 1 in the text field next to **Show a warning if the skew value of very big tables exceeds**. Example: 0.7. The default is 0.25.

Notes:

- What is considered to be *very big* varies with the size of your IBM Integrated Analytics System. On smaller systems, the tables that will cause a warning are smaller and vice versa. The threshold is somewhere in the range between 50 million and 100 million table rows.
- The actual skew value can be greater than 1, and thus you could enter a value greater than one, but for the tables that are affected by the skew threshold (very big accelerator-shadow tables), 1 is already a bad value.

To be able to issue a warning, IBM Db2 Analytics Accelerator Studio must compare the specified skew threshold with the actual skew values of the tables on your accelerators. The actual values are obtained by running the **SYSPROC.ACCEL_GET_TABLES_INFO** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

- To disable checking for unsupported columns, clear the **Check for unsupported columns** check box. By default, this feature is enabled because it tells you beforehand why a table cannot be added or why certain columns were skipped.

Important: For the deactivation to take effect, you must disconnect, then reconnect to the Db2 subsystem. To do so:

- a. Go to the Administration Explorer.
 - b. In the **All Databases** folder, navigate to the appropriate Db2 subsystem icon.
 - c. Right-click the icon.
 - d. Select **Disconnect** from the menu.
 - e. Double-click the icon to reconnect.
2. Click **Apply** to save the changes or **Restore Defaults** to undo the changes.
 3. Click **OK** to save your settings and exit the window or just exit the window.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Updating accelerator tables continually

The incremental update function of IBM Db2 Analytics Accelerator for z/OS allows you to update accelerator-shadow tables continually. Changes to the data in original Db2 for z/OS tables are thus propagated to the corresponding target tables with a high frequency and just a brief delay. This way, query results from an accelerator are always extracted from recent, close-to-realtime data.

The information about changes to be transferred comes from the Db2 for z/OS log.

In the past, data in accelerator-shadow tables could only be updated by use of the table load function. With this function, you can only update entire tables and entire table partitions, and the latter only if the tables are partitioned by range or partitioned by growth. The incremental update function, in contrast, updates individual table rows. Advantages and disadvantages are connected with either method. Table 24 compares the advantages and disadvantages.

Table 24. Comparison of table update methods

Manual update of entire table using load function	Manual update of entire partitions using load function	Incremental update
For all types of tables	Only for range-partitioned tables and tables partitioned by growth	For all types of tables
Potentially long-running process, depending on the size of the tables	Potentially long-running process, depending on the size of partitions and the distribution of data changes across partitions (the more partitions are involved, the more time-consuming the update becomes)	Short-running process after initial load
Fast (up to one TB per hour)	Fast (up to one TB per hour)	Slower (in the range of GBs per hour)
Redundant load of table rows that have not changed, resulting in waste of processing resources (MIPS)	Redundant load of table rows that have not changed, resulting in waste of processing resources (MIPS)	No redundant load of table rows, no waste of processing resources.
Just a snapshot of the original data at some point in the past	Just a snapshot of the original data at some point in the past	Continual updates, recent data

Use cases

Incremental updates are the preferable method with regard to the following points of interest:

Realtime business intelligence

If realtime data is very important for your business intelligence operations, incremental updates are the preferable choice. Incremental updates do not guarantee that query data is the most recent available, but the latency is low, and this makes incremental updates a cost-efficient alternative.

Quick reporting on operational data

Incremental updates provide an uncomplicated, easy-to-use, and fast way to reporting on data in an operational data store (ODS) because the function is deeply integrated into your Db2 for z/OS and IBM Db2 Analytics Accelerator for z/OS setup. No up-front extraction and transformation of data is required to obtain results.

Cost-efficiency

Incremental updates are more cost-efficient than traditional update methods because there is no redundancy during updates and hence no waste of processing resources.

In addition, internal tests have shown that running incremental update processes affect the query response time only marginally.

High availability

Incremental updates can be configured in such a way that data changes in a Db2 for z/OS table are propagated to more than one accelerator-shadow table on multiple accelerators. If one accelerator fails, another one can take over to process incoming queries. This happens with just a short delay.

Making queries wait for incremental updates

To minimize the gap in data recency between the original Db2 for z/OS tables and incrementally updated accelerator-shadow tables (replication latency), you can delay the execution of queries until the latest incremental updates have been applied. This guarantees that an accelerated query against incrementally updated tables always "sees" the latest changes that have been committed in Db2 for z/OS.

This capability fulfills the criteria for Hybrid Transactional Analytical Processing (HTAP). It allows you to run transactional processes as well as analytical queries on a unified data store with a minimized time lag between transactional completion and analytical insight. Informed business decisions can be made much faster, on the basis of much more recent data.

To activate the delay, you must add a special register to your queries. You can also use a ZPARM or bind option to the same effect. The special register is:

```
SET CURRENT QUERY ACCELERATION WAITFORDATA = n.m.
```

where *n.m.* is a decimal number denoting seconds. The minimum is 0.0 (no delay). The maximum is 3600.0 (a delay of one hour).

Important: If you want to use a delay, specify at least 20 seconds. Lower values might work, but you must take the time into account that is needed to scan and read the SYSLOG and to report the findings. Experience has shown that realistic delays are around 20 seconds long.

Example

```
SET CURRENT QUERY ACCELERATION = ENABLE;  
SET CURRENT QUERY ACCELERATION WAITFORDATA = 120.0;  
SELECT * FROM XYZ WHERE [...]
```

The first special register in line 1 enables query acceleration. The second special register sets the delay to 120 seconds. What follows is the SQL code for the query. The execution of the query will be postponed for 120 seconds. During that time, the replication engine tries to apply all committed changes that have accumulated before the query started. If the replication engine cannot finish the job during that period, the query will not be accelerated. The further handling depends on the option set in the first special register and on the expiration behavior configured on the IBM Db2 Analytics Accelerator Console. The first special register is set to ENABLE. That means the query fails if one of the conditions for query acceleration is not met. Let's assume that the conditions can be met; so normally, the query would run on the accelerator. However, the default expiration behavior is to abort a query if the incremental updates cannot be applied in time. Supposing the default behavior is set, the query would indeed fail if not all of the pending incremental updates could be applied in 120 seconds.

How to calculate the proper delay time

You certainly ask yourself how to determine the proper delay time (WAITFORDATA value) for your environment. There are two values by which an

initial delay time or starting point can be calculated: the *replication latency* and the *replication velocity*. The replication latency is the time that the incremental update process is lagging behind. The replication velocity is a measure for the speed at which data changes could be applied in the past. You get these two values by running the following Db2 command:

```
-DIS ACCEL(<name>) DETAIL
```

where <name> is the name of the accelerator attached to the Db2 subsystem. If incremental updates have been configured for this pair of Db2 subsystem and accelerator, you find the two values in the command output. For example:

```
...
CURRENT REPLICATION LATENCY FOR THIS DB2 SYSTEM = 2000 MS
NUMBER OF SUCCESSFUL QUERY REQUESTS WITH DELAY PROTOCOL FOR ALL DB2 SYSTEMS = 0
NUMBER OF EXPIRED QUERY REQUESTS WITH DELAY PROTOCOL FOR ALL DB2 SYSTEMS = 1
REPLICATION VELOCITY = 0.04 DB2 LOG SECONDS APPLIED PER SECOND
...
```

In this example, The latency is 2000 milliseconds or 2 seconds, and the velocity is 0.04. You get the proper WAITFORDATA value by dividing latency by velocity. For the example, this gives you a WAITFORDATA value of 50 seconds to start with.

Conditions for the success or failure of a query

Whether a query fails depends on the setting of the SET CURRENT QUERY ACCELERATION special register. It also depends on the type of the table. For accelerator-only tables, the WAITFORDATA option is always ignored because it can be assumed that the data in these tables is always the most recent data available (there is no other data in a Db2 for z/OS source table).

If the WAITFORDATA option has been set for queries that reference incrementally updated tables as well as tables not enabled for incremental updates (loaded tables), the query might fail if loaded tables are to be excluded.

For information on how to configure the treatment of loaded tables, see *Including or excluding loaded tables from WAITFORDATA queries* in the *IBM Db2 Analytics Accelerator for z/OS: Installation Guide*.

Table 25 lists the possible combinations of settings and their effect with regard to query processing. It is assumed that an incoming query satisfies the conditions for query acceleration.

Table 25. Settings that influence WAITFORDATA query processing

CURRENT QUERY ACCELERATION	WAITFORDATA	Loaded tables(This has an effect only if loaded tables are referenced by the query)	Query processing
NONE	N/A	N/A	Db2 for z/OS

Table 25. Settings that influence WAITFORDATA query processing (continued)

CURRENT QUERY ACCELERATION	WAITFORDATA	Loaded tables(This has an effect only if loaded tables are referenced by the query)	Query processing
ENABLE / ELIGIBLE / ALL	0.0	Include	Accelerator
		Exclude	Query fails
	> 0.0	Include	Query fails if outstanding incremental updates cannot be applied within WAITFORDATA period.
		Exclude	Query fails
ENABLE WITH FAILBACK	0.0	Include	Accelerator
		Exclude	Db2 for z/OS
	> 0.0	Include	Db2 for z/OS
		Exclude	Db2 for z/OS

Error codes in connection with WAITFORDATA processing

If an error occurs in connection with WAITFORDATA processing, an error message with SQLCODE = -904 is returned in the format:

WAITFORDATA rsn=<reason code>:<reason code text>

where

<reason code>

An integer (positive, negative, or 0)

<reason code text>

The message text

Example

SQLCODE = -904, ERROR: UNSUCCESSFUL EXECUTION CAUSED BY AN UNAVAILABLE RESOURCE. REASON 00E7000E, TYPE OF RESOURCE 00001080, AND RESOURCE NAME WAITFORDATA rsn=1:Table(s) are in invalid state.

The message shows rsn=1 as the <reason code>, and Table(s) are in invalid state. as the <reason code text> .

Table of error codes

Table 26 on page 122 lists the possible reason codes and message texts. It also provides a description of the most common causes.

Table 26. Reason codes and texts in WAITFORDATA error messages

RSN code	Message text	Common causes
1	Table(s) are in invalid state.	Not all tables to be processed are in the <i>replicationInProgress</i> state. This state, however, is a precondition for WAITFORDATA processing. Check the Last Load column in the Tables section of the Accelerator view in IBM Db2 Analytics Accelerator Studio.
2	Subscription state is invalid.	Incremental updates have not been started for the accelerator on which the tables are located. In IBM Db2 Analytics Accelerator Studio, check the Replication status in the header of the Accelerator view. The status must be <i>Started</i> . If necessary, click Start .
3	Set of tables is empty or invalid.	<p>Some tables are empty or have not been enabled for incremental updates.</p> <p>Make sure that none of the tables in the set are empty. If you find empty tables, reload them, or exclude them from the set.</p> <p>Check the Replication Since column in the Tables section of the Accelerator view in IBM Db2 Analytics Accelerator Studio. The column must show a date for all the tables to be processed. If necessary, click Replication > Enable Replication.</p>
5	Timeout for WAITFORDATA reached.	Pending incremental updates could not be applied within the WAITFORDATA period.

Dependencies on other processing states, configurations, or product features

Other processing states or the use of certain configuration settings or other product features might affect query processing if WAITFORDATA is enabled (value > 0.0):

- If one of the tables referenced by the query is currently being reloaded, or in the spilling or draining state, the query will fail or be processed by Db2 for z/OS (ENABLE WITH FAILBACK). This is because the end of these processes or table states cannot be predicted, and in most cases, they take much longer than the WAITFORDATA period.

- If workload balancing is used in a high-availability setup, a WAITFORDATA setting > 0.0 might lead to successful or failing queries. The decisive factor is the replication latency of the accelerator that a query is sent to. If the latency is higher than the WAITFORDATA period, the query fails. The problem is that one does not know beforehand to which accelerator a query is sent. The accelerator is picked at random.
- If the product software on the accelerator is back-level, and does not support the WAITFORDATA protocol, a query does not necessarily fail. If CURRENT QUERY ACCELERATION is set to ENABLE, ELIGIBLE, or ENABLE WITH FAILBACK, the query might be processed by Db2 for z/OS. If, on the other hand, the special register is set to ALL, the query will fail. The same applies to the equivalent ZPARM or bind option settings.
- You cannot change the distribution key or the organizing keys of a table while incremental updates are enabled. To enable the **Alter Keys** function, you must first disable replication, and that means that a setting of WAITFORDATA > 0.0 will have no effect.
- A failure of the database engine on the accelerator has the effect that all queries with a WAITFORDATA setting will fail. However, these queries will be restarted and processed as usual after the database engine has recovered or after a replacement instance has been started.
- Treatment of date, time, and timestamp values: An accelerated query that retrieves dates might give you the same values as Db2 for z/OS, but a query that selects time values or timestamps will probably produce different results. The reason is that the WAITFORDATA period and the difference between the z/OS system clock and the accelerator's system clock (=< 7 seconds) are added to these values while the incremental updates are being applied.

Related concepts:

“Accelerator view” on page 64

Related tasks:

“Including or excluding tables from incremental updates” on page 125

Starting or stopping incremental updates

When a Db2 subsystem has been configured for incremental updates, you can start or stop incremental updates for attached accelerators from the corresponding accelerator views in IBM Db2 Analytics Accelerator Studio. Using the functional links in the header starts or stops incremental updates for an entire accelerator.

About this task

When a subsystem has been configured for incremental updates, the header of accelerator view for regular accelerators shows an additional entry, which is labeled **Replication**. Next to this label, you find a status indicator and a functional link. Status indicator and link vary according to the current status of the accelerator. For example, if incremental updates have been stopped, the status indicator shows *Stopped* and a functional link labeled **Start**. This is reversed to *Started* and **Stop** when incremental updates have been started. The following statuses are possible:

Disabled

The incremental update function has not been configured for this accelerator and is therefore not available.

Error The incremental update function has been configured, but an error occurred. In most cases, this means that a component cannot be reached.

Therefore, it cannot be started or controlled. The event log might provide useful information about the root cause of the error.

Started

The incremental update function is running and updates are applied to accelerator-shadow tables that have been included in the process.

Stopped

The incremental update function is available and accelerator-shadow tables have been included in the process. However, the function is not running right now.

See Figure 10.

Accelerator: VMNPS04 @ DWEDA11

Acceleration: Unknown Start Stop	Credentials valid since: 11/13/12 6:56 PM Update
Status: Unknown	Trace: DEFAULT / OFF Configure Save Clear
Used space: N/A	Active queries: N/A
Replication: Stopped Start	Replication latency: Stopped Show events

► Query Statistics
► About

▼ Tables (3 of 3 loaded / 3 of 3 enabled for acceleration)

+ Add... ✎ Alter Keys... ✖ Remove ⬆ Load... ⚙ Acceleration ▼ 💾 Storage Saver ▼ 🔄 Replication ▼ 🔴

Name	Size	Rows	Acceleration	Last Load	Replication Since
IUTEST	-	- 3 of 3	3 of 3 tables	3 of 3 tables	3 of 3
CUSTOMERA	-	- Enabled	11/16/12 1:21 PM	11/16/12 1:21 PM	11/16/12 1:21 PM
CUSTOMERA2	-	- Enabled	11/16/12 1:21 PM	11/16/12 1:21 PM	11/16/12 1:21 PM
CUSTOMERA3	-	- Enabled	11/16/12 1:21 PM	11/16/12 1:21 PM	11/16/12 1:21 PM

Figure 10. Replication status and functional link in the header of the accelerator view

Important: Using the functional links starts or stops the incremental update process for all tables that have been enabled for incremental updates on the selected accelerator.

