5.1

IBM Db2 SQL Performance Analyzer User's Guide



Note:

Before using this information and the product it supports, read the "Notices" topic at the end of this information.

September, 2019

This edition applies to Version 5 Release 1 of IBM Db2 SQL Performance Analyzer for z/OS (product number 5697-W51) and to all subsequent releases and modifications until otherwise indicated in new editions.

[©] Copyright IBM Corporation 2002, 2019.

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

[©] Copyright International Business Machines Corporation .

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

Contents

About this information	ix
Obertand, Dho Col Derferman Angleren angliste	
Chapter 1. Db2 SQL Performance Analyzer overview	1
What's new in Db2 SQL Performance Analyzer	1
Db2 12 function level support	1
What does Db2 SQL Performance Analyzer do?	2
Db2 SQL Performance Analyzer features and benefits	2
Generic plan tables	2
Catalog access	3
Db2 SQL Performance Analyzer and QMF	3
Remote operation through DRDA	3
Accelerated table and SQL report processing	4
SQL PA stored procedure capability	4
SUL Advisor	4
EXClude WHERE CURRENT OF statements from Query Limit reports	5 -
Actual query cost	5 0
DD2 SQL Performance Analyzer components and facilities	ð
Parser and catalog interfaces	8
Resource estimator	8
QMF Intercept	9
CAF and QMF features that enhance SQL PA	9
Plan Table	10
Db2 COL Devformance Analyzev evaluations and process flow	1U
Db2 SQL Performance Analyzer architecture and process flow	±±
DD2 SQL Performance Analyzer operating environments	∠⊥
Service updates and support information	⊥3 1 2
Accessibility features	כב גר
Accessibility leatures	14
Chapter 2. Customizing SOL PA	15
Prenaring to customize SOL PA	15
Set up your environment prior to customization	17
Plan table configurations for DB2 FXPI AIN output	18
Requirements for using the SOL Performance Analyzer stored procedures	<u>10</u> 18
Worksheets: Gathering required data set names	10
Worksheets: Gathering narameter values for Tools Customizer	20
Starting and preparing Tools Customizer for use	<u>-</u> 0 41
Starting Tools Customizer	41
Modifying Tools Customizer user settings	
Changing display options	45
Customizing Db2 SOL Performance AnalyzerDb2 Autonomics Director.	
Roadmap: Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first	۵. 47
Roadmap: Customizing a new version of Db2 SOL Performance AnalyzerDb2 Autonomics	
Director from a previous customization.	
Roadmap: Recustomizing Db2 SOL Performance AnalyzerDb2 Autonomics Director	
Specifying the metadata library for the product to customize	50
Discovering Db2 SQL Performance AnalyzerDb2 Autonomics Director information automatical	50 v 52
Creating and associating Db2 entries	
Defining parameters	

Generating customization jobs	64
Submitting customization jobs	64
Browsing parameters	67
Copying Db2 entries	67
Removing Db2 entries	68
Deleting Db2 entries	69
Displaying customization jobs	
Maintaining customization jobs	
Using Tools Customizer in a multiple-LPAR environment	70
Optional: Creating generic plan tables	71
Optional: Interfacing with DB2 Query Monitor	
Optional: Configuring Db2 SQL Performance Analyzer for use with QMF	73
Optional: Interfacing with DB2 Administration Tool	77
Optional: Set up ANLCSPA for use from Browse or Edit sessions	78
Optional: Enabling the stored procedure customization for SQL Performance Analyzer	79
Important requirements for using the stored procedure	79
Requirements for allocating the stored procedure load library	79
Customizing the DSNSPAS or WLMSPAS startup procedure	79
Customizing the stored procedure ANLCNTL parameters	
Calling the Db2 SQL Performance Analyzer stored procedure	80
Specifying user parameters to the SQL Performance Analyzer stored procedure	82
Migrating SQL Performance Analyzer from one Db2 version to another Db2 version	83
Migrating SQL Performance Analyzer to Db2 10	84
Migrating SQL Performance Analyzer to Db2 11 or Db2 12	84
Chapter 3. Running SQL PA	
Using SQL PA in batch	87
SQL PA Batch JCL	87
Comparing old and new packages in batch	88
Testing current access paths for packages in batch	
Generating JCL for batch processing	
Required ddnames	
Using debug batch JCL	
Sort Query Limits Report batch JCL	
Running SQL PA using the ISPF Interface	
Starting SQL PA USING ISPF	
Navigating the DD2 SQL Performance Analyzer panels	
Del COL Derfermence Analyzer primery many panel	
Dbz SQL Performance Analyzer primary menu panel	95
Medifying Db2 SOL Performance Analyzer parameters	
Changing the CSV report entions	90 07
Basic processing entions	
Comparing old and new packages	
Testing current access naths for packages	101
Viewing the last set of generated reports	103
ANI CSPA edit macro	103
What if? analysis	106
Db2 SQL Performance Analyzer statistics migration	112
Db2 SQL Performance Analyzer reports menu	114
Using SOL PA in OMF	
OMF intercept exit	
Db2 primary options menu	
SQL PA cost summary under OMF	
Run queries that exceed limits in OMF	
QMF batch processing	119
Using the SQL PA stored procedure	

Accessing the SQL PA stored procedure	120
Plan Table report levels	121
EEEPATH table	
SQL data file: user_ID.ANLEEE.SQL	123
VIADRDA remote connect	
Protecting sensitive user table data	124
Chapter 4. SQL PA run parameters	127
Impact on cost estimates	
Parameter file characteristics	
Specifying ANLPARM user parameters	129
Specifying threshold control parameters	129
Specifying ANLCNTL configuration parameters and selecting the target host system	130
Enabling the predicate analysis report	130
Optimizing the table and SQL report processing	131
The difference between SANLDATA and SANLPARM	131
Qualifying unqualified objects in packages	132
Qualifying unqualified objects in DBRMs and SQL files	132
Chapter 5. Plan Table report input specifications	133
Explain process	133
Configuring table access	133
Plan table access	134
Input command requirements	134
Input commands	135
Input command definitions	140
Subparameter definitions	142
Specifying language and explaining data from another user	148
Selected authorization identification	149
User authorization table	149
User defaults data set	
Transferring complex queries to SQL PA	151
Chapter 6. SQL PA Reporting capabilities	153
Db2 SQL Performance Analyzer cost summary report	
Resource consumption estimates	154
QUNITS values	155
Charge back cost estimates	155
Warnings and notifications	155
Db2 SQL Performance Analyzer enhanced explain report	156
Table space scans	158
Cluster matching index scans	160
Cluster nonmatching index scans	162
Random matching index scans	
Random nonmatching index scans	
Multiple index scans	
Joining tables	
Subquery processes	
Sorting results	
Insert, Update, Delete, Select, and Ketresh	
DDZ SQL Performance Analyzer detail trace report	
Forecasting query processing	
LISE OF THE DDZ SQL PERFORMANCE ANALYZER INSTALL PARAMETERS	
FIUCESSIIIg Steps	U81 100
1/0 estimates Path longths and CPU time	۲۵۷ ۱۵۶
r ann tenguns and Oro unite Physical and logical I/O	103 105

Elapsed wait time	
Sample Db2 SQL Performance Analyzer detail trace report	
QLIMIT (installation limits) report	
Predicate Analysis information	
Host Variables information	
Checking host variables	
The Plan Table report	
Package reports	200
Remote packages reports	
Explain report review	
Explain tabular report reference	
Explain condensed report reference	
The CSV report	
Chapter 7 Troublesheeting SOL DA	24.2
Chapter 7. Housteshouling SQL FA	
Gathering diagnostic information	
Resetting the DD2 SQL Performance Analyzer REGISTRY	
ISU Interfacing notes for installers	
DD2 SQL Performance Analyzer variables in the ISPF user profile	
Host variable support from any 150 input medium	
SQL PA supports upper and tower case	214
Applying user indexes to tables	214
Chanter 8 Tools Customizer troubleshooting	217
Gathering diagnostic information	217
Determining the trace data set name	217
Chapter 9. How to look up message explanations	219
Chapter 10. Reference information	221
Chapter 10. Reference information Tools Customizer reference	221 .221
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets	221
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters	221
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters	221
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF	221
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents	221
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table	221
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEWORK table	221 221221226230241242242242242242242242
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEWORK table The EEEDBRM table	221 221 221 226 230 241 242 242 242 242 242
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEWORK table The EEEDBRM table The EEEDBRM table The EEEHINT table	221 221 221 221 226 230 241 242 242 242 242 242 243 243
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEWORK table The EEEDBRM table The EEEDBRM table The EEEHINT table The EEEPLAN table	221 221 221 226 230 241 242 242 242 242 242 243 243 243
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEAUTH table The EEEDBRM table The EEEDBRM table The EEEPLAN table The EEEPLAN table The EEEPATH table	221 221 221 226 230 241 242 242 242 242 243 243 243 244 245
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEWORK table The EEEDBRM table The EEEHINT table The EEEPLAN table The EEEPLAN table The EEEPATH table PLAN_TABLE contents	221 221 221 226 230 241 242 242 242 242 242 243 243 243 243 243
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEAUTH table The EEEDBRM table The EEEDBRM table The EEEPLAN table The EEEPLAN table The EEEPATH table PLAN_TABLE contents Migration from IBM Db2 Path Checker to SQL Performance Analyzer	221 221 221 226 230 241 242 242 242 242 242 243 243 243 243 244 245 246 246
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 243 243 243 243 243 244 245 246
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 243 243 243 243 243 243
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 243 243 243 243 243 244 243 244 245 246 246 246 249
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 242 243 243 243 244 245 246 246 246 246 246 246 246
Chapter 10. Reference information Tools Customizer reference Tools Customizer terminology and data sets ANLCNTL configuration parameters ANLPARM user parameters ANLPARM user parameter settings for QMF Plan Table table contents The EEEAUTH table The EEEWORK table The EEEDBRM table The EEEDBRM table The EEEPLAN table The EEEPLAN table The EEEPATH table PLAN_TABLE contents Migration from IBM Db2 Path Checker to SQL Performance Analyzer Db2 SQL Performance Analyzer messages and return codes Output report messages Warnie de und element	221 221 221 226 230 241 242 242 242 242 242 243 243 243 244 245 246 246 246 246 246 246 249 249 249
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 242 243 243 243 244 245 246 246 246 246 246 246 249 249 249 249 249 249 249 249 249 249
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 242 242 243 243 243 243
Chapter 10. Reference information Tools Customizer reference	221 221 221 226 230 241 242 242 242 242 242 242 243 243 243 243
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 242 242 243 243 243 243
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 242 244 245 244 245 246 246 246 246 246 246 249 249 249 249 249 249 249 249 249 249
Chapter 10. Reference information	221 221 221 226 230 241 242 242 242 242 242 242 243 243 243 243

Tools Customizer messages	
Tools Customizer messages	
Notices	403
Index	

About this information

IBM[®] Db2[®] SQL Performance Analyzer for z/OS[®](also referred to as Db2 SQL PA or SQL PA) provides Db2application programmers and database designers with resource usage information and costs associated with SQL queries without having to run them in Db2.

These topics provide instructions for installing, configuring, and using SQL PA.

These topics are designed to help application programmers and database designers perform the following tasks:

- Customize SQL PA
- Produce and interpret cost reports
- Tune queries to achieve maximum performance
- Use the Plan Table report to better understand access path selection
- Run SQL PA in different environments
- Diagnose and recover from SQL PA problems

To use these topics, you should have a working knowledge of the following:

- The z/OS operating system
- ISPF
- SMP/E

Tip: To find the most current version of this information, always use <u>IBM Knowledge Center</u>, which is updated more frequently than PDF books.

Chapter 1. Db2 SQL Performance Analyzer overview

IBM Db2 SQL Performance Analyzer for z/OS (also referred to as SQL Performance Analyzer or SQL PA) is a query analysis tool that you can use to optimize the performance of your SQL statements for Db2.

Topics:

- "What does Db2 SQL Performance Analyzer do?" on page 2
- "Db2 SQL Performance Analyzer features and benefits" on page 2
- "Db2 SQL Performance Analyzer components and facilities" on page 8
- <u>"Db2 SQL Performance Analyzer architecture and process flow" on page 11</u>
- "Db2 SQL Performance Analyzer operating environments" on page 12
- "Product documentation and updates" on page 13
- "Accessibility features" on page 14

What's new in Db2 SQL Performance Analyzer

This topic summarizes the technical changes for this edition.

New and changed information is indicated by a vertical bar (|) to the left of a change. Editorial changes that have no technical significance are not noted.

September, 2019

- New examples have been added to show how customers have typically set parameter values in the ANLCNTL and ANLPARM members. See <u>"ANLCNTL configuration parameters" on page 226</u> and "ANLPARM user parameters" on page 230 for more information on the examples.
- New fields on COMPARE and TEST panels. **Package Cost Filter** limits processing to a specified difference in total cost of all statements within a package. **Statement Cost Filter** limits processing to a specified difference in total cost of each statement within a package. **Show Invalid Packages** determines whether a warning message will be shown for packages flagged as invalid.

Db2 12 function level support

When you activate new Db2 12 function levels in a Db2 subsystem or data sharing group, enhancements might become available that impact IBM Db2 SQL Performance Analyzer.

The levels of function level support are defined as follows:

Tolerated

The product works as it did on a previous release or function level of Db2 for z/OS, but it does not support the new features of this function level.

Supported

The product supports most, but not necessarily all, of the new function-level features that IBM deems the most significant for this product.

The following function levels are tolerated or supported by Db2 SQL Performance Analyzer and are provided with the corresponding PTF, if any. For more information on Db2 features at each function level, select the link in the **Db2 12 function level** column.

Table 1. Db2 SQL Performance Analyzer PTFs in support of Db2 12 function levels			
Db2 12 function level Toleration PTF Support PTF			
FL508 UI57237 No additional PTF require		No additional PTF required	
FL507	UI57237	No additional PTF required	

Table 1. Db2 SQL Performance Analyzer PTFs in support of Db2 12 function levels (continued)			
Db2 12 function level Toleration PTF Support PTF		Support PTF	
FL506	UI57237	No additional PTF required	
FL505	UI57237	No additional PTF required	
FL504	UI57237	No additional PTF required	
FL503	No PTF required	No PTF required	
FL502	No PTF required	No PTF required	
FL501	No PTF required	No PTF required	
FL500	No PTF required	No PTF required	

What does Db2 SQL Performance Analyzer do?

Db2 SQL Performance Analyzer provides you with an extensive analysis of SQL queries without running them. This analysis helps you to tune your queries to achieve maximum performance.

Db2 SQL Performance Analyzer prevents problems associated with long-running queries by calculating the cost of queries before processing them. Db2 SQL Performance Analyzer tells you how long queries will take before you run them, before resources are consumed, and before the query is terminated by a governor. Db2 SQL Performance Analyzer determines the cost of running a query under the attach facilities of IMS, CICS[®], batch, TSO, SPUFI, and QMF. The monetary cost of each query is presented and a cost estimate in familiar units: CPU time, I/O count, elapsed time, and as QUNITS (a single number that represents the overall cost). Data and content are presented in easily readable reports.

The Plan Table report provides you with information about how Db2 accesses data for a given SQL statement. The Plan Table report is produced in both TSO and batch. Programming language data sets can be scanned directly to extract SQL for analysis.

Db2 SQL Performance Analyzer features and benefits

SQL PA includes several performance features that provide for fast and efficient processing without affecting the system.

Topics:

- "Generic plan tables" on page 2
- <u>"Catalog access" on page 3</u>
- "Db2 SQL Performance Analyzer and QMF " on page 3
- "Remote operation through DRDA" on page 3
- "Accelerated table and SQL report processing" on page 4
- "Db2 SQL Performance Analyzer stored procedure capability " on page 4
- "SQL Advisor" on page 4
- <u>"Actual query cost" on page 5</u>
- "Exclude WHERE CURRENT OF statements from Query Limit reports" on page 5

Generic plan tables

To eliminate the need for each user to have their own *authid*.PLAN_TABLE for the EXPLAIN output, Db2 SQL Performance Analyzer can use a set of generic plan tables in a specifically designed segmented table space.

Using generic plan tables provides for both immediate and simultaneous access, scratch, and reuse of the generic tables with mass delete (a single I/O to drop all entries). By registering each generic copy in use,

you can avoid conflicts during concurrent use of Db2 SQL Performance Analyzer. Concurrent Db2 SQL Performance Analyzer users can process against these generic plan tables without conflict. You can change the settings during configuration. You can also create generic DSN_STATEMNT_TABLE tables and DSN_FUNCTION_TABLE tables.

If you are not using secondary authorization IDs and generic plan tables, you can use SQL PA under a primary authorization ID, if a valid plan table exists under that authorization ID. Db2 SQL Performance Analyzer automatically detects this situation and switches from the generic to the primary plan table. Users in this situation should avoid assigning query numbers above 100 000 000 for their own EXPLAIN operations, because Db2 SQL Performance Analyzer deletes all queries above this number upon invocation.

The authorization ID that is used to customize Db2 SQL Performance Analyzer should not be a generic ID. Generic IDs should not be used outside of Db2 SQL Performance Analyzer. For example, generic IDs should not be the owner of a package that might contain static bind information that the user would want to keep. The EXPLAIN tables of generic IDs will be cleaned up upon invocation and are intended to be "owned" by SQL PA.

To provide direct selection of specific PLAN_TABLE, DSN_STATEMNT_TABLE, and DSN_FUNCTION_TABLE tables, Db2 SQL Performance Analyzer provides the USEPLAN and SETPLAN parameters.

Catalog access

Db2 SQL Performance Analyzer accesses the catalog for information about the table spaces, tables, indexes, and relationships that are determined by each access path, and retrieves critical size and cardinality data that is used as input to the cost assessment process.

Db2 SQL Performance Analyzer accesses the catalog only once for each object sought, even if many queries use the same tables or indexes in their access path. Db2 SQL Performance Analyzer stores the details for hundreds of each object type in memory, and checks this information cache before attempting to access the catalog again. Catalog access is available for the SYSTABLESPACE, SYSTABLES, SYSINDEXES, SYSKEYS, SYSRELS, SYSCOLUMNS, SYSSYNONYMS, and SYSROUTINES tables.

Db2 SQL Performance Analyzer and QMF

Db2 SQL Performance Analyzer interfaces with QMF by using the QMF governor exit, and contains code that is specifically designed for optimum performance with minimal elapsed time. For example, virtually all I/Os to disk are eliminated and are replaced by memory resident arrays.

The PL/I environment is essential to running the Db2 SQL Performance Analyzer cost components. The PL/I environment is set up once upon entry into QMF, and remains ready for processing each query. When QMF terminates, the PL/I environment is disabled.

All nonessential code has been removed from the load module ANLQMF. Also, the source code for the exit interface is provided so that you can define the Db2 SQL Performance Analyzer parameters that describe the environment directly into the exit. By doing so, you can eliminate the processor usage that is required to read the parameters from a file for each run.

Remote operation through DRDA

Db2 SQL Performance Analyzer can run a remote copy of Db2 SQL Performance Analyzer on another Db2 subsystem, on the same z/OS server, or on another z/OS server by using the DRDA protocol and the SQL CONNECT statement.

The remote locations must contain fully installed versions of Db2 SQL Performance Analyzer, including a REGISTRY and the following explain tables:

- PLAN_TABLE
- DSN_STATEMNT_TABLE
- DSN_FUNCTION_TABLE
- DSN_DETCOST_TABLE

DSN_PREDICAT_TABLE

Users can initiate Db2 SQL Performance Analyzer on any system that contains a valid copy of the product and run their SQL by using any other valid copy and Db2 subsystem through DRDA Connect facilities. The location at which the explain will be performed and the location for the package information can be set differently. Results are returned back to the local system for presentation, storing report files, and other tasks. This feature provides the ability to quickly diagnose production problems.

This feature operates in TSO and batch modes only.

Accelerated table and SQL report processing

The FASTTBL parameter provides you with the ability to bypass the Explain processing so that you can produce table reports faster.

The FASTTBL parameter is one of the system parameters. Depending on the value of the parameter certain information might not be shown. For example, information about shared or exclusive locks is not shown when FASTTBL=YES.

Db2 SQL Performance Analyzer stored procedure capability

Db2 SQL Performance Analyzer can operate as a stored procedure, which enables users from anywhere in the network to call Db2 SQL Performance Analyzer for a cost estimate in real time, from Db2 applications, IMS, CICS, remote workstations, PCs, or any other compatible connection.

Users receive a quick and comprehensive view of how long their queries are likely to run, along with any warning flags about exceeding the limits that are defined in the configuration parameters.

The IBM Resource Limit Facility (RLF) governs the processor time that is spent for dynamic queries only, and only within the Db2 address spaces. If the limit is exceeded, the query ends while in process. However, with the Db2 SQL Performance Analyzer stored procedure, you can develop applications with a built-in governor for queries that can preempt any RLF cancellations. This feature provides you with control and flexibility to handle larger requests.

Whether the query is running under IMS, CICS, or is from a remote client workstation, the following call to the Db2 SQL Performance Analyzer stored procedure is required:

EXEC SQL CALL ANLPRCR (<varname>)

The parameter list includes the SQL statement to be analyzed, its length, any Db2 SQL Performance Analyzer user parameters that you want to work with, and several output host variables that are populated with Db2 SQL Performance Analyzer cost estimates and warning flags.

Db2 SQL Performance Analyzer provides a stored procedure for use with WLM-controlled address spaces. This stored procedure is named ANLPRCR and uses the Recoverable Resource Manager Services Attach Facility (RRSAF).

A PL/I sample program and a COBOL sample program are provided for the stored procedure environment. Use these sample programs to learn about and design programs that use the WLM stored procedure. Both the source code and the executable modules of the PL/I sample programs are included.

SQL Advisor

The SQL Advisor finds and presents opportunities for fine-tuning performance.

It uses a comprehensive set of warnings, alerts, and recommendations to convey this information. The SQL Advisor can also issue guidelines, notes, and other informational messages to assist you in understanding the logic behind certain design decisions and tuning choices. All advice is integrated into the Explain and Detail Trace reports. You control how much information is generated.

Exclude WHERE CURRENT OF statements from Query Limit reports

The WHERE CURRENT OF clause can cause values that exceed limits set for acceptable query statements. If you are familiar with all of your WHERE CURRENT OF clauses, you can exclude them from your Query Limit reports.

The DSPWCOC parameter provides you the ability to exclude the WHERE CURRENT OF statements from the Query Limits report. Set this parameter to NO to have WHERE CURRENT OF statements excluded from the Query Limits report.

Actual query cost

The total impact of SQL queries that are processed by Db2 is difficult to assess. One reason is because obtaining and recording all of the cost data in one place for a specific query is difficult.

For example, Db2 records some of the processor time that is used by a specific query in the Type 101 Accounting SMF record. However, this time does not represent all of the processor time that was used by the query.

The Sequential and List Prefetch, Asynchronous Writes, Thread Management, part of Lock Management, and other processes are performed by Db2 for you, and charged to one of its own address spaces. Consider the Db2 processor time distribution, as shown in the following figure.



Figure 1. Processor time distribution

Processor time distribution has the following four levels.

LEVEL 1

This level includes the entire ASCB CPU times that are recorded for a typical transaction, including the IRLM and Application A/S (CICS, IMS). This level represents the full cost of a transaction in processor consumption terms.

LEVEL 2

This level includes that portion of the CPU times that are reported by Db2 as CPU times (TCB, SRB, and Total) in SMF 101 of the individual user.

LEVEL 3

This level includes the CPU times that Db2 does not report for each user. The CPU times are reported for the entire system (ASCB CPU Time) in SMF 100.

LEVEL 4

This level includes the application init/term/overheads and program processing that is not recorded by Db2.

As the previous figure shows, assembling all the components of Db2 processor time is not a trivial matter. Looking at the processor time in the SMF Type 101 accounting record yields only part of the processor time that is used by the query. The remainder of processor time is included in the SMF Type 100 statistics record, along with the usage for many other queries, recorded at periodic time intervals. Some components of Db2 processor times are not recorded at all. Although RMF can provide the total resource consumption by address space, only Db2 SQL Performance Analyzer can correlate these costs to the specific users who caused them.

Consider the following breakdown of processor time.

• Within the Db2 DATABASE SERVICES address space:

SRB (always run before any TCB):

- 1. Database writes (async write)
- 2. Sequential and list prefetch
- 3. Part of thread creation

TCB (in priority sequence):

- 1. VSAM open and close
- 2. Space management
- 3. BSDS activity
- 4. Statistics recording
- Within the Db2 SYSTEM SERVICES:

SRB work:

- 1. Checkpoint logging
- 2. Physical logging

TCB work:

- 1. Archiving
- 2. Space management
- Within the Db2 IRLM:

TCB Work:

- 1. Deadlock detection
- Within the USER (Application) address space:

TCB activity:

- 1. SQL processing
- 2. Sync reads
- 3. Sync writes
- 4. Lock requests or acquire
- 5. Lock release
- 6. Logical logging

Much of the processing that happens outside the scope of the address space of the user is absorbed by Db2. The true cost of each Db2 query might not be adequately represented by the SMF 101 accounting record or by the DB2PM accounting report (or its equivalents). Because Db2 absorbs some of the query costs, some users consider the processor usage of Db2 to be high. The bulk of this processor usage is merely Db2 working on behalf of its users and absorbing the costs internally. Even I/O that is performed by the Media Manager under Database Services A/S is not recorded reliably by SMF within the records (including Job and Step termination) of the user.

In a SQL Performance Analyzer cost summary, all of the processor time factors are considered. The fully loaded costs of each query are represented in a forecast of resource consumption that is based on each particular form of query.

SQL Performance Analyzer can split the cost analysis into incremental components, so that you can see exactly where each query is spending most of its time. SQL Performance Analyzer does analysis without having to run the query, which is valuable for long-running queries against large databases.

Along with the processor time for asynchronous I/O, SQL Performance AnalyzerSQL Performance Analyzer forecasts the actual counts of sequential and list prefetch, and synchronous read I/O counts, all of which are sometimes difficult to predict. Even more so, the asynchronous writes of those pages that are updated by inserts, updates, and deletes, and the Db2 logging activity that is associated with those writes, are almost never shown in the Type 101 Accounting record. The writes are not shown because they occur after the query terminates. SQL Performance Analyzer can correlate them directly to the query that caused them. The writes are used to determine a realistic response time that includes the processor usage of the delivery vehicle, be that TSO SPUFI, or IMS Wait for Input. Consider the elapsed time distributions that are shown in the following figure.



Figure 2. Elapsed time distribution

The entire elapsed time of the transaction includes some portions that are not captured in Db2 class 1 elapsed time, such as create and terminate thread, plus any DBMS processor usage (CICS/IMS init/term/ scheduling).

Class 2 elapsed time is a subset of class 1 and represents the time spent in Db2 asynchronous and synchronous only. Class 3 elapsed time includes the synchronous I/O and lock wait times, and is a subset of class 2. It also includes the asynchronous I/O wait times, and Service Task (SRB) and Archive Log waits.

The last component of elapsed time is application time, which is partially captured in Db2 class 1 for SQL-related activities, and partially not captured for subsequent processing of the rows after fetch.

SQL Performance Analyzer provides an end-to-end view of elapsed time for each of your queries. It also highlights the work done by the system, after your query ends, in support of insert, update, and delete pages that are logged and rewritten to DASD on your behalf. SQL Performance Analyzer can provide a

more resource-intensive view of how your SQL is processed, and a direct comparison back to SMF accounting records validates part of the cost that is presented by SQL Performance Analyzer.

Db2 SQL Performance Analyzer components and facilities

Db2 SQL Performance Analyzer is comprised of several main components. Db2 SQL Performance Analyzer uses the Db2 call attachment facility (CAF) or the recovery resources manager attach facility (RRSAF) to interface directly with Db2.

Topics:

- "Parser and catalog interfaces" on page 8
- "Resource estimator" on page 8
- "QMF Intercept" on page 9
- "CAF and QMF features that enhance SQL PA" on page 9
- "Plan Table report" on page 10
- "Tools Customizer overview" on page 10

Parser and catalog interfaces

The parser and the catalog interfaces find important SQL and store it in a list.

SQL parser routine

The SQL parser routine finds the executable SQL in any sequential file, PDS member, or DBRM module (all DBRMs are supported). The SQL parser extracts those statements and assigns Explain query numbers to them. The following numbering convention is used:

- Sequential SQL begins with 100 000 001.
- SQL statements that are extracted from DBRM modules are identified by the number 100 000 000 + the DBRM statement number.

The role of the SQL parser is to parse and store information about each query, such as the type and number of predicates, sort clauses, columns, and correlation names. This information is used later during the costing phase. The Explain-ready SQL statements are then passed to Db2 for evaluation by use of the EXPLAIN statement.

Catalog Interface

The Catalog Interface component of Db2 SQL Performance Analyzer looks up the access paths that are chosen for each query, the tables and indexes involved, referential integrity relationships, space information, and other Explain data, and then disposes of the generic plan table entries with a mass delete.

Db2 SQL Performance Analyzer accesses the catalog for information about the table spaces, tables, indexes, keys, and relationships discovered by each access path, and retrieves critical size and cardinality data that are essential to the cost assessment process.

Db2 SQL Performance Analyzer accesses the catalog only once for each object sought, even if many queries are using the same tables or indexes in their access path. Db2 SQL Performance Analyzer stores the details about each object type in memory, and checks this information cache before attempting to access the catalog again. Catalog access is mainly confined to the SYSTABLESPACE, SYSTABLES, SYSINDEXES, SYSKEYS, SYSRELS, SYSCOLUMNS, SYSSYNONYMS, and SYSROUTINES tables.

Resource estimator

The resource estimator component of Db2 SQL Performance Analyzer takes the parser and catalog information and determines the cost of each SQL statement.

The resource estimator determines costs by using its knowledge of the algorithms for the Db2 optimizer, and of the hundreds of benchmarked path lengths that represent the incremental costs of Db2 operations.

The costing module of the resource estimator determines a fully loaded cost for each query by using all of the intelligence data that is provided by the parser, the catalog sizing information in the catalog, and the evaluation techniques of the optimizer.

For example, the resource estimator calculates the filtering effects of each predicate in the WHERE clause to determine a final estimate of the number of rows retrieved. It also determines the portion of the data table and indexes that were accessed to provide these rows. In this processing, Db2 SQL Performance Analyzer mimics the algorithms of the optimizer and uses its costing methods to arrive at the estimate that is provided by Db2 SQL Performance Analyzer. The resource estimator is also sensitive to user and installation parameters, and can hypothesize and predict virtually any Db2 operational situation or environment that you can envision. Db2 SQL Performance Analyzer can also use the internal estimates of the optimizer for path length by using data from the DSN_STATEMNT_TABLE table.

QMF Intercept

The QMF Intercept program, ANLQMF, is a condensed version of ANLSQLPA.

The QMF Intercept program manages single SQL statements under QMF and has minimal reporting requirements.

Db2 SQL Performance Analyzer interfaces with QMF by using the QMF governor exit and contains customized code that is designed for optimum performance with minimal elapsed time. Among the performance features is the elimination of virtually all I/Os to disk by replacing them with memory resident cache.

A PL/I environment is required to run the Db2 SQL Performance Analyzer cost components. That environment is set up once, upon entry into QMF. It remains ready for the processing of each query until QMF ends, which eliminates the constant processor usage of re-initializing the PL/I environment.

All nonessential code was removed from the load module, and the source code for the exit interface is provided so you can hard code the parameters that describe the environment into the exit. Hard coding the parameters avoids reading them in for each operation.

Generally, the QMF Intercept program is installed with options that show costs for only those SQL statements that exceed the guidelines that are defined in the configuration parameters. Db2 SQL Performance Analyzer runs in the background and only displays information if your queries are resource-intensive.

CAF and QMF features that enhance SQL PA

You can use the call attachment facility (CAF) and the QMF governor exit to enhance how Db2 SQL Performance Analyzer functions.

CAF

Db2 SQL Performance Analyzer uses the CAF to communicate with Db2 instead of using IKJEFT01 because CAF uses fewer resources.

Db2 SQL Performance Analyzer provides its own set of generic plan tables for collecting each user's transient Explain information, and provides a REGISTRY table to control contention and provide maximum concurrency. These tables are built under Db2 on the systems that Db2 SQL Performance Analyzer is installed on. The secondary authorization ID exit programs that are provided with Db2(DSN3@ATH and DSN3@SGN) are used to introduce a set of Db2 SQL Performance Analyzer secondary authorization IDs that own the set of Db2 SQL Performance Analyzer generic plan tables.

QMF governor exit

The QMF governor exit that is provided with Db2 can intercept all QMF queries and perform a cost analysis on them before the database is accessed. The Db2 SQL Performance Analyzer stored procedure is loaded into the DSNSPAS or WLMSPAS address spaces for access from anywhere in the Db2 network.

Db2 SQL Performance Analyzer supplements the normal facilities of TSO ISPF by adding to the panel and by adding CLIST and message libraries that are commonly allocated by ISPF to plug Db2 SQL Performance Analyzer into the framework of daily operations.

Plan Table report

The Plan Table report helps the Db2 application designer and programmer understand the access path that was selected by Db2 for a given SQL statement.

The Plan Table report displays information that is produced by the SQL EXPLAIN statement and additional relevant Db2 catalog data.

The Plan Table report has been fully integrated into the TSO ISPF interface of Db2 SQL Performance Analyzer. The Plan Table report can also be generated in a batch job.

Information that is produced by the SQL EXPLAIN function is useful in application design and tuning. However, the SQL EXPLAIN function information is not sufficient to determine and understand the access path that is chosen by Db2 and documented by the EXPLAIN function.

Typically, the application designer and programmer must extract Db2 catalog information to understand and validate the chosen access path.

The main purpose of the Plan Table report is to combine information that is returned by the SQL EXPLAIN statement and information that is extracted from the Db2 catalog and to document that data in an understandable way.

The Plan Table report can be generated for a query number of a previous EXPLAIN operation, the name of a package that was bound with the EXPLAIN option, a QMF query, or any Data Manipulation Language (DML) statement. For a QMF query or a DML statement, the statement is explained and an explanation of the selected access path is provided. If the package was not bound with the EXPLAIN option, a dynamic EXPLAIN of the DML SQL statements in the package is performed.

Using current EXPLAIN statement data as input to the SQL Performance Analyzer cost analysis provides you the opportunity to ensure that the Db2 catalog represents an accurate view of this application environment. For an application in the early stages of development, that view could be the projected production environment or it could represent a new indexing option that is being considered to boost the performance of an existing production application.

Tools Customizer overview

IBM Tools Customizer for z/OS (also referred to as Tools Customizer) standardizes many of the customization processes that are required to customize IBM Tools that run on z/OS.

Tools Customizer provides a consistent ISPF interface to ensure that the customization process is the same for all IBM Tools products and solution pack components. It also provides the ability to "discover" parameter values from products or solution pack components that you previously customized manually or by using Tools Customizer.

Features and benefits

Tools Customizer provides the following features:

- A single, consistent ISPF interface ensures that the customization process is the same for all IBM Tools products and solution pack components.
- A Discover EXEC discovers values for common product, LPAR, and Db2 parameters from a product or solution pack component that you previously customized manually or by using Tools Customizer. Each IBM Tools product and solution pack component has a unique Discover EXEC. The discovered parameters are stored in the data store. If the product or solution pack component that you want to customize exists in the Tools Customizer data store, Tools Customizer issues a warning before it overwrites existing values. Use the Discover EXEC by issuing the DISCOVER command on the **Customizer Workplace** panel.
- The data store retains discovered and manually specified parameter values. Because the parameter information is persistently stored, you have to manually specify or discover parameter values only once. Tools Customizer uses these parameter values where they are applicable.

- A metadata repository contains the members that define the following customization attributes for products and solution pack components:
 - Parameters, tasks, and steps for the product or solution pack component to be customized. Some product or solution pack parameters, tasks, and steps are required.
 - LPAR parameters for the local LPAR. All of the LPAR parameters are required.
 - Db2 parameters for the Db2 subsystem, Db2 group attach name, or Db2 data sharing member on which you will customize the product or solution pack component. All of the Db2 parameters are required.
- Default values are provided for product parameters and solution pack component parameters, LPAR parameters, and Db2 parameters. The default values show examples of how to complete fields.

Db2 SQL Performance Analyzer architecture and process flow

Db2 SQL Performance Analyzer (SQL PA) processes information from various input sources based on user and system parameters that you specify and then produces a series of reports that you use in your analysis.

SQL PA firsts reads and sets user and system-related parameters, and then reads SQL statements. The source of the SQL statements can be any of the supported input sources, such as a single DBRM or a whole DBRM library, a PLAN or PACKAGE from the system catalog, SQL that is contained in a sequential or partitioned data set, or SQL that is embedded in source code.

For each of the SQL statements, SQL PA determines if a plan exists. If a plan does not exist, SQL PA completes the following steps:

- 1. Performs a Db2 PREPARE and EXPLAIN to create the necessary plan information.
- 2. Fetches the rows from various EXPLAIN tables for analysis.
- 3. Fetches the system catalog statistics that are associated with Db2 objects that are being analyzed.
- 4. Analyzes the information collected.
- 5. Produces a number of reports.

The following figure shows the basic process flow for Db2 SQL Performance Analyzer.



Figure 3. The Db2 SQL Performance Analyzer process

Db2 SQL Performance Analyzer operating environments

Db2 SQL Performance Analyzer operates in four primary environments: batch, TSO, QMF, and as a stored procedure.

From the batch or TSO environment, Db2 SQL Performance Analyzer can evaluate queries that are targeted to run from any attach facility. You determine how your application's queries attach to Db2 by setting a user parameter that you supply to Db2 SQL Performance Analyzer. To evaluate how your application runs under the many versions of IMS or CICS, run Db2 SQL Performance Analyzer in batch mode to produce a series of reports that detail the forecasted performance of each query.

The same facilities that are available to users of Db2 SQL Performance Analyzer in batch mode are provided online under TSO. When you are designing an application, you can use Db2 SQL Performance Analyzer under TSO iteratively to refine the design of that application. With Db2 SQL Performance Analyzer under TSO, you also can target a few specific SQL statements from a file, and do a detailed cost analysis on these. By using this method, you can examine the effects of various options in the design process. Similarly, you can examine large files of SQL statements or scan DBRM modules.

Db2 SQL Performance Analyzer under TSO and batch provides identical report formats. Therefore, if the system is busy, consider submitting a batch job; if you are fine-tuning an application, do it online in real time by using Db2 SQL Performance Analyzer under TSO.

In the QMF environment, Db2 SQL Performance Analyzer is designed to intercept ad hoc queries and notify you of their cost before processing. Its primary purpose is to identify long-running queries, and give you the opportunity to cancel them before they run.

You can cancel queries that exceed the guidelines for processor use, I/O use, and elapsed time. You can configure queries to be canceled automatically or at your discretion. You can also redirect long-running queries to QMF batch where they can run to completion. When queries are unrestricted by the governor process, you can obtain the results set that you want.

Db2 SQL Performance Analyzer does not intercept IMS or CICS transactions in flight because these systems do not have a common facility to present the cost and solicit a response to continue or cancel. However, IMS and CICS can call the WLM stored procedure (ANLPRCR), which provides a convenient way to govern virtually all Db2 applications, both dynamic and static SQL. You can call Db2 SQL Performance Analyzer as a stored procedure, and pass the SQL statement for evaluation. Db2 SQL Performance Analyzer returns the cost values, warning flags, and other information in real time, so that the application can decide whether to proceed with the query or try an alternative query.

Db2 SQL Performance Analyzer can parse and extract the relevant SQL statements from the DBRM modules that are created by the Db2 precompiler. A cost analysis is then performed (either from TSO or batch mode), and even IMS- or CICS-related costs are included in the overall view of each query. For example, you can evaluate an entire application system that is running on IMS to determine which SQL statements are likely to run the longest. You can then fine-tune those statements to enhance their performance. All valid methods of attaching to Db2 are supported by Db2 SQL Performance Analyzer, along with their individual attendant processor usage. The processor usage, including the path lengths of attaching to Db2, is available through private benchmarks.

Service updates and support information

Service updates and support information for this product, including software fix packs, PTFs, frequently asked questions (FAQs), technical notes, troubleshooting information, and downloads, are available from the web.

To find service updates and support information, see the following website:

http://www.ibm.com/support/entry/portal/Overview/Software/Information_Management/ DB2_Tools_for_z~OS

Product documentation and updates

DB2 Tools information is available at multiple places on the web. You can receive updates to DB2 Tools information automatically by registering with the IBM My Notifications service.

Information on the web

The DB2 Tools Product Documentation web page provides current product documentation that you can view, print, and download. To locate publications with the most up-to-date information, refer to the following web page:

http://www.ibm.com/software/data/db2imstools/db2tools-library.html

You can also access documentation for many DB2 Tools from IBM Knowledge Center:

http://www.ibm.com/support/knowledgecenter

Search for a specific DB2 Tool product or browse the **Information Management > DB2 for z/OS family**.

IBM Redbooks[®] publications that cover DB2 Tools are available from the following web page:

http://www.redbooks.ibm.com

The Data Management Tools Solutions website shows how IBM solutions can help IT organizations maximize their investment in DB2 databases while staying ahead of today's top data management challenges:

Receiving documentation updates automatically

To automatically receive emails that notify you when new technote documents are released, when existing product documentation is updated, and when new product documentation is available, you can register with the IBM My Notifications service. You can customize the service so that you receive information about only those IBM products that you specify.

To register with the My Notifications service:

- 1. Go to http://www.ibm.com/support/mysupport
- 2. Enter your IBM ID and password, or create one by clicking register now.
- 3. When the My Notifications page is displayed, click **Subscribe** to select those products that you want to receive information updates about. The DB2 Tools option is located under **Software > Information Management**.
- 4. Click **Continue** to specify the types of updates that you want to receive.
- 5. Click **Submit** to save your profile.

How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other IBM product documentation, use one of the following options:

- Use the online reader comment form, which is located at http://www.ibm.com/software/data/rcf/.
- Send your comments by email to <u>comments@us.ibm.com</u>. Include the name of the book, the part number of the book, the version of the product that you are using, and, if applicable, the specific location of the text you are commenting on, for example, a page number or table number.

Accessibility features

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use a software product successfully.

The major accessibility features in this product enable users to perform the following activities:

- Use assistive technologies such as screen readers and screen magnifier software. Consult the assistive technology documentation for specific information when using it to access z/OS interfaces.
- Customize display attributes such as color, contrast, and font size.
- Operate specific or equivalent features by using only the keyboard. Refer to the following publications for information about accessing ISPF interfaces:
 - z/OS ISPF User's Guide, Volume 1
 - z/OS TSO/E Primer
 - z/OS TSO/E User's Guide

These guides describe how to use the ISPF interface, including the use of keyboard shortcuts or function keys (PF keys), include the default settings for the PF keys, and explain how to modify their functions.

Chapter 2. Customizing SQL PA

After you install Db2 SQL Performance Analyzer (SQL PA) by following the installation instructions in the Program Directory, you must run Tools Customizer to specify the variables for each Db2 subsystem and to customize the configuration and user parameters.

To complete the customization process by using Tools Customizer, see <u>"Preparing to customize SQL PA"</u> on page 15.

Preparing to customize SQL PA

Before you start to customize SQL PA, determine all of the customization values that you need to specify during the customization process, and familiarize yourself with all of the customization tasks.

The following checklist lists and describes each significant customization step. Use this checklist to guide you through the entire customization process.

Tip: Print the following checklist and the data set names and parameter values worksheets. Use the worksheets to record your values, and refer to them during the customization process.

Task	Link to detailed instructions	Status	
Tools Customizer basics			
Prior to beginning the customization process, familiarize yourself with Tools Customizer terminology and data sets, and other basic information about Tools Customizer.	"Tools Customizer reference" on page 221		
Software requirements			
Verify that your environment meets the minimum software requirements.	"Verify that your environment meets software requirements" on page 17		
 To install and use Db2 SQL Performance Analyzer, your environment must be running a supported version of the z/OS operating system and of Db2 for z/OS. 			
 To use the QMF governor, you must be using a supported version of QMF. 			
 To interface with Db2 Administration Tool for z/OS, Db2 Path Checker for z/OS, and Db2 Query Monitor for z/OS, you must be using supported versions of these products. 			
SMP/E installation			
Verify that Db2 SQL Performance Analyzer has been installed correctly. Db2 SQL Performance Analyzer is installed by using standard SMP/E processing.	"Verify that Db2 SQL Performance Analyzer has been installed successfully" on page 17		
Verify that Tools Customizer for z/OS has been installed correctly. Tools Customizer for z/OS is installed by using standard SMP/E processing.	"Verify that Tools Customizer has been installed successfully" on page 18		
Db2 components			
Verify that the DSNTIAD sample program, which is used to run SQL statements, is available.	None.		
Plan table configurations for Db2 EXPLAIN output			
Understand the differences between generic, alternative, or unique EXPLAIN plan tables.	"Plan table configurations for Db2 EXPLAIN output" on page 18		
Db2 SQL Performance Analyzer stored procedure requirem	ents		
Ensure that the requirements for using the stored procedures are met.	"Requirements for using the SQL Performance Analyzer stored procedures" on page 18		
Gather data set names			

ask Link to detailed instructions		Status	
During the customization process, you must specify names for the following types of data sets:	mes <u>"Worksheets: Gathering required data set names" on page</u>		
Tools Customizer			
Db2 SQL Performance Analyzer			
Gather parameter values			
During the customization process, you must specify parameter values for Db2 SQL Performance Analyzer, for Db2, and for your LPAR.	"Worksheets: Gathering parameter values for Tools Customizer" on page 20		
Customize Db2 SQL Performance Analyzer			
Start Tools Customizer by running a REXX EXEC from the ISPF Command Shell panel.	"Starting Tools Customizer" on page 41		
Set up Tools Customizer user settings. If you are running Tools Customizer for the first time, you must modify several user settings to suit your environment. Otherwise, if the user settings that you have already established are still appropriate, skip this step.	"Modifying Tools Customizer user settings" on page 42		
Complete the steps in the appropriate customization roadmap	based on the type of customization that you are performing.		
Customizing Db2 SQL Performance Analyzer for the first time	"Roadmap: Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time" on page 47		
Follow this roadmap if you do not have a customized version of Db2 SQL Performance Analyzer and you need to customize it for the first time.			
Customizing a different version of Db2 SQL Performance Analyzer	"Roadmap: Customizing a new version of Db2 SQL Performance AnalyzerDb2 Autonomics Director from a previous customization" on page 48		
Follow this roadmap if you have already customized a version of Db2 SQL Performance Analyzer and you want to use the same parameter values to customize a different version.			
Recustomizing Db2 SQL Performance Analyzer	"Roadmap: Recustomizing Db2 SQL Performance		
Follow this roadmap if you have a customized version of Db2 SQL Performance Analyzer but you want to change one or more parameter values.	AnalyzerDb2 Autonomics Director on page 49		
Some customization options require you to manually complete additional tasks after you have used Tools Customizer. If you generated jobs in Tools Customizer that correspond to the following customization options, complete the additional tasks before you submit the job In some cases, an optional task can be completed either by using Tools Customizer or by manually completing tasks without using Tools Customizer.			
Optional: Creating generic plan tables			
Create secondary authorization IDs that will act as the owners of the generic Db2 SQL Performance Analyzer plan tables.	"Optional: Creating generic plan tables" on page 71		
Optional: Interfacing with DB2 Query Monitor			
Enable Db2 SQL Performance Analyzer to interface with Db2 Query Monitor.	"Optional: Interfacing with Db2 Query Monitor" on page 73		
Optional: Configuring Db2 SQL Performance Analyzer for use with QMF			
Create the interface between Db2 SQL Performance Analyzer and QMF, and customize the interface by modifying the parameters that define the target host system.	"Optional: Configuring Db2 SQL Performance Analyzer for use with QMF" on page 73		
Interfacing with Db2 Administration Tool			
Add Db2 SQL Performance Analyzer to the Db2 Admin Launchpad.	"Optional: Interfacing with Db2 Administration Tool" on page 77		

Task Link to detailed instructions		Status
Optional: Set up ANLCSPA for use from Browse or Edit sessions		
Either use Tools Customizer to generate the job or run the ANLCALLC CLIST."Optional: Set up ANLCSPA for use from Browse or Edit sessions" on page 78		
Optional: Enabling the stored procedure customization for Db2 SQL Performance Analyzer		
Customizing and calling the stored procedure address space startup procedure.	"Optional: Enabling the stored procedure customization for Db2 SQL Performance Analyzer" on page 79	

Set up your environment prior to customization

Prior to beginning the customization process, ensure that your environment meets all requirements, that you have installed all prerequisite software, and that you have considered how you want to customize optional features.