Procedure

1. Start IBM Db2 Analytics Accelerator Studio and connect to a Db2 subsystem.
2. In the Administration Explorer, select the **Accelerators** folder.
3. In the Object List Editor on the right, double-click the accelerator that you want to include in the incremental update process.
4. Start or stop incremental updates by taking one of the following actions:
 - To start incremental updates, click **Start** in the header of the accelerator view. The **Replication status** in the header of the accelerator view changes to *Started*.
 - To stop incremental updates, click **Stop** in the header of the accelerator view. This opens the Stop Replication window, in which you can select between the following options for stopping the incremental update process:

Controlled

Completes all work in progress before stopping the process.

Immediate

Stops the process without completing work in progress. Compared with the **Controlled** option, using this option causes a restart of the process to take longer.

Click **OK** after selecting an option. You return to the accelerator view, which now shows a **Replication status** of *Stopped*.

Including or excluding tables from incremental updates

Incremental updates can be enabled or disabled on the table level. This allows you to select the tables that take part in the process. When a subsystem has been configured for incremental updates, the toolbar of the accelerator view for regular accelerators shows an extra button, which works like a drop-down menu.

Before you begin

An accelerator-shadow table can only be included in the incremental update process if it contains data, that is, if it has been loaded.

When loading a table, make sure that the lock mode **TABLE_SET** is selected because this guarantees that data changes will not go amiss during the time between unloading the table data from Db2 to the accelerator and the point where the incremental update process sets in.

You can select lock mode **ROW** for a better performance. This lock mode effects a faster propagation of changes because the incremental updates are not put on halt when a table load starts and continue after the load, but are written to a spill queue, which is drained in the course of events. However, this comes at the price of a slower load process because lock mode **ROW** prevents the loading of the temporary table, which eventually replaces the former accelerator-shadow table. That is, loads are executed as a series of row changes. Decide whether faster loads or faster incremental updates are more important to you.

Attention:

- Tables to be included must have a unique constraint (primary key or primary index). If that is not the case, IBM Db2 Analytics Accelerator adds a hidden column with unique values to the accelerator-shadow table. This column, called *informational unique constraint*, serves as the primary key or primary index.
- When lock mode ROW is set, Db2 data is unloaded at isolation level CS (cursor stability). Whether a row, a page, or an entire table is locked is determined by the LOCKSIZE defined for the table space of the source tables. To avoid deadlocks during the draining of the spill queue, set this table space to LOCKSIZE ROW if you want to use lock mode ROW for your incrementally updated tables.
- If you update Db2 for z/OS tables by running the LOAD utility rather than an INSERT, UPDATE, or DELETE operation, you must set the following parameters (for the LOAD utility):
 - SHRLEVEL CHANGE
 - LOG YES

Otherwise, the changes that were made by the LOAD utility are not detected by the incremental update function, and will thus not be reflected in your accelerator-shadow tables.

- With the introduction of IBM Db2 Analytics Accelerator for z/OS Version 3.1, the Db2 attribute DATA CAPTURE is added to the tables on an accelerator. The attribute can carry the value Y or N (default), for *yes* or *no*. When incremental updates are enabled for a table, the DATA CAPTURE attribute of the table is set to the value Y. Once set, this attribute value persists, even if the table is disabled at a later time. Bear this in mind, especially if you run applications that use the DATA CAPTURE attribute.
- You might have to reload or even remove tables from an accelerator after the execution of an ALTER TABLE or ALTER TABLESPACE statement in Db2 for z/OS. For more information, see *What to do after changing a table or table space in Db2 for z/OS*. You find a **Related reference** link at the end of this topic.
- IBM InfoSphere Data Replication for z/OS uses a metadata table called DMMD_SIGNALS . This table is a kind of an interface for the entire process. There is absolutely no reason to select it for incremental updates. Ensure that you do not select it inadvertently as doing so will lead to unpredictable and unwanted side-effects.

About this task

Two different situations must be considered before adding tables to an incremental update process:

- An accelerator-shadow table has not yet been loaded, that is, it is in *Initial load pending* state. If you enable such a table, and load it immediately thereafter, the tables in Db2 and on the accelerator are in sync. No data changes will be lost because table contents are the same at the outset and the incremental update process captures all future changes and propagates these to the accelerator. Tables of this type show the same timestamp in the **Last Load** and **Replication Since** columns of the accelerator view. See the entry for the REP1 table in Figure 11 on page 127.
- An accelerator-shadow table has already been loaded when it is added to the incremental update process. In this case, it is likely that the tables are out of sync because data changes might have been applied to the original Db2 table, and not been reflected in the corresponding accelerator-shadow table because the last manual update of the table was before these changes happened. If a safety mechanism was not in place, such tables would continue to be out of sync

because the incremental update process can only capture and propagate changes that occur in the future. To avoid this, the status of such tables changes to *Initial load pending* when incremental updates are enabled. This state enforces a complete reload of the tables before they can be used, so that the synchronization gap is closed before the incremental update process sets in.

Important: If the load fails for some of the selected tables, incremental updates are automatically disabled for these tables.

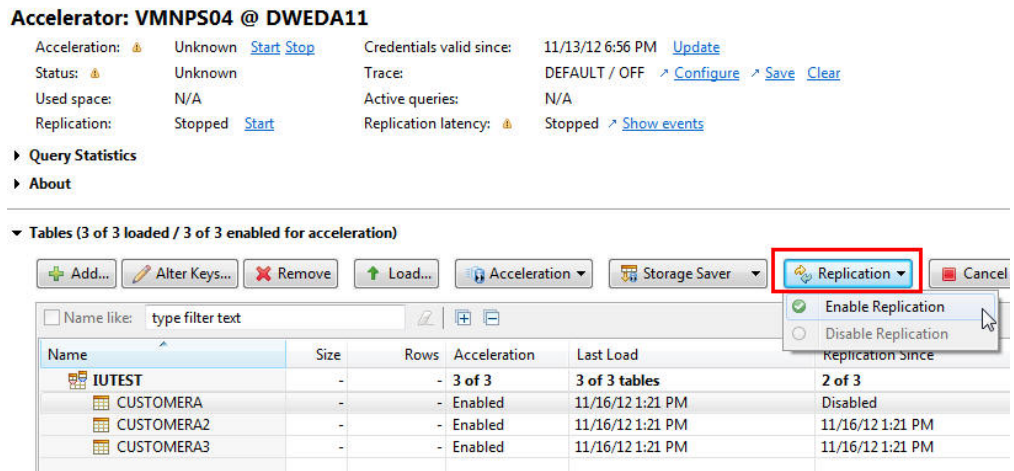


Figure 11. Timestamp columns and **Replication** button in the accelerator view

The **Replication Since** column shows the following states in the order indicated as soon as a load process begins:

1. *Suspended*
2. *Refreshing*

The *Refreshing* state is shown until the last change has been drained from the spill queue. When you move the mouse pointer over the *Refreshing* label, the current processing phase is displayed in a tool-tip window. The refresh process consists of the following phases:

Pending

The table load process was started, but no data has yet been transferred. Therefore, only the data that has been transferred before the refresh is available for accelerated queries.

Spilling

The table load is in progress, and data is being transferred. Incremental updates that occur during this time are written to the spill queue. Therefore, only the recently transferred load data is available for accelerated queries.

Draining

The load has finished and the changes that have been collected in the spill queue are being applied. Only the data changes that have already been transferred to the table are available for accelerated queries.

Tip: If a table remains in the draining phase for several hours, you are probably in the very rare situation in which new changes arrive in the spill

queue faster than they can be drained from it, so that the draining phase never ends (infinite drain). In this case, stop incremental updates completely for a while.

Restriction: It is not possible to enable tables if the table name or the schema name contains GB18030 characters (Simplified Chinese) of Unicode plane 2 (U+20000-U+2FFFF: Supplementary Ideographic Plane). Trying to do so results in an error.

Procedure

1. Select the accelerator-shadow tables to be included or excluded.
2. Include or exclude tables from the process:
 - To include selected tables in the incremental update process, click **Replication > Enable Replication**. See Figure 11 on page 127 for the location of the button.
 - To exclude selected tables from the incremental update process, click **Replication > Disable Replication**.
3. If you have disabled the selected tables, click **Start** in the header of the accelerator view to restart the incremental update process for the remaining enabled tables. Also note that incremental updates are not restarted automatically if you enable just one table and the steps involved are completed in this order:
 - a. Define the accelerator-shadow table.
 - b. Enable incremental updates for this table.
 - c. Load the accelerator-shadow table.
 - d. Start incremental updates (**Replication > Start**).
 - e. Reload the accelerator-shadow table.

Normally, when this is done for more than one table, incremental updates continue or are restarted for all tables that are not being loaded or where loading has been finished, which finally results in an automatic restart for all tables. In this special case (just one table), however, you must restart incremental updates manually after the reload.

Related reference:

“What to do after changing a table or table space in Db2 for z/OS” on page 133

Related information:



CREATE TABLESPACE

Monitoring incremental updates

You can monitor the changes to accelerator tables that result from incremental updates. The header of the accelerator view in IBM Db2 Analytics Accelerator Studio provides an additional line of information when incremental updates have been configured for the accelerator. This line of information is labeled **Replication latency**.

One of the following values is displayed next to this label:

Stopped

Incremental updates have been stopped for the entire accelerator.

Low It will take less than three minutes to propagate recent changes to the corresponding accelerator-shadow tables.

Medium

Recent changes to the original Db2 tables will be propagated to the corresponding accelerator-shadow tables within three and six minutes.

High It will take six minutes or longer to apply recent changes to Db2 tables to the corresponding accelerator-shadow tables.

When you move the mouse pointer over the latency value, a pop-up window shows the following information:

Target Agent

Operations completed by the incremental update function.

Inserts

Number of row inserts in accelerator-shadow tables that have been included in the incremental update process.

Deletes

Number of row deletions from accelerator-shadow tables that have been included in the incremental update process.

Remember:

- An UPDATE is a DELETE followed by an INSERT, so these two figures account for all incremental updates that have been applied.
- Since an UPDATE in Db2 is a DELETE followed by an INSERT on the accelerator side, the values can be different from those in Db2 for z/OS (see **Source Agent**). Furthermore, multiple operations on a single row within a short time might be reported as just one single operation.
- If a load operation is in progress and lock mode **Row** has been selected, the original Db2 tables are not locked during the load. Hence it is possible to INSERT new rows into the Db2 tables even though a load is taking place. If incremental updates are enabled for the same tables, the incremental update function stores these new rows in the spill queue until the load has finished, and applies these changes thereafter (draining the spill queue). In the course of a drain, a DELETE and an INSERT operation is carried out for each row in the affected accelerator-shadow tables. The DELETE deletes the row that was inserted by the previous load operation, and the INSERT adds this row again (applies the latest change) . This has the effect that the number **Deletes** increases in cases like this.

See Figure 12 on page 130.

Accelerator: VMNPS04 @ DWEDA11

Acceleration: 🚩 Unknown [Start](#) [Stop](#) Credentials valid since: 11/13/12 6:56 PM [Update](#)
 Status: 🚩 Unknown Trace: DEFAULT / OFF [Configure](#) [Save](#) [Clear](#)
 Used space: N/A Active queries: N/A
 Replication: Started [Stop](#) Replication latency: Low [Show events](#)

► Query Statistics
 ► About

▼ Tables (3 of 3 loaded / 3 of 3 enabled for acceleration)

[Add...](#) [Alter Keys...](#) [Remove](#) [Load...](#) [Acceleration](#) [Storage Saver](#) [Replication](#) [Cancel](#)

☐ Name like: type filter text

Name	Size	Rows	Acceleration	Last Load	Replication Since
📁 IUTEST	-	-	3 of 3	3 of 3 tables	2 of 3
📄 CUSTOMERA	-	-	Enabled	11/16/12 1:21 PM	Disabled
📄 CUSTOMERA2	-	-	Enabled	11/16/12 1:21 PM	11/16/12 1:21 PM
📄 CUSTOMERA3	-	-	Enabled	11/16/12 1:21 PM	11/16/12 1:21 PM

Figure 12. Replication latency in the accelerator view

Using the event viewer for incremental updates

The event viewer shows system messages that are related to incremental update processes. It is equipped with various filtering options to facilitate the search for particular messages.

About this task

The event viewer shows a log entry for each system event that occurred in connection with incremental updates. The entries are listed in a table, which contains the following columns:

Type The type of the event or message connected with the event. Possible types are **Error**, **Warning**, and **Information**.

Message

The message code and the starting text of the event message. The full message is displayed in the **Message** window on the lower right.

Event ID

The unique identifier of the event.

Originator

The component that issued a message.

Time The time at which an event occurred.

Procedure

- To open the event viewer:
 - Open the Accelerator view of the appropriate accelerator. This must be an accelerator that is included in the incremental update process. That is, incremental updates must be configured for the Db2 subsystem to which the accelerator is attached. The current replication status of the accelerator (*Started*, *Stopped*, or *Error*) is irrelevant.
 - Click the **Show events** link in the header of the Accelerator view.
- Selecting a message in the Replication Events window cause the main column values to be displayed in the **Event Details** section on the lower left, and the full message text to be displayed in the **Message** window on the lower right.

- Using the **View** drop-down list on the upper right, you can hide all messages of an event type that is different from the selected type. To apply this type of filter, click the **Run** button after selecting an event type from the drop-down list. For example, if you select **Error** and **Run**, only error messages and messages with a severity higher than *error* are displayed. The underlying mechanism is the same as for the classification of trace information. The lower the severity of the selected information type, the more verbose is the information that is displayed. While the **Error** type provides information about errors and even severer events only, the **Warning** type provides information about warnings, errors, and severer events. That is, each information type with a lower severity includes the information of the types with a higher severity.
- Using the **Show** drop-down list, you can select a time frame for the messages to be displayed. To apply this type of filter, click the **Run** button after selecting a time frame from the drop-down list. For example, if you select **Last 24 hours** and **Run**, the event viewer shows all events that happened during the last 24 hours.
 1. Selecting **Custom interval** from the **Show** drop-down list opens the Define Custom Interval window, in which you can define a time frame of your own:
 - To use a predefined time frame (same as using the **Show** drop-down list in the Replication Events window), leave the **Show events within** radio button as is, and select a suitable time frame from the adjacent drop-down list.
 - To define a custom time frame, select **Show events** and:
 - a. In the line starting with **From**, select the start of your time frame by using the date picker and the time spin buttons on the right.
 - b. In the line starting with **To**, select the end of your time frame by using the date picker and the time spin buttons on the right.
 2. Click **OK** to return to the Replication Events window.
- Typing a search term in the search field on the top of the table reduces the messages being displayed to those messages that contain the search term. For example, if you know the error code belonging to a particular message, you can type in this error code to show just the messages for one particular type of error.
- Close the event viewer by clicking the cross-shaped icon on the window tab.

Warnings related to incremental update processes

The following warnings might occur in connection with incremental updates.

InsufficientLockModeWarning

This warning is issued if you did not select the option **Lock Db2 tables while loading** for the initial load of an accelerator-shadow table taking part in the incremental update process, or if you selected the lock modes **NONE** or **PARTITION**.

If locks were disabled for the initial load of an accelerator-shadow table (**Lock Db2 tables while loading** not selected), the following message is displayed:

One of the tables has replication enabled. This table must be locked while loading to avoid an inconsistency of data between Db2 and the accelerator.

If the lock modes **NONE** or **PARTITION** were used, data consistency between tables in Db2 and accelerator-shadow tables cannot be guaranteed because table changes can be applied to the Db2 tables during the unload process. If possible, use the lock mode **TABLE_SET**.