Verify that your environment meets software requirements

Ensure that you are using z/OS V1.12 (5694-A01) or later.

Ensure that you are using one of the following supported versions of Db2 for z/OS:

- Db2 V10 (5605-DB2)
- Db2 Value Unit Edition V10.1 (5697-P31)
- Db2 V11 (5615-DB2)
- Db2 Value Unit Edition V11.1 (5697-P43)
- Db2 V12 (5650-DB2)
- Db2 Value Unit Edition V12.1 (5770-AF3)

Attention: Enabling-new-function mode (ENFM) is not supported.

To use the QMF governor, ensure that you are using one of the following supported versions of Db2 QMF:

- Db2 QMF V9.1
- Db2 QMF V10.1

To use Db2 SQL Performance Analyzer to expand the analysis of the SQL statements that are selected by IBM Db2 Path Checker for z/OS, ensure that you are using one of the following supported versions of Db2 Path Checker:

- Db2 Path Checker V4.1
- Db2 Path Checker V4.2

To add Db2 SQL Performance Analyzer to the Db2 Administration Tool Launchpad, ensure that you are using one of the following supported versions of Db2 Administration Tool for z/OS:

- Db2 Administration Tool V10.1
- Db2 Administration Tool V10.2
- Db2 Administration Tool V11.1

To set up the interface between Db2 SQL Performance Analyzer and Db2 Query Monitor for z/OS, ensure that you are using one of the following supported versions of Db2 Query Monitor:

- Db2 Query Monitor V3.1
- Db2 Query Monitor V3.2

Verify that Db2 SQL Performance Analyzer has been installed successfully

See the program directory for IBM Db2 SQL Performance Analyzer for z/OS, GI13-4609 for installation instructions.

Verify that Tools Customizer has been installed successfully

Tools Customizer for z/OS is a component of IBM Tools Base for z/OS (5655-V93), which is available free of charge. Tools Customizer for z/OS provides a standard approach to customizing IBM Db2 for z/OS Tools.

See the program directory for IBM Tools Base for z/OS, GI10-8819 for installation instructions.

Plan table configurations for Db2 EXPLAIN output

You can set up generic, alternative, or unique EXPLAIN plan tables.

You can define a unique *authorization_ID*.PLAN_TABLE for each user for Db2 EXPLAIN output. You can optionally choose to create generic plan tables, or you can specify alternative EXPLAIN tables for Db2 EXPLAIN output.

Creating generic plan tables is preferred to specifying alternative EXPLAIN tables. Generic plan tables provide better performance. Generic plan tables also allow important user entries to be overlaid if plan tables under the primary authorization ID of a user are used with Db2 SQL Performance Analyzer (SQL PA). Generic plan tables conserve space and allow SQL PA to handle ad hoc users who do not have their own EXPLAIN output tables. You can set up an arbitrary number of generic plan tables for general use. These plan tables are shared by all SQL PA online and batch users.

Tip: You might want to run SQL PA without adding generic secondary authorization IDs when you first start using SQL PA. In this case, all SQL PA users must have their own plan tables (*authorization_ID*.PLAN_TABLE) created before they can use SQL PA, or they must select one by specifying alternative EXPLAIN tables.

If you cannot set up generic plan tables due to restrictions for creating secondary authorization IDs, you can specify alternative EXPLAIN tables by using the USEPLAN and SETPLAN parameters.

You can limit SQL PA to use only the generic plan tables by setting the value of the SETPLAN system parameter to GENERIC.

Requirements for using the SQL Performance Analyzer stored procedures

SQL Performance Analyzer has some important requirements for running SQL Performance Analyzer as a stored procedure.

The stored procedure facility allows a specialized program or procedure to be called from any Db2 application program, including Db2 clients from PC servers that arrive through DDF, CICS, IMS, or other OS/390[®] regions. The procedure is called in the same way as a subroutine or subprogram. The procedure accepts host variables as input and returns host variables as output. Rows can also be included as part of the answer set that is returned to the caller.

Tip: SQL Performance Analyzer provides stored procedures that are managed by WLM (ANLPRCR and ANLPRER) and sample access programs for the ANLPRCR and ANLPRER stored procedures in PL/I and the ANLPRCR stored procedure in COBOL.

The SQL Performance Analyzer stored procedures can run in the WLM address space. z/OS is not aware that the stored procedures are Db2-related programs. Stored procedures must be created and linked with the Language Environment[®] libraries (LE/370), which is a common set of link-edit and runtime routines that support Assembler, COBOL, PL/I, FORTRAN, and C. These libraries are normally named SYS1.CEE.*xxxx*.

SQL Performance Analyzer users can issue a stored procedure call from any z/OS address space to the ANLPRCR stored procedure to determine the cost of the query before they run it. SQL Performance Analyzer returns the costs of the query, along with warnings and return codes that indicate whether the query runs within the defined limits.

The SQL Performance Analyzer stored procedure components are distributed in the libraries that are shown in the following table:

Table 2. Stored procedure component library distribution		
Library	Stored procedures	
hlq.SANLLOAD	The files here exist in SANLLOAD before customization for stored procedures. After customization for stored procedures, the files are copied to SANLLODS. Use the files when they are in the SANLLODS library.	
	ANLPRCR	
	ANLPRER	
	Before the ANLSTPR and ANLSTER programs can be used, you must have completed product customization by using Tools Customizer.	
hlq.SANLDBRM	ANLPRCR, ANLSTPR, ANLPRER, ANLSTER	
hlq.SANLSAMP	ANLSTPR, ANLSTCR, ANLSTER	
hlq.SANLJCL	The following files will be in SANLJCL after customization is complete:	
	 ssidSTER and ssidSTPR (execution JCL) 	
	 ssidWLM (create stored procedures) 	
	 member name that was chosen for WLM application environment during customization 	

Worksheets: Gathering required data set names

Identify and record the data set names that will be used during the customization process and make sure that requirements for certain data sets are met.

Tip: Print the following worksheets and refer to them during the customization process.

Data set names for Tools Customizer

Identify and record the following Tools Customizer data set names:

Data set name	Description	Special requirements	Your data set name
SCCQEXEC	EXEC library for Tools Customizer	None.	
SCCQDENU	Metadata library for Tools Customizer	None.	
SCCQLOAD	Executable load module library for Tools Customizer	None.	
SCCQMENU	ISPF messages for Tools Customizer	None.	
SCCQPENU	ISPF panels for Tools Customizer	None.	
SCCQSAMP	Sample members for Tools Customizer	None.	
SCCQTENU	Table library for Tools Customizer	None.	

Data set names for Db2 SQL Performance Analyzer

Identify and record the following Db2 SQL Performance Analyzer data set names. During the customization process, you will enter the following values on the Tools Customizer panels.

Data set name	Description	Special requirements	Your data set name
SANLBASE	Sample library for Db2 SQL Performance Analyzer	None.	
SANLCLST	CLIST library for Db2 SQL Performance Analyzer	You must have write access to this data set.	
SANLDBRM	DBRM library for Db2 SQL Performance Analyzer	None.	
SANLDENU	Metadata libraryDb2 SQL Performance Analyzer	None.	
SANLLOAD	Executable load module library for Db2 SQL Performance Analyzer	None.	
SANLMENU	ISPF messages for Db2 SQL Performance Analyzer	None.	
SANLPENU	ISPF panels for Db2 SQL Performance Analyzer	None.	
SANLSAMP	Sample members for Db2 SQL Performance Analyzer	You must have write access to this data set.	
SANLSQL	SQL storage library	You must have write access to this data set.	

Worksheets: Gathering parameter values for Tools Customizer

During the customization process, you provide parameter values for the product that you are customizing, for Db2, and for your LPAR.

Use the worksheets in this topic to record the appropriate parameter settings for your purposes, and then use these worksheets during the customization process. The worksheets are organized based on the order of the customization panels in Tools Customizer.

Tip: Print the following worksheets and refer to them during the customization process.

Settings for Tools Customizer

Description

Use the following worksheet to identify and record the values for Tools Customizer settings. During the customization process, you will enter these values on the **Tools Customizer Settings** panel (CCQPSET).

For more information about the parameters in this section, see <u>"Data sets that Tools Customizer uses</u> during customization" on page 224

Product Customization Settings						
Parameter	Sample or default value	Your value				
Customization library qualifier The high-level qualifier that is used as the prefix for the customization library.	DB2TOOL.PRODUCT.CUST					
Use DB2 group attach Determines the value that is used in the CONNECT statements in the generated customization jobs.	YES					

Tools Customizer Library Settings					
Parameter	Sample or default value	Your value			
Metadata library The name of the data set that contains the metadata that is used to display the Db2 and LPAR parameters.	DB2TOOL.CCQ110.SCCQDENU				
Discover output data set The name of the data set in which the output from the a product or a component Discover EXEC is stored.	DB2TOOL.CCQ110.DISCOVER				
Data store data set The name of the data set where Tools Customizer stores information about product or component, LPAR, and Db2 parameter values.	DB2TOOL.CCQ110.DATASTOR				

User Job Card Settings for Customization Jobs						
Parameter	Sample or default value	Your value				
The job card information to be inserted into the generated jobs for customizing a product or component.	The job statement information from the ISPF Batch Selection panel.					

Metadata library for Db2 SQL Performance Analyzer

Description

Use the following worksheet to identify and record the value of the metadata library for Db2 SQL Performance Analyzer. During the customization process, you will enter this value on the **Specify the Metadata Library** panel (CCQPHLQ).

Db2 SQL Performance Analyzer metadata library						
Parameter	Sample or default value	Discovered?	Your value			
Metadata library The name of the Db2 SQL Performance Analyzer metadata library.	hlq.SANLDENU	No				

Customization values for the Discover EXEC

Complete this worksheet only if you are recustomizing a product that has previously been customized by using Tools Customizer.

Description

Use the following worksheet to identify and record the customization values for the Tools Customizer Discover EXEC. During the customization process, you will enter these values on the **Discover Customized Product Information** panel (CCQPDSC).

Discover EXEC for Extracting Information from an Already Customized Product					
Parameter	Sample or default value	Your value			
Discover EXEC library The fully qualified data set name that contains the product Discover EXEC. For more information, see <u>"Data sets</u> <u>that Tools Customizer uses during</u> customization" on page 224.	<i>hlq</i> .SANLCLST				
Discover EXEC name The name of the Discover EXEC.	ANLCDISC				

Discover EXEC for Extracting Information from an Already Customized Product (continued)						
Parameter	Sample or default value	Your value				
Discover output data set The fully qualified name of the data set for the output from the product Discover EXEC. For more information, see <u>"Data sets</u> that Tools Customizer uses during customization" on page 224.	The name of the Discover output library that you entered on the Tools Customizer Settings (CCQPSET) panel.					

Information for Discover EXEC					
Parameter	Sample or default value	Your value			
SQL PA high-level qualifier from the previous customization of SQL PA Enter the SQL PA high-level qualifier from the previous customization of SQL PA.	SYSX.ANL510				

Product to Customize section

Description

The parameters that are listed in the Product to Customize section are read-only. They contain information that was provided on other panels, by Tools Customizer, or by the SQL PA metadata data set.

Read-only fields in the Product to Customize section on Tools Customizer panels				
Parameter	Discovered?	Source of this value		
Product metadata library This value is the library that you specified on the Specify the Product to Customize panel. This field is scrollable. Place your cursor anywhere on the field and press PF11 to view its full contents.	No	This value is specified on the Specify the Metadata Library panel (CCQPHLQ).		
LPAR The LPAR field displays the LPAR on which you are customizing Db2 SQL Performance Analyzer.	No	This value is provided by Tools Customizer.		
Product name This value displays the product that is being customized. In this example, Db2 SQL Performance Analyzer should be displayed in this field. This field is scrollable. Place your cursor anywhere on the field and press PF11 to view its full contents.	No	This value is provided by the product metadata file.		
Version The Version field displays the version, release and maintenance of the product you are customizing in the format Vn.Rn.nn.	No	This value is provided by the product metadata file.		
Product customization library This value displays the name of the data set in which the generated library customization jobs will be stored.	No	This value is derived from the user-specified customization library qualifier on the Tools Customizer Settings panel (CCQPSET).		

Required parameters section

Description

The parameters in this task are required for all customizations. During the customization process, you will enter these values on the Product Parameters panel (CCQPPRD).

Parameters in the Required parameters section on the Product Parameters panel (CCQPPRD)					
Parameter Required? Discovered? Default value Your value					
HLQ of Db2 SQL Performance Analyzer installation The high-level qualifier of the SQL PA libraries.	Yes	Yes	SQLPA.ANL510		

Task: Create files needed for TSO and batch execution

Description

Sets up the necessary files to be able to invoke SQL PA through TSO or batch.

This task is required.

Jobs generated

The steps in this task generate the following customization jobs:

- ANLDRO*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLDROPD template and is in the *job_sequence_number*DROPDB2_entry_ID member.
- ANLDCR*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLDCRE8 template and is in the *job_sequence_number*DCREDB2_entry_ID member.
- ANLPCR*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLPCRE8 template and is in the *job_sequence_number*PCRE*DB2_entry_ID* member.
- ANLICR*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLICRE8 template and is in the *job_sequence_number*ICRE*DB2_entry_ID* member.
- ANLJCR*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLJCRE8 template and is in the *job_sequence_number*JCRE*DB2_entry_ID* member.
- ANLBND*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLBNDGR template and is in the *job_sequence_number*BNDG*DB2_entry_ID* member.

Required authority

The following list indicates the job and the type of authority that is required to run the job:

ANLDROab job The user ID must have Db2 system administrator authority to use the customization ID.

ANLDCRab job

The user ID must have Db2 system administrator authority to use the customization ID.

ANLPCRab job

The required authority depends on the following factors:

- To create the *hlq*.SANLPARM and *hlq*.SANLJCL libraries, the user ID must have the authority to create them by using the customization library high-level qualifier.
- If the **Copy stored procedure load modules** step in the **Set up stored procedures** task is selected, the user ID must have the authority to create libraries by using the installation high-level qualifier and the customization library high-level qualifier.
- If the **Create SANLVCLS** step in the **Set up ANLCSPA for use from Browse or Edit sessions** task is selected, the user ID must have the authority to create libraries by using the installation high-level qualifier.

ANLICRab job

The user ID must have the authority to create members in the installation libraries and the customization libraries.

ANLJCRab job

The user ID must have the authority to create members in the customization libraries.

ANLBNDab job

The user ID must have Db2 system administrator authority to use the customization ID.

Steps and parameters for the Create files needed for TSO and batch execution task				
Step or parameter	Required?	Discovered?	Default value	Your value
Drop Db2 objects Drops Db2 objects from a previous customization or SQL PA version.	No	No	Not selected	
Create files, create Db2 objects, and perform binds and grants Sets up the parameter files and the initialization files needed to start SQL PA. As part of that process, it creates the SANLJCL library if necessary, creates the Db2 objects, and performs the necessary binds and grants for the SQL PA packages and plans.	Yes	No	Selected	
Prefix for output reports and work files This value sets the prefix to be used for the output reports and work files. The default value is your TSO user ID. If multiple sessions of SQL PA are to be run concurrently, the prefix must be unique for each session, which can be established by incorporating the system variable, ZSCREEN, into the value.	Yes	Yes	The default value is your TSO user ID.	
High-level qualifier of DB2 Path Checker The PATHLIB system parameter indicates the high- level qualifier of the libraries in which DB2 Path Checker is installed. If you do not have DB2 Path Checker installed, accept the default value.	Yes	Yes	SYSX.DB2PATHC	

Task: Create the generic plan tables

Description

Creates the generic plan tables for use by all users.

This task is optional.

Jobs generated

This task generates the following jobs:

- ANL3@Aab, where ab are alphanumeric characters assigned by Tools Customizer. This job is based on the ANL3@ATH template and is in the *job_sequence_number*3@ATDB2_entry_ID member.
- ANL3@Sab, where ab are alphanumeric characters assigned by Tools Customizer. This job is based on the ANL3@SGN template and is in the *job_sequence_number*3@SGDB2_entry_ID member.

Required authority

The user ID that runs the ANL3@Aab and the ANL3@Sab jobs must have Db2 system administrator authority to use the customization ID.

Steps and parameters for the Create the generic plan tables task					
Step or parameter	Required?	Discovered?	Default value	Your value	
Create new DSN3@ATH Creates the modified DSN3@ATH load module.	Yes	No	Selected		
Create new DSN3@SGN Creates the modified DSN3@SGN load module.	No	No	Not selected		

Task: Set up interface to QMF

Complete this worksheet only if you intend to use SQL Performance Analyzer with DB2 QMF.

Description

Sets up the QMF governor exit that is used with SQL Performance Analyzer.

Jobs generated

This task generates the following jobs:

- ANLGVQ*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLGVQMF template and is in the *job_sequence_number*GVQMDB2_entry_ID member.
- ANLXTQ*ab*, where *ab* are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLXTQMF template and is in the *job_sequence_number*XTQM*DB2_entry_ID* member.
- ANLQMFCL. This job is based on the ANLQMFCL template and is in the *job_sequence_number*QMFCL member.

Required authority

The following list indicates the type of authority that is required to run the job:

ANLGVQab job

The user ID must have Db2 system administrator authority to use the customization ID.

ANLXTQab job

The user ID must have Db2 system administrator authority to use the customization ID.

ANLQMFCL job

The user ID must have the authority to use the installation high-level qualifier.

Steps and parameters for the Set up interface to QMF task							
Step or parameter	Required?	Discovered?	Default value	Your value			
Create new ANLQMF and DSQUEGV1 Creates the new ANLQMF load module and creates the modified DSQUEGV1 load module.	Yes	No	Selected				
QMF user library The data set name of the QMF user library.	Yes	Yes	SYS3.SDSQUSRE				
QMF load library The data set name of the QMF load library.	Yes	Yes	SYS3.SDSQLOAD				
Create ANLCDSQ1 Creates ANLCDSQ1 that can be used to invoke QMF.	No	No	Not selected				
QMF ISPF panel library The data set name of the QMF ISPF panel library.	No	Yes	SYS3.SDSQPLBE				
QMF ISPF message library The data set name of the QMF ISPF message library.	No	Yes	SYS3.SDSQMLBE				
QMF ISPF skeleton library The data set name of the QMF ISPF skeleton library.	No	Yes	SYS3.SDSQSLBE				
QMF REXX execution library The data set name of the QMF REXX execution library.	No	Yes	SYS3.SDSQEXCE				
QMF CLIST library The data set name of the QMF CLIST library.	No	Yes	SYS3.SDSQCLTE				
QMF panel library The data set name of the QMF panel library.	No	Yes	SYS3.SDSQPNLE				
QMF GDDM map group The data set name of the GDDM map group.	No	Yes	SYS3.DSQMAPE				

Steps and parameters for the Set up interface to QMF task (continued)							
Step or parameter	Required?	Discovered?	Default value	Your value			
QMF GDDM chart group The data set name of the GDDM chart group.	No	Yes	SYS3.DSQCHART				
QMF GDDM data The data set name of the GDDM data.	No	Yes	SYS3.ADMCDATA				
QMF GDDM form definitions The data set name of the GDDM form definitions.	No	Yes	SYS3.ADMGDF				
QMF GDDM symbols The data set name of the GDDM symbols.	No	Yes	SYS3.ADMSYMBL				
QMF output print file The data set name of the QMF print file.	No	Yes	SYS3.ADMPRINT. REQUEST.QUEUE				
QMF GDDM definitions The data set name of the GDDM definitions.	No	Yes	SYS3.GDDMDEF1				

Task: Set up interface to Query Monitor

Complete this worksheet only if you intend to use SQL Performance Analyzer with IBM DB2 Query Monitor.

Description

Creates the CLIST necessary to interface with Query Monitor.

Jobs generated

This task generates the ANLQMONC job. This job is based on the ANLQMONC template and is in the *job_sequence_number*QMONC member.

Required authority

The user ID that runs the ANLQMONC job must have the authority to use the installation high-level qualifier.

Steps and parameters for the Set up interface to Query Monitor task						
Step or parameter	Required?	Discovered?	Default value	Your value		
Create ANLCIQM Creates the ANLCIQM CLIST that is used by Query Monitor to invoke SQL PA.	Yes	No	Selected			

Task: Set up stored procedures

Complete this worksheet only if you intend to use the SQL PA stored procedures..

Description

Sets up the SQL PA stored procedures.

Jobs generated

This task generates the following customization jobs:

- ANLCLODS. This job is based on the ANLCLODS template and is in the job_sequence_numberCLODS member.
- ANLWLMab, where ab are alphanumeric characters assigned by Tools Customizer. This job is based on the ANLWLMSP template and is in the *job_sequence_number*WLMSDB2_entry_ID member.

Required authority

ANLCLODS

The user ID must have the authority to use the installation high-level qualifier.
ANLWLM*ab*

The user ID must have Db2 system administrator authority to use the customization ID.

Steps and parameters for the Set up stored procedures task				
Step or parameter	Required?	Discovered?	Default value	Your value
Copy stored procedure load modules Copies the stored procedure load modules from the SANLLOAD library to the SANLLODS library.	Yes	No	Selected	
Set up the stored procedure using RRSAF Sets up the stored procedure to use RRSAF.	No	No	Not selected	

Task: Set up ANLCSPA for use from Browse or Edit sessions

Complete this worksheet only if you intend to set up ANLCSPA for use from Browse or Edit sessions. ANLCSPA is a special edit macro that allows you to selectively pick a portion of the file for analysis, choosing one or several statements.

Description

Sets up ANLCSPA for use from Browse or Edit sessions.

Jobs generated

This task generates the following customization jobs:

- ANLCVCLS. This job is based on the ANLCVCLS template and is in the *job_sequence_number*CVCLS member.
- ANLCCSPA. This job is based on the ANLCCSPA template and is in the *job_sequence_number*CCSPA member.

Required authority

The user ID that runs the ANLCVCLS and ANLCCSPA jobs must have the authority to use the installation high-level qualifier and the customization library high-level qualifier.

Steps and parameters for the Set up ANLCSPA for use from Browse or Edit sessions task					
Step	Required?	Discovered?	Default value	Your value	
Create SANLVCLS Creates a variable block library for the CLISTs and copies the members.	No	No	Not selected		
Tip: Do not mix the formats of the libraries in SYSPROC. If your SYSPROC allocation uses libraries with variable block formatting, select this step. ANLCALLC will allocate the appropriate library, and, if a mismatch occurs, a message will be issued and the libraries will not be allocated.					
Copy ANLCSPA to SYSPROC library Copies ANLCSPA to a library that is allocated to SYSPROC.	No	No	Not selected		
SYSPROC library for ANLCSPA Specify the data set name of the SYSPROC library to which ANLCSPA is to be copied.	No	Yes	SYSX.SYSPROC		

Task: Add SQL PA to the IBM DB2 Admin Launchpad

Complete this worksheet only if you intend to add SQL PA to the IBM DB2 Admin Launchpad.

Description

Adds SQL PA to the IBM DB2 Admin Launchpad.

Jobs generated

ANLLCHPD. This job is based on the ANLLCHPD template and is in the *job_sequence_number*LCHPD member.

Required authority

The user ID that runs the ANLLCHPD job must have the authority to run the ADBDMTI EXEC in DB2 Admin.

Steps and parameters for the Add SQL PA to the IBM DB2 Admin Launchpad task					
Step or parameter Required? Discovered? Default value Your value					
Add SQL PA to Launchpad Adds SQL PA to the IBM DB2 Admin Launchpad.	Yes	No	Selected		
HLQ of DB2 Admin installation The high-level qualifier of the DB2 Admin libraries.	Yes	Yes	SYSX.DB2ADMI		

LPAR Parameters section

Description

This section contains LPAR parameters. All parameters are required. During the customization process, you will enter these values on the **LPAR Parameters** panel (CCQPLPR).

Parameter	Doguirod2	Discovered	Default value	Vourvalue
Message library The data set name of the ISPF message library. You can specify multiple values for this parameter.	Yes	Yes	ISPF.SISPMLIB	
Panel library The data set name of the ISPF panel library. You can specify multiple values for this parameter.	Yes	Yes	ISPF.SISPPLIB	
Skeleton library The data set name of the ISPF skeleton library. You can specify multiple values for this parameter.	Yes	Yes	ISPF.SISPSLIB	
ISPF table input library The data set name of the ISPF table input library. You can specify multiple values for this parameter.	Yes	Yes	ISPF.SISPTLIB	
Link list library The data set name of the link list library. You can specify multiple values for this parameter.	Yes	No	ISP.SISPLOAD	
Command procedures library The data set name of the ISPF command procedures library. You can specify multiple values for this parameter.	Yes	Yes	SYS1.PROCLIB	

Language Environment Libraries				
Parameter	Required?	Discovered?	Default value	Your value
Language environment high-level qualifier The high-level qualifier of the language environment library.	Yes	Yes	CEE	

DB2 Parameters section

Description

This section contains DB2 parameters. All parameters are required. During the customization process, you will enter these values on the **DB2 Parameters** panel (CCQPDB2).

General Db2 Information					
Parameter	Required?	Discovered?	Default value	Your value	
Mode The mode in which the Db2 subsystem is running. The following values are valid:	Yes	Yes	NFM		
 CM is compatibility mode on all listed Db2 versions except Db2 10. 					
 CM8 is conversion mode from Db2 V8 on Db2 10. 					
 CM9 is conversion mode from Db2 Version 9.1 on Db2 10. 					
 NFM is new-function mode on all listed Db2 versions. 					
 Level number The version, release, and modification level of the Db2 subsystem. The following values are valid: 101 is valid only for CM8, CM9 or NFM. 111 is valid only for CM or NFM. 	Yes	Yes	blank		

Db2 Libraries					
Parameter	Required?	Discovered?	Default value	Your value	
Load library The data set name of the Db2 load library. You can specify multiple values for this parameter.	Yes	Yes	DSN.SDSNLOAD		
Run library The data set name of the Db2 run library. You can specify multiple values for this parameter.	Yes	Yes	DSN.RUNLIB.LOAD		
Exit library The data set name of the Db2 exit library. You can specify multiple values for this parameter.	Yes	Yes	DSN.SDSNEXIT		
Sample library The data set name of the Db2 sample library. You can specify multiple values for this parameter.	Yes	Yes	DSN.SDSNSAMP		
Macro library The data set name of the Db2 macro library. You can specify multiple values for this parameter.	Yes	Yes	DSN.SDSNMACS		

Db2 Utilities				
Parameter	Required?	Discovered?	Default value	Your value
Plan name for the DSNTIAD utility The plan name for the DSNTIAD utility. The value must be 8 characters or less.	Yes	Yes	DSNTIAD	

SQL PA Db2 Objects				
Parameter	Required?	Discovered?	Default value	Your value
Storage group name The name of the storage group that will be used for creating Db2 objects for customization. The value must be 128 characters or less.	Yes		SYSDEFLT	
Database name The name of the database in which Db2 objects will be created for customization. The value must be 8 characters or less.	Yes	No	ANLDBASE	
Table space nameThe name of the table space in which Db2 objectswill be created for customization. The value mustbe 24 characters or less.	Yes	Yes	ANLSPC08	
Database buffer pool name The name of the buffer pool to be used for the database during customization. The value must be 6 characters or less. The default is blank, which means that Db2 will choose the buffer pool.	No	No		
Index buffer pool name The name of the buffer pool to be used for indexes during customization. The value must be 6 characters or less. The default is blank, which means that Db2 will choose the buffer pool.	No	No		
Table space buffer pool nameThe name of the buffer pool to be used for thetable space during customization. The value mustbe 6 characters or less. The buffer pool size mustbe larger than 4K. If the value is blank, BP8K0 willbe used.	No	No	BP8K0	

SQL PA Db2 General Information					
Parameter	Required?	Discovered?	Default value	Your value	
Drop database when dropping DB2 objects To drop the SQL PA database when all SQL PA Db2 objects are dropped, specify YES. To drop all SQL PA Db2 objects except the database, specify NO.	Yes	Yes	YES		
Proc name for WLM application environment Specify the name of the procedure for WLM application environment.	Yes	Yes	DSN8WLMR		
Name of WLM APPLENV for stored procedure Specify the name of the WLM application environment to use for the stored procedure.	Yes	Yes	WLMENVR		
(DATAGRP) Group authorized for table data The user group that has the authority to view user table data, such as HIGH2KEY and LOW2KEY. If the value is +0FF+, all users will be able to view the user table data.	No	Yes	+OFF+		
(DSGROUP) Number of members in data sharing group The DSGROUP system parameter indicates the number of members in the data sharing group. This parameter is used to adjust the overall processing power of the system to reflect excess processing time that is incurred by data sharing operations. The result is a more accurate cost assessment.	Yes	Yes	0		

SQL PA Db2 General Information (continued)				
Parameter	Required?	Discovered?	Default value	Your value
 (BUFFHIT) Buffer hit percentage The BUFFHIT user parameter indicates the ratio of physical to logical I/O, expressed as a percentage from 0 to 100. Specify a value for the percentage of pages that are found in the buffer pool (that is, for which disk I/O is not necessary). For example, specifying 20 means that 20% of the time the required pages are found in the buffer pool, whereas 80% of the time a physical I/O to disk is required. Specifying 000 means that 100% of time the I/O goes to disk, and no pages are assumed to be in the pool 	Yes	Yes	0	
(BUFFERS) The number of 4 KB buffers in 4 KB buffer pool Set the BUFFERS system parameter to the total number of 4 KB buffers in the 4 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Specify the most prominent 4 KB buffer pool here, even if several are used.	Yes	Yes	2000	
(BUFF08K) The number of 8 KB buffers in 8 KB buffer pool Set the BUFF08K system parameter to the total number of 8 KB buffers in the 8 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Specify the most prominent 8 KB buffer pool here, even if several are used.	Yes	Yes	500	
(BUFF16K) The number of 16 KB buffers in 16 KB buffer pool Set the BUFF16K system parameter to the total number of 16 KB buffers in the 16 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Specify the most prominent 16 KB buffer pool here, even if several are used.	Yes	Yes	250	
(BUFF32K) The number of 32 KB buffers in 32 KB buffer pool Set the BUFF32K system parameter to the total number of 32 KB buffers in the 32 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Specify the most prominent 32 KB buffer pool here, even if several are used.	Yes	Yes	100	

SQL PA Installation Parameters					
Parameter	Required?	Discovered?	Default value	Your value	
Customization ID to use for set up The DB2 user ID that is to be used to customize SQL PA. This ID owns the SQL PA objects that are created for the customization and must have SYSADM authority.	Yes	Yes	ANLUSERO		

SQL PA Installation Parameters (continued)					
Parameter	Required?	Discovered?	Default value	Your value	
Customized library HLQ The high-level qualifier of the SQL PA customized libraries of SANLDATA, SANLJCL, and SANLPARM. If the parameter is not specified, it defaults to the high-level qualifier of the SQL PA installation.	Yes	Yes	SQLPA.ANL510		
Customized library volume The name of the volume of SANLPARM, SANLJCL, SANLDATA, SANLLODS and SANLVCLS customized libraries. The default is blank, which means that SMS will select the volume.	Yes		blank		
Load library lower level qualifier This parameter indicates the lower level of qualification, if the load modules are copied to a library in the target libraries other than the default library of SANLLOAD. The default is SANLLOAD.	Yes	Yes	SANLLOAD		
Secondary authids for SQL PA Specifies the optional secondary authorization IDs to use as the owners of the generic plan tables.	Yes	Yes	blank		
QMF user ID If you have QMF installed and plan to process QMF statements in SQL PA (option 1.7) specify the owner of the QMF DB2 tables. If you do not have QMF installed, leave the parameter value blank.	Yes	Yes	blank		
(USRPARM) Allow users to have a copy of the user parameters The USRPARM system parameter authorizes the use of the user-level user parameters.	Yes	No	NO		
 (CONNECT) Attach facility The CONNECT user parameter specifies the expected application connection to Db2 when the SQL is started in production. This value helps SQL PA account for excess processing that is caused by various attach facilities, including thread management and application processing. SQL Performance Analyzer lists the total MIPS rate for a processor, and the calculated speed per engine, which is the number that is used for most CPU computations. Db2 SQL Performance Analyzer uses the (CONNECT) Attach facility parameter to determine which attachment facility has been chosen to connect to Db2. The Detail Trace Report includes information about what Db2 SQL Performance Analyzer assumes is the processor usage for that connection, which is included once for each query issued. 	Yes	Yes	CAF		

SQL PA Installation Parameters (continued)					
Parameter	Required?	Discovered?	Default value	Your value	
(STORAGE) Storage device The STORAGE user parameter reflects the most prominent DASD storage medium for an application's databases. If your system uses several device types, select the device that holds most of your databases. The default value is 3390-3.	Yes	Yes	3390-3		
Valid values for disk device types are 3390, 3390-1, 3390-2, 3390-3, 3390-9, 6390, 6390-3, 6390-5, 6390-9, 7390-1, 7390-2, 7390-3, or 7390-F.					
Valid values for solid-state devices are 4305, 4380, 6110, 6110-1, 6110-2, 6110-4, 6680-1, 7900, ICEBRG, STK, EMCX, EMCBAR, EMCELT, RAMAC, RAMAC2, RAMAC3, ESSE20, ESSF20, ESS800, and ESS80T (turbo).					
If several device types are on your system, specify the device that holds most of your databases.					
(NEWSTOR) Newly defined storage device The NEWSTOR user parameter indicates that you are defining a new storage device. Although a number of device types are already defined, including ESS, RAMAC, EMC, ICEBERG, and other solid state devices, you can define your own device type by using this parameter. Follow this parameter with the NEWSEEK, NEWROTO, and NEWXFER parameters to define the new device. Either this parameter or a STORAGE NEWDSK setting indicates that new disk storage is being defined.	Yes	Yes	blank		
(NEWROTA) Newly defined rotational delay The NEWROTA user parameter specifies the average rotational delay (half of a revolution, also called latency) in seconds for the definition of a new storage device. This parameter is effective only when the NEWSTOR parameter is also set or when the STORAGE user parameter is set to NEWDSK.	Yes		blank		
(NEWSEEK) Newly defined seek time The NEWSEEK user parameter specifies the average seek time in seconds for the definition of a new storage device. This parameter is effective only when the NEWSTOR parameter is also set or when the STORAGE user parameter is set to NEWDSK.	Yes	Yes	blank		
(NEWXFER) Newly defined transfer rate The NEWXFER user parameter specifies the average transfer rate of the device over the channel in kilobytes per second. This parameter is effective only when the NEWSTOR parameter is also set or when the STORAGE parameter is set to NEWDSK.	Yes	Yes	blank		
SQL PA Batch Processing Parameters					

Parameter	Required?	Discovered?	Default value	Your value
First line of the jobcard of batch JCL Specify the first of four lines that will be used in the jobcard when you generate batch JCL.	Yes	Yes	//ANLBATCH JOB ,	

SQL PA Batch Processing Parameters (continued)				
Parameter	Required?	Discovered?	Default value	Your value
Second line of the jobcard of batch JCL Specify the second of four lines that will be used in the jobcard when you generate batch JCL.	Yes	Yes	// USER=&SYSUID,NOTIFY =&SYSUI	
Third line of the jobcard of batch JCL Specify the third of four lines that will be used in the jobcard when you generate batch JCL.	Yes	Yes	// MSGLEVEL=(1,1), REGION=0M,T	
Fourth line of the jobcard of batch JCL Specify the fourth of four lines that will be used in the jobcard when you generate batch JCL.	Yes	Yes	/*JOBPARM S=*	
(QUALIFY) Default qualifier for objects The QUALIFY user parameter specifies the qualifier that will be used to qualify unqualified objects in an SQL statement. If no value is specified, the current SQL ID will be used.	Yes	Yes	blank	

SQL PA Cost Parameters				
Parameter	Required?	Discovered?	Default value	Your value
(CPUCOST) Cost of one hour of CPU time Set the CPUCOST system parameter to the cost of one hour of CPU time, in the national currency, for this configuration. This parameter is used by SQL PA for charge back and financial cost determination.	Yes	Yes	500	
(IOSCOST) Cost of 1000 I/O calls Set the IOSCOST system parameter to the cost of 1000 I/O calls, in the national currency, for this configuration. This parameter is used by SQL PA for charge back and financial cost determination.	Yes	Yes	10	
(TIMCOST) Cost of one hour of connect time Set the TIMCOST system parameter to the cost of one hour of connect time, in the national currency, for this configuration. This parameter is used by SQL PA for charge back and financial cost determination.	Yes	Yes	10	
(MONEYIS) National currency Set the MONEYIS system parameter to the national currency, in descriptive form, such as DOLLARS, POUNDS, STERLING, DRACHMA, or KRONA.	Yes	Yes	DOLLARS	
(CURRSYM) Currency symbol Set the CURRSYM system parameter to the single character symbol that represents the national currency that corresponds to the MONEYIS parameter.	Yes	Yes	\$	

SQL PA EXPLAIN Parameters				
Parameter	Required?	Discovered?	Default value	Your value
(SETPLAN) Allow users to set the plan table owner The SETPLAN system parameter determines whether SQL PA uses the generic plan tables or the plan tables that are qualified by USEPLAN.	Yes	Yes	NOTALLOW	
 ALLOW SQL PA users are authorized to use USEPLAN. 				
 NOTALLOW (Default) SQL PA users are not authorized to use USEPLAN. In this case, the generic IDs are used to qualify plan tables. If no generic ID is available, the user's TSO ID is used to qualify the plan table. 				
 GENERIC Only generic IDs are used to qualify the plan tables. 				
Note: In previous releases, YES and NO were used for the SETPLAN parameter and will still be processed if already in place. Starting with V5.1, only the ALLOW, NOTALLOW, and GENERIC values will be accepted.				
(KEEPLAN) Retain plan table information The KEEPLAN user parameter indicates whether plan records that are owned by the value of the USEPLAN parameter are retained when SQL PA runs in batch. SQL PA deletes all plan records at the conclusion of each run, and it deletes all plan records with QUERYNO greater than 100 M on startup. If this parameter is set to YES, the deletion of plan records in your permanent plan tables is prevented. This parameter does not affect online operation, and it does not affect the generic plan tables.	Yes	Yes	NO	
(PRECISE) User DB2 optimizer estimates The PRECISE user parameter provides an extra level of precision to the cost estimate. If this parameter is set to YES, SQL PA replaces its internal cost estimate and QUNIT result with the values from Db2. It also uses the optimizer's path length in calculations. If this parameter is set to ALL, the cost estimate includes additional factors and the Predicate Analysis report is generated.	Yes	Yes	YES	
(NLSCODE) DBCS or mixed data names in SQL The NLSCODE user parameter enables the processing of SQL statements that include DBCS or mixed data names, such as those that are found in the Korean, Japanese, and Chinese languages. If this parameter is set to +OFF+, SQL is converted to uppercase before processing. This translation might alter the DBCS and mixed data names. The lowercase-to-uppercase translation is not performed with any other setting.	Yes	Yes	+OFF+	
(PROCESS) DBRM sequence numbers out of order Set the PROCESS user parameter to NOSEQ if your DBRM was created or modified by third- party software products that have caused the sequence numbers of the statements to be out of order. Without this setting, message ANL2003E will be issued because of synchronization errors that result from the numbers being out of sequence.	Yes	Yes	+OFF+	

SQL PA EXPLAIN Parameters (continued)					
Required?	Discovered?	Default value	Your value		
Yes	Yes	ONE			
Yes	Yes	NO			
Yes	Yes	+OFF+			
Yes	Yes	NO			
Yes	Yes	NO			
	Required? Yes Yes	Required?Discovered?YesYesYesYesYesYesYesYesYesYesYesYesYesYes	Required?Discovered?Default valueYesYesONEYesYesNOYesYesNOYesYes+OFF+YesYesNOYesYesNO		

SOL PA	Report	Parameters
JŲLFA	Report	Falameters

Parameter	Required?	Discovered?	Default value	Your value
(ACCTYPE) Plan report access type filter Set the ACCTYPE user parameter to filter the statements of plans and packages for the Explain Explanation report according to the types of access path. Specify ALL or A to show all statements. Use HASH or H to show only statements that use a hash access path. Specify MATCHING or M to show only statements that use a matching index scan access path. Specify NONMATCH or N to show only statements that use a non-matching index scan access path. Specify TABSCAN or T to show only statements that use a table space scan access path.	Yes	Yes	ALL	
(FORMAT) Format SQL in plan report The FORMAT user parameter indicates how Explain formats the SQL statements of the plans and packages. Set this parameter to YES if you want each SQL keyword, such as SELECT, INTO, FROM, and WHERE, to start on a new line. Set this parameter to NO if you want only certain keywords, such as SELECT and UNION, to start on a new line. Setting this parameter to NO consumes less page space.	Yes	Yes	YES	

SQL PA Report Parameters (continued)				
Parameter	Required?	Discovered?	Default value	Your value
(HOSTVAR) Host variables in plan report The HOSTVAR user parameter indicates whether Explain lists the host variable information of the plans and packages. If this parameter is set to YES, the Explain report lists up to 100 host variables and their definitions. If this parameter is set to NO, no host variable definitions are listed in the Explain report.	Yes	Yes	NO	
(INDEX) Plan report index level The INDEX user parameter controls the level of index information in the Explain report. When this parameter is set to ALL or CONDENSE, information for all indexes that were created for the table is shown. When this parameter is set to NO, no index information is shown. When this parameter is set to TABLESPACE or YES, detailed index information is shown.	Yes	Yes	NO	
(PLANINF) Plan report plan info level The PLANINF user parameter controls the printing of the plan and DBRM/package report after the first miniplan in a plan of the Explain report. If this parameter is set to YES, the plan and DBRM/ package report are printed if specified in the authorization table. If this parameter is set to MINI, a short report that shows the plan parameters is printed. If this parameter is set to NO, a plan or DBRM/package report is not printed.	Yes	Yes	YES	
 (LEVEL) Plan report format The LEVEL user parameter controls the level of information in the plan table report. DETAIL shows a detailed report. SQLSUMMARY shows only the SQL statement and access path messages. SUMMARY shows only the summary page with one line for each SQL statement. TABULAR shows a brief report with plan table information in tabular format. For more information, see the following options: ACCESSPATH, ALLINDEX, KEY, NOCATALOG, NOKEYS, TINO, TIYES, XOFF, and XON.	Yes	Yes	DETAIL	
(TURNOFF) Exclude messages from output reports The TURNOFF user parameter excludes specified messages from the SQL PA output. Specify the 4- digit numerical message number to turn off the message. You can enter as many TURNOFF parameters as you want, but you must enter each one on a separate line. The default is to show all messages.	Yes	Yes	blank	
(REPORTS) Level of SQL PA reports The REPORTS user parameter specifies the level of reporting. If this parameter is set to LOG, the cost report will be generated, which provides a summary of query cost plus any warning messages. If this parameter is set to EXP, the Enhanced Explain report is generated with the cost report. If this parameter is set to DET or ALL, the detailed trace information is generated with the cost report and the Enhanced Explain report.	Yes	Yes	ALL	

SQL PA Report Parameters (continued)	SQL PA Report Parameters (continued)					
Parameter	Required?	Discovered?	Default value	Your value		
(SHOWALT) Show alternate indexes Set the SHOWALT user parameter to YES to display information about alternate indexes that are not selected by the optimizer in the Enhanced Explain and detail trace reports. The information is presented for access paths of I, N, and R only, and is not presented for multiple index or direct access using row ID. The information includes pertinent catalog statistics about the size and shape of each alternate index, and information about the index key's columns.	Yes	Yes	NO			
(QLIMSQL) Display SQL error in query limits report Set the QLIMSQL user parameter to YES to display an SQL statement that received a negative SQLCODE in the QLIMITS report.	Yes	Yes	NO			
(OBJECTS) Display tables and indexes in SQL report The OBJECTS user parameter determines whether the list of objects (tables and indexes) that are used will be included in the SQL report. If this parameter is set to YES, the list of objects is displayed. If this parameter is set to NO, the list of objects is not displayed.	Yes	Yes	NO			
(DSPPARM) Display parameters in output reports The DSPPARM user parameter determines whether the SQL PA parameters and their values are displayed at the top of the enhanced explain and detail trace reports. If this parameter is set to YES, the parameters are displayed. If this parameter is set to NO, the parameters do not display.	Yes	Yes	YES			
(DSPVARS) Display host variables in SQL The DSPVARS user parameter determines whether the host variable names or parameter markers are displayed in SQL statements in the cost reports. If this parameter is set to YES, the host variables are displayed. If this parameter is set to NO, the parameter markers are displayed.	Yes	Yes	NO			
(NUMBERS) Input file created with NUMBERS ON The NUMBERS user parameter specifies whether the input file was created with NUMBERS ON.	Yes	Yes	YES			

SQL PA Message Handling Parameters					
Parameter	Required?	Discovered?	Default value	Your value	
(ADVISOR) ADVISOR level for output reports The ADVISOR user parameter displays additional information about the performance and design of each query and about the tables and indexes that are accessed by the query. Warnings and Alerts are always displayed, even when ADVISOR is set to NO. Notes and Recommendations are displayed when ADVISOR is set to YES, which can reveal potential performance tuning opportunities. A value of ALL is ideal for those who want to receive Guideline messages and confirmation that tasks have concluded successfully.	Yes	No	NO		

SQL PA Message Handling Parameters (continued)				
Parameter	Required?	Discovered?	Default value	Your value
(RETCODE) Return code reflects SQL PA warnings The RETCODE user parameter governs whether the return code reflects the warnings, error messages, or cost overruns from SQL PA. If this parameter is set to NO, the program return code reflects the normal processing return code of the SQL PA programs. If this parameter is set to YES, values of 4, 8, 12, or 16 are returned as program return codes from normal operation that encountered SQL PA messages.	Yes	Yes	NO	
(NOSTATS) Warn when statistics not set Set the NOSTATS user parameter to YES to be notified when the default was used for the statistics of any table, index, table space, or column because the catalog statistics were not set.	Yes	Yes	YES	

SQL PA Warning Message Enablement Parameters				
Parameter	Required?	Discovered?	Default value	Your value
(CPUTIME) Maximum CPU time The CPUTIME system parameter indicates the maximum CPU time, in seconds, that is permitted before the SQL statement is flagged as exceeding the installation's CPU time limit. The maximum value is 86400, which is equivalent to 24 hours. A setting of 0 results in unlimited CPU time.	Yes	Yes	10	
(COSTING) Maximum query cost The COSTING system parameter indicates the maximum monetary value, in your currency, that is permitted before the SQL statement is flagged as exceeding the installation's monetary limit. The maximum value is 999999999. A setting of 0 results in no monetary limit.	Yes	Yes	15	
(IOCALLS) Maximum I/O calls The IOCALLS system parameter indicates the maximum physical I/O calls that are permitted before the SQL statement is flagged as exceeding the installation's physical I/O limit. The maximum value is 999999999. A setting of 0 results in unlimited physical I/O.	Yes	Yes	1000	
(COSTQUN) Maximum query units The COSTQUN system parameter indicates the maximum number of QUNITS that are permitted before the SQL statement is flagged as exceeding the installation's QUNIT limit. The maximum value is 999999999. A setting of 0 results in no QUNIT limit.	Yes	Yes	200	
(ELAPSED) Maximum elapsed time The ELAPSED system parameter indicates the maximum elapsed time, in seconds, that is permitted before the SQL statement is flagged as exceeding the installation's elapsed time limit. The maximum value is 86400, which is equivalent to 24 hours. A setting of 0 results in unlimited elapsed time.	Yes	Yes	60	

SQL PA Warning Message Enablement Parameters (continued)				
Parameter	Required?	Discovered?	Default value	Your value
(INLISTS) Mximum number of IN (LIST) elements The INLISTS user parameter specifies the maximum number of elements that are shown in an IN (LIST) predicate. If the maximum number of elements is exceeded, SQL PA issues a notification message.	Yes	Yes	10	
(ISCANPG) Max IX leaf pages read (matching scan) The ISCANPG user parameter indicates the maximum number of index leaf pages that are read with a matching index scan. If the maximum number of pages is exceeded, SQL PA issues a notification message.	Yes		1000	
(NONIXPG) Max IX leaf pages read (non-match scan) The NONIXPG user parameter indicates the maximum number of index leaf pages that are read with a non-matching index scan. If the maximum number of pages is exceeded, SQL PA issues a notification message.	Yes	Yes	5000	
(IXUPDAT) Maximum indexes on table update The IXUPDAT user parameter indicates the maximum number of indexes that can exist on a table that is being updated. If the maximum number of indexes is exceeded, SQL PA issues a notification message.	Yes	Yes	5	
(JOINTAB) Maximum tables to be joined The JOINTAB user parameter indicates the maximum number of tables that can be joined together. If the maximum number of tables is exceeded, SQL PA issues a notification message.	Yes	Yes	10	
(MATCOLS) Minimum index columns matched The MATCOLS user parameter limits the number of columns in an index that are used in a matching index scan. The parameter value expresses the fraction of columns in the index key. For example, the default value of 0.50 states that at least half of the columns in the index key should be used in a matching index scan. If not, SQL PA will issue a notification message that indicates that the index is not being used to its fullest potential.	Yes	Yes	0.50	
(PREDPCT) Maximum predicate filtering percentage The PREDPCT user parameter limits the fraction of the number of total rows to be retained after predicate filtering. When this limit is exceeded, a notification message is issued that indicates that the predicates are not effectively restrictive. For example the default value of 0.15 states that if more than 15% of the rows pass filtering, SQL PA will issue a message.	Yes	Yes	0.15	
(TSCANPG) Maximum data pages for table space scan The TSCANPG user parameter indicates the limit of the number of data pages that are read with a table space scan. If the maximum number of pages is exceeded, SQL PA issues a notification message.	Yes	Yes	50000	

Starting and preparing Tools Customizer for use

Use the provided REXX EXEC to start Tools Customizer. The first time that you use Tools Customizer, you must modify the settings that Tools Customizer uses to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Starting Tools Customizer

Start Tools Customizer by running a REXX EXEC from the ISPF Command Shell panel.