NoUniqueConstraintsDefinedWarning

A table does not have a unique constraint, such as a unique key or unique index. This would result in a poor performance when UPDATE or DELETE changes are propagated to an accelerator-shadow table (because it takes longer to identify the changed table rows).

To avoid a performance degradation, the product inserts informational unique constraints into the accelerator-shadow table. That is, a hidden column with unique values is inserted to serve as the unique index.

OrganizingKeysAndUniqueConstraintsMismatchWarning

The unique constraints of a table in Db2 for z/OS differ from those of the corresponding accelerator-shadow table or exceed the number of matching organizing keys. This might lead to a slow propagation of data changes to the accelerator. In the case of different unique constraints, it just takes longer to identify the table rows to be updated or deleted. A mismatch between the unique constraints of the table and the organizing keys affects the change propagation negatively if there are more unique constraints than matching organizing keys. If extents cannot be located for certain table rows because there is no matching organizing key, table scans take longer.

Collecting CDC trace information

None of the options in the Save Trace window of IBM Db2 Analytics Accelerator Studio collects trace information about the CDC Capture Agent. This information, however, might be the key to a problem with the incremental update function.

About this task

If you are facing a problem with the incremental update function, provide as much information about the related software components as possible the support team. This includes information about the CDC Capture Agent, which is the part of IBM InfoSphere Data Replication for z/OS that runs on z/OS. The support team might even ask you for this type of information. Here is a description on how to obtain it.

Procedure

1. The CDC Capture Agent is just one of the started tasks running on your z/OS system. If you do not know the name of the started task, ask a z/OS systems programmer which task that is.
2. In the SDSF program on z/OS, type XDC in front of the relevant address space output to copy its contents to a sequential data set.
3. Transfer this data set in ASCII mode to your Linux or Windows workstation and add the resulting .txt file to your problem management record (PMR).
4. If a trace dump has been generated, include that information as well. Transfer this data in TERSED format to your workstation because this retains the mainframe-readable formatting of the data during the transfer to a non-mainframe system, and thus ensures that the data remains useful for IBM support. Use an extension of .trs for the files. The TERSED format also reduces the file size. See the documentation for the AMATERSE program if you need further assistance.

What to do after changing a table or table space in Db2 for z/OS

In a number of cases, an ALTER TABLE or ALTER TABLESPACE statement in Db2 for z/OS does not involve maintenance work on your accelerator-shadow tables. There are cases, however, in which it does. See which steps you might have to take.

NOT Programming Interface information

ALTER TABLE

- If a column was added, changed, or deleted by an ALTER TABLE operation on a Db2 table, queries against this table are not accelerated if they reference the new, changed, or deleted column (reason code 14). Therefore, you must complete the following steps after an ALTER TABLE operation:

1. Remove the corresponding accelerator-shadow table from all accelerators on which it is defined.

In most cases, a REORG operation is not required before the table can be redefined on the accelerator. However, a REORG is required in the following cases, although this seems to be just a recommended action, as the advisory *REORG-pending* (AREO) state suggests:

- The ALTER TABLE operation added a column.
- The ALTER TABLE operation changed column data types.
- The ALTER TABLE operation dropped columns in *Db2 11 for z/OS*.

2. Redefine the table on the accelerators from which you previously removed it.
3. Load the redefined table on all accelerators.
4. Enable query acceleration for the redefined table on all accelerators.

ALTER TABLESPACE

If a table space was changed in Db2 for z/OS (ALTER TABLESPACE), and the tables of that table space also exist on an accelerator, actions are required under the following conditions:

- If the maximum number of partitions has changed (MAXPARTITIONS), you only have to reload the new or changed partitions. It is not required to reload existing partitions that have not changed.
- If a table space was changed from *segmented* to *partitioned by growth*, you must remove the accelerator-shadow tables of that table space from all accelerators, then redefine and re-prepare these for accelerated queries. Follow steps 1 through 4 to do that.

End of NOT Programming Interface information

Updating IBM Db2 Analytics Accelerator Studio

Follow the instructions here to download and install an upgrade of IBM Db2 Analytics Accelerator Studio.

About this task

To download a refresh pack for IBM Db2 Analytics Accelerator Studio, a workstation with an internet connection is required.

If IBM Installation Manager is installed on a workstation without an internet connection, the program cannot automatically locate the repositories for updates of

already installed products or additional plugins. Therefore, an administrator has to download the update repository to a central location that users without an internet connection have access to.

Workstation users without an internet connections can then let their local IBM Installation Manager installations point to the repository in the central location or copy the entire installation package from the central location to their workstations and go on from there.

You can use the newer versions of IBM Db2 Analytics Accelerator Studio with older versions of accelerators and stored procedures. However, the user interface might not show new functions until you update the stored procedures to the next higher or newest version.

Procedure

- To download a refresh pack for yourself or for others from a computer with an internet connection:
 1. Click the following link:

Select fixes: IBM Db2 Analytics Accelerator Studio
(All releases, All platforms)
 2. Select the proper refresh pack and click the **Continue** button above the list.
 3. Sign in with your IBM ID or click **Create IBM id** and follow the instructions to create an IBM ID first.
 4. Select your preferred download method:
 - **Download using Download Director** (requires Java enabled browser)
 - **Download using bulk FTP**
 - **Download using your browser (HTTPS)**
 5. Leave the check box **Include prerequisites and co-requisite fixes (you can select the ones you need later)** selected.
 6. In the View and accept terms window, click **I agree** to agree to the terms and conditions.
 7. Click **Download now** and follow the instructions.
 8. Extract the content of the downloaded archive to a directory on your local workstation or to the central location (for access or download by other users).
- To update an IBM Db2 Analytics Accelerator Studio installation:
 1. Start IBM Installation Manager.
 2. Update IBM Db2 Analytics Accelerator Studio by using one of the following procedures:
 - The following approach works if you can still connect to a previously accessed IBM Installation Manager repository, and if that repository has been updated (contains the update):
 - a. Click **Update**.
 - b. Follow the instructions in the wizard. Deselect components that you do not want to update.
 - If the first method did not work, but you can access the repository file on your company's network or on your local workstation:
 - a. From the main menu of IBM Installation Manager, select **File > Preferences > Repositories**.
 - b. Click **Add Repository**.

- c. Click **Browse** and navigate to the directory that contains the extracted update package, either in the central location, or on your local workstation.
- d. Select the `repository.config` file and click **Open**.
- e. Click **OK** twice to return the main window of IBM Installation Manager.
- f. Click **Install**.
- g. Follow the instructions in the installation wizard.

Installing updates

Refer to the appropriate sections to update individual components or migrate from one version to another.

It is rarely necessary to update all components. However, if you must update more than one component, follow the suggested order.

Order in which to update components

1. IBM Db2 Analytics Accelerator software. When in doubt: It is better if the accelerator software is at a newer level than the stored procedures. See *Updating IBM Db2 Analytics Accelerator software* in the *Installation Guide*.
2. Prerequisite PTFs for Db2 11 for z/OS or Db2 12 for z/OS. See *Installing prerequisite PTFs for Db2 for z/OS* in the *Installation Guide*.
3. Stored procedures. See *Updating the IBM Db2 Analytics Accelerator stored procedures* in the *Installation Guide*.
4. Db2 Analytics Accelerator on Z. See *Updating Db2 Analytics Accelerator on Z* in the *Installation Guide*.
5. IBM Db2 Analytics Accelerator Studio. Actually, the order does not matter for IBM Db2 Analytics Accelerator Studio. You can update it any time. See “Updating IBM Db2 Analytics Accelerator Studio” on page 133

Updating IBM Db2 Analytics Accelerator software

These are software components that run on the accelerator hardware, that is, the IBM Integrated Analytics System. This chapter does not apply to updates for Db2 Analytics Accelerator on Z. Components on the IBM Integrated Analytics System are the IBM Db2 Analytics Accelerator, the SQL engine, and the operating system of the accelerator.

Installing update packages for the accelerator

IBM Db2 Analytics Accelerator for z/OS software includes specific libraries and other code to be installed on the IBM Integrated Analytics System.

Before you begin

Make sure that the following conditions apply:

- Make sure that IBM Db2 Analytics Accelerator for z/OS has been successfully connected to Db2 for z/OS.
- IBM UNIX System Services is installed on your z/OS data server.
- You have a user ID and a password to log on to your z/OS data server. The user ID has read access to the z/OS UNIX file system (zFS).
- The user ID is authorized to run the `SYSPROC.ACCEL_UPDATE_SOFTWARE2` stored procedure.

About this task

The entire installation or update procedure consists of the following steps:

1. Downloading the fix pack or update package.
2. Transferring the update packages to the accelerator.
3. Activating the newly installed software.

This topic covers step 1 only.

Procedure

Downloading the fix pack or update package. You have two options here:

- Download the fix pack or update package from IBM Fix Central to your workstation and then upload the package from your workstation using IBM Db2 Analytics Accelerator Studio. Place the checksum file for installation verification in the same directory on your workstation.

There is a checksum file for every supported algorithm. Supported algorithms are SHA256, SHA1 and MD5. The corresponding files on IBM Fix Central are:

- sha256sums.txt
- sha1sums.txt
- md5sums.txt

Any of these files contains one line for each package. Every line consists of two values. The first value is the hash used for verification; the second value is the corresponding package name.

Important: During the process, a large global temporary table is created in Db2 for z/OS. This table requires up to 25 GB of disk space. Make sure that this amount of disk space is available.

- Download the fix pack or update package from IBM Fix Central and save it directly in the directory specified by the AQT_HOST_PACKAGE_DIRECTORY environment variable. To use this second option, you must:
 1. Log on to your z/OS data server using a remote client, such as IBM Personal Communications.
 2. Set the AQT_HOST_PACKAGE_DIRECTORY environment variable to a directory path in the z/FS. Then save the update packages in this directory. Place the checksum file for installation verification in the same directory on your workstation (for details, see the previous description of the first download option).

Important: An update packages requires up to 25 GB of disk space. Make sure that enough space is available in the z/FS directory that AQT_HOST_PACKAGE_DIRECTORY points to.

Important: The directory that AQT_HOST_PACKAGE_DIRECTORY points to requires multiple gigabytes (GB) of disk space. For example, if you work with multiple accelerators in a rolling upgrade scenario, you must have multiple instances of the installation packages in the file system. So make sure that the directory can accommodate all the files.

Results

Your local workstation or z/FS directory (as set by the AQT_HOST_PACKAGE_DIRECTORY environment variable) on your z/OS data server now contains the accelerator update package *version.tar.z*.

Transferring update packages for the accelerator

Transfer update packages for IBM Db2 Analytics Accelerator for z/OS on an IBM Integrated Analytics System by completing the steps in this section.

Before you begin

Make sure that the following conditions apply:

- You have downloaded suitable accelerator installation packages and their respective checksum files to a workstation that is connected to your z/OS data server, or such packages exist in a z/FS directory on that server. For the latter option, the AQT_HOST_PACKAGE_DIRECTORY environment variable must point to the z/FS directory.
- The value of the AQT_HOST_PACKAGE_DIRECTORY environment variable determines where IBM Db2 Analytics Accelerator Studio looks for software updates. If it is not set correctly, the packages cannot be located and thus cannot be transferred or installed.
- The user ID has read and write access to the directory that the AQT_HOST_PACKAGE_DIRECTORY environment variable points to and to all its subdirectories. You specify this variable in the AQTENV data set. The AQTENV data set must be referenced by the Workload Manager (WLM) environment that was set up for the IBM Db2 Analytics Accelerator for z/OS stored procedures. To avoid a setup with obsolete or wrong entries, use the sample AQTENV data set that comes with IBM Db2 Analytics Accelerator for z/OS Version 7.1.0.

About this task

You can transfer update packages from the z/OS UNIX file system (zFS) to the accelerator, or from a workstation on which IBM Db2 Analytics Accelerator Studio is installed, and which is connected to the z/OS data server. The first method requires that the packages have been placed in the proper z/FS directory. To get them there, you can, for example, download packages from IBM Fix Central directly to the zFS of the mainframe. To this end, you need a client application on the mainframe that is capable of handling FTPS (FTP over TLS) traffic. The download site at IBM Fix Central offers an option called *Download using bulk FTPS*. The second method requires that the update packages have been downloaded to the client workstation before. To actually use a transferred package, you must activate it. How to do this is described in a later topic. See the hint at the end of this description.

Installing a software update invokes the **SYSPROC.ACCEL_UPDATE_SOFTWARE2** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Procedure

1. Start IBM Db2 Analytics Accelerator Studio.
2. In the Administration Explorer, select the **Accelerators** folder.
3. In the Object List Editor on the right, double-click the accelerator.
4. If necessary, expand the **About** section.
5. In the **About** section, click the **Transfer updates** link.
6. In the Transfer Updates window, you can see all software packages that are available in the zFS of your z/OS data server. Select the appropriate check boxes in the first column of the table to mark the packages that you want to transfer.

Attention:

- Make sure that you select the proper packages, that is, packages belonging to the release level that you want to upgrade to. The list in the Transfer Updates window might be confusing, especially if it also contains older packages. To find the correct package numbers, see the closing information or the release notes for the latest program temporary fix (PTF). You find the closing information or the release notes for a PTF on the support home page, in the category *Plan and install documentation* (see link under **Related information** at the end of this topic).
 - If the list contains too many packages so that selecting the proper ones becomes awkward, select unwanted package names and click **Remove**. This will just remove the packages from the list or view, but won't delete the package files from the z/FS or your client workstation.
7. Choose the appropriate method to transfer the selected packages:

Transfer file from client

To transfer one or more downloaded packages from the connected client workstation.

Transfer file from IBM Z

To transfer one or more packages from the z/FS of your z/OS data server.

What to do next

To save space in the SMP/E target directory of the zFS, consider deleting sub-directories that contain already transferred packages. Be careful, however, if you have more than one accelerator. Do not delete packages if these are still needed for other accelerators.

If the AQT_HOST_PACKAGE_DIRECTORY environment variable does not point directly to the SMP/E target directories but to a copy, you might also want to delete the copy.

Related information:

IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference

Activating an updated accelerator software version

To use a recently transferred software package, you must first activate that package by selecting the version that the package contains.

Before you begin

- You cannot activate a software version as long as a IBM Db2 Analytics Accelerator Console window is open. Therefore, close all console windows. If you are not sure whether there are any open console windows, click **List Tasks** in the relevant Accelerator view of IBM Db2 Analytics Accelerator Studio and cancel all active console tasks from the List Tasks window.
- If incremental updates are enabled, stop these for each Db2 subsystem connected to the accelerator. To do this from IBM Db2 Analytics Accelerator Studio:
 1. In the Administration Explorer window, open the **Accelerators** folder.
 2. Select a relevant connection and open its Accelerator view.
 3. In the header section of the Accelerator view, look for **Replication:** and click **Stop**.
 4. Repeat the preceding steps for each Db2 subsystem connection to the same accelerator.

Note: If you have connections to Db2 data sharing groups (DSGs) and prefer to run the SYSPROC.ACCEL_SET_REPLICATION stored procedure from a client command line: It is not necessary to stop incremental updates for each individual member. A DSG is treated like a single connection and stopping incremental updates for the primary member will also stop these for the other members.

About this task

- Earlier product versions allow you to switch between older and newer versions of the IBM Db2 Analytics Accelerator software. This is not possible with IBM Db2 Analytics Accelerator for z/OS Version 7.1.0. Once you have activated a particular version, you cannot go back to an older one.
- The application of an update affects all Db2 subsystems that are connected to an accelerator.
- The activation of a different accelerator software version causes a restart of the accelerator.