Before you begin

Tools Customizer must be SMP/E installed. You must know the high-level qualifier of where the Tools Customizer libraries reside. The high-level qualifier is considered to be all the segments of the data set name except the lowest-level qualifier, which is SCCQEXEC.



Attention: Ensure that Tools Customizer load libraries are not APF authorized. APF authorizing Tools Customizer libraries results in an abend.

About this task

To run the REXX EXEC, you must either change the placeholder in the EXEC for the high-level qualifier of the Tools Customizer EXEC library or pass the high-level qualifier as a parameter when you run the EXEC. The REXX EXEC is in the CCQTCZ member of the EXEC library.

Procedure

1. Optional: Change the placeholder for the high-level qualifier in the REXX EXEC:

- a) Find the EXEC library data set for Tools Customizer. The name of the data set is *high_level_qualifier*.SCCQEXEC.
- b) Edit data set member CCQTCZ and replace the <TCZ HLQ> string with the high-level qualifier of the EXEC library data set.

For example, if the name of the Tools Customizer EXEC library is CCQTCZ.USABSAND.SCCQEXEC, replace <TCZ HLQ> with CCQTCZ.USABSAND.

You have to change the placeholder for the high-level qualifier only once. When you run the REXX EXEC, you do not have to pass the high-level qualifier as a parameter.

- 2. Run the REXX EXEC (CCQTCZ):
 - a) From the ISPF Primary Option Menu, select option 6.

The ISPF Command Shell panel is displayed.

b) Specify the EX command to run the REXX EXEC.
 For example, if the Tools Customizer EXEC library is CCQTCZ.USABSAND.SCCQEXEC and you changed the placeholder for the high-level qualifier in the REXX EXEC, specify:

EX 'CCQTCZ.USABSAND.SCCQEXEC(CCQTCZ)'

If you did not change the placeholder for the high-level qualifier in the REXX EXEC, specify:

EX 'CCQTCZ.USABSAND.SCCQEXEC(CCQTCZ)' 'CCQTCZ.USABSAND'

You can also specify a trace data set name and a user profile when you run the REXX EXEC.

• The default trace data set name is *USERID*.CCQ.TRACE. To specify a different trace data set name, append the trace data set name to the command. For example, to specify a trace data set name of CCQTCZ.MYTRACE, enter:

EX 'CCQTCZ.USABSAND.SCCQEXEC(CCQTCZ)' 'CCQTCZ.USABSAND, CCQTCZ.MYTRACE'

• To specify a user profile other than your own, append the user profile name to the command. For example, to specify a user profile of SHRPROF, enter:

CCQTCZ.USABSAND.SCCQEXEC(CCQTCZ)' 'CCQTCZ.USABSAND, ,CCQTCZ.SHRPROF'

Tools Customizer will use the settings from the specified profile. This profile will be updated when you exit Tools Customizer, but your own profile will remain unchanged.

Results

The IBM Customizer Tools for z/OS main menu panel is displayed.

What to do next

If you are running Tools Customizer for the first time, you must modify the Tools Customizer user settings. If you have already set the Tools Customizer user settings, either customize or recustomize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Modifying Tools Customizer user settings

Before you can customize Db2 SQL Performance AnalyzerDb2 Autonomics Director with Tools Customizer, you must review the settings that Tools Customizer uses. You might have to change the default values to suit your environment. In most cases, you can change the Tools Customizer values at any time. For example, after you have customized Db2 SQL Performance AnalyzerDb2 Autonomics Director and are customizing a different product or solution pack, you might have to change the settings.

Procedure

1. On the **IBM Tools Customizer for z/OS** main panel (CCQPHME), specify option 0, **User settings for Tools Customizer**.

The **Tools Customizer Settings** panel (CCQPSET) is displayed, as shown in the following figure:

```
CCOPSET
                Tools Customizer Settings
                                                  14:03:51
Command ===>
Enter the settings for customizing a product or press End to save and exit.
Commands: SAVE OPTIONS
Product Customization Settings
 *Customization library qualifier . . DB2TOOL.PRODUCT.CUST
 Tools Customizer Library Settings
 *Metadata library . . . . DB2TOOL.CCQ110.SCCQDENU
 *Discover output data set . DB2T00L.CCQ110.DISCOVER
 Volume serial . . . .
User Job Card Settings for Customization Jobs
===> //
              JOB
(ACCOUNT), 'NAME',
```

Figure 4. The Tools Customizer Settings panel (CCQPSET)

Note: An asterisk next to a field indicates that the field is required.

2. Review the values for the following required fields. Use the default value or specify your own value. You must have appropriate READ and WRITE access to the data sets that are specified.

Customization library qualifier

The high-level qualifier that is used as the prefix for the customization library. The customization library is a data set in which the generated jobs to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director are stored. WRITE access to this qualifier is required.

For each product to be customized, the first value that is specified for the qualifier is always used, even if you change it after you have generated the customization jobs. For example, if you customize a product and then specify a new qualifier for recustomization, although the new qualifier is saved and displayed, the original value is used.

To maintain multiple instances of Tools Customizer, specify a unique customization library qualifier for each instance of Tools Customizer.

Data set names that exceed 42 characters must be enclosed in single quotation marks (').

Volume serial

The volume name in which the customization library will reside. If you don't specify a volume name, it will be assigned by the system.

Use DB2 group attach

Determines the value that is used in the CONNECT statements in the generated customization jobs. Specify YES for data sharing environments, which causes the group attach name to be used. Specifying NO, in most cases, causes the SSID to be used in the Db2 CONNECT statement.

Important: This field has no effect when you are customizing a product on a Db2 subsystem that is not a member of a data sharing group. In this case, the Db2 subsystem ID (SSID) is always used in the CONNECT statements in the generated customization jobs.

When you are customizing a product on a Db2 subsystem that is a member of a data sharing group, how the Db2 subsystem is defined and the value of the **Use DB2 group attach** field determines the value that is used in the CONNECT statements in the generated jobs. The following table shows whether the SSID or the group attach name is used:

Db2 subsystem definition	Value of the Use DB2 group attach field	Value that is used in the CONNECT statements
The Db2 subsystem is defined with an SSID.	Yes	Group attach name
	No	SSID ¹
The Db2 subsystem is not defined with an SSID.	Yes or No	Group attach name

Table 3. The effect of the value of the **Use DB2 group attach** field in a data sharing environment

Note 1: If you generate jobs for multiple Db2 subsystems that are defined with an SSID and belong to the same data sharing group, the SSID of the first Db2 subsystem that is selected is used.

For example, assume that on the **Customizer Workplace** panel, you generated jobs for the following Db2 subsystems:

- DB2C, which is a stand-alone Db2 subsystem
- DB2A, which is a Db2 subsystem that is a member of data sharing group DSG1
- A Db2 subsystem that was not defined with an SSID that is a member of data sharing group DSGA

The following figure shows how these Db2 entries might be listed on the **Customizer Workplace** panel:

Associated DB2 Entries and Parameter Status Line commands: G - Generate jobs E - Edit B - Browse C - Copy R - Remove Cmd SSID GrpAttch Lv1 Mode User ID Date Status Message DB2C -- 121 NFM SYSADM 2017/11/09 Ready to Customize DB2A DSG1 121 NFM SYSADM 2017/11/09 Ready to Customize -- DSGA 121 NFM SYSADM 2017/11/09 Ready to Customize End of DB2 entries The following table shows which values are used in the CONNECT statements in the generated jobs, based on the value of the **Use DB2 group attach** field.

Table 4. Value that is used in the CONNECT statements in the generated jobs			
SSID	GrpAttch	Value of the Use DB2 group attach field	Value that is used in the CONNECT statements
		Yes	SSID
	No	SSID	
		Yes	Group attach name
DDZA	DSGI	No	SSID
DECA	Yes	Group attach name	
	DJGA	No	Group attach name

Tools Customizer metadata library

The name of the data set that contains the metadata that is used to display the Db2 and LPAR parameters. The parameters that are displayed on the **LPAR Parameters** panel and the **DB2 Parameters** panel depend on the parameters that you define and the tasks and steps that you select on the **Product Parameters** panel for the product that you are customizing. For example, the Db2 parameters that are required, based on the selected tasks and steps, are displayed on the **DB2 Parameters** panel, and you can edit them. If they are not required, they are not displayed. Read access to this data set is required. Data set names that exceed 42 characters must be enclosed in single quotation marks (').

Discover output data set

The name of the data set in which the output from the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC is stored. Each product has its own Discover EXEC. The Discover EXEC retrieves the product, LPAR, and Db2 parameters from a previously customized product. Write access to this data set is required. Data set names that exceed 42 characters must be enclosed in single quotation marks (').

Volume serial

The volume name in which the discover output data set will reside. If you don't specify a volume name, it will be assigned by the system.

Data store data set

The name of the data set where Tools Customizer stores information about product, LPAR, and Db2 parameter values. Information about which products are associated with which Db2 entries (Db2 subsystems, Db2 group attach names, and Db2 data sharing members) is also stored in this data set. Data set names that exceed 42 characters must be enclosed in single quotation marks ('). The specified data store data set can be used with only one invocation of Tools Customizer at a time. Data set names that exceed 42 characters must be enclosed in single quotation marks (').

Volume serial

The volume name in which the data store data set will reside. If you don't specify a volume name, it will be assigned by the system.

User job card settings for customization jobs

The job card information to be inserted into the generated jobs for customizing a product. The default value is the job statement information from the **ISPF Batch Selection** panel.

The first line of the job card automatically begins with the following information:

// JOB

where characters 3 - 10 are reserved by Tools Customizer for the job name and includes a blank space after JOB. This name cannot be edited. Information that you specify on the first line of the

job card cannot exceed 57 characters. This character limit includes a continuation character. All other lines of the job card cannot exceed 72 characters.

3. Press End to save and exit.

If the Discover output data set and the data store data set that you specified do not exist, Tools Customizer creates them.

Important: If the ISPF sessions unexpectedly ends before you exit Tools Customizer, the fields on the **Tools Customizer Settings** panel (CCQPSET) will be repopulated with default values, and you will be required to review them or specify new values again.

Results

The values are saved, and the **IBM Tools Customizer for z/OS** main menu panel (CCQPHME) is displayed again.

What to do next

You are ready to customize or recustomize Db2 SQL Performance AnalyzerDb2 Autonomics Director or to change parameter settings.

Related concepts

Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director

Using Tools Customizer to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director consists of identifying the product to customize; defining any required Db2 SQL Performance AnalyzerDb2 Autonomics Director, LPAR, and Db2 parameters; generating the customization jobs; and submitting the jobs.

Changing display options

You can choose which types of information to show on Tools Customizer panels. You can also copy your user profile to another data set so that it can be shared with other users.

About this task

By using the OPTIONS command, you can choose to show or hide the following information on Tools Customizer panels:

- The instructions on all panels
- The Product to Customize section on the **Customizer Workplace** panel (CCQPWRK)
- The Usage Notes section on the Product Parameters panel (CCQPPRD)

The OPTIONS command also allows you to copy your user profile to another data set so that it can be shared with other users. By sharing a copy of your profile, other uses can customize the same products that you initially customized or started to customize.

Procedure

1. On any Tools Customizer panel, issue the OPTIONS command.

The **Miscellaneous Options** panel (CCQPOPT) is displayed, as shown in the following figure. By default, all panel display options are preselected with a slash (/), which means that they will be displayed.

CCQP Comm	POP1 Mi nand ===>	scellaneous Options
Ente Pane / /	er a / to select options a el Display Options Show the panel instructi Show the Product to Cust Show the Usage Notes sec	and press Enter. To cancel, press End. Cons comize section ction
User -	r Profile Sharing Option Copy user profile to ano	other data set

Figure 5. The Panel Display Options panel (CCQPOPT)

- 2. To hide the panel instructions, the Product to Customize section, or the Usage Notes section, remove the slash from the appropriate option or options.
- 3. To copy your user profile to another data set so that it can be shared with other users:
 - a) Type a slash in the **Copy user profile to another data set** field and press Enter.
 - b) Specify the fully qualified name of the data set into which you want to copy the current user profile. If the data set name exceeds 42 characters, enclose the name in quotation marks. ALTER or UPDATE authorization to this data set is required.
 - c) Optionally specify a volume name in which the user profile data set will reside. If you don't specify a volume name, it will be assigned by the system.
- 4. Press Enter to save your changes.

Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director

Using Tools Customizer to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director consists of identifying the product to customize; defining any required Db2 SQL Performance AnalyzerDb2 Autonomics Director, LPAR, and Db2 parameters; generating the customization jobs; and submitting the jobs.

Customization roadmaps describe the steps that you must complete to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director. Separate roadmaps are provided for the three most common types of customizations.

Use the following table to determine which roadmap corresponds to your environment.

Environment description	Roadmap
You do not have a customized version of Db2 SQL Performance AnalyzerDb2 Autonomics Director, and you need to customize it for the first time.	"Roadmap: Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time" on page 47
You have already customized a version of Db2 SQL Performance AnalyzerDb2 Autonomics Director, and you want to use the same parameter values to customize a different version.	"Roadmap: Customizing a new version of Db2 SQL Performance AnalyzerDb2 Autonomics Director from a previous customization" on page 48
You have a customized version of of Db2 SQL Performance AnalyzerDb2 Autonomics Director, but you want to change one or more parameter values.	"Roadmap: Recustomizing Db2 SQL Performance AnalyzerDb2 Autonomics Director" on page 49

Table 5. Customization roadmaps

Roadmap: Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time

This roadmap lists and describes the steps that are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time.

If you are customizing a previous version of Db2 SQL Performance AnalyzerDb2 Autonomics Director, see "Roadmap: Customizing a new version of Db2 SQL Performance AnalyzerDb2 Autonomics Director from a previous customization" on page 48.

Before you complete these steps, ensure that the following prerequisites have been met:

- All of the product customization steps that must be done before Tools Customizer is started are complete.
- The LPAR ISPF libraries that are required to submit the jobs are known.
- Tools Customizer is started.
- The Tools Customizer settings have been reviewed or modified, and saved.

Complete the steps in the following table to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time.

Step	Description	Instructions	
1	Specify the product metadata library for the product that you want to customize. The name of this library is <i>hlq</i> .SANLDYXDENU.	"Specifying the metadata library for the product to customize" on page 50	
2	Create new Db2 entries and associate them with Db2 SQL Performance AnalyzerDb2 Autonomics Director.	"Creating and associating Db2 entries" on page 53	
3	Define the required parameters.	"Defining parameters" on page 55	
4	Generate the customization jobs for the product or for the Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director is ready to be customized.	"Generating customization jobs" on page 64	
5	Submit the generated customization jobs.	"Submitting customization jobs" on page 64	

Table 6. Steps for customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time

The following table lists some of the common administrative tasks that you might need to do during the customization process.

Table 7. Administrative tasks

Description	Instructions
Browse the different types of parameters.	"Browsing parameters" on page 67
Copy an existing Db2 entry to the list of Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director can be customized.	"Copying Db2 entries" on page 67
Remove one or more Db2 entries from the associated list.	"Removing Db2 entries" on page 68
Delete one or more Db2 entries from the master list.	"Deleting Db2 entries" on page 69
Display a list of customization jobs that have been previously generated.	"Displaying customization jobs" on page 69

Table 7. Administrative tasks (continued)		
Description	Instructions	
Maintain the customization jobs in the customization library.	"Maintaining customization jobs" on page 70	

Roadmap: Customizing a new version of Db2 SQL Performance AnalyzerDb2 Autonomics Director from a previous customization

This roadmap lists and describes the steps for customizing a new version of Db2 SQL Performance AnalyzerDb2 Autonomics Director based on the existing customization values of a previous version of the same product.

Use this roadmap even if the previous version of Db2 SQL Performance AnalyzerDb2 Autonomics Director was not customized by using Tools Customizer.

Before you complete these steps, ensure that the following prerequisites have been met:

- All of the product customization steps that must be done before Tools Customizer is started are complete.
- Tools Customizer is started.
- The Tools Customizer settings have been reviewed or modified, and saved.

Complete the steps in the following table to customize a new version of Db2 SQL Performance AnalyzerDb2 Autonomics Director from a previous customization.

Step	Description	Instructions
1	Specify the product metadata library for the product that you want to customize. The name of this library is <i>hlq</i> .SANLDYXDENU.	"Specifying the metadata library for the product to customize" on page 50
2	Use the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC to discover information about the version of Db2 SQL Performance AnalyzerDb2 Autonomics Director that you previously customized manually.	"Discovering Db2 SQL Performance AnalyzerDb2 Autonomics Director information automatically" on page 52
3	Define the required parameters.	"Defining parameters" on page 55
4	Generate the customization jobs for the product or for the Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director is ready to be customized.	"Generating customization jobs" on page 64
5	Submit the generated customization jobs.	"Submitting customization jobs" on page 64

Table 8. Steps for customizing a new version of Db2 SQL Performance AnalyzerDb2 Autonomics Director from a previous customization

The following table lists some of the common administrative tasks that you might need to do during the customization process.

Table 9. Administrative tasks

Description	Instructions
Browse the different types of parameters.	"Browsing parameters" on page 67

Table 9. Administrative tasks (continued)			
Description	Instructions		
Copy an existing Db2 entry to the list of Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director can be customized.	"Copying Db2 entries" on page 67		
Remove one or more Db2 entries from the associated list.	"Removing Db2 entries" on page 68		
Delete one or more Db2 entries from the master list.	"Deleting Db2 entries" on page 69		
Display a list of customization jobs that have been previously generated.	"Displaying customization jobs" on page 69		
Maintain the customization jobs in the customization library.	"Maintaining customization jobs" on page 70		

Roadmap: Recustomizing Db2 SQL Performance AnalyzerDb2 Autonomics Director

This roadmap lists and describes the steps to change parameter values and regenerate customization jobs for Db2 SQL Performance AnalyzerDb2 Autonomics Director after you have customized it for the first time.

The new customization jobs will replace the customization jobs that were previously generated and stored in the customization library. Part of the recustomization process includes selecting or deselecting optional tasks or steps, changing the definitions of parameters that have already been defined, or both. Use the method in this roadmap instead of deleting customization jobs from the customization library.

Before you complete these steps, ensure that the following prerequisites have been met:

- All of the product customization steps that must be done before Tools Customizer is started are complete.
- Tools Customizer is started.

Complete the steps in the following table to recustomize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Step	Description	Instructions	
1	Specify the product metadata library for the product that you want to recustomize. The name of this library is <i>hlq</i> .SANLDYXDENU.	"Specifying the metadata library for the product to customize" on page 50	
2	Edit the specific tasks, steps, or parameters that need to be changed.	 <u>"Defining Db2 SQL Performance</u> AnalyzerDb2 Autonomics Director parameters" on page 56 <u>"Defining LPAR parameters" on page 59</u> <u>"Defining Db2 parameters" on page 61</u> 	
3	Generate the customization jobs for the product or for the Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director is ready to be customized.	"Generating customization jobs" on page 64	
4	Submit the new generated customization jobs.	"Submitting customization jobs" on page 64	

Table 10. Required steps for recustomizing Db2 SQL Performance AnalyzerDb2 Autonomics Director

The following table lists some of the common administrative tasks that you might need to do during the customization process.

Table 11. Administrative tasks

Description	Instructions
Browse the different types of parameters.	"Browsing parameters" on page 67
Copy an existing Db2 entry to the list of Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director can be customized.	"Copying Db2 entries" on page 67
Remove one or more Db2 entries from the associated list.	"Removing Db2 entries" on page 68
Delete one or more Db2 entries from the master list.	"Deleting Db2 entries" on page 69
Display a list of customization jobs that have been previously generated.	"Displaying customization jobs" on page 69
Maintain the customization jobs in the customization library.	"Maintaining customization jobs" on page 70

Specifying the metadata library for the product to customize

You must specify a metadata library for the product that you want to customize.

About this task

The product metadata library contains the information that determines which tasks, steps, and parameters are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director. This information controls what is displayed on the **Product Parameters** panel, the **LPAR Parameters panel**, and the **DB2 Parameters** panel.

After Db2 SQL Performance AnalyzerDb2 Autonomics Director has been SMP/E installed, the default name of the product metadata library is *high_level_qualifier*.SANLDYXDENU, where *high_level_qualifier* is all of the segments of the data set name except the lowest-level qualifier.

Procedure

1. Specify option 1 on the **Tools Customizer for z/OS** panel.

The **Specify the Product or Pack Metadata Library** panel is displayed. This panel contains a list of the product metadata libraries that you specified most recently. If you are using Tools Customizer for the first time, this list is empty, as shown in the following figure:

CCQPHLQ Specify the Product or Pack Metadata Library 15:12:22 Type the name of the metadata library for the product or pack in the Metadata library field, or select the library in the list of previous libraries and press Enter to populate the field. Press Enter to continue. The default name of the metadata library after the product or pack has been SMP/E installed is <hlq>.SxxxDENU, where <hlq> is the high-level qualifier for the product or the pack, and xxx is the three-character prefix for the product or the pack. Product or pack metadata library . ANL.ALIAS.SANLDENU Version Metadata Name Library => => => => CCQPHLQ Specify the Metadata Library HH:MM:SS Command ===> Scroll ===> PAGE Type the name of the metadata library for the product or the pack in the Metadata library field, or select the library in the list of previous libraries and press Enter to populate the field. Press Enter to continue. The default name of the metadata library after the product or pack has been SMP/E installed is <hlq>.SxxxDENU, where <hlq> is the high-level qualifier for the product or the pack, and xxx is the three-character prefix for the product or the pack. Product or pack metadata library . DYX.DYX160.SDYXDENU Name Version Metadata Library => => =>

Figure 6. The Specify the Metadata Library panel

- 2. Use one of the following methods to specify the product metadata library:
 - Type the name of a fully qualified partitioned data set (PDS) or an extended partitioned data set (PDSE) in the **Product or pack metadata library** field. Double quotation marks (") cannot be used around the name. Single quotation marks (') can be used but are not required. If you are customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time, you must use this method.
 - Place the cursor in any column of the Recent Metadata Libraries list, and press Enter to populate Product or pack metadata library field. Press Enter again to select product or pack for customization.

Results

If you are customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time, the **Run Discover EXEC** panel is displayed. Otherwise, the **Customizer Workplace** panel is displayed.

What to do next

- Complete the steps that correspond to your environment:
 - **Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director for the first time** Do not run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC. Press End. The **Customizer Workplace** panel is displayed. If your environment requires associated Db2 entries, ensure that they are created and associated. If your environment does not require

associated Db2 entries, skip this step, and edit Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters.