Activating a software update invokes the **SYSPROC.ACCEL_UPDATE_SOFTWARE** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

Attention:

- The activation of a new version of IBM Db2 Analytics Accelerator for z/OS software components might delete the query history from the accelerator if the new version introduces changes to the query history function.
- The sequence in which you activate the different software components is important. The sequence depends on the source and on the target version. To find the proper sequence for your particular update, see “Order in which to update components” on page 135.
- When the accelerator is restarted with a new accelerator software version, accelerated queries cannot be processed on that accelerator before it reaches the *online* state. The only way to avoid a delay is to use a second accelerator in a workload balancing setup. That is, a second accelerator which is connected to the same Db2 subsystem, and which contains the same set of tables. Note that the tables must be loaded and enabled.

Procedure

1. In the Administration Explorer, select the **Accelerators** folder.
2. In the Object List Editor on the right, double-click the accelerator.
3. In the Accelerator view, from the **Refresh** drop-down list in the upper right, select **Automatic off**. Otherwise, you might see warnings during the activation of the new software saying that the accelerator cannot be contacted.
4. If necessary, expand the **About** section.
5. In the **About** section, click the **Apply other software version** link.
6. In the Apply Software Version wizard, you can see all software packages on the accelerator that are currently available for the selected component. To read information about a particular version before you activate it, select the appropriate entry in the list. The information is provided in the **Details of selected version** text box at the bottom.
7. Activate a version by selecting the appropriate radio button in the **Switch To** column.
8. Click **Finish**.

Results

In some cases, migration processes require some time to finish, especially after a restart of the database container. It is therefore quite normal that, after you click the **Finish** button, the accelerator is not instantly ready to process queries. You can check the status of an accelerator at the top of the corresponding Accelerator view. The status must be *online* for an accelerator to process queries.

What to do next

To avoid a cluttered Apply Software Version window, you can remove packages from the accelerator that you no longer need.

For instructions on how to complete this task, follow the appropriate link under **Related tasks**.

Related information:



IBM DB2 Analytics Accelerator for z/OS: Stored Procedures Reference

Removing obsolete software packages from an accelerator

To free up space on an accelerator and reduce the number of versions that are displayed in the Apply Software Version window, you can remove packages that you no longer need.

About this task

The removal wizard can remove all packages that have been deployed with the help of the transfer wizard, that is:

- Accelerator software packages
- Database software packages

Note: You can only remove software packages from regular accelerators.

Procedure

1. Open the appropriate Accelerator view.
2. If necessary, expand the **About** section.
3. Click the **Remove** link in the server subsection.
4. In the Remove Software Versions window, select the software packages that you want to remove. To do so, select the appropriate check boxes in the first column. As a decision aid, you might want to read the information under **Details of selected version**, which is displayed for each package.
5. Click **Remove**.

What to do next

The removal wizard that is described in this topic can only remove packages from an accelerator. To free up space in the z/OS UNIX file system (zFS) of your data server, manually delete no longer needed IBM Db2 Analytics Accelerator for z/OS packages from the zFS. To do so, open a UNIX System Services shell and delete the files by using the **rm** command.

Opening a service request

You might run into a situation in which you have to contact IBM support because you cannot solve an issue by yourself. A reason might be that specific updates or patches are required to successfully complete maintenance or repair jobs on the accelerator. Such jobs are usually carried out by an IBM service engineer. Follow the steps here to open a service request (formerly called problem management record or PMR) and provide IBM support with the necessary information.

1. In IBM Db2 Analytics Accelerator Studio, enable tracing for the accelerator that you want to update. A trace level of DEFAULT is sufficient.
For more information, see *Tracing* in the *IBM Db2 Analytics Accelerator Studio: User's Guide*.
2. Save the trace information to a file.
3. Open a service request at <https://www.ibm.com/support/servicerequest/Home.action>. Consider that IBM support needs some time to analyze your system, coordinate maintenance actions, and update components as required.
4. Attach the trace file to the service request.

Transferring maintenance updates

A task that is closely related to service requests is the transfer of maintenance updates. When you have submitted the service request, IBM support will tell you the proper download location and the names of

the packages that you have to download. After finishing the download, transfer the update packages as a preparation for the IBM service personnel, who will help you install the updates on your IBM Integrated Analytics System.

Before you begin

Update packages must exist in the download directory for packages of this type in the z/OS UNIX file system (zFS). The target system must be able to access this directory.

Important: The AQT_HOST_PACKAGE_DIRECTORY environment variable points to this download directory. It specifies an absolute path (starting from the root directory). The AQT_HOST_PACKAGE_DIRECTORY environment is set in the <HLQSP>.SAQTSAMP(AQTENV) data set, where <HLQSP> is the chosen high-level qualifier for stored-procedure libraries. The value of this environment variable determines where IBM Db2 Analytics Accelerator Studio looks for updates. If it is not set correctly, the packages cannot be located and thus cannot be transferred to the IBM Integrated Analytics System. To avoid a setup with obsolete or wrong entries, use the sample AQTENV data set that comes with IBM Db2 Analytics Accelerator for z/OS Version 7.1.0.

About this task

In general, maintenance update packages are not installed automatically. They are just transferred to a directory on the IBM Integrated Analytics System, from where you must install them manually with the help of IBM support.

Transferring an update invokes the **SYSPROC.ACCEL_UPDATE_SOFTWARE2** stored procedure on your data server. For information about the privileges that are required to run this procedure and further details, see the appropriate section in the *IBM Db2 Analytics Accelerator for z/OS: Stored Procedures Reference*. A link to this document is provided under **Related reference** at the end of this section.

To start an update transfer process, follow the steps in this section.

Procedure

1. Start IBM Db2 Analytics Accelerator Studio.
2. In the Administration Explorer, select the **Accelerators** folder.
3. In the Object List Editor on the right, double-click the accelerator.
4. If necessary, expand the **About** section.
5. In the **About** section, click the **Transfer updates** link.
6. In the Transfer Updates window, you can see all software packages that are available in the zFS of your z/OS data server. Select the appropriate check boxes in the first column of the table to mark the packages that you want to transfer.

Attention:

- Make sure that you select the proper packages, that is, packages belonging to the release level that you want to upgrade to. The list in the Transfer Updates window might be confusing, especially if it also contains older packages. To find the correct package numbers, see the closing information or the release notes for the latest program temporary fix (PTF). You find the closing information or the release notes for a PTF on the support home page, in the category *Plan and install documentation* (see link under **Related information** at the end of this topic).
 - If the list contains too many packages so that selecting the proper ones becomes awkward, select unwanted package names and click **Remove**. This will just remove the packages from the list or view, but won't delete the package files from the z/FS or your client workstation.
7. Choose the appropriate method to transfer the selected packages:

Transfer file from client

To transfer one or more downloaded packages from the connected client workstation.

Transfer file from IBM Z

To transfer one or more packages from the z/FS of your z/OS data server.

Installing maintenance updates

While updates for IBM Db2 Analytics Accelerator for z/OS and the database are installed automatically at the end of the transfer process or by using the **Apply** function, other updates have to be installed manually with the help of IBM support.

Before you begin

Make sure that the following conditions apply:

- You have opened a service request as described in “Opening a service request” on page 140.
- The IBM Integrated Analytics System must be fully operational. Hardware problems must not exist. Replace defective hard disks before installing the update.
- Remote access from z/OS OMVS to the accelerator via secure shell (ssh) must be enabled.
- IBM support will ask you to open or allow a screen-sharing session on your workstation. Make sure that this complies with the security guidelines of your organization or ask the responsible staff for a special permit.

About this task

An update installation requires you to stop the database container for approximately 30 minutes. The accelerator cannot be used during this time.

Procedure

1. Contact IBM support. IBM support will ask you for remote screen-sharing access to the client machine.
2. With IBM assistance, open a login shell to the accelerator. There are several ways to do that:
 - If an ssh server and an ssh client are installed in the z/FS of your IBM Z, and if server and client have been activated, you can connect with a network client application, such as PuTTY, from your client workstation to the accelerator by opening the following connections:
 - a. From your workstation to the IBM Z server by ssh. Use your user ID and password to log in. See an example of an ssh login from a PuTTY window:

```
login as: willy
willy@boedcal's password:
```

Important: If your z/OS version is V2R2 or higher, make sure that the address space for the Cryptographic Service Facility (CSF) has been started. Otherwise, the ssh connection to z/OS will fail.

- b. From the IBM Z to the accelerator by ssh. See the following example:

```
ssh -p 2222 root@10.104.9.3
```

where:

2222

is the listening port on the accelerator

10.104.9.3

is the IP address of the accelerator

- If an ssh server is installed and active on your IBM Z, and if port-forwarding is enabled for that server, you can connect to the accelerator by an ssh tunnel, which looks like a single connection

from your client workstation to the accelerator. To enable port-forwarding, set `AllowTcpForwarding` yes in the `/etc/ssh/sshd_config` file of your zFS and restart the ssh server for the change to take effect.

Example: The commands for a tunnel connection would look similar to these:

a. `ssh -L 10022:10.0.0.1:2222 user@systemz`

where:

-L 10022

is the local port on the client workstation

10.0.0.1

is the IP address of the accelerator

2222

is the remote port or listening port on the accelerator

user@systemz

user is ID with which you log on to the UNIX System Services ssh server and systemz is the host name of that server.

b. `ssh -p 10022 root@localhost`

where:

-p 10022

is the local port on the client workstation

root@localhost

root is the user ID on the accelerator and localhost is the host name of the client workstation

Tip: IBM used to provide a tool package called IBM Ported Tools for z/OS. See:

https://www-03.ibm.com/systems/z/os/zos/features/unix/port_tools.html

The package does not exist anymore, but more specific packages with similar functionality are now offered by the Rocket software company (<http://www.rocketsoftware.com/zos-open-source>). These packages usually include an OpenSSH component, which contains an ssh client that you can use to connect from the UNIX System Services file system to the accelerator. Note that you only need the client component of OpenSSH for maintenance updates (called `ssh`).

The resulting login screen looks similar to this:

```
Using keyboard-interactive authentication.
Secure token for service password
(this token must be sent to IBM together with the serial# the first time you request
a service password):

MHu2cK9FLC2pIkyAEb0oyDxK1yU7UQ01VPyXVviVxwGRtAC3gBiXiio5YLXzQTCQRfL/TRyuG8zFBC2CkSe3
GXPrFjM9KgB41gEFY0oxSEhjXY0S89fKGa7E/M+/VCdnsfHX2k2GpBq0XpyVPps6o0Q==

Enter Service Password (Date: '20170506' Serial#: '7130001' Rev: '4'):
```

3. As you enter the command to log on to the machine, the secure token, which contains the encrypted machine password, is displayed on the screen. In the previous example, this is the cryptic string that starts with `MHu2cK`. For the first service session, submit this token and the machine serial number to IBM support. IBM support will then generate a service password. You have to submit the secure token only for the first service session. The token is stored in an IBM service database for subsequent requests. This database cannot be accessed from the internet. If a further service session is required, you just have to give IBM support the serial number of the machine.
4. IBM support will enter the service password for you. The password will be valid exclusively for your accelerator machine and for the current date only.

5. Navigate to the directory on the accelerator in which the transferred update files are stored.
6. To proceed, follow the instructions of IBM support.
7. Having installed the updates, delete the files that are not needed anymore.
8. Terminate the ssh session by entering `exit` at the command prompt.

Chapter 7. Status information about the system and the incremental update function

When the system has been properly configured, a heartbeat connection is established between the accelerator and the connected Db2 subsystems. Every 30 seconds, the heartbeat connection provides the Db2 subsystems with status information about the accelerator. You can view most of this information by using the `-DIS ACCEL Db2` commands. Detailed information about specific events cannot be viewed in this way, but is written to the z/OS system log (SYSLOG), in the form of DSNX881I messages. This includes messages that are related to the incremental update function.

To establish a heartbeat connection, an accelerator must have been paired successfully with a Db2 subsystem. In addition, the accelerator must have been started for that particular subsystem by the `-START ACCEL` command or by use of the corresponding function in IBM Db2 Analytics Accelerator Studio.

To receive incremental-update-related event messages in a SYSLOG, IBM InfoSphere Data Replication for z/OS must have been installed and properly configured for use with the incremental update function of IBM Db2 Analytics Accelerator.

Support model

IBM Db2 Analytics Accelerator is a solution that consists of various hardware and software components. Each of these components might issue a DSNX881I message. If the message indicates a hardware or software problem, open a problem management record (PMR) with component ID (CompID) 5697AQT00.

Never open a service request or PMR for the IBM Integrated Analytics System. Always use the IBM Db2 Analytics Accelerator CompID, even if the problem is related to Db2 for z/OS or to the attached accelerator hardware.

Your service request or PMR will be routed internally to the proper support request queue. Make sure that the request contains an IBM Db2 Analytics Accelerator trace file that was obtained by using the **Save Trace** function in IBM Db2 Analytics Accelerator Studio. Such a trace file does not only contain software trace messages, but also a complete set of diagnostic hardware information.

The Save Trace window in IBM Db2 Analytics Accelerator Studio allows you to select an **Accelerator** and an **Appliance** option (apart from other options). It is important to notice the difference between the two:

Accelerator

Collects trace information from components running inside the Docker container

Appliance

Collects trace information from the physical system on which the Docker container is running

For details, follow the link to the description of the **Save Trace** function at the end of this topic.

Consider using automatic DSNX881I notifications. This function can inform administrators of critical events and create PMRs automatically.

DSNX881I message structure

Each DSNX881I message is made up of the following parts, which occur in the order indicated:

```
DSNX881I -<SSID> <MESSAGE-ID> <SEVERITY>  
<ACCELERATOR_MESSAGE_COUNTER> (<ACCELERATOR-TIMESTAMP>)  
ACCELERATOR-NAME(ACCELERATOR-IP) <MESSAGE-TEXT>
```

The placeholders have the following meaning:

SSID

Is the Db2 subsystem ID (SSID)

MESSAGE-ID

A numeric ID for the specific error message. This ID can be used for system monitoring.

Messages with IDs from 1 to 19 and from 21 to 2999

Accelerator heartbeat (system state) messages.

Messages with ID 20

These are incremental-update-related messages generated by IBM InfoSphere Data Replication for z/OS (CDC). They can be informational messages, warnings, or errors. Do not configure automatic notifications and service requests (PMRs) based on these messages, as the messages do not take the overall system state into account. In most cases, these messages are issued alongside messages in the range from ID3000 to ID3999. Refer to the latter type of messages because these contain proper instructions and do take the overall system state into account.

Message with IDs from 3000 to 3999

These are messages related to the incremental update function. In contrast to the ID20 messages, these messages are more specific or might ask for a concrete intervention (manual or automated) on your part. They can be used to trigger automatic notifications and PMRs.

SEVERITY

I Information message

W Warning message

E Error message

ACCELERATOR_MESSAGE_COUNTER

An internal counter that increases with every additional error on the accelerator. If the text after the DSNX881I qualifier is longer than 255 characters, another DSNX881I message is issued. All messages belonging together will have the same <ACCELERATOR_MESSAGE_COUNTER> value. The <MESSAGE-TEXT> block of the each subsequent message contains a sequel to the information in the previous message.

ACCELERATOR-TIMESTAMP

The time when the error occurred on the accelerator. The internal clock of the accelerator is synchronized with the first Db2 subsystem that was connected to the accelerator.

ACCELERATOR-NAME

The name of the accelerator on which the error occurred.

ACCELERATOR-IP

The IP address of the accelerator on which the error occurred.

MESSAGE

A textual description of the error.

The length of a DSNX881I message does not exceed 255 characters. If more characters are needed, additional DSNX881I messages are written to the SYSLOG. If an LPAR contains multiple Db2 subsystems that are connected to the same physical accelerator, error messages are issued for every subsystem. That is, you see the same messages multiple times in the log, each time with a different subsystem ID (SSID).

If an accelerator is paired with a data sharing group (DSG), all members of the group can write messages to group's system logs (SYSLOGs), provided that the -START ACCEL command has been issued for all members. In this case, make sure that applications are in place that monitor the SYSLOGs. If all members of the DSG are located in the same logical partition (LPAR), there is only one SYSLOG to monitor (see Figure 13 on page 148). However, if the members are located in different LPARs, you need to monitor the SYSLOGs of all LPARs involved (see Figure 14 on page 149).

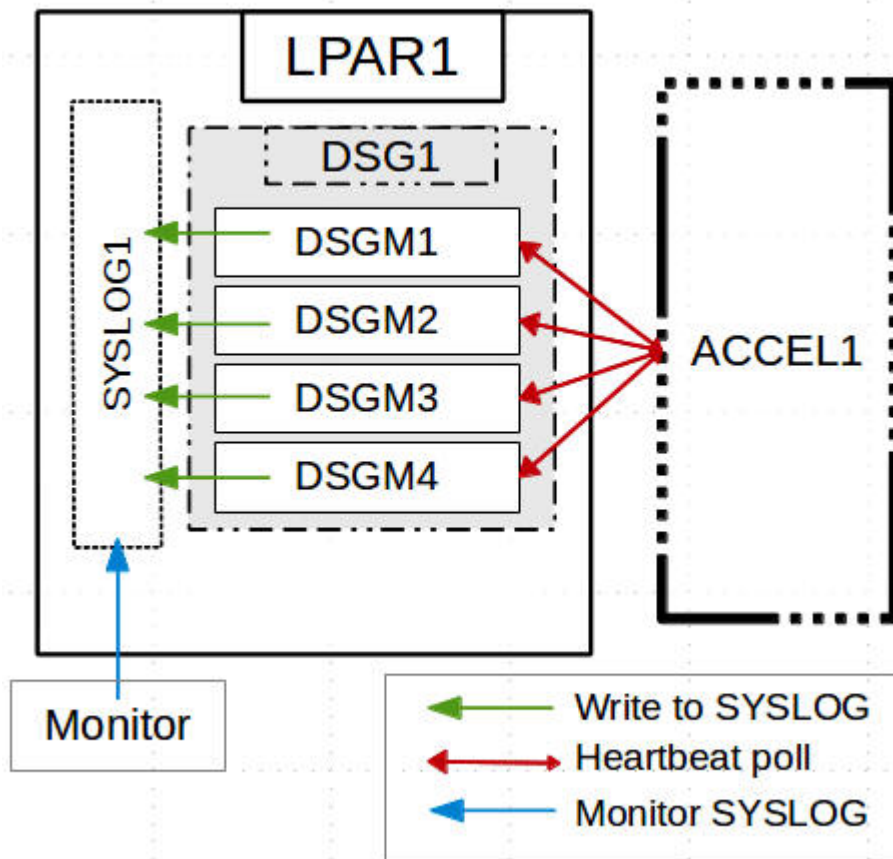


Figure 13. All members in the same LPAR

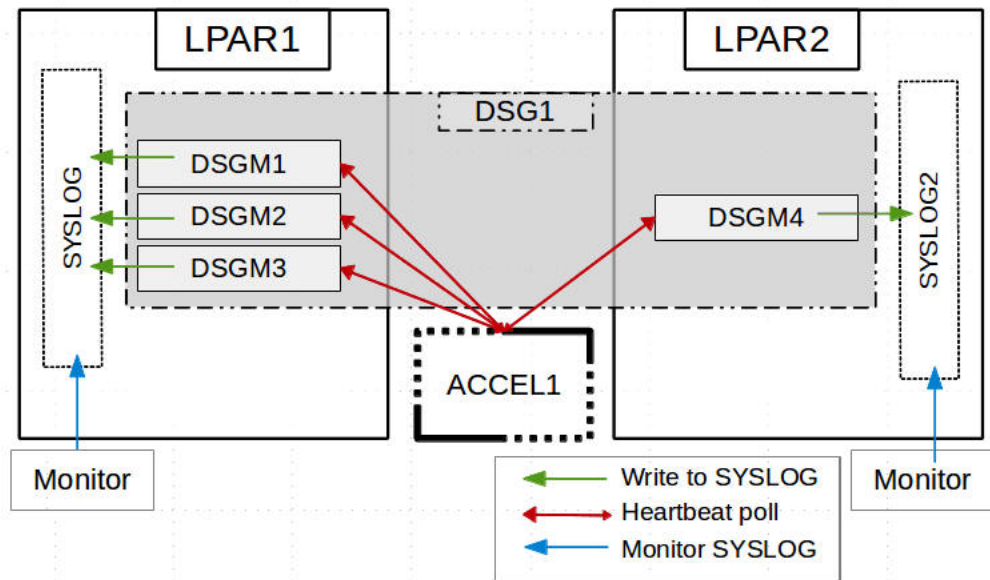


Figure 14. Members in two different LPARs

Note: It might look as if only one member writes messages to the SYSLOG, but this is actually a synchronization issue. If one member is always the first to issue a heartbeat request, then this member will receive all the messages and write these to the SYSLOGs. After that, the messages are deleted from the accelerator queue. The other members that send their heartbeat requests later will not receive these messages because the queue is empty. You might also see that only a few members write messages to the SYSLOG. This just means that the first member to send a heartbeat request is (always) found among this subset of members. The underlying mechanism is the same.

If an error occurs while an accelerator is in the *Stopped* state, that is, the -STOP ACCEL command has been issued before, an error message is stored on the accelerator. As soon as the accelerator becomes available again in Db2, the stored error messages are sent to the Db2 subsystem, provided that -START ACCEL has been issued for the subsystem, or, in case of a data sharing group, for at least one member of the group. It might happen that a DSNX881I message reports a past problem that has already been fixed.

For a list of message IDs, severity indicators, and message texts, see *Structure of DSNX881I Messages* (link under **Related information**).

Related tasks:

“Saving trace information” on page 114

Related information:

 [Prerequisites and Maintenance for IBM DB2 Analytics Accelerator for z/OS Version 6.1](#)

 [Structure of DSNX881I Messages](#)

Chapter 8. Troubleshooting

In the following sections, you find descriptions of known IBM Db2 Analytics Accelerator for z/OS problems. The author and the development team have tried to provide a solution wherever possible. However, a solution might be unavailable because the cause of a problem cannot be clearly identified. This is mostly the case if multiple causes can lead to the same symptom. It can also be that a solution has not yet been found. In such cases, contact IBM support.

Important: During a customer session with IBM support, system parameters might be changed. For the parameter changes to take effect, a restart of an accelerator is often required. In such situations, the accelerator will be unavailable for operational tasks until it has come back online.

Firewall blocks operations because crucial ports are closed

If product functions are not executed, check your firewall settings. IBM Db2 Analytics Accelerator requires certain ports to be open. See the table in this topic.

Symptoms

- You cannot accelerate queries.
- You cannot run functions from the IBM Db2 Analytics Accelerator Console.
- Incremental updates do not work or IBM InfoSphere Data Replication for z/OS cannot be configured.
- You cannot open a secure-shell (ssh) connection to an accelerator.
- Services are unavailable or switches (routers) cannot be reached because ICPM (Internet Control Message Protocol) traffic is blocked.

Causes

Required networking ports are closed.

Resolving the problem

Table 27. Required networking ports for IBM Db2 Analytics Accelerator

Function or protocol	Required port	Direction
Query execution (DRDA protocol)	1400	Both ways
Communication between accelerator and stored procedures	1401	Both ways
IBM Db2 Analytics Accelerator Console	1600	Both ways
Secure DRDA (stunnel)	11400	Both ways
Secure communication between accelerator and stored procedures (stunnel)	11401	Both ways
Incremental update communication	11301-11350	Outbound (from z/OS to accelerator)

Table 27. Required networking ports for IBM Db2 Analytics Accelerator (continued)

Function or protocol	Required port	Direction
CDC configuration	5999	Inbound (generally from accelerator to z/OS; depends on the configuration of the CDC Capture Agent)
DB2 DDF port (for continuous incremental updates)	446 (default)	Inbound (from accelerator to z/OS)
Secure shell connection (ssh)	2222	Both ways
ICMP traffic	N/A	Both ways

IBM Db2 Analytics Accelerator Studio does not start

If IBM Db2 Analytics Accelerator Studio does not start, the reason might be that the name of the starting user contains a number sign (#).

Symptoms

IBM Db2 Analytics Accelerator Studio does not start.

Causes

At installation time, IBM Db2 Analytics Accelerator Studio creates a workspace directory in

@user.home\IBM\Db2 Analytics Accelerator Studio 2.1\workspace

where @user.home is the user's home directory, which is usually the same as the user name.

This means, if the user name contains a number sign (#), the workspace directory path contains a number sign, too. IBM Db2 Analytics Accelerator Studio cannot handle workspace directory paths that include number signs.

Resolving the problem

1. If the user name contains a number sign (#), open the file `datastudio.ini` in a text editor. This file is located in the installation directory of IBM Db2 Analytics Accelerator Studio.
2. Remove the following entries:
`-data`
`@user.home\IBM\Db2 Analytics Accelerator Studio 2.1\workspace`
3. Save and close the file.
4. The next time you start IBM Db2 Analytics Accelerator Studio, you are asked for the workspace directory path. This allows you to enter an alternative path that does not contain a number sign.

IBM Db2 Analytics Accelerator Studio on Linux does not start

IBM Db2 Analytics Accelerator Studio does not start from certain versions of Linux, particularly SUSE Linux Enterprise Server 10 with service pack 4.

Symptoms

An error message similar to the following is displayed:

```
/opt/tools/isaostudio/./jre/bin/javaw: symbol lookup error
/usr/lib/xulrunner-1.9.2.17/libxul.so: undefined symbol
gdk_screen_get_resolution
```

Causes

An updated version of a required library (XULRunner) is not compatible with IBM Db2 Analytics Accelerator Studio.

Resolving the problem

Use an earlier version of the XULRunner library. The problem also occurs in connection with IBM Rational® Performance Tester. You can apply the suggested solution analogously. See:

<http://www.ibm.com/support/docview.wss?uid=swg21393436>

Package not found when running a stored procedure from IBM Db2 Analytics Accelerator Studio

You receive a message saying that a package was not found when you try to run an IBM Db2 Analytics Accelerator for z/OS stored procedure from IBM Db2 Analytics Accelerator Studio.

Symptoms

You receive a message similar to this one:

```
SQL0805N  Package "<location>.NULLID.SYSSTAT.5359534C564C3031"
was not found.  SQLSTATE=51002
```

Causes

The package has not been bound due to an IBM Db2 Analytics Accelerator Studio installation error.

Resolving the problem

Bind the package manually. The following methods can be used:

- From the Db2 command-line client:

1. `cd <Db2-client-install-folder>\bnd`
where <Db2-client-install-folder> is the fully qualified path to the installation folder of the Db2 command-line client.
2. `db2 connect to <database-name>`
where <database-name> is the database to which the stored procedure belongs.
3. `db2 bind @db2cli.lst grant public`

- Using the Db2Binder utility from a Windows command-prompt:

1. Adjust the following command as needed. Then press the Enter key.
`cd /d <idaa-studio-install-directory>`
`\plugins\com.ibm.datatools.db2_2.1.403.v20120228_2105\driver`

where <idaa-studio-install-directory> is the drive and installation directory of IBM Db2 Analytics Accelerator Studio on your local workstation, for example C:\Program Files\IBM\ IBM Db2 Analytics Accelerator Studio 2.1. The full name of the com.ibm.datatools.db2_ directory changes with each new driver. So make sure that you choose the correct directory.

2. Enter:

```
..\..\..\jre\bin\java -cp db2jcc4.jar;  
db2jcc4_license_cisuz.jar;  
db2jcc4_license_cu.jar com.ibm.db2.jcc.Db2Binder  
-url jdbc:db2://<server>:<port>/<location>  
-user <user-id> -password <password>
```

where

<server>

Is the host name of the Db2 data server

<port>

Is the port on which the Db2 data server listens to JDBC requests

<location>

Is the unique name of the database server. An application uses the location name to access a Db2 database server. A database alias can be used to override the location name when accessing a remote server.

<user-id>

Is a user ID with the privilege of running the Db2 Binder utility

<password>

Is the password belonging to <user-id>

Long-running tasks are not completed

Tasks started from IBM Db2 Analytics Accelerator Studio are not completed if they need more than two minutes of time, as, for example, the loading of an accelerator-shadow table.

Symptoms

The progress indicator never reaches the end or runs in an endless loop. You never receive a message that reports the successful completion or failure of the task.

Causes

A Db2 timeout setting (IDLE THREAD TIMEOUT parameter) prematurely ends the process (DDF thread) before the task is completed. The default value of this parameter is 2 minutes.

Resolving the problem

Increase the Db2 timeout value. For instructions, follow the link at the end of this topic.

Related information



IDLE THREAD TIMEOUT field (IDTHTOIN subsystem parameter)

SQL code -430 from IBM Db2 Analytics Accelerator for z/OS stored procedures

A stored procedure of IBM Db2 Analytics Accelerator for z/OS ends abnormally, and you receive an error message with SQL code -430.

Symptoms

IBM Db2 Analytics Accelerator for z/OS stored procedures end abnormally. In IBM Db2 Analytics Accelerator Studio, the Administration Explorer returns a message window like this one:

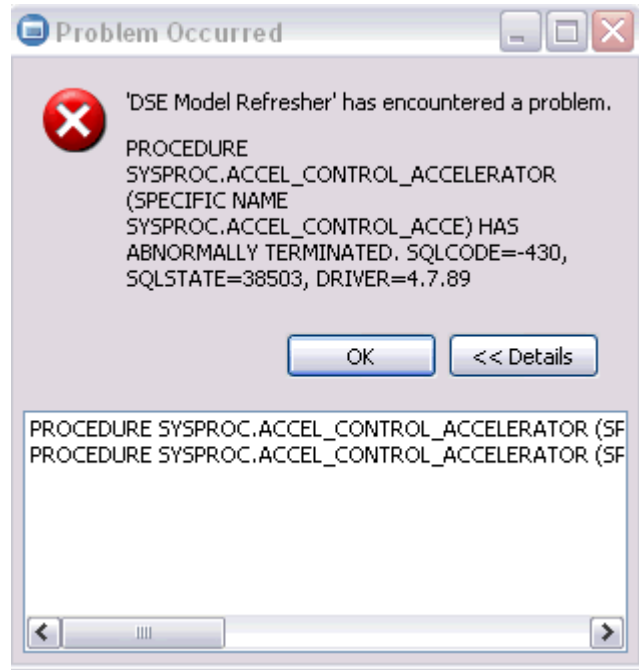


Figure 15. SQL code -430 message window

Causes

This might be a configuration problem.

Diagnosing the problem

1. Verify that the IBM Db2 Analytics Accelerator for z/OS stored procedures run in a separate Workload Manager (WLM) environment. Each application environment must be set up according to the instructions in *Setting up a WLM application environment for IBM Db2 Analytics Accelerator for z/OS stored procedures* in the *IBM Db2 Analytics Accelerator for z/OS: Installation Guide*. In particular, the ADMIN_INFO_SYSPARM and DSNUTILU stored procedures must run in different WLM environments and NUMTCB must be set to the correct value.
2. If the problem persists, collect the following diagnostic information for IBM support:
 - CEEDUMP in the WLM job
 - Stored procedure trace files in the z/FS directory /tmp. The names of these files follow this pattern:

aqt-trace-<YYYYMMDD-HHMM>-<procedure name>-*

Example:

aqt-trace-20171113-1028-ACCEL_TEST_CONNECTION-ghbHxQ

Transfer the trace files in binary mode.