Customizing Db2 SQL Performance AnalyzerDb2 Autonomics Director from a previous or current customization

Press Enter to run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC. The **Discover Customized Product Information** panel is displayed. Specify the required information for running the EXEC.

Discovering Db2 SQL Performance AnalyzerDb2 Autonomics Director information automatically

You can use the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC to discover information from a previous or current customization of Db2 SQL Performance AnalyzerDb2 Autonomics Director.

About this task

Tip: Using the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC to discover information from a previous or current customization saves time and reduces errors that can occur when parameters are specified manually.

Db2 SQL Performance AnalyzerDb2 Autonomics Director provides the Discover EXEC that you will run. Therefore, the information that can be discovered depends on Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Parameter values that are discovered and parameter values that are specified manually are saved in the data store. If parameter values for the product that you want to customize exist in the data store, Tools Customizer issues a warning before existing values are replaced.

Procedure

1. On the **Customizer Workplace** panel, issue the DISCOVER command.

If you chose to run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC on the pop-up panel after you specified the product to customize, skip this step.

Tip: You can run any Tools Customizer primary command by using either of the following methods:

- Place the cursor on the name of the primary command, and press Enter.
- Type the primary command name in the command line, and press Enter.

The **Discover Customized Product Information** panel is displayed, as shown in the following figure:

CCQPDSC	Discover Customized Product Information	13:18:16		
For the product you are customizing, the Discover EXEC retrieves product information from an already customized product. Specify the required information, and press Enter to run the Discover EXEC. Press End to cancel.				
Product to Customiz Product metadata Product name	ze library : ANL.ALIAS.SANLDENU > L : DB2 SQL Performance Analyz > V	PAR : MVS1 /ersion . : 5.1.0		
Discover EXEC for E Discover EXEC lib Discover EXEC name Discover output da	Extracting Information from an Already Cus rary ANL.ALIAS.SANLCLST e ANLCDISC ata set DB2TOOL.CCQ110.DISCOVER	tomized Product		
Information for Dis SQL PA High Level	scover EXEC Qualifier from the previous installation SYSX.ANL420	>		

Figure 7. The Discover Customized Product Information panel

2. Either accept the default values for the following input fields that Tools Customizer generates, or replace the default values with your own values:

Discover EXEC library (CLIST library)

The fully qualified data set name that contains the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC.

Discover EXEC name

The name of the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC.

Discover output data set

The fully qualified data set where output from the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC is stored.

3. Either accept or change the default values in the Information for Discover EXEC fields.

These fields are generated by Db2 SQL Performance AnalyzerDb2 Autonomics Director. They show the information that is required to run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC.

4. Issue the RUN command to run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC.

Alternatively, save your information without running the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC by issuing the SAVE command.

If you issue the RUN command to run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC, the parameter information is discovered for Db2 SQL Performance AnalyzerDb2 Autonomics Director, and the **Customizer Workplace** panel is displayed.

Results

The discovered parameter values for Db2 SQL Performance AnalyzerDb2 Autonomics Director replace any existing values.

What to do next

The next step depends on your environment:

- If Db2 entries were not discovered, or if you need to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director on new Db2 entries, create and associate the entries.
- If Db2 entries were discovered and you want to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director on only these entries, define the parameters.

Related tasks

Creating and associating Db2 entries

You can create new Db2 entries and associate them with Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Defining parameters

To customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, you must define Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters, LPAR parameters, and Db2 parameters, if your customization requires Db2 entries.

Creating and associating Db2 entries

You can create new Db2 entries and associate them with Db2 SQL Performance AnalyzerDb2 Autonomics Director.

About this task

The list of associated Db2 entries is on the **Customizer Workplace** panel.

Procedure

1. Issue the ASSOCIATE command on the **Customizer Workplace** panel.

The Associate DB2 Entry for Product panel is displayed, as shown in the following figure:

```
CCOPDAD
                      Associate DB2 Entry for Product
                                                                16:36:42
                                                        Scroll ===> PAGE
Command ===>
Select any of the following DB2 entries to add them to the Customizer
Workplace panel. You use the Customizer Workplace panel to choose the DB2
subsystems, data sharing members, and group attach names on which to
customize the product.
Commands: CREATE - Create new DB2 entries
DB2 Entries
 Line commands: A - Associate C - Copy D - Delete
 Cmd SSID GrpAttch
    *
        *
----- End of DB2 entries -----
CCQPDAD
                     Associate DB2 Entry for Product
                                                                16:36:42
                                                       Scroll ===> PAGE
Command ===>
Select any of the following DB2 entries to add them to the Customizer
Workplace panel. You use the Customizer Workplace panel to choose the DB2
subsystems, data sharing members, and group attach names on which to
customize the product.
Commands: CREATE - Create new DB2 entries
DB2 Entries
 Line commands: A - Associate C - Copy D - Delete
 Cmd SSID GrpAttch
    * *
             ----- End of DB2 entries -----
```

Figure 8. The Associate DB2 Entry for Product panel

2. Create Db2 entries.

If you need to associate Db2 entries that are already in the master list, skip this step and go to step 3.

a) Issue the CREATE command to create one Db2 entry, or issue CREATE *nn* to create multiple Db2 entries, where *nn* is the number of new entries to be created.

The **Create DB2 Entries** panel is displayed, as shown in the following figure:

```
      CCQPCDB
      Create DB2 Entries
      Row 1 of 1

      Command ===>
      Scroll ===> PAGE

      Specify the SSID, the group attach name, or both in the appropriate columns for each new DB2 entry and press Enter. To create additional entries, issue the Inn line command, where nn is the number of entries to be inserted, and press Enter. To cancel, press End.

      New DB2 Entries
      Line commands: I - Insert into list

      Line commands: I - Insert into list
      R - Remove from list

      Cmd
      SSID

      GrpAttch
      Message

      -------
      End of DB2 entries
```

Figure 9. The Create DB2 Entries panel

b) In the appropriate columns, specify a Db2 subsystem ID, Db2 group attach name, or Db2 data sharing member name for the Db2 entry that you want to create, and press Enter.

Valid values are 1 - 4 characters. You can use symbolic characters. You cannot use blanks.

Tips:

- To insert multiple Db2 entries, specify the Inn line command, where nn is the number of Db2 entries to be inserted.
- You will define specific parameters for these new Db2 entries on the DB2 Parameters panel. This
 panel is displayed after you select these new Db2 entries and issue the line command to generate

the jobs, after you issue the primary command to generate the jobs for all associated Db2 entries, or when you manually edit the Db2 parameters.

The **Associate DB2 Entry for Product** panel is displayed, and the new Db2 entry is displayed in the master list, as shown in the following figure:

```
CCQPDAD
                       Associate DB2 Entry for Product
                                                              Row 1 to 3 of 3
                                                             Scroll ===> CSR
Command ===>
Select any of the following DB2 entries to add them to the Customizer
Workplace panel. You use the Customizer Workplace panel to choose the DB2
subsystems, data sharing members, and group attach names on which to
customize the product.
Commands: CREATE - Create new DB2 entries
DB2 Entries
 Line commands: A - Associate C - Copy D - Delete
 Cmd SSID GrpAttch
      DBAA --
      DBAB --
     DBAC --
                     ----- End of DB2 entries -----
```

Figure 10. Associate DB2 Entry for Product panel

- c) Repeat steps b and c for each Db2 entry that you want to create.
- d) When you have created all the Db2 entries, associate them with Db2 SQL Performance AnalyzerDb2 Autonomics Director, or press End to display the **Customizer Workplace** panel.
- 3. Associate Db2 entries.
 - a) Specify A against one or more Db2 entries in the master list, and press Enter to associate them with Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Results

The **Customizer Workplace** panel is displayed with the associated Db2 entries displayed in the associated list.

What to do next

Define the parameters. **Related concepts**

Tools Customizer terminology

Tools Customizer uses several unique terms that you should be familiar with before you begin to use Tools Customizer.

Defining parameters

To customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, you must define Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters, LPAR parameters, and Db2 parameters, if your customization requires Db2 entries.

About this task

You must define the Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters first for the following reasons:

- If you ran the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC, you must review the values that were discovered.
- If you select optional tasks and steps on the **Product Parameters** panel that affect the Db2 entry on which you will customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, additional parameters might be displayed on the **DB2 Parameters** panel.

• If other steps must be completed in a specific sequence, customization notes on the **Product Parameters** panel will display the correct sequence.

Defining Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters

Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters are specific to Db2 SQL Performance AnalyzerDb2 Autonomics Director.

About this task

If you ran the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC, you must review the parameters that were discovered.

Procedure

1. Specify E next to the **Product parameters** field on the **Customizer Workplace** panel, and press Enter.

The **Product Parameters** panel is displayed, as shown in the following figure. If other steps must be completed in a specific sequence before you define the Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters, a note labeled **Important** will display the correct sequence on this panel.

```
CCOPPRD
                                   Product Parameters
                                                                               13:26:32
  Command ===>
                                                                        Scroll ===> PAGE
  Complete the following tasks to customize the products. The required tasks,
  required steps within a required or selected task, and required parameters
  are preceded by an asterisk (*). Ensure that values are specified for the
  required parameters. Press End to save and exit.
  Commands: SAVE VERIFYOFF
  Line Commands: / - Select
  Product to Customize
    Product metadata library : ANL.ALIAS.SANLDENU >
                                                                     LPAR. . . : MVS1
    Product name . . . . . : DB2 SQL Performance Analyz > Version . : 5.1.0
  Product customization library : ANL.PRODUCT.CUST.$MVS1$.ANL510
                                                                             More:
                                                                                        +
   Required parameters
    HLQ of SQL PA installation . . . . . . . ANL.ANL510
  / Create files needed for TSO and batch execution
      Drop DB2 objects
    / Create files, create DB2 objects, and perform binds and grants
Prefix for output reports and work files
          High-level qualifier of DB2 Path Checker
    Create generic plan tables
    / Create new DSN3@ATH
      Create new DSN3@SGN
 TCUSTMZR
                      Product Parameters: DB2 Autonomics
                                                                             Top of data
                                                                       Scroll ===> CSR
 Command ===>
Complete the following tasks to customize the products. The required tasks, required steps within a required or selected task, and required parameters are preceded by an asterisk (*). Ensure that values are specified for the
 required parameters. Press End to save and exit.
Commands: SAVE VERIFYOFF
Line Commands: / - Select
 Product customization library : TWUSR.WRK0160.$RS22$.DYX160
                                                                             More:
                                                                                        +
 Common parameters
  *DB2 Autonomics high-level qualifier
     DB2T00L.DYX160
  *Started Task PROCLIB . . DYX.PROCLIB
 * Create or Drop DB2 Autonomics tables
     Drop DB2 Autonomics objects
   * Create or update repository
    *DB2 Autonomics temporary package . . . DYX16TMK
    *DB2 Autonomics temporary plan . . . . DYX16TMP
 * Bind and free packages and plans
     Free packages and plans
   * Bind packages and plans
 * Grant EXECUTE privilege
   * Grant EXECUTE privilege on the plans
```

Figure 11. The **Product Parameters** panel

You can use the following primary commands on this panel:

SAVE

Saves the specified product or component parameter values.

VERIFY / VERIFYOFF

Use the VERIFY and VERIFYOFF commands to turn on and off parameter verification of product or component parameters. Before you can generate customization jobs, you must verify that all required parameters are set to a valid value. The product or component parameter status of Verify Values on the Customize Workplace panel indicates that the values have not been verified.

Enter these commands either by typing them in the command field and pressing Enter or by positioning the cursor on the command and pressing Enter. When VERIFY is active, VERIFYOFF is displayed so that you can toggle between the two states. By default, verification is turned on when you display the **DB2 Parameters** panel, and the verification state is reset to VERIFY every time you exit the Customization Workplace panel by pressing PF3.

Turning verification off is useful when you need to exit the panel before you have entered all of the required parameters, but you want to save the parameters that you have specified. When you disable verification, it is disabled only for the Product or Component Parameters Values panel.

2. Select any required tasks and steps, and specify values for any parameters. After you select a task or step with a slash (/), put the cursor in the selected field and press Enter.

If tasks, steps, and parameters are required, they are preselected with a slash (/). Otherwise, they are not preselected.

All of the required parameters have default values, which you can either accept or change.

Tips:

- In the command line, specify the KEYS command, and map EXPAND to one of the function keys.
- For a detailed description of all input fields, put the cursor in the field, and press F1 or the key that is mapped to Help.
- The following elements apply to specific fields:
 - Add is displayed when parameters can have multiple values but currently have only one value. To specify multiple values in these fields, place the cursor on Add, and press Enter. Use the displayed panel to add or delete additional values.
 - **List** is displayed when the complete list of valid values for the fields is too long to be displayed on the panel. To see the complete list of values, place the cursor on **List**, and press F1 or the key that is mapped to Help.
 - **More** is displayed when input fields contains multiple values. To see all of the values in the field, place the cursor on **More**, and press Enter.
- 3. Optional: Select other tasks and steps with a slash (/) and press Enter to activate the input fields. Either accept or change the default values that are displayed.
- 4. Press End to save your changes and exit, or issue the SAVE command to save your changes and stay on the **Product Parameters** panel.

Results

The **Customizer Workplace** panel is displayed, and the status of the product parameters is Ready to Customize.

What to do next

If the status of other parameters on the **Customizer Workplace** panel is Incomplete, Verify Values, or Discovered, edit these parameters. **Related tasks**

Defining LPAR parameters

LPAR parameters are parameters on the local LPAR that are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Defining Db2 parameters Db2 parameters are parameters for a Db2 entry.

Defining LPAR parameters

LPAR parameters are parameters on the local LPAR that are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Procedure

1. Specify E next to the LPAR parameters field, and press Enter.

The LPAR Parameters panel is displayed, as shown in the following figure:

CCQPLPR LPAR Parameters 13:28:55 Enter values for all of the LPAR parameters. Press End to save and exit. Commands: SAVE - Save parameter values Product to Customize Product metadata library : ANL.ALIAS.SANLDENU >LPAR. . . : MVS1Product name : DB2 SQL Performance Analyz >Version . : 5.1.0 **ISPF** Libraries Message library ISPF.ISPPLIB > Add... > Add... > Add... > Add... > Add... Command procedures library SYS1.PROCLIB > Add... Language Environment Libraries Language environment high-level qualifier CEE >

Figure 12. The LPAR Parameters panel

You can use the following primary commands on this panel:

SAVE

Saves the specified product or component parameter values.

VERIFY / VERIFYOFF

Use the VERIFY and VERIFYOFF commands to turn on and off parameter verification of LPAR parameters. Before you can generate customization jobs, you must verify that all required parameters are set to a valid value. The LPAR parameter status of Verify Values on the Customize Workplace panel indicates that the values have not been verified.

Enter these commands either by typing them in the command field and pressing Enter or by positioning the cursor on the command and pressing Enter. When VERIFY is active, VERIFYOFF is displayed so that you can toggle between the two states. By default, verification is turned on when you display the LPAR Parameters panel, and the verification state is reset to VERIFY every time you exit the Customization Workplace panel by pressing PF3.

Turning verification off is useful when you need to exit the panel before you have entered all of the required parameters, but you want to save the parameters that you have specified. When you disable verification, it is disabled only for the LPAR Parameters Values panel.

2. Specify values for all required parameters that are displayed.

Many parameters have default values, which you can either accept or change.

Tips:

- In the command line, specify the KEYS command, and map EXPAND to one of the function keys.
- For a detailed description of all input fields, put the cursor in the field, and press F1 or the key that is mapped to Help.
- The following elements apply to specific fields:
 - Add is displayed when parameters can have multiple values but currently have only one value. To specify multiple values in these fields, place the cursor on Add, and press Enter. Use the displayed panel to add or delete additional values.
 - List is displayed when the complete list of valid values for the fields is too long to be displayed on the panel. To see the complete list of values, place the cursor on List, and press F1 or the key that is mapped to Help.
 - **More** is displayed when input fields contains multiple values. To see all of the values in the field, place the cursor on **More**, and press Enter.

The following LPAR parameters can contain 1 - 64 values:

- LPAR macro library
- Message library
- Panel library
- Skeleton library
- ISPF table input library
- ISPF user profile library
- File tailoring output library
- Link list library
- Command procedures library
- Macro library
- Link-edit library
- Load library
- Started task library name
- 3. Press End to save your changes and exit, or issue the SAVE command to save your changes and stay on the same panel.

Results

The **Customizer Workplace** panel is displayed, and the status of the LPAR parameters is Ready to Customize.

What to do next

If the status of other parameters on the **Customizer Workplace** panel is Incomplete, Verify Values, or Discovered, edit these parameters.

Related tasks

Defining Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters are specific to Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Defining Db2 parameters Db2 parameters are parameters for a Db2 entry.

Defining Db2 parameters

Db2 parameters are parameters for a Db2 entry.

About this task

If you did not run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC, you must create and associate one or more Db2 entries before you can define the Db2 parameters. For more information, see "Creating and associating Db2 entries" on page 53.

Procedure

1. Specify E next to one or more Db2 entries in the associated list, which is in the Associated Db2 Entries and Parameter Status section on the **Customizer Workplace** panel, and press Enter.

The **DB2 Parameters** panel is displayed, as shown in the following figure:

DB2 Parameters CCOPDB2 13:36:01 Enter values for all of the DB2 parameters. Press End to save and exit. Commands: SAVE - Save parameter values Product to Customize Product metadata library : ANL.ALIAS.SANLDENU > LPAR. . . : MVS1 Product name : DB2 SQL Performance Analyz > Version . : 5.1.0 More: + Group attach name General DB2 Information
 Mode
 .
 .
 .
 .
 NFM
 (CM,CM)

 Level
 Number
 .
 .
 .
 .
 101
 (101,111)
 (CM, CM8, CM9, NFM) DB2 Libraries Add... Load Library DSN.SDSNLOAD >

 Run Library
 ...
 DSN.SDSNRLIB

 Exit Library
 ...
 DSN.SDSNEXIT

 Sample library
 ...
 DSN.SDSNSAMP

 Add... > > Add... > Add... Macro library DSN.SDSNMACS > DB2 Bufferpools Add... Name of the 4 KB bufferpool BP0 Name of the 8 KB bufferpool BP8K0 DB2 Utilities Plan name for the DSNTIAD utility SQL PA Db2 Objects Storage group name SYSDEFLT >

TCUSTMZR DB2 Parameters: DB2 Autonomics 21:35:51 Command ===> Scroll ===> CSR Ensure that values are specified for the required DB2 parameters. Press End to save and exit. Commands: SAVE VERIFYOFF More: + Group attach name General DB2 Information - common DB2 Libraries - common *Load library DSN.SDSNLOAD *Run library DSN.RUNLIB.LOAD *Exit library DSN.SDSNEXIT Add hhA bbA DB2 Buffer Pools - common *Name of the 4 KB buffer pool BP0 *Name of the 8 KB buffer pool BP8K0 DB2 Utilities - common SYSAFF for DB2 utilities . *Plan name for the DSNTIAD utility DSNTIAD DB2 Autonomics BIND Parameters *BIND owner ID DB2USER *User ID for GRANT statement PUBLIC >

Figure 13. The **DB2 Parameters** panel

You can use the following primary commands on this panel:

SAVE

Saves the specified product or component parameter values.

VERIFY / VERIFYOFF

Use the VERIFY and VERIFYOFF commands to turn on and off parameter verification of Db2 parameters. Before you can generate customization jobs, you must verify that all required
parameters are set to a valid value. The Db2 parameter status of Verify Values on the Customize Workplace panel indicates that the values have not been verified.

Enter these commands either by typing them in the command field and pressing Enter or by positioning the cursor on the command and pressing Enter. When VERIFY is active, VERIFYOFF is displayed so that you can toggle between the two states. By default, verification is turned on when you display the **DB2 Parameters panel**, and the verification state is reset to VERIFY every time you exit the Customization Workplace panel by pressing PF3.

Turning verification off is useful when you need to exit the panel before you have entered all of the required parameters, but you want to save the parameters that you have specified. When you disable verification, it is disabled only for the **DB2 Parameters Values** panel.

2. Specify values for all parameters that are displayed.

Tips:

- In the command line, specify the KEYS command, and map EXPAND to one of the function keys.
- For a detailed description of all input fields, put the cursor in the field, and press F1 or the key that is mapped to Help.
- The following elements apply to specific fields:
 - Add is displayed when parameters can have multiple values but currently have only one value. To specify multiple values in these fields, place the cursor on Add, and press Enter. Use the displayed panel to add or delete additional values.
 - List is displayed when the complete list of valid values for the fields is too long to be displayed on the panel. To see the complete list of values, place the cursor on List, and press F1 or the key that is mapped to Help.
 - **More** is displayed when input fields contains multiple values. To see all of the values in the field, place the cursor on **More**, and press Enter.

Many parameters have default values, which you can either accept or change.

3. Press End to save your changes and exit, or issue the SAVE command to save your changes and stay on the same panel.

Results

The status of the Db2 entries that you selected on the **Customizer Workplace** panel is Ready to Customize.

What to do next

If the status of other parameters on the **Customizer Workplace** panel is Incomplete, Verify Values, or Discovered, edit these parameters.

Related tasks

Defining Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters

Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters are specific to Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Defining LPAR parameters

LPAR parameters are parameters on the local LPAR that are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Generating customization jobs

To generate customization jobs for Db2 SQL Performance AnalyzerDb2 Autonomics Director and any associated Db2 entries, issue the GENERATEALL command, or select one or more Db2 entries on which to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Procedure

Generate the customization jobs by using one of the following methods.

- If you want to generate customization jobs at the product level and for any associated Db2 entries, issue the GENERATEALL command, and press Enter.
- If you want to generate customization jobs for specific Db2 entries, select the Db2 entries by specifying the G line command against them, and press Enter. The available Db2 entries are in the associated list in the Associated Db2 Entries and Parameter Status section.

Important: Regenerating customization jobs will replace any existing jobs, including jobs that you might have manually modified after they were generated.

Results

If the status is Incomplete or Discovered for Db2 SQL Performance AnalyzerDb2 Autonomics Director parameters, LPAR parameters, or Db2 parameters, Tools Customizer automatically starts an editing session for the types of parameters that are required. The session continues until the panel for each type of required parameter has been displayed.

What to do next

If an automatic editing session is started, accept the displayed parameter values or define values for the required types of parameters, select optional parameters, tasks, or steps for your environment, and save the parameter values. Otherwise, the customization jobs are generated, and you can submit them.

Tip: If the customization jobs are generated, but you are not ready to submit them, you can see them later by issuing the JOBLIST command on the **Customizer Workplace** panel. The JOBLIST command displays the **Finish Product Customization** panel, which you can use to submit the jobs.

Submitting customization jobs

Submit the customization jobs to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Before you begin

Ensure that the correct jobs are generated.

About this task

The following figure shows part of the **Finish Product Customization** panel. The table on this panel shows the customization jobs that are generated by Tools Customizer. They are grouped by job sequence number.

CCQPCST Finish Product Customization Row 1 to 23 of 41 Submit the members in the order in which they apply to each DB2 entry. To submit the job, edit the member and issue the TSO SUBMIT command, or edit the customized library and submit the jobs from there. Product to Customize Product metadata library : ANL.ALIAS.SANLDENU > LPAR . . : MVS1 Product name : DB2 SQL Performance Analyz > Version . : 5.1.0 Line Commands: E - Edit B - Browse Product customization library . : ANL.PRODUCT.CUST.\$MVS1\$.ANL510 > Cmd Member SSID GrpAttch Template Generated Description A0DROPAA DB01 --ANLDROPD 2013/06/25 Drops DB2 objects from a previo AUDROPAA DB01 --A1DCREAA DB01 --A2PCREAA DB01 --A3ICREAA DB01 --A4JCREAA DB01 --A5BNDGAA DB01 --ANLDCRE8 2013/06/25 Creates the necessary DB2 objec ANLPCRE8 2013/06/25 Creates all required SQL PA lib ANLICRE8 2013/06/25 Creates ssidCNTL, ssidPARM, and ANLJCRE8 2013/06/25 Creates the JCL that is stored ANLBNDGR 2013/06/25 Binds the packages and plans, a B1QMONC DB01 --ANLQMFGV 2013/06/25 Creates the CLIST that is used TCUSTMZR Finish Product Customization Row 1 to 9 of 15 Scroll ===> CSR Command ===> For a first-time customization, submit the jobs in the members in the order in which they apply to the DB2 entries. Otherwise, submit only the necessary jobs that were generated after changes were made. To submit jobs, browse the members and issue the TSO SUBMIT command. Line Commands: E - Edit B - Browse Product customization library: TWUSR.WRK0160.\$RS22\$.DYX160 Cmd Member New SSID GrpAttch Template Date Description * * * * * * * A0DROPAB YES SS01---DYXDROP 2017/09/26 Drop DB2 Autonomics tablesA1DDL1AB YES SS01--DYXDROP 2017/09/26 Create DB2 Autonomics ObjecA2FREEAB YES SS01--DYXFREE 2017/09/26 Free packages and plansA3BINDAB YES SS01--DYXBIND 2017/09/26 Binds packages and plansA4GRANAB YES SS01--DYXGRANT 2017/09/26 Grants EXECUTE authorityA5WLMAB YES SS01--DYXRUN1 2017/09/26 Create WLM address space foA6RUN1AB YES SS01--DYXRUN1 2017/09/26 Create batch job to run DB2A7TSNAAB YES SS01--DYXSRVCF 2017/09/26 Create batch job to run RTSA8SRVCF YES ----DYXAGTCF 2017/09/26 Create agent config file foB0SSLCR YES ----DYXSLCR 2017/09/26 Create agent config file foB0SSLCR YES ----DYXAGTJB 2017/09/26 Create agent job for DB2 AuB2SRVJB YES ----DYXAGTJB 2017/09/26 Create server job for DB2 AuB3AGTST YES ----DYXAGTST 2017/09/26 Create agent STC for DB2 Au -----> B2SRVJB YES -- --B3AGTST YES -- --B4SRVST YES -- --DYXAGTST 2017/09/26 Create agent STC for DB2 Au DYXSRVST 2017/09/26 Create server STC for DB2 A ----- End of customized jobs -

Figure 14. The Finish Product Customization panel

The member-naming conventions depend on whether the customization jobs are for Db2 entries, and LPAR, or the product.

The New filed indicates if the job member is newly created/updated. It is either YES or NO. YES indicates the job member is newly created or updated, and it needs to be submitted for customization. NO indicates the job member is not newly created/updated, it does not need to be submitted for customization.

Customization jobs for Db2 entries

The members use the following naming convention:

```
<job_sequence_number><job_ID><DB2_entry_ID>
```

where

job_sequence_number

Two alphanumeric characters, A0 - Z9, that Tools Customizer assigns to a customization job. The number for the first template in the sequence is A0, the number for the second template is A1, and so on.

job_ID

Characters 4 - 7 of the template name, if the template name contains five or more characters. Otherwise, only character 4 is used. Db2 SQL Performance AnalyzerDb2 Autonomics Director assigns the template name.

DB2_entry_ID

Two alphanumeric characters, AA - 99, that Tools Customizer assigns to a Db2 entry.

For example, the XYZBNDDB2_entry_ID_1 and XYZBNDDB2_entry_ID_2 jobs are generated from the XYZBNDGR template, and the XYZ4DB2_entry_ID_1 and XYZ4DB2_entry_ID_2 jobs are generated from the XYZ4 template. If the jobs are generated on two Db2 entries, the following member names are listed sequentially: A0BNDGAA, A0BNDGAB, A14AA, A14AB.

Customization jobs for an LPAR or the product

The members use the following naming convention:

<job_sequence_number><job_ID>

where

job_sequence_number

Two alphanumeric characters, A0 - Z9, that Tools Customizer assigns to a customization job. The number for the first template in the sequence is A0, the number for the second template is A1, and so on.

job_ID

Characters 4 - 8 of the template name, if the template name contains five or more characters. Otherwise, only character 4 is used. For example, for the XYZMAKE template, the job ID is MAKE. For the XYZM template, the job ID is M. Db2 SQL Performance AnalyzerDb2 Autonomics Director assigns the template name, and it is displayed in the Template column.

For example, the XYZBNDGR job is generated from the XYZBNDGR template, and the XYZ4 job is generated from the XYZ4 template. The following member names are listed sequentially: AOBNDGR, A14.

Procedure

- 1. Submit the generated customization jobs by following the process that you use in your environment or by using the following method:
 - a) Specify B or E against a customization job or the product customization library, and press Enter. An ISPF browsing or editing session is started.
 - b) Browse the customization job or each member in the library to ensure that the information is correct.
 - c) Run the TSO SUBMIT command.
- 2. Press End.

Results

Db2 SQL Performance AnalyzerDb2 Autonomics Director is customized, and the **Customizer Workplace** panel is displayed. The status is Customized for the Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director was customized.

What to do next

You can generate more customization jobs for other Db2 entries, view a list of customization jobs that you previously generated, or recustomize Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Browsing parameters

You can browse the product or component parameters, the LPAR parameters, and the Db2 parameters in read-only mode.

Procedure

- 1. On the **Customizer Workplace** panel, specify B next to the **Product parameters** field, the **LPAR parameters** field, or the Db2 entry that you want to browse, and press Enter.
 - The panel that corresponds to your specification is displayed.
- 2. Press End to exit.

Copying Db2 entries

You can copy associated and not associated Db2 entries to other Db2 entries or to new Db2 entries.

About this task

Go to the step that applies to your environment:

- To copy an associated Db2 entry to another associated Db2 entry or to an entry that is not associated, go to step 1.
- To copy an associated Db2 entry to a new entry, go to step 2.
- To copy a Db2 entry that is not associated to a new entry, go to step 3.

Procedure

- 1. To copy an associated Db2 entry to another associated Db2 entry or to an entry that is not associated, complete the following steps:
 - a) Specify C against a Db2 entry in the associated list of Db2 entries on the **Customizer Workplace** panel, and press Enter.

The Copy Associated DB2 Entry panel is displayed.

b) Select one or more Db2 entries to which information will be copied by specifying the / line command, and press Enter.

The Associated column indicates whether the Db2 entry is associated.

Tip: To copy information into all of the Db2 Entries in the list, issue the SELECTALL primary command, and press Enter.

The Copy DB2 Parameter Values panel is displayed.

c) Specify an option for copying common and product-specific Db2 parameter values.

Common Db2 parameter values apply to all Db2 entries for all products that you have customized by using Tools Customizer. Product-specific Db2 parameter values apply only to the product that you are currently customizing.

- To copy the common Db2 parameter values and the product-specific Db2 parameter values, specify option 1, and press Enter.
- To copy only the product-specified Db2 parameter values, specify option 2, and press Enter.

In some cases, the Db2 parameter values might contain the Db2 subsystem ID as an isolated qualifier in data set names. For example, in the DB01.DB01TEST.DB01.SANLLOAD, data set name, the DB01 subsystem ID is isolated in the first and third qualifiers but is not isolated in the second qualifier. When the Db2 subsystem ID is an isolated qualifier in data set names, the **Change DB2 Subsystem ID in DB2 Parameter Values** panel is displayed. Otherwise, the **Customizer Workplace** panel is displayed.

d) If the **Change DB2 Subsystem ID in DB2 Parameter Values** panel is displayed, specify an option for changing the subsystem IDs. Otherwise, skip this step.

- To change the subsystem ID in isolated qualifiers in data set names, specify option 1, and press Enter.
- To use the same subsystem ID in all values, specify option 2, and press Enter.

The **Customizer Workplace** panel is displayed with the copied associated entry in the list.

- 2. To copy an associated Db2 entry to a new entry, complete the following steps:
 - a) Specify C against a Db2 entry in the associated list of Db2 entries on the **Customizer Workplace** panel, and press Enter.

The Copy Associated DB2 Entry panel is displayed.

b) Issue the CREATE command.

The Create DB2 Entries panel is displayed.

c) Specify the SSID, the group attach name, or both in the appropriate columns for each new Db2 entry, and press Enter.

Tip: To add rows for additional entries, specify the Inn line command, where nn is the number of entries to be created, and press Enter.

The **Copy Associated DB2 Entry** panel is displayed with the new entries in the list. The new entries are preselected.

d) Press Enter to complete the copy process.

The **Customizer Workplace** panel is displayed with the copied entries in the list.

- 3. To copy a Db2 entry that is not associated to a new entry, complete the following steps:
 - a) Issue the ASSOCIATE command on the **Customizer Workplace** panel.

The Associate DB2 Entry for Product panel is displayed.

- b) Select one or more Db2 entries by specifying the / line command, and press Enter. The **Copy a DB2 Entry** panel is displayed.
- c) Specify the SSID, the group attach name, or both in the appropriate columns for the new Db2 entry, and press Enter.

The Associate DB2 Entry for product panel is displayed with the copied entry in the list.

d) If you want to associate the copied entry, specify A against it, and press Enter. The **Customizer Workplace** panel is displayed with the copied entries in the list.

What to do next

Edit any of the parameters or generate the jobs. **Related concepts**

Tools Customizer terminology

Tools Customizer uses several unique terms that you should be familiar with before you begin to use Tools Customizer.

Removing Db2 entries

You can remove Db2 entries from the associated list.

About this task

When you remove Db2 entries from the associated list, any customization jobs for the entries are removed from the list of jobs on the **Finish Product Customization** panel, and they are deleted.

Procedure

On the **Customizer Workplace** panel, specify R next to one or more Db2 entries that you want to remove, and press Enter.

The selected Db2 entries are removed from the associated list and added to the master list on the **Associate DB2 Entry for Product** panel, and the customization jobs are deleted.

Related concepts

Tools Customizer terminology

Tools Customizer uses several unique terms that you should be familiar with before you begin to use Tools Customizer.

Deleting Db2 entries

You can delete Db2 entries from the master list.

About this task

When you delete Db2 entries from the master list, any associations and all customization jobs for products that are customized on the entries will be deleted.

Procedure

1. On the **Customizer Workplace** panel, issue the ASSOCIATE command.

The Associate DB2 Entry for Product panel is displayed.

2. Specify D next to one or more Db2 entries that you want to delete, and press Enter.

If the entry is associated with any products, the **Delete Associated DB2 Entry** panel for the first Db2 entry that you selected is displayed. Otherwise, the **Delete DB2 Entry** panel is displayed.

3. To delete the Db2 entries, press Enter.

If the Db2 entries are associated with any products in the table on the **Delete Associated DB2 Entry** panel, any associations and all customization jobs for the products that are customized on it are deleted. Otherwise, only the Db2 entries are deleted. If you selected multiple Db2 entries to delete, the next Db2 entry that you selected is displayed on either the **Delete Associated DB2 Entry** panel or the **Delete DB2 Entry** panel. Otherwise, the **Associate DB2 Entry for Product** panel is displayed.

What to do next

If you selected multiple Db2 entries to delete, repeat step 3 until all selected entries are deleted. Then, continue the customization process.

Displaying customization jobs

You can view a list of the members that contain the customization jobs before or after you submit the jobs.

About this task

The customization jobs that you generate for one Db2 entry are also displayed when you customize Db2 SQL Performance AnalyzerDb2 Autonomics Director for another Db2 entry later.

Procedure

On the **Customizer Workplace** panel, issue the JOBLIST command.

The **Finish Product Customization** panel is displayed. This panel shows the list of jobs that you have previously generated. They are grouped by job sequence number. Use this panel to browse or edit the generated jobs before you submit them.

Maintaining customization jobs

Instead of deleting customization jobs outside of Tools Customizer, you can maintain the correct jobs for Db2 SQL Performance AnalyzerDb2 Autonomics Director by completing the steps for recustomization.

About this task

You cannot delete or rename customization jobs from the customization library by starting an ISPF browse or edit session from the **Finish Product Customization** panel. If you try to delete customization jobs by using this method, the CCQC034S message is issued. If you try to rename customization jobs, the CCQC035S message is issued.

If you delete or rename customization jobs from the customization library by using ISPF outside of Tools Customizer, Tools Customizer will not recognize that the jobs were deleted, and the **Finish Product Customization** panel will still display them. If you browse or edit jobs that were deleted from the library outside of Tools Customizer, the CCQC027S message is issued.

Procedure

To maintain the correct customization jobs in the customization library, complete the steps for recustomization.

Using Tools Customizer in a multiple-LPAR environment

Currently, Tools Customizer supports only the local LPAR; however, you can propagate customizations to additional LPARs by using either of two different methods.

About this task

In a multiple-LPAR environment, Tools Customizer identifies the LPAR to which you are logged on. Tools Customizer uses this LPAR name for several different parameter settings, one of which is the data store. When you use the data store during the customization of Db2 SQL Performance AnalyzerDb2 Autonomics Director that is on a different LPAR, Tools Customizer issues message CCQD586S, which indicates that the product has already been customized based on values from the data store on the first LPAR. This message is issued to prevent the data store from becoming corrupted.

This behavior occurs in the following conditions:

- Tools Customizer is installed on a DASD device that is shared by multiple LPARs.
- After a product is customized by using Tools Customizer, the data store is copied to another LPAR.

Procedure

To customize products running against a Db2 subsystem on an LPAR where Tools Customizer is not installed, consider using one of the following methods:

Install one instance of Tools Customizer on one LPAR

If you intend to reuse the customization values for all the instances of your products on all LPARs, use this method.

- a. Associate all the Db2 entries in this one instance of Tools Customizer. The LPARs on which the Db2 subsystems reside do not matter.
- b. Generate the customization jobs for each Db2 entry.
- c. Copy the generated customization jobs to the LPAR to run against the specific Db2 entries. Some LPAR-specific edits might be required. You can make these edits in the customized jobs that you copied. Note that this situation is one of the few situations where you might need to make manual changes to the jobs that are customized by Tools Customizer.

Install one instance of Tools Customizer on each LPAR

If you do not want to reuse previous customization values and you want to start new customizations, use this method.

Important: This method will likely not be the preferred approach for most organizations because most organizations tend to use similar or identical customization values for each product instance on all LPARs.

Optional: Creating generic plan tables

Optionally, you can create generic plan tables by using Tools Customizer. Tools Customizer will create the necessary JCL, but you must manually complete some steps before and after you submit the customization job.

Before you begin

Ensure that you selected the optional task to create generic plan tables on the **Product Parameters** panel in Tools Customizer and that Tools Customizer generated the customization job.

About this task

Before you can create generic plan tables, you must create secondary authorization identifiers (authorization IDs) that will act as the owners of the generic SQL Performance Analyzer plan tables. These secondary authorization IDs allow SQL Performance Analyzer to switch any user from their current authorization ID to one of the special SQL Performance Analyzer secondary authorization IDs, which are shared by all Db2 users.

Before you submit the customization job, you must complete the following steps manually.

Procedure

1. Define the secondary authorization IDs in the DSN3@ATH exit.

The sample exit code for the DSN3@ATH exit is distributed with Db2 as member DSN3SATH in the DSNxxx.SDSNSAMP sample library.

- a) Obtain the most current copy of DSN3SATH and copy it to *hlq*.SANLSAMP. Many sites customize this exit by adding their own code to the Db2 sample program.
- b) Copy the SQL Performance Analyzer modifications from *hlq*.SANLSAMP(ANLAUTH), and paste them into the DSN3@ATH exit immediately after the SATH090 label. The following example shows the code with the secondary authorization IDs after the SATH090 label:

SATH090	DS	OH	
* ======= * APPENI * =======	===== D THE =====	ANL SECONDARY AUTHIDS TO	THE END OF THE SECONDARY LIST *
* THERE A * AS MAN * THE NUN * ENTRIES	ARE 24 Y ANL MBER (S IN /	45 POSSIBLE ENTRIES ON TH IDS AS WILL FIT ON THE EN DF ENTRIES AND THE ANL USI ANL REGISTRY TABLE. THE N	E SECONDARY AUTHID LIST. INSERT * ND, WITHOUT EXCEEDING 245 LIMIT * ER NAMES MUST CORRESPOND TO THE * JMBER OF IDS (10) IS ARBITRARY. *
	LA L LA SR BC BC	R4,AIDLSEC R0,AIDLSCNT R3,=X'F5' R3,R0 2,ANLOK ENDANI	R4 ADDRESS OF SECONDARY AREA R0 NUMBER OF 2NDARY SLOTS USED R3 CONSTANT 245 MAXIMUM SLOTS R3 HOLDS REMAINING SLOTS UNUSED OK IF >0 RESULT (CONDITION = 2) ELSE NO SLOTS CANNOT INSEPT IDS
ANLOK	MH	R0,=X'0008'	MULTIPLY SLOTS * 8 = BYTES USED
ANLSEC	AR CLC BE LA CR BNL B	R0,R4 0(8,R4),=CL8'' ADDANL1 R4,8(R4) R4,R0 ENDANL ANLSEC OU	RO STORE STOPPING ADDRESS HERE IS THIS A BLANK SLOT? GOOD, ADD SQL/PA SECONDARIES BUMP UP AND CONTINUE HAVE WE EXHAUSTED THE LIST? YES, NO MORE ROOM AVAILABLE NO, TRY NEXT SLOT
ADDANLI	DS MVC LA BCT B	0(8,R4),=CL8'ANLUSER1' R4,8(R4) R3,ADDANL2 ENDANL	ADD SECONDARY ID 1 BUMP TO NEXT YES, ROOM FOR ANOTHER NO. WE ARE DONE
ADDANL2	MVC LA BCT	0(8,R4),=CL8'ANLUSER2' R4,8(R4) R3,ADDANL3	ADD SECONDARY ID 2 BUMP TO NEXT YES, ROOM FOR ANOTHER

	-			
	В	ENDANL	NO, WE ARE DONE	
ADDANL3	MVC	0(8,R4),=CL8'ANLUSER3'	ADD SECONDARY ID 3	
	LA	R4,8(R4)	BUMP TO NEXT	
	BCT	R3, ADDANL4	YES, ROOM FOR ANOTHER	
	В	ENDANL	NO, WE ARE DONE	
ADDANI 4	MVC	0(8,R4),=CL8'ANLUSER4'	ADD SECONDARY TD 4	
	IΔ	$R_{4} = 8(R_{4})$	BUMP TO NEXT	
	BCT		VES ROOM FOR ANOTHER	
	B	ENDANI	NO WE ARE DONE	
	MVC	O(Q DA) = C(Q'AN) C E DE'		
ADDANES		$D_1 \circ (D_1)$	RIMD TO NEVT	
	DCT			
			NO WE ADE DONE	
			NU, WE ARE DUNE	
ADDANLO	INVC	0(8, R4),=CL8 ANLUSER0	ADD SECUNDARY ID O	
	LA	R4,8(R4)		
	BUI	R3, ADDANL/	YES, ROUM FOR ANOTHER	
	В		NU, WE ARE DONE	
ADDANL7	MVC	$\Theta(8, R4)$,=CL8'ANLUSER7'	ADD SECONDARY ID 7	
	LA	R4,8(R4)	BUMP TO NEXT	
	BCI	R3, ADDANL8	YES, ROOM FOR ANOTHER	
	В	ENDANL	NO, WE ARE DONE	
ADDANL8	MVC	0(8,R4),=CL8'ANLUSER8'	ADD SECONDARY ID 8	
	LA	R4,8(R4)	BUMP TO NEXT	
	вст	R3,ADDANL9	YES, ROOM FOR ANOTHER	
	В	ENDANL	NO, WE ARE DONE	
ADDANL9	MVC	0(8,R4),=CL8'ANLUSER9'	ADD SECONDARY ID 9	
	LA	R4,8(R4)	BUMP TO NEXT	
	BCT	R3,ADDANLA	YES, ROOM FOR ANOTHER	
	В	ENDANL	NO, WE ARE DONE	
ADDANLA	MVC	0(8,R4),=CL8'ANLUSERA'	ADD SECONDARY ID 10	
ENDANL	DS	OH		
*				*
* =====	=====			: *
*				*

2. Optional: Define the secondary authorization IDs in the DSN3@SGN exit.

Consider modifying the DSN3@SGN exit with the same set of IDs that you added to the DSN3@ATH exit. Even though the DSN3@SGN exit is used only for IMS and CICS sign-on authorizations, keeping these two exits consistent is a best practice. This step is identical to step 1 except that you add the secondary authorization IDs to the DSN3@SGN exit immediately after the SSGN090 label.

- 3. Submit the customization job.
- 4. Test the modified DSN3@ATH and DSN3@SGN authorization exits in isolation before putting the exits into production. To facilitate testing in isolation, load the DSN3@ATH and DSN3@SGN exits into the SYS1.DSNEXIT library or a similar stand-alone library so that they can be copied into SYS1.DSNLINK or SYS1.DSNLOAD during a test window and verified.
 - a) Assemble and link the modules into SYS1.DSNEXIT or an equivalent library that does not currently contain the production exits.



Attention: If the library is already on the link list, users have immediate access.

- b) During a test window or on a test system, rename the current exits (if they exist) and copy the new modules into SYS1.DSNLINK or SYS1.DSNLOAD.
- c) From a TSO session, enter the DB2I command. Select an empty file for input and enter the following Db2 statement: SET CURRENT SQLID = 'ANLUSER1'. If the authorization exits have been correctly modified, an SQLCODE of 0 is returned. If no valid secondary authorization ID by that name exists, Db2 returns an SQLCODE of -553, which means that the exit modifications are not working. If additional coding errors are present, the exits might abend. If you encounter a problem, correct the error and retest the exits after you rename the old production exits to their proper names.
- d) If the SQLCODE is 0, set all of the SQL PA secondary authorization IDs that you defined. This step verifies that all the secondary authorization IDs are spelled correctly.
- e) Migrate the DSN3@ATH and DSN3@SGN modules to a production library on the link list (refer to SYS1.PARMLIB member LNKLST*xx*), such as SYS1.DSNLINK or SYS1.DSNLOAD, for access by all users.

Optional: Interfacing with Db2 Query Monitor

You can use an optional task in Tools Customizer to set up the interface between Db2 SQL Performance Analyzer and Db2 Query Monitor. If you are also installing Db2 Query Monitor, you might want to enable this interface.

About this task

This optional configuration task allows Db2 Query Monitor to call Db2 SQL Performance Analyzer for further analysis of problem SQL.

Tools Customizer creates the ANLCIQM CLIST in the *high_level_qualifier*.SANLCLST library for interfacing Db2 SQL Performance Analyzer with the Db2 Query Monitor.

Procedure

To enable Db2 SQL Performance Analyzer to interface with Db2 Query Monitor:

- 1. Select and specify values for all of the required **Set up interface to Query Monitor** task parameters. The ANLCIQM CLIST is created for you.
- 2. Set QDEBUG to NO.

The remaining parameters are set by Db2 Query Monitor upon invocation.

- 3. In Db2 Query Monitor, export SQL text of interest to a data set through the Export SQL Text to DSN panel, shown in Figure 15 on page 73.
- 4. Specify Y in the **Execute SQL PA against exported data set** field, to invoke Db2 SQL Performance Analyzer against the data set where you are exporting SQL text.

Access the Export SQL Text to DSN panel when you view operational summaries of activity, exceptional query activity, or in-process activity.

Example

CQM\$EXPT Export SQL Text to DSN					
Option ===>					
Export to data	<pre>set (Required if data set is a PDS)</pre>				
Execute SQL/PA	against exported data set Y/N Y				
Press Enter to	process request or PF3/CANCEL to exit				

Figure 15. Query Monitor panel to invoke Db2 SQL Performance Analyzer against an exported data set containing SQL

Optional: Configuring Db2 SQL Performance Analyzer for use with QMF

You can optionally configure Db2 SQL Performance Analyzer to interface with QMF. Tools Customizer creates the necessary JCL, but some of the following steps must be completed before running the JCL.

Procedure

- 1. Create the interface between Db2 SQL Performance Analyzer and QMF.
 - a) Copy the contents of the ANLQMF member in the *high_level_qualifier*.SANLSAMP library.
 - b) In the DSQUEGV1 member of the *high_level_qualifier*.SDSQUSRE library, paste the ANLQMF code between the code that is labeled GET ADDRESSES OF DXEXCBA AND DXEGOVA CNTL BLOCKS, which is located before the USING WORK, WORKPTR statement, and the code that is labeled SET

SCOPE VALUE TO ZERO ON INITIALIZATION, which is located before the CLI GOVFUNCT, GOVINIT statement, in the governor exit program in any supported version of QMF.

c) Save the changes to DSQUEGV1.