Database access problems after updating IBM Db2 Analytics Accelerator Studio or IBM Optim Query Tuner

IBM Db2 Analytics Accelerator Studio or IBM Optim[™] Query Tuner no longer run successfully after an upgrade to a new version. A message says that privileges to access the database are missing.

Symptoms

A text similar to the following is displayed as part of the error message:
STEFANA DOES NOT HAVE THE PRIVILEGE TO PERFORM OPERATION EXECUTE PACKAGE
ON OBJECT DB 20SC.DSN50ADM. SQLCODE=-551, SQLSTATE=42501, DRIVER=3.58.81

Causes

A bind step has not been executed or users lack an execution privilege.

Diagnosing the problem

A bind step must be executed before a new version of IBM Db2 Analytics Accelerator Studio or IBM Optim Query Tuner can be employed. However, after carrying out the bind step, only the user who ran the command has the privilege to access the database from IBM Db2 Analytics Accelerator Studio or IBM Optim Query Tuner. If other users want to use IBM Db2 Analytics Accelerator Studio or IBM Optim Query Tuner, you must grant them this privilege.

Resolving the problem

1. To create and bind the Db2 packages, follow the instructions on this website:
https://www.ibm.com/support/knowledgecenter/SS62YD_3.1.1/com.ibm.datatools.qrytune.configothers.doc/topics/enabledb2zfromclient_ds.html
2. To grant the EXECUTE privilege to other users, proceed as follows:
https://www.ibm.com/support/knowledgecenter/SS7L9Q_4.1.0/com.ibm.datatools.qrytune.workloadtune.doc/topics/verify.html

DRDA connection does not work

You can ping the accelerators, but you cannot establish a distributed relational database access (DRDA) connection between your database management system and the accelerator.

Symptoms

Db2 commands or IBM Db2 Analytics Accelerator for z/OS stored procedures cannot establish a TCP/IP connection with the accelerator. Running the SYSPROC.ACCEL_TEST_CONNECTION stored procedure reveals this issue.

Resolving the problem

Make sure that the distributed data facility (DDF) of Db2 for z/OS uses the same TCP/IP stack as the ping program.

Different results for floating-point aggregates

Queries that involve floating-point aggregates produce different results when processed by IBM Db2 Analytics Accelerator for z/OS rather than Db2 for z/OS.

Symptoms

The very same query is processed twice, once by IBM Db2 Analytics Accelerator for z/OS and once by Db2 for z/OS. The results are different.

Causes

The result differences are differences in precision or error tolerances, which are caused by different methods of representation for floating-point numbers.

Resolving the problem

The problem cannot be solved. The differences must be accepted.

Related concepts

Stored procedure calls fail with message AQT10206I while temporary files are being created

Some or all running instances of the IBM Db2 Analytics Accelerator for z/OS stored procedures fail during the creation of temporary files.

Symptoms

Message AQT10206I is displayed for each failed stored procedure call:

AQT10206I - The OPEN NEW operation on the "/tmp/aqt-stored-procedure-trace-XXX" file, data set, or pipe failed. Diagnostic information: mkstemp() failed with rc = -1 and errno = 133

Causes

IBM UNIX System Services provides a z/OS UNIX file system (zFS) for z/OS. This file system, which contains the directory for the storage of temporary files (/tmp), is full. No more temporary files can be created for the execution of stored procedures.

Resolving the problem

Delete no longer needed files from the zFS. For example, you can safely delete stored-procedure trace-files. Note that you need superuser privileges in IBM UNIX System Services to do that. Follow these steps to delete the stored-procedure trace-files:

1. Log in to the IBM UNIX System Services shell of the logical partition (LPAR) from where you started the IBM Db2 Analytics Accelerator for z/OS stored procedure that returned the error. To log in, you can open a secure shell (ssh) window and use the **su** command to log in with superuser privileges. You can also use the IBM UNIX System Services shell to log in directly with an ID that has the required superuser privileges.
2. Change to the /tmp directory:
`cd /tmp`
3. Delete the stored-procedure trace files:
`rm aqt-stored-procedure-trace*`

Query history is not displayed - error message returned

The query history is not displayed in the Accelerator view and a corresponding error message is returned.

Symptoms

You do not see valid values in the columns of the **Query Monitoring** section of the Accelerator view and a message similar to the following is displayed:

```
[2013-05-25 05:14:09,790] [ERROR] [HistoryCol] Query History collection
impossible com.ibm.db2.jcc.am.SqlException: UNSUCCESSFUL EXECUTION CAUSED
BY AN UNAVAILABLE RESOURCE. REASON 00C900D1, TYPE OF RESOURCE 00000907,
AND RESOURCE NAME. SQLCODE=-904, SQLSTATE=57011, DRIVER=4.15.79
```

Causes

The value of the Db2 for z/OS parameter LOBVALA might be too low.

Resolving the problem

Increase the value of the LOBVALA parameter. The default value is 10240. Set it to 20480. Increase the value further if this does not solve the problem.

Db2 for z/OS returns SQLCODE -30040

A query cannot be executed and SQLCODE -30040 is returned by Db2 for z/OS.

Symptoms

Db2 for z/OS reports SQLCODE -30040, after IBM Db2 Analytics Accelerator has reported a `DRDARessourceLimitError` (DRDA code point 0x1233) and one of the following reason codes:

Reason 0x00001304 Type 0x00001409

The allowed maximum of concurrently running queries has been reached or exceeded. Your actual maximum depends on your hardware resources. The minimum is 100, the (theoretical) maximum is 240 queries running in parallel.

Reason 0x00000301 Type 0x00001409

The accelerator state forbids the execution of the query. For example, if the state is *Initializing* or *Maintenance*, queries cannot be processed.

Resolving the problem

1. Depending on the reason code, wait for earlier queries to finish or wait for the accelerator to change to a state that permits query execution.
2. Resubmit the failed query.

Appendix A. Impact of special register, ZPARM, or bind option settings on accelerator-only tables

SQL statements that reference accelerator-only tables can only be executed on an accelerator. However, certain combinations of special register and ZPARM settings might prohibit the execution of a statement on an accelerator. The same is true for the QUERYACCELERATION bind option for static SQL queries. The table in this section shows how these settings affect the statement execution.

Legend

The statement examples in Table 28 on page 160 use abbreviations that indicate the type of a table:

AOT Accelerator-only table

ACC Regular accelerator table

Db2 Db2 table (not on accelerator)

Remember that the ZPARM setting QUERY_ACCEL_OPTIONS=2 enables INSERT ... SELECT FROM ... statements. With accelerator-only tables, the effect is often different from the effect it has in connection with regular accelerator tables. In many cases, the setting leads to an error. When applied successfully, it mostly leads to an acceleration of the INSERT portion of an INSERT ... SELECT FROM ... statement.

Table 28. Statement execution depending on the settings of CURRENT QUERY ACCELERATION, QUERY_ACCELERATION, or QUERYACCELERATION and QUERY_ACCEL_OPTIONS=2

Statement	Special register SET CURRENT QUERY ACCELERATION= ZPARM QUERY_ACCELERATION= (ZPARM) Bind option QUERYACCELERATION=	QUERY_ACCEL_OPTIONS=2	Statement execution
SELECT * FROM T1_AOT	NONE (1)	No	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	Accelerator
SELECT * FROM T1_AOT, T5_ACC	NONE (1)	No	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	Accelerator
SELECT * FROM T1_AOT, T3_Db2	NONE (1)	No	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	Error, reason code 12
SELECT * FROM T5_ACC, T3_Db2	NONE (1)	No	Db2
	ALL (5)	No	Error, reason code 12
	ELIGIBLE (4)	No	Db2
	ENABLE (2)	No	Accelerator
INSERT INTO T1_AOT VALUES (...)	NONE (1)	No	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	Accelerator
INSERT INTO T1_AOT SELECT FROM T2_AOT	NONE (1)	No	Error, reason code 2
		Yes	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	INSERT portion accelerated
		Yes	INSERT portion accelerated

Table 28. Statement execution depending on the settings of CURRENT QUERY ACCELERATION, QUERY_ACCELERATION, or QUERYACCELERATION and QUERY_ACCEL_OPTIONS=2 (continued)

Statement	Special register SET CURRENT QUERY ACCELERATION= ZPARM QUERY_ACCELERATION= (ZPARM) Bind option QUERYACCELERATION=	QUERY_ACCEL_OPTIONS=2	Statement execution
INSERT INTO T1_A0T SELECT FROM T5_ACC	NONE (1)	No	Error, reason code 2
		Yes	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	INSERT portion accelerated
		Yes	INSERT portion accelerated
INSERT INTO T1_A0T SELECT FROM T3_Db2	NONE (1)	No	Error, reason code 2
		Yes	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	Error, reason code 12
		Yes	Error, reason code 12
INSERT INTO T1_A0T SELECT FROM T2_A0T, T3_Db2, T5_ACC	NONE	No	Error, reason code 2
		Yes	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	Error, reason code 2
		Yes	Error, reason code 12
<ul style="list-style-type: none"> INSERT INTO T5_ACC SELECT FROM T1_A0T INSERT INTO T3_Db2 SELECT FROM T1_A0T 	NONE (1)	No	Error, reason code 2
		Yes	Error, reason code 2
	ENABLE (2) ELIGIBLE (4) ALL (5)	No	SELECT portion accelerated
		Yes	SELECT portion accelerated

Table 28. Statement execution depending on the settings of CURRENT QUERY ACCELERATION, QUERY_ACCELERATION, or QUERYACCELERATION and QUERY_ACCEL_OPTIONS=2 (continued)

Statement	Special register SET CURRENT QUERY ACCELERATION= ZPARM QUERY_ACCELERATION= (ZPARM) Bind option QUERYACCELERATION=	QUERY_ACCEL_OPTIONS=2	Statement execution
<ul style="list-style-type: none"> • INSERT INTO T5_ACC SELECT FROM T5_ACC • INSERT INTO T3_Db2 SELECT FROM T5_ACC 	NONE (1)	No	Db2
		Yes	Db2
	ALL (5)	No	Error, reason code 17
		Yes	SELECT portion accelerated
	ELIGIBLE (4)	No	Db2
		Yes	SELECT portion accelerated
	ENABLE (2)	No	Db2
		Yes	Entirely in Db2 or SELECT portion accelerated (based on estimated response-time reduction)
		No	Error, reason code 17
		Yes	Error, reason code 12
		No	Db2
		Yes	Db2
<ul style="list-style-type: none"> • UPDATE T1_AOT (SELECT FROM T2_AOT) • UPDATE T1_AOT (SELECT FROM T5_ACC) • DELETE T1_AOT (SELECT FROM T2_AOT) • DELETE T1_AOT (SELECT FROM T5_ACC) 	NONE (1)	No	Error, reason code 2
	ENABLE (2)	No	Accelerator
	ELIGIBLE (4)		
	ALL (5)		
<ul style="list-style-type: none"> • UPDATE T1_AOT (SELECT FROM T3_Db2) • DELETE T1_AOT (SELECT FROM T3_Db2) 	NONE (1)	No	Error, reason code 2
	ENABLE (2)	No	Error, reason code 12
	ELIGIBLE (4)	No	Error, reason code 12

Table 28. Statement execution depending on the settings of CURRENT QUERY ACCELERATION, QUERY_ACCELERATION, or QUERYACCELERATION and QUERY_ACCEL_OPTIONS=2 (continued)

Statement	Special register SET CURRENT QUERY ACCELERATION= ZPARM QUERY_ACCELERATION= (ZPARM) Bind option QUERYACCELERATION=	QUERY_ACCEL_OPTIONS=2	Statement execution
<ul style="list-style-type: none"> • UPDATE T5_ACC (SELECT FROM T1_A0T) • UPDATE T3_Db2 (SELECT FROM T1_A0T) • DELETE T5_ACC (SELECT FROM T1_A0T) • DELETE T3_Db2 (SELECT FROM T1_A0T) 	NONE (1) ENABLE (2) ELIGIBLE (4) ALL (5)	No	Error, reason code 22
		No	Error, reason code 22

Appendix B. Using telnet to log on to the IBM Db2 Analytics Accelerator Console

A telnet logon to the IBM Db2 Analytics Accelerator Console is still possible for compatibility reasons, but not recommended. Consider using the more secure ssh method instead.

Procedure

1. Start a client or emulator session (using, for example, IBM Personal Communications) to communicate with the z/OS system on which your Db2 subsystem is located.
2. Log on to TSO/ISPF.
3. Enter the following command:
tso telnet <hostname> 1600

where

<hostname>

Is the IP address of the accelerator that is connected to the Db2 for z/OS data server.

1600

Is the number of the port configured for accessing the IBM Db2 Analytics Accelerator Console using a telnet connection between the Db2 for z/OS data server and the accelerator.

For example:

tso telnet 10.101.8.8 1600

4. When prompted, enter acceladm as the user ID, followed by the console password. The initial password is dwa-1234. You must change this password at the first logon.

The console also offers an option to create additional console users with different passwords.

5. Press the Pause key, then Enter to display the following screen:

```
*****
*           Welcome to the IBM Db2 Analytics Accelerator Console
*****

You have the following options:
(0) - (Menu) Manage Configuration Console Users
(1) - (Menu) Run Accelerator Functions
(2) - (Menu) Manage Incremental Updates
(3) - (Menu) Manage Encryption of Data in Motion

-----
(x) - Exit the Configuration Console
```

What to do next

Type x and press Enter to exit the console and close the telnet session.

Appendix C. Using ssh to log on to the IBM Db2 Analytics Accelerator Console

A secure shell (ssh) connection ensures that the information exchanged between your IBM Z[®] system and the accelerator console is encrypted.

Before you begin

- An ssh client must be installed on the mainframe computer (IBM Z). Starting with z/OS V2R2, the OpenSSH package, which contains an ssh client, is part of the operating system.

If you still use z/OS V2R1, you can obtain an ssh client by installing the respective component from BM Ported Tools for z/OS. See:

https://www-03.ibm.com/systems/z/os/zos/features/unix/port_tools.html

Mind that IBM does not offer this package anymore. However, the functionality lives on in more specific packages provided by the Rocket software company (<http://www.rocketsoftware.com/zos-open-source>). These packages usually include the OpenSSH component that contains the ssh client, so that you can connect from the z/FS file system to the accelerator. Note that you only need the client component of OpenSSH for maintenance updates (called ssh).

- An ssh client must also be installed on the workstation used to access the IBM Db2 Analytics Accelerator for z/OS Console. You can use a tool like PuTTY, which comes free of charge.

Procedure

1. On your client workstation, open an ssh shell and connect to the mainframe computer by using the following command:

```
ssh <mainframe-ip>
```

where <mainframe-ip> is the IP address or host name of the IBM Z system that is connected to the accelerator.

2. Enter the logon user ID and password for the IBM Z system.
3. When connected, enter the following commands from the shell to connect to the IBM Db2 Analytics Accelerator Console:

```
ssh -p 2222 configuration-console@<accelerator-ip>
```

where

-p 2222

Specifies a non-default connection port.

<accelerator-ip>

Is the IP address of the accelerator.

For IBM Db2 Analytics Accelerator on an IBM Integrated Analytics System, this is the virtual IP or wall IP address.

For Db2 Analytics Accelerator on Z, this is the IP address of the network that you labeled DB2 in the Appliance Installer.

4. When prompted, enter `accladm` as the user ID, followed by the console password. The initial password is `dwa-1234`. You must change this password at the first logon.

The console also offers an option to create additional console users with different passwords.

5. Press the Pause key, then Enter to display the following screen:

```
*****
*           Welcome to the IBM Db2 Analytics Accelerator Console
*****

You have the following options:
(0) - (Menu) Manage Configuration Console Users
(1) - (Menu) Run Accelerator Functions
(2) - (Menu) Manage Incremental Updates
(3) - (Menu) Manage Encryption of Data in Motion

-----
(x) - Exit the Configuration Console
```

What to do next

Type x and press Enter to exit the console and close the **ssh** session.

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
USA*

For license inquiries regarding double-byte character set (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. 19-21,
Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

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 jurisdictions 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.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those

websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide 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 Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
USA*

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.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

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

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to

IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks 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 "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.



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

Microsoft and Windows 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.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display

or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

GDPR considerations

For PID(s): 5697-DA7, 5697-DA5, 5697-DAB

Notice: This document is intended to help you in your preparations for GDPR readiness. It provides information about features of IBM Db2 Analytics Accelerator for z/OS that you can configure, and aspects of the product's use, that you should consider to help your organization with GDPR readiness. This information is not an exhaustive list, due to the many ways that clients can choose and configure features, and the large variety of ways that the product can be used in itself and with third-party applications and systems.

Clients are responsible for ensuring their own compliance with various laws and regulations, including the European Union General Data Protection Regulation. Clients are solely responsible for obtaining advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulations that may affect the clients' business and any actions the clients may need to take to comply with such laws and regulations.

The products, services, and other capabilities described herein are not suitable for all client situations and may have restricted availability. IBM does not provide legal, accounting, or auditing advice or represent or warrant that its services or products will ensure that clients are in compliance with any law or regulation.

1. GDPR Overview
2. "Product Configuration for GDPR" on page 174
3. "Data Life Cycle" on page 175
4. "Data Collection" on page 176
5. "Data Storage" on page 176
6. "Data Access" on page 177
7. "Data Processing" on page 177
8. "Data Deletion" on page 178
9. "Data Monitoring" on page 178
10. "Responding to Data Subject Rights" on page 178

GDPR

General Data Protection Regulation (GDPR) has been adopted by the European Union ("EU") and applies from May 25, 2018.

Why is GDPR important?

GDPR establishes a stronger data protection regulatory framework for processing of personal data of individuals. GDPR brings:

- New and enhanced rights for individuals
- Widened definition of personal data
- New obligations for processors
- Potential for significant financial penalties for non-compliance
- Compulsory data breach notification

Read more about GDPR

- EU GDPR Information Portal (<https://www.eugdpr.org/>)
- [ibm.com®/GDPR website \(http://ibm.com/GDPR\)](http://ibm.com/GDPR)

The EU General Data Protection Regulation (GDPR) regulates data privacy in the European Union (EU). For details, see: <https://www.eugdpr.org>

The regulations laid out in the GDPR apply if the data controller (an organization that collects data from EU residents), or the processor (an organization that processes data on behalf of a data controller, such as a cloud service provider), or the data subject (an individual) is based in the EU (source: Wikipedia. See: https://en.wikipedia.org/wiki/General_Data_Protection_Regulation). A violation of GDPR regulations might incur severe financial penalties (fines) and reputation damages. This documentation is intended for data controllers or processors who use Db2 for z/OS together with IBM Db2 Analytics Accelerator for z/OS with the aim of storing personal data of EU residents (individuals). As regards IBM Db2 Analytics Accelerator for z/OS, the IBM company is neither the data controller, nor the data processor. Nevertheless, the IBM company feels obliged to give such entities guidance on how to follow GDPR regulations.

Your responsibilities as a data controller or processor

If you act on behalf of a data controller or data processor, see the frequently asked questions (FAQ) on the official GDPR website to find out what your responsibilities are:

<https://www.eugdpr.org/gdpr-faqs.html>

Furthermore, refer to chapter 4 in the GDPR regulations:

<https://www.eugdpr.org/article-summaries.html>

Product Configuration for GDPR

The following sections provide considerations for configuring IBM Db2 Analytics Accelerator for z/OS to help your organization with GDPR readiness.

How to configure the offering so that it can be used in a GDPR environment

The following sections describe how personal data is stored and processed by IBM Db2 Analytics Accelerator for z/OS (called 'accelerator' in the text that follows).

If personal data is to be stored on an accelerator, the following configuration steps might help you meet your compliance objectives with regard to GDPR:

- Activate the encryption of data in motion (data that is transferred over a data network between Db2 for z/OS and the accelerator).
- Activate the encryption of data at rest (encryption of disks and other storage devices).
- Do not share passwords for the operation of the IBM Db2 Analytics Accelerator Console between multiple persons. Instead, create separate user IDs with different passwords for each console user.
- Give read access to personal data (the privilege to run queries) only to authorized Db2 for z/OS users. To this end, define appropriate Db2 authorizations (SELECT, and so on) in the Db2 for z/OS catalog.

- Restrict the right to unload data from and to create image copies in Db2 for z/OS. Give this right to authorized administrators only.
- Restrict access to the 'Save Trace' function in Db2 Analytics Accelerator for z/OS because the collected trace data might contain personal data. Access should be given to authorized administrators only.
- Restrict access to the table SYSIBM.USERNAMES in the catalog of the Db2 for z/OS communication database because this table contains authentication credentials. If these credentials are read or copied by unauthorized users, these users might obtain access to sensitive personal data.

Data Life Cycle

You can create copies of any Db2 for z/OS table on an accelerator (accelerator-shadow tables). As soon as you create a copy of a Db2 for z/OS table and load it with personal data, the GDPR regulations apply in the same way as they apply to personal data in the original Db2 for z/OS tables.

An accelerator also allows you to load and store external data outside of Db2 for z/OS, such as the data of VSAM files, in accelerator-shadow tables. Products like the Db2 Analytics Accelerator Loader can be used to this end. If the external data contains personal information, the GDPR regulations apply to the data that is transferred to the accelerator.

What types of data?

The administrative environment of an accelerator (client and configuration software) does not store personal data, except for the user IDs and authentication tokens of its administrators. However, the GDPR regulations apply to all personal data that you store in database tables on the accelerator.

Where in the process?

An accelerator is loaded with data in the course of individual load operations or by the incremental update function (automated load of table updates). Adding a table to an accelerator does not yet copy any data. It merely defines the table's structure (metadata) on the accelerator. However, if you load the table or enable it for incremental updates, the data of the source table - which might contain personal data - is copied to the accelerator.

For what purpose?

An accelerator, like Db2 for z/OS, is a general database management system for SQL queries. Such queries are run with the aim of retrieving stored information and gaining further insights (analytical queries that uncover the dependencies or connections between sets of values). An accelerator can return the results of SQL queries to authorized and authenticated users of the system. The results usually consist of subsets or aggregations of numerous data records. All these records might include personal data.

Personal data used for online contact with IBM

IBM Db2 Analytics Accelerator for z/OS clients can submit online comments/feedback/requests to contact IBM about accelerator subjects in a variety of ways, primarily:

- Public comments area on pages in the IBM Integration community on IBM developerWorks®

- Public comments area on pages of the IBM Db2 Analytics Accelerator for z/OS product documentation in the IBM Knowledge Center
- Feedback forms in the IBM Integration community

Typically, only the client name and email address are used, to enable personal replies for the subject of the contact, and the use of personal data conforms to the *IBM Online Privacy Statement* (<http://www.ibm.com/privacy/cc/>).

Data Collection

IBM Db2 Analytics Accelerator for z/OS can be used to collect personal data. When assessing your use of IBM Db2 Analytics Accelerator for z/OS and the requirements of GDPR, you should consider the types of personal data which in your circumstances are stored on an accelerator.

The use of an accelerator does not impose new requirements for the collection of personal data. It does not introduce new users or new privileges, but makes use of existing users and their privileges as defined in Db2 for z/OS. The use of an accelerator does not introduce additional obligations for the safekeeping or administration of data if you already observe the GDPR regulations for the storage of data in Db2 for z/OS.

Make sure that the process of collecting data and storing it in Db2 for z/OS is GDPR-compliant. The use of an accelerator will then be equally compliant.

Data Storage

The use of IBM Db2 Analytics Accelerator for z/OS involves the storage of data in a persistent manner.

An accelerator is a general-purpose data processing and retrieval engine. It is not aware of the meaning of data or any meaning that can be inferred from it. It is thus the responsibility of the data controller or processor to classify the data as personal data - for example account data - and make sure that the collection, access and processing of this data is in accordance with the GDPR regulations.

Storage in backups:

An accelerator has no backup function. However, image copies are created in Db2 for z/OS when the accelerator's archiving function is used (high performance storage saver). If these image copies contain personal data, the GDPR rules apply. This means, for example, that if individuals request their data to be deleted, it must also be deleted from all the image copies that may exist.

Storage in archives:

The high performance storage saver deletes table data from Db2 for z/OS and stores it exclusively on an accelerator. If this data contains personal information, the GDPR regulations might require that you delete the personal information from a number of records. The current version of the accelerator software does not allow DELETE operations on archived partitions. So in case you must delete records, first restore the archived partition to Db2 for z/OS, then delete then delete the personal data from Db2 for z/OS. Finally, archive the partition again (which now does not include the personal data anymore).

Data Access

Access to data on an accelerator requires a Db2 for z/OS user ID and a password. The accelerator recognizes the privileges defined in Db2 for z/OS and does not allow a user to run functions that she or he is not entitled to use.

As long as the Db2 for z/OS authorizations, such as SELECT, INSERT, UPDATE, or DELETE privileges, are defined in accordance with the GDPR regulations, access to data on the accelerator will likewise be in accordance with these regulations.

So it is crucial that the definition of roles and access rights in Db2 for z/OS effectuates an adequate separation of duties and responsibilities.

One accelerator-specific function might provide access to personal data on the accelerator: This is the "Save Trace" function, which collects diagnostic information for troubleshooting. The resulting trace files might contain personal data because they sometimes include the code of SQL queries to which personal information has been passed in the form of query literals. Trace files can also include memory dumps that contain personal data.

It is therefore important that access privileges to data, such as SELECT privileges or superuser privileges like SYSADM and DBADM, are granted with great care. The permission to use the "Save Trace" function should be given to selected administrators only.

The privilege to create image copies in Db2 for z/OS must also be handled with care. It is required, for example, if a user wants to archive table partitions on an accelerator.

Data Processing

Privileges defined in the Db2 for z/OS catalog control data processing on an accelerator.

In addition to that, the encryption features (encryption of data in motion and encryption of data at rest) should be activated. This prevents an unauthorized person without proper privileges from gaining direct access to personal data by interfering with the network communication or by obtaining physical access to the storage devices (hard disks or solid state disks).

Encryption features and safeguarding precautions:

Encryption of data in motion:

This feature encrypts the network traffic between an IBM Z system and an accelerator.

Encryption of data at rest:

This feature encrypts the data on an accelerator's physical storage devices (hard disks and solid state disks). Depending on the model of the hardware, the feature might already be turned on by default.

Encryption key ownership:

Restrict access to the encryption keys because the keys enable a person to read or decrypt personal data on a network or storage device. Put the safekeeping and maintenance (rotation) of the encryption keys in the hands of a trusted administrator.

Data Deletion

When data is deleted from Db2 for z/OS, it might still exist in table copies on the accelerator. To delete the data on an accelerator, you have the following options:

The data in accelerator-shadow tables is deleted:

- If the data in the original Db2 for z/OS tables has been deleted, and if the affected accelerator-shadow tables have been reloaded (with the contents of the now empty Db2 tables).
- If the accelerator-shadow tables are removed from the accelerator.
- If the incremental update function is used and the function replicates empty table content to the accelerator because the original Db2 for z/OS table data has been deleted.

The data in an accelerator-only table (AOT) is deleted:

- If the AOT is removed from the accelerator.
- If an explicit DELETE statement is submitted on the Db2 for z/OS side, which references the AOT and contains a predicate that specifies the row to be deleted.

The data in partitions archived by the high-performance storage saver (HPSS) is deleted:

- If the table partitions are removed from the accelerator.
- If an archived partition is restored to Db2 for z/OS.
- If the data is deleted from Db2 for z/OS because this means that the archived partition is reloaded or that the partition is archived again after the deletion from Db2 for z/OS (empty table content will be reloaded or archived).

Data Monitoring

Db2 for z/OS provides powerful auditing and monitoring functions that allow you to track the processing of personal data. Because IBM Db2 Analytics Accelerator for z/OS is a logical component of Db2 for z/OS, and keeps only copies of Db2 for z/OS data, the monitoring functions for data processing in Db2 for z/OS are sufficient.

Responding to Data Subject Rights

To identify an individual's data that is stored on an accelerator, you can submit SQL queries from Db2 for z/OS. These queries ought to search for the name, the account number, or for other identifying information about the subject using SQL syntax. Since an accelerator works on copies of Db2 for z/OS data, the same queries can be run to retrieve the data either from Db2 for z/OS or from the accelerator. Specific advice or examples of such queries can not be provided here because the structure and the content of the database tables, as defined by the data controller or data processor, is not known.

Does the offering facilitate being able to meet data subject rights?

Right to Access

Can the client provide individuals access to their data?

Db2 for z/OS and accelerators can process SQL queries in which an individual's ID can be specified as a search criterion. This way, all data

stored about that individual can be retrieved. In Db2 for z/OS, a client's data access can be restricted, so that the client can only search and retrieve a subset of the available data records, that is, those records that belong to a specific group of individuals. To this end, a feature called label based access control (LBAC) can be employed. IBM Db2 Analytics Accelerator provides a similar functionality, which restricts access to a selected set of rows based on the CURRENT SQLID special register.

Can the client provide individuals information about what data the client has about the individual?

Yes. To identify an individual's data stored on an accelerator, the client can submit SQL queries from Db2 for z/OS. These queries ought to search for the name, the account number, or for other identifying information about the subject using SQL syntax.

Right to Modify

Can the client allow an individual to modify or correct their data?

Db2 for z/OS allows a client to update the data that is stored about an individual. Depending on the architecture of the application, it might even be possible to authorize the individuals, so that these can update their data by themselves. After an update in Db2 for z/OS, the modified data is copied to the accelerators attached to the database during the next reload operation, which can be started manually or automatically (by the incremental update function). During that process, incorrect data will be overwritten.

Can the client correct an individual's data for them?

Yes (see the previous question *Can the client allow an individual to modify or correct their data?*).

Right to Restrict Processing

Can the client stop processing an individual's data?

Yes. In Db2 for z/OS, a client can submit DELETE statements to remove data records of individuals. The removal will be reflected in the data that is stored on an attached accelerator when the next reload operation (manual or automatic) is taking place. During that process, records previously deleted from Db2 for z/OS will also be deleted from the accelerator. Furthermore, an individual's data can be excluded from SQL queries through the use of filters in query predicates. For example, a query that reads `SELECT CUSTOMER_DATA FROM CUSTOMER_TABLE where custid NOT IN (a,b,...c)` returns only data records that are not related to (do not belong to) the users in the specified group (a,b,...c).

Right to Object

Same as "Right to Restrict Processing."

Right to Be Forgotten

Can the client delete an individual's data?

Yes.

Right to Data Portability

Can the customer provide an individual with the information that they have about the individual in a user-friendly/machine readable format?

Yes.

Glossary

This glossary includes terms and definitions for IBM Db2 Analytics Accelerator for z/OS.

The following cross-references are used in this glossary:

- *See* refers you from a term to a preferred synonym, or from an acronym or abbreviation to the defined full form.
- *See also* refers you to a related or contrasting term.

A

accelerator-shadow table

A table on an accelerator whose definition is derived from a Db2 for z/OS table and that can be used for accelerated queries after loading (copying data from the original Db2 table to the accelerator-shadow table).

access plan graph

A visual representation of a query that shows the database objects that are accessed by the query and the order in which this is done.

APF See authorized program facility.

authorized program facility (APF)

In a z/OS environment, a facility that permits the identification of programs that are authorized to use restricted functions.

C

cluster

- Number of configured accelerators.
- Table rows that are grouped together because they have the same value in a particular column.

colocated join

A join method where columns of an accelerator-shadow table must be on the same processing node. Leads to shorter query response times.

coordinator node

The primary node of the accelerator. It distributes the workload among the other nodes.

D

data slice

A portion of an accelerator-shadow table that is processed by a worker node.

DDF See distributed data facility (DDF).

distributed data facility (DDF)

A set of Db2 for z/OS components by which Db2 for z/OS communicates with another relational database management system.

Dynamic Statement Cache

A pool in which Db2 saves prepared SQL statements that can be shared

among different threads, plans, and packages. By sharing these statements, applications can avoid unnecessary preparation processes and thus improve performance.

F

fact table

A database table, commonly located at the center of a star schema surrounded by dimension tables, that contains measures of a business process, and foreign keys from the dimension tables.

failback

An automatic operation that returns the processing of a query to Db2 for z/OS if an error occurs while the query is processed by an accelerator.

friendly arithmetic

When an exception, such as a buffer overflow, a division by zero, or an out-of-range value occurs, Db2 returns the value NULL combined with a +802 warning, provided that the faulty expression is in the outermost SELECT statement and affects only one row.

I

inhouse processing

A query is processed by Db2 for z/OS and not routed to an accelerator.

L

lock A means of preventing uncommitted changes made by one application process from being perceived by another application process and for preventing one application process from updating data that is being accessed by another process. A lock ensures the integrity of data by preventing concurrent users from accessing inconsistent data.

P

port-forwarding

A networking mechanism that allows Secure Shell access to a host in a private network from the outside.

Q

query conversion

Db2 for z/OS and the query engine of IBM Db2 Analytics Accelerator handle certain SQL expressions differently. In some cases, IBM Db2 Analytics Accelerator does not support an SQL expression that is supported by Db2 for z/OS. Therefore, changes to the mode of processing are sometimes inevitable when a query is routed to an accelerator.

R

row See table row.

rowset

A set of rows for which a cursor position is established.

S

Secure Shell (SSH)

A UNIX-based command interface and protocol for securely getting access to a remote computer.

S-FTP See SSH File Transfer Protocol.

skew An imbalanced distribution of table rows among the processing nodes of the accelerator.

Sockets Secure (SOCKS)

A client/server architecture that transports TCP/IP traffic through a secure gateway. A SOCKS server performs many of the same services that a proxy server does.

SOCKS

See Sockets Secure.

SSH File Transfer Protocol

A network protocol that provides the ability to transfer files securely over any reliable data stream.

SSH tunnel

A secure and encrypted path through a network.

SSH See Secure Shell.

systems programmer

A programmer who plans, maintains, and controls the use of an operating system with the aim of improving the overall productivity of an installation.

T

table row (row)

The horizontal component of a table, consisting of a sequence of values, one for each column of the table.

table space

A logical unit of storage in a database. In Db2 for z/OS, a table space is a page set and can contain one or more tables.

temporary table

A table that holds temporary data. Temporary tables are useful for holding or sorting intermediate results from queries that contain many rows. The two kinds of temporary tables, which are created by different SQL statements, are the created temporary table and the declared temporary table.

throughput

A measure of the amount of information transmitted over a network in a given period of time. Throughput is generally measured in bits per second (bps), kilobits per second (Kbps), or megabits per second (Mbps).

V

virtual IP address

An IP address that is shared among multiple domain names or multiple servers. Virtual IP addressing enables one IP address to be used either when insufficient IP addresses are available or as a means to balance traffic to multiple servers.

W

WLM application environment

A z/OS Workload Manager attribute that is associated with one or more procedures. The WLM application environment determines the address space in which a given procedure runs.

Index

Numerics

24:00:00 41, 44

A

About section, Accelerator view 64
accelerating static SQL queries
 bind option 5
 bind options 18
acceleration conditions 17
accelerator
 adding to configuration 75
 canceling 109
 canceling tasks 109
 defining 75
 maintenance 140
 repair 140
 status 81
 task 109
Accelerator view
 About section 64
 header 64
 Monitoring section 64
 Query Monitoring section 64, 103
 Query monitoring section, columns 99
 refresh interval 116
 Tables section 64
accelerator-only tables 11, 64, 79, 90, 99, 159
 bind option for static SQL 18
 dynamic SQL queries against 17
 restrictions 47
 special register settings 17
 static SQL queries against 18
accelerator-shadow tables 2
 removing 3
accelerators
 adding 1
 basic concepts 5
 networking details 78
 removing 112
 updating software 135
access plan graph
 example 6
access, Db2 subsystem 71
accessibility features for this product xiii
activation of software updates 138
Add Tables wizard 79
adding
 accelerators 1
administration client 57
Administration Explorer
 folders 59
 menus 59
aggregate SQL functions 26
aliases 50
ALTER
 TABLE, required actions 133
 TABLESPACE, required actions 133
ALTER TABLE operation, incremental update function 125
analyze queries 94

array data type 50
ASCII 22
authentication 75
automatic notifications and creation of PMRs 145

B

basic concepts
 accelerators 5
 access plan graph 6
 distribution keys 8
 selecting tables for query acceleration 6
 system table entries 17
 tables 5
 unsupported data types 5
binary strings 43
bind options for accelerating static SQL queries
 QUERYACCELERATION 18, 159
broadcast 2
buffer overflow 41
built-in functions
 CUME_DIST 35
 FIRST_VALUE 35
 LAG 35
 LAST_VALUE 35
 LEAD 35
 NTH_VALUE 35
 NTILE 35
 PERCENT_RANK 35
 RATIO_TO_REPORT 35

C

call-home function 145
canceling tasks 109
CAST specification 26
CCSID
 1200 22
 1208 22
 367 22
 37 22
 819 22
 ASCII 22
 EBCDIC 22
 UNICODE 22
CDC
 event messages 145
changing
 distribution key 90
CHAR (n) 41
coded character set identifier
 See CCSID
CODEUNITS16 26
CODEUNITS32 26
conditions
 query routing to an accelerator 21
conditions for query acceleration 17
configuration, trace behavior 113
configure tracing 64, 113

- connection
 - data server 1
 - database 1
 - database server 73
 - profile 71
- connection profile 1
- create objects in connected databases 93
- creating PMRs
 - PMRs 145
- credentials 75, 165, 167
- CUME_DIST 35
- CURRENT QUERY ACCELERATION 159
- CURRENT QUERY ACCELERATION special register 17

D

- data consistency 40
- data loss, incremental update function 125
- data privacy 173
- data sharing group 61, 64, 82
- data slices 5
- data types
 - dashDB 22
 - Db2 for z/OS 22
 - ranges 22
 - supported 22
 - unsupported 22
- database
 - installing updates 135
 - update packages 137
- database connection profile 1, 71
- database engine
 - supported functions 35
- database request modules 21
- database server
 - connection 73
 - exploring 74
- Db2 10 for z/OS
 - new features 50
- Db2 11 for z/OS
 - aliases 50
 - array data type 50
 - global variables 50
 - grouping sets 50
 - named memory variables 50
 - new features 50
 - supergroups 50
 - synonyms 50
- Db2 Dynamic Statement Cache 17
- Db2 for z/OS, incompatibilities 113
- DECLARE CURSOR 21
- defining
 - accelerator-shadow tables 79
 - data 2
- delete trace information 64
- diagnostic information 113
- distributing the workload to multiple accelerators 16
- distribution keys
 - changing 90
 - impact
 - join performance 8
 - query acceleration 8
 - overview 2
 - specifying 90
- division by zero 41
- DNS_QUERYINFO_TABLE 94
- DOUBLE 54

- DRDA 21
- DSNX881I messages
 - structure of 145
- SYSLOG 145

E

- EBCDIC 22
- enhancements in this version xi
- environment variables
 - AQT_HOST_PACKAGE_DIRECTORY 135, 137, 141
- European Union (EU) 173
- evaluation, predicate 43
- event viewer 130
- event viewer for incremental updates 130
- exceptions 43
- EXPLAIN
 - information 94
 - tables 94
 - tables, creating 94
- exploring a database server 74
- export of table specifications 110

F

- FAILBACK processing
 - overview 17
 - restrictions 17
- firewall 151
- FIRST_VALUE 35
- FLOAT 41, 54
- floating-point error tolerance 54
- friendly arithmetic 41

G

- GB18030 characters, incremental update function 125
- GDPR 173
- General Data Protection Regulation
 - See* GDPR
- global variables 50
- glossary 181
- grouping sets 50
- GUI elements 59

H

- hash partitioning 8
- header, Accelerator view 64
- health, system 145
- heartbeat 145
- HFP 54
- HTAP 117
 - See* WAITFORDATA
- Hybrid Transactional Analytical Processing (HTAP)
 - See* WAITFORDATA

I

- IBM Db2 Analytics Accelerator for z/OS
 - Console 75, 165, 167
 - installing updates 135
 - ssh 167
 - telnet 165
 - update packages 137

- IBM Db2 Analytics Accelerator for z/OS *(continued)*
 - updates, installing 135
- IBM Db2 Analytics Accelerator Studio 57
- IBM InfoSphere Data Replication for z/OS
 - See* CDC
- IDT
 - See* in-database transformation
- IEEE754 54
- importing table specifications 111
- improvements in this version xi
- in-database transformation 11
- incompatibilities 41, 47
- incremental updates
 - ALTER TABLE operation 125
 - data loss 125
 - description 117
 - event messages 145
 - excluding tables (disabling) 125
 - GB18030 characters 125
 - including tables (enabling tables for) 125
 - incompatibilities 47
 - monitoring 128
 - REORG 125
 - ROTATE PARTITION 125
 - start 123
 - stop 123
 - use cases 117
 - using the event viewer 130
 - WAITFORDATA 119
 - warnings 131
- inhouse query processing 17
- installation, updates 135
- interface elements
 - Administration Explorer 59
 - menu bar 59
- introduction 1
- isolation levels 40

L

- LAG 35
- LAST_VALUE 35
- LEAD 35
- length, rows 43
- LIKE operator 43
- list tasks 109
- load data 2
- load tables 83
- loadaccelerator-only tables 11
- LOBVALA 157
- location name 77
- loss of data, incremental update function 125

M

- maintenance 140
 - installing updates 142
 - transferring updates 141
- maximum row length of tables 41
- MBCS 43
- MEDIAN function 26
- menu bar 59
- mixed data 43
- modification of connected databases 93
- monitoring 128
- monitoring incremental updates 128

- Monitoring section, Accelerator view 64
- multibyte data 43

N

- named memory variables 50
- networking details 78
- new features xi
- NTH_VALUE 35
- NTILE 35

O

- Object List Editor 61
- OCTETS 26
- out-of-range values 41

P

- pairing code 75
- parameter markers 43
- PERCENT_RANK 35
- PMR 113
 - See* problem management record
- PMRs, creating 145
- port-forwarding 142
- ports 151
- precision of timestamp 50
- predicate evaluation 41, 43
- preferences 116
- privacy
 - See* data privacy
- problem management record 113
- process flow 4

Q

- queries, running 93
- query
 - evaluation 5
 - matching 5
 - redirection 5
- query acceleration
 - disabling 2, 3, 89
 - enabling 2, 3
 - enabling tables 88
 - reason codes 94
 - selecting tables 6
- query history 157
- query monitoring 103
 - columns 99
- Query Monitoring section in Accelerator view 157
- Query Monitoring section, Accelerator view 64
- query routing to an accelerator
 - conditions 17, 21
 - configuration 17
 - floating-point error tolerance 54
 - incompatibilities 41
 - isolation levels 40
 - restrictions
 - result data types 39
 - special registers 39
 - subtypes 39
 - result data types 39
 - supported functions of database engine 35

- query routing to an accelerator (*continued*)
 - supported queries 21
 - supported special registers 37
 - supported SQL functions 26
- query routing to an accelerator, incompatibilities with Db2 for z/OS 50
- QUERY_ACCEL_OPTION 159
- QUERY_ACCELERATION 159
- QUERY_ACCELERATION ZPARM 17
- QUERYACCELERATION 18, 159

R

- random distribution 2, 8
- RATIO_TO_REPORT 35
- REAL 54
- reason code 159
- reason codes 94
- redistribution 2
- refresh interval of Accelerator view 116
- remove
 - accelerators 112
 - tables 110
- removeaccelerator-shadow tables 2
- REORG in connection with incremental updates 125
- repair 140
 - installing updates 142
 - transferring updates 141
- replication events 130
- restrictions
 - accelerator-only tables 47
- ROTATE PARTITION, incremental update function 125
- row length 43
- running queries 93

S

- save trace information 64, 115, 132
- scalar SQL functions 26
- schema filters 71
- selecting tables
 - access plan graph 6
 - basic concepts 6
 - overview 2
- service request 140
- SET CURRENT QUERY ACCELERATION 159
- sharing credentials 75, 165, 167
- software update transfer 64
- software version
 - activation 138
 - switching 64
- special registers
 - equivalent dashDB functions 37
 - replacing 37
- SQL scripts 93
- ssh logon to console 167
- starting incremental updates 123
- states of tables 9
- static SQL queries 5
- static SQL queries, bind options 18, 159
- status of an accelerator 81
- stopping incremental updates 123
- strings in TIME function 45
- supergroups 50
- supported built-in functions 35
- supported SQL functions 26

- synchronization of table data 40
- synonyms 50
- system
 - health 145
 - monitoring 145
- system table entries 17

T

- table skew threshold 116
- table specifications
 - exporting 110
 - importing 111
- tables
 - defining 79
 - disabling query acceleration 89
 - enabling for query acceleration 88
 - loading 83
 - possible operations depending on state 9
 - removing from accelerator 110
 - selecting 6
 - states 9
- Tables section, Accelerator view 64
- task completion order 4
- tasks
 - overview 71
 - prerequisites 71
- telnet logon to console 165
- temporal tables 50
- time
 - representations 41
 - values 41
- TIME function
 - strings 45
- TIMESTAMP 45
- timestamp precision 50
- tracing 113
 - CDC Capture Agent 132
 - configuration 113
 - IBM InfoSphere Data Replication for z/OS 132
 - incremental updates 132
 - performance impact 113
 - profiles 113
 - save information 115
 - trace file location 115
- transferring update
 - IBM Db2 Analytics Accelerator for z/OS software 137
 - maintenance 141
 - repair 141
- troubleshooting 151
- truncation 41

U

- UNICODE 22
- Unicode SQL statements 42
- unified store 117
 - See* WAITFORDATA
- update
 - authentication 1
- update, credentials 64
- updating
 - credentials 75, 165, 167
 - database software 135
 - IBM Db2 Analytics Accelerator software 135

updating (*continued*)
 IBM Db2 Analytics Accelerator Studio from IBM
 Installation Manager 133
user IDs 71

W

WAITFORDATA 117
 description 119
warnings 131
wildcard characters (% or _ 43
workload balancing 16

X

XMLAGG function 26

Z

ZPARMs
 QUERY_ACCEL_OPTION 159
 QUERY_ACCELERATION 159



Product Number: 5697-DA7

SH12-7082-05