- 2. Customize the interface between Db2 SQL Performance Analyzer and QMF by modifying the parameters that define the target host system.
 - a) Open the Db2 SQL Performance Analyzer (SQL PA) Governor Interface program, ANLGOV1, which is located in the *high_level_qualifier*.SANLSAMP library, and find the ANLPARM parameters list near the end of the program code.

The following code shows the default ANLPARM settings in ANLGOV1:

ANLPARM ANLCARDS D D D D D D D D D D D D D D D D D D		H'800' OCL800 CL16'SUBSYST DSN ' CL16'SUBSYST DSN ' CL16'SEEMSGS NO ' CL16'VERSION V10R1' CL16'SUBVERS V10NFM' CL16'SUBVERS V10NFM' CL16'SUFHIT 000' CL16'BUFFHIT 000' CL16'DEGREES ANY' CL16'REFRESH ANY' CL8'USEPLAN ' CL8'USEPLAN ' CL8'USEPLAN ALL0W' CL8'ALL0W' CL8'ALL0W' CL16'BUFFERS 00002000' CL16'BUFFERS 00002000' CL16'BUFFERS 00002000' CL16'BUFFERS 00002000' CL16'BUFFISX 00001000' CL16'BUFF32K 00000100' CL16'BUFF32K 00001000' CL16'CPUTIME 00010' CL16'COSTING 00000015' CL16'COSTQUN 100' CL16'COSTQUN 100' CL16'COSTQUN 100' CL16'COSTQUN 100' CL16'CNTMCOST 0010.000' CL16'CNTMCOST 0010.000' CL16'CNTMCOST 0010.000' CL16'CNTMCOST 0010.000' CL16'DSGROUP 0' CL16'DBTRACE NO ' CL16'AUTHIDO ' CL240'	STORE UP TO 50 SQL PA PARMS HERE PASSED TO PLI PGM ANLQMF REQUIRED SUBSYTEM ID FOR CAF MAX= WARNINGS ONLY; ANY=ALL QUERIES SEE INFORMATIONAL ERROR MSGS? CURRENT DB2 RELEASE V10R1 V11R1 DB2 LEVEL IS V10NFM V10CM8 DASD POOL IS 3390 MOD 3 ALL I/O IS PHYSICAL (BP HIT RATIO) ENABLE PARALLEL I/O: ANY OR ONE 1 REFRESH MQTS USE FOR APS: ANY NO SETS THE QUALIFIER FOR UNQUALIFIED OBJECTS SETS THE EXPLAIN TABLE OWNER UNLESS THE VALUE IS 'NOT USED' ALLOWS USEPLAN TO FUNCTION IF THE VALUE IS 'ALLOW' 4K APPLICATION BUFFER POOL 8K APPLICATION BUFFER POOL 16K APPLICATION BUFFER POOL 32k APPLI		
<pre>************************************</pre>					



b) Modify each parameter for your target host system.

The bottom slots are reserved for the future and must not be changed. Any errors in modifying this parameter list can cause unpredictable or incorrect results. The length of the parameter list must remain at 800 characters. Be sure to note the spacing and spelling of parameters.

Tip: Refer to <u>"ANLPARM user parameter settings for QMF" on page 241</u> for the required settings for using Db2 SQL Performance Analyzer with QMF. For optimal results, modify the parameters in the exit to correspond to the specific host machine that is running QMF with Db2 SQL Performance Analyzer.

3. Optional: Force an unconditional cancellation of queries that exceed the defined limits.

If you do not complete this step, Db2 SQL Performance Analyzer gives you the option of canceling each query that exceeds the defined limits. To force an unconditional cancellation of queries that exceed the defined limits:

a) Find the following code fragment in the ANLGOV1 program:

LOGUSER ST	R15,XCBERRET	PUT RETURN CODE IN XCB
MVC	XCBMGTXT,ANLER01	ANLGOV1 LIMITS MESSAGE
MVI	XCBLOGM,XCBLOGMY	SET QMF TO LOG MESSAGE
* (ALT) B	EXIT16	FOR ALWAYS CANCEL, TAKE EXIT16 NOW

- b) Delete the asterisk (*) and the (ALT) from the EXIT16 instruction.
- 4. Optional: To use a Generic ID as the PLAN_TABLE owner, set 'SETPLAN GENERIC' or 'SETPLAN NOTALLOW'.

If there is no Generic ID available and 'SETPLAN NOTALLOW' is specified, the TSO ID is used as the PLAN_TABLE owner. If there is no Generic ID available and 'SETPLAN GENERIC' is specified, an error is reported.

- 5. Save the changes to ANLGOV1.
- 6. Assemble and link-edit ANLGOV1 and DSQUEGV1.
 - a) Edit the JCL for the ANLEXIT procedure, which is located in the *job_sequence_IDGVQMDB2_entry_ID* member of the Tools Customizer jobs.
 - Check the ANLEXIT procedure for valid system library names.
 - Optional: To use a test library for final testing of the exit modifications instead of using the production load library for QMF, change the target output library in the OUTLIB member and locate the two locations within the Tools Customizer JCL that points to that same target output library.

For reference, the processing portion of the JCL is:

```
//*
//ANL1 EXEC ANLEXIT
// MBR=ANLGOV1
//LNK.SYSIN DD *
INCLUDE SYSLIB(ANLQMF)
NAME ANLGOV1(R)
```

b) Save the changes to *job_sequence_IDGVQMDB2_entry_ID*.

- c) Edit the JCL for the QMFEXIT procedure, which is located in the *job_sequence_IDXTQMDB2_entry_ID* member of the Tools Customizer jobs.
 - Check the QMFEXIT procedure for valid system library names.
 - Optional: To use a test library for final testing of the exit modifications instead of using the production load library for QMF, change the target output library in the OUTLIB member and locate the two locations within the Tools Customizer JCL that points to that same target output library.

For reference, the processing portion of the JCL is:

```
//*
//QMF1 EXEC QMFEXIT
// MBR=DSQUEGV1
//LNK.SYSIN DD *
NAME DSQUEGV1(R)
```

d) Save the changes to *job_sequence_IDXTQMDB2_entry_ID*.

e) Submit job_sequence_IDGVQMDB2_entry_ID and job_sequence_IDXTQMDB2_entry_ID.

- 7. Optional: Test the QMF governor exits before putting them into production.
 - a. Preallocate ISPLLIB (or the ISPF load library) to the test library that you chose by using the standard TSO mode without ISPF and entering the following command or equivalent:

ALLOC F(ISPLLIB) DA(test.library) SHR REUS

- b. Restart ISPF and select the QMF option that starts the ANLDSQ1 CLIST or equivalent. When you invoke a QMF query, the Db2 SQL Performance Analyzer Governor Interface program intercepts the query and provides the cost analysis directly to the panel.
- c. Make any further changes to the settings for the governor exits based on your test results.
- d. When you have completed the changes, migrate the ANLGOV1 and DSQUEGV1 modules into the production library for QMF, which is typically SYS1.SDSQLOAD.
- 8. Verify that the ANLCDSQ1 CLIST includes the data sets that are currently used for your QMF and GDDM environments.

Depending on which release of QMF you are currently running, the actual library names and high-level qualifiers might differ. In the ANLCDSQ1 CLIST, SYS3 is used as the high-level qualifier for most data sets.

Table 12. Sample data set names and allocations for QMF files				
File name Sample data set name		Allocation		
ISPPLIB	SYS3.SDSQPLBE	QMF ISPF panel library		
ISPMLIB	SYS3.SDSQMLBE	QMF ISPF message library		
ISPSLIB	SYS3.SDSQSLBE	QMF ISPF skeleton library		
SYSEXEC	SYS3.SDSQEXCE	TSO REXX execution library		
SYSPRINT	* (terminal)	ANLGOV1 and ANLQMF messages		
DSQPRINT	SYSOUT(X)	Print data output to hold class (X)		
DSQDEBUG	SYSOUT(X)	Trace dump output to hold class (X)		
DSQEDIT	work space	Edit transfer file (temp)		
DSQUDUMP	SYSOUT(X)	Output snap dump to hold class (X)		
DSQSPILL	work space	Overflow spill file (temp)		
DSQUCFRM	DSQUCFRM	Saves user-defined ICUFORMS		
DSNPNLE	SYS3.DSQPNLE	QMF panel file		
ADMGGMAP	SYS3.DSQMAPE	GDDM map group for QMF		
ADMCFORM	SYS3.DSQCHART	GDDM chart forms for QMF		
ADMCDATA	SYS3.ADMCDATA	GDDM data file		
ADMGDF	SYS3.ADMGDF	GDDM form definitions file		
ADMSYMBL	SYS3.ADMSYMBL	GDDM symbols file		
ADMPRNTQ	SYS3.ADMPRINT. REQUEST.QUEUE	GDDM output print file		
ADMDEFS	SYS3.GDDMDEF1	GDDM definitions file		

9. Optional: Set up Db2 SQL Performance Analyzer so that you can start it from an existing QMF interface, rather than starting QMF through Db2 SQL Performance Analyzer.

a) Add the following allocations to your startup CLIST (or REXX exec):

ALLOC FI(DSQLLIB) DA('QMF####:SDSQLOAD', 'sysx.hiqual.SANLLOAD', + 'DSN####:SDSNEXIT', 'DSN####:SDSNLOAD') SHR REUSE ISPEXEC LIBDEF ISPLLIB LIBRARY ID(DSQLLIB) ALLOC FI(SYSPRINT) DA(*) SHR

These statements cause the QMF load library, Db2 SQL Performance Analyzer program load library, and Db2 exit and load libraries to become available.

- b) If you install the modified QMF Governor exit into a different library, add the library to the list of allocated libraries as the first entry. The SYSPRINT allocation is required by Db2 SQL Performance Analyzer to direct warning messages to the terminal.
- c) Optionally, use the following code at the end of your QMF CLIST (or REXX exec) to free the allocated libraries.

```
ISPEXEC LIBDEF ISPLLIB LIBRARY ID( )
FREE FI(DSQLLIB)
```

Optional: Interfacing with Db2 Administration Tool

If Db2 Administration Tool (Db2 Admin) is installed, you can integrate it with Db2 SQL Performance Analyzer by using Tools Customizer to run the Add SQL PA to Db2 Admin Launchpad task.

About this task

A REXX exec, ANLCADBI, is provided in the *high_level_qualifier*.SANLCLST library which allows Db2 SQL Performance Analyzer to interface with Db2 Admin.

Procedure

- 1. Start Tools Customizer.
- 2. Navigate to the Product Parameters panel from the Customizer Workplace panel.
- 3. Specify values for all of the parameters listed under the Add SQL PA to DB2 Admin Launchpad task.
- 4. Generate the customization job.
- 5. Optional. If Db2 SQL Performance Analyzer will be the first entry to the Db2 Admin, run Db2 Admin's ADBL with the parameter DMT.

EXEC hlq.SADBCLST(ADBL) 'DMT'

A dialog box displays. Press enter to confirm the initial table values and PF3 to exit.

This step builds the required Data Management Tools table, and ANLCADBI adds the Db2 SQL Performance Analyzer entry to the Db2 Admin Launchpad

6. Run the customization job.

Example

Running the Db2 Admin CLIST ADBL displays the Data Management Tools Launchpad panel, as shown in the following figure.

DB2 Tools Launchpad Row 1 from 8 Command ===> Scroll ===> CSR					
Specify DB2 SSID (opt) ===> (Enter '?' for a list of active SSIDs)					
Select the DB2 tool you wish to launch or enter its code in the command line.					
Sel Code Tool Name Rel Prog No.					
ADM DB2 Administration Tool 102 5697-L90					
APPLICATION MANAGEMENT TOOLS No table entries in this category					
CDA COL Devicementation Analysis Analysis Color File File File File File File File File					
SPA SUL PETIOIMANCE ANALYZET 510 5697-W51					
RECOVERY AND REPLICATION MANAGEMENT TOOLS No table entries in this category ************************************					

Figure 17. Data Management Tools Launchpad

Optional: Set up ANLCSPA for use from Browse or Edit sessions

To invoke SQL PA from a browse or edit session, you can use Tools Customizer to run the "Set up ANLCSPA for use from Browse or Edit sessions" task, or you can manually run a CLIST to allocate the CLIST library to SYSPROC.

About this task

If you want to use Tools Customizer, the **Set up ANLCSPA for use from Browse or Edit sessions** task contains the following optional steps, which you can implement in any combination that you want.

- Create a library that is variable length and it copies all the members in SANLCLST into the new library, which is named SANLVCLS. You can use this library in an allocation to SYSPROC.
- Copy ANLCSPA to a library that is already allocated to SYSPROC. If you do not want to allocate the entire library, you can copy ANLCSPA to a different library.

To invoke ANLCSPA from any browse or edit session, the CLIST library must be allocated to SYSPROC. The preferred method to invoke ANLCSPA is through the TSO logon procedure. The CLIST ANLCALLC is provided for you to use, if the allocation is necessary after the logon procedure.

Procedure

- 1. Optional: Use Tools Customizer to generate one or more jobs that will help to set up ANLCSPA.
 - a) Start Tools Customizer.
 - b) Navigate to the **Product Parameters** panel from the **Customizer Workplace** panel.
 - c) Select the Set up ANLCSPA for use from Browse or Edit sessions task.
 - d) Select one or more of the optional steps.
 - e) Specify values for all of the required parameters.
 - f) Generate the customization jobs.
 - g) Run the customization jobs.
- 2. Optional: Manually run ANLCALLC in the SANLCLST library to allocate the SANLCLST library to SYSPROC.

Optional: Enabling the stored procedure customization for Db2 SQL Performance Analyzer

You must complete several tasks if you used the optional Tools Customizer task to enable the use of stored procedures for SQL Performance Analyzer.

Important requirements for using the stored procedure

Using the stored procedure in Db2 SQL Performance Analyzer has some requirements.

A requirement for using the Db2 SQL Performance Analyzer stored procedure is that programs that call any stored procedure must be compiled and link edited with Language Environment (LE) compatible libraries. If LE is not used, the programs fail to initialize properly. See the Language Environment Programming Guide for more information.

To maintain good program performance, the maximum size of an SQL statement supported by the SQL PA stored procedure is 32,000 bytes. For larger SQL statements, consider using the TSO or batch program interfaces for evaluation.

Requirements for allocating the stored procedure load library

You can change the load library that is allocated for the stored procedure.

WLM stored procedure address spaces are started by MVS[™] automatically, but they also have STEPLIB allocations. Stored procedures must run from an allocated STEPLIB, which must not be the *high_level_qualifier*.SANLLOAD library. The Db2 SQL Performance Analyzer stored procedure load modules are copied to the *high_level_qualifier*.SANLLODS library during customization.

Restrictions:

- Do not place ANLPRCR in a library that is on the MVS link list.
- This library must not be APF-authorized.
- Stored procedures must run from an allocated STEPLIB, which must not be the *high_level_qualifier*.SANLLOAD library.

Customizing the stored procedure address space startup procedure

You can customize the Db2 stored procedure address space to run the Db2 SQL Performance Analyzer (SQL PA) stored procedure.

About this task

SQL PA provides a sample startup procedure named *ssid*WLMR in the *high_level_qualifier*.SANLJCL library.

The stored procedure address space startup procedure JCL must contain an ANLCNTL DD statement that refers to the set of standard ANLCNTL parameters that are normally used by SQL PA. Also, the address space startup procedure must contain DD cards for SDSNLOAD, SDSNEXIT, and SCEERUN.

For a stored procedure to read a non-DB2 file, the file must be allocated in the stored procedure address space startup JCL. The stored procedures, ANLPRER and ANLPRCR, can write messages to SYSPRINT. However, SYSPRINT must be allocated to the stored procedure address space to receive those messages. The load library that contains the SQL PA stored procedure must be part of the STEPLIBs allocated, and be separate from the TSO/batch load library.

To customize this startup procedure (if you do not use *ssid*WLMR):

Procedure

• If the ANLPRER or ANLPRCR stored procedure was loaded into a library that is not already defined by the STEPLIB DD statement, concatenate the library that contains the stored procedure to the STEPLIB DD statement.

Restrictions:

- Do not place ANLPRCR or ANLPRER in a library that is on the MVS link list.
- This library must not be APF-authorized.
- Stored procedures must run from an allocated STEPLIB, which must not be the *high_level_qualifier*.SANLLOAD library.
- Add a //SYSPRINT DD SYSOUT=* statement to the address space startup procedure to direct any messages written by the SQL PA stored procedure.
- Add a //ANLCNTL DD DSN=, DISP=SHR statement to the address space startup procedure to allocate a file that contains a set of configuration parameters for use by all SQL PA stored procedure users.

The data set cannot be the same ANLCNTL data set that is used elsewhere for SQL PA batch and TSO operations.

After the address space is active, the data set (*file_name*) is considered in use and cannot be modified. Edit the *ssid*CNTL member, which is located in *high_level_qualifier*.SANLDATA, and customize those parameters to describe the host system for the stored procedure.

• Insert rows into the catalog.

Results

After you have customized the startup procedure, customize the stored procedure ANLCNTL parameters.

Customizing the stored procedure ANLCNTL parameters

After you have created, customized, and inserted rows into the catalog for the Db2 SQL Performance Analyzer stored procedures, customize the parameters for ANLCNTL.

Procedure

Examine the parameters in *ssid*CNTL in the *high_level_qualifier*.SANLDATA library that you used when you customized the stored procedure address space startup procedure, and customize them to describe the target host configuration that you want, set the limits, warnings, and cost algorithm.

For more information about the ANLCNTL configuration parameters, refer to <u>"ANLCNTL configuration</u> parameters" on page 226.

What to do next

Start the stored procedure address space.

Calling the Db2 SQL Performance Analyzer stored procedure

After the stored procedure address space has been started, you can call the Db2 SQL Performance Analyzer stored procedure.

About this task

A call to the Db2 SQL Performance Analyzer stored procedure for cost analysis of any SQL statement is accomplished by passing a parameter string. The parameter string for ANLPRCR consists of the names of 11 host variables. The host variables are positional, which means that the stored procedure expects a specific data type and value for a variable depending on its position in the string. The host variable names that are used are arbitrary.

Restriction: All programs that call the Db2 SQL Performance Analyzer stored procedure must be compiled and link edited with LE/370 compatible libraries. If not, the program fails to initialize properly.

The following example shows a typical call to the ANLPRCR stored procedure.

EXEC SQL CALL ANLPRCR (:SQL_STMT, :SQL_LEN, :ANL_PARM, :ANL_CPU, :ANL_ELAP, :ANL_IOC, :ANL_QUN, :ANL_MONY, :ANL_WARN, :ANL_CODE, :SQL_CODE);

The following table uses the names in the example to describe the characteristics and use of each host variable.

Table 13. SQL PA stored procedure call parameters					
Host variable	Input or output	Data type	Contents		
1 SQL_STMT	Input	VARCHAR(32000)	Contains the SQL SELECT, INSERT, UPDATE, or DELETE statement to be evaluated, up to 32000 characters long.		
2 SQL_LEN	Input	SMALLINT	Contains the length of the SQL statement.		
3 ANL_PARM	Input	CHAR(240)	Contains up to 15 Db2 SQL Performance Analyzer user parameters. See <u>"Specifying</u> user parameters to the SQL Performance Analyzer stored procedure" on page 82.		
4 ANL_CPU	Output	DOUBLE	ANLPRCR returns the CPU time estimate in seconds in floating point format (8 byte, double precision).		
5 ANL_ELAP	Output	DOUBLE	ANLPRCR returns the elapsed time estimate in seconds in floating point format.		
6 ANL_IOC	Output	DOUBLE	ANLPRCR returns the I/O count estimate in floating point format.		
7 ANL_QUN	Output	DOUBLE	ANLPRCR returns the Db2 SQL Performance Analyzer QUNITS (query units) estimate in floating point format.		
8 ANL_MONY	Output	DOUBLE	ANLPRCR returns the monetary cost estimate in floating point format.		
9 ANL_WARN	Output	CHAR(5)	ANLPRCR returns a string of five (5) Y N flags for the CPU time, elapsed time, I/O count, QUNITs, and money estimates, in that order. If an estimate exceeds the installation limit, the flag contains a 'Y'.		
10 ANL_CODE	Output	INTEGER	Contains the return code from ANLPRCR. The code value is 0 if no limits were exceeded, or 16 if one or more limits were exceeded. On certain errors, it might contain another positive integer.		
11 SQL_CODE	Output	INTEGER	Contains the last SQL code from the Db2 SQL Performance Analyzer PREPARE or EXPLAIN statement.		

The call parameter list for the EXPLAIN-capable stored procedures (ANLPRER) is slightly different, as shown in the following figure. The fields marked with an asterisk (*) are unique to the ANLPRER stored procedure.

	EXEC SQL CALL ANLPRE	ER :/ :/ :/	(:SQL_STMT, :SQL_LEN, :ANL_PARM, ANL_CPU, :ANL_ELAP, :ANL_IOC, ANL_QUNT, :ANL_COST, :ANL_WARN, ANL_CODE, :SQL_CODE, :DB2_EXPL, DB2_RECS);
* *	SQL_STMT CHAR(32000) VAR SQL_LEN BIN FIXED (15) ANL_PARM CHAR(240) ANL_CPU BIN FLOAT (53) ANL_ELAP BIN FLOAT (53) ANL_OUNT BIN FLOAT (53) ANL_COST BIN FLOAT (53) ANL_COST BIN FLOAT (53) ANL_CODE BIN FIXED (31) SQL_CODE BIN FIXED (31) DB2_EXPL CHAR(31920) DB2 PECS BIN FIXED (15)		CONTAINS THE SQL STATEMENT INPUT CONTAINS LENGTH OF SQL_STMT INPUT CONTAINS USER PARAMETERS: 15 MAX INPUT CONTAINS OUT CPU TIME ESTIMATE CONTAINS OUT ELAPSED TIME ESTIMATE CONTAINS OUT I/O COUNT ESTIMATE CONTAINS OUT QUNITS ESTIMATE CONTAINS OUT MONETARY COST ESTIMATE HOLDS 5 WARNING FLAGS FOR THE ABOVE SQL/PA RETURN CODE: 0 GOOD; +N BAD SQL CODE FOR LAST PREPARE/EXPLAIN CONTAINS PLAN & STAT RECORDS OUTPUT CONTAINS COUNT OF THE PLAN+STAT PATPS

Figure 18. ANLPRER call parameter list

Specifying user parameters to the SQL Performance Analyzer stored procedure

The SQL Performance Analyzer stored procedure call parameter list contains a list of SQL Performance Analyzer user parameters in the third host variable. You can specify up to 15 user parameters in this list.

Procedure

- 1. To specify user parameters to the SQL Performance Analyzer stored procedure rather than using the default settings, perform the following steps:
 - a) Enter the name and the value for the SQL PA parameter, preceded and followed by at least one blank.

The following set of user parameters are supplied by default:

```
REPORTS STP (STP is required)
VERSION V10R1
STORAGE 3390-3
REFRESH ANY
DEGREES 1
QUALIFY authid
CONNECT CAF (or RRSAF)
PRECISE YES
```

Requirement: The first parameter must be the REPORTS STP parameter. The list is 240 bytes long. You can supply up to 15 parameter and value sets. Each can be a maximum of 16 characters long, including blanks.

The default value for the VERSION parameter, which indicates the version of Db2 that you are using, is V10R1. If you use a different version of Db2, you must edit the VERSION parameter, especially if no PLAN_TABLE exists. Otherwise, the parameters are automatically created in Db2 10 format.

2. To report access plans and catalog statistics in the ANLPRER program, add the RETPLAN YES parameter to the passed parameter list.

Example

The following figure shows a sample of the formatted output.

```
* CAF OPEN RETCODE IS
                                                                 0
     EXPLAIN PLAN FOR (LENGTH 480)
         SELECT COUNT(*) FROM
*
         SYSIBM.SYSPACKSTMT A, SYSIBM.SYSPACKDEP B
WHERE A.LOCATION = ? AND
B.DLOCATION = ? AND
*
*
*
                  B.DCOLLID = ? AND
*
                  A.COLLID <> ?
* ANLPRER SQLCODE IS 0
* ANLPRER RETURNS ==> WARNING FLAGS: ---Y-
                         10.34282 CPU TIME:
     ELAPSED:
                                                                          7.24000
   I/O COUNT:
                                     6
                                               QUNITS:
                                                                          160886
                                    1.61
   MONETARY:
                                                SQL CODE:
   ANL CODE:
                                                                                    0
                                       0
     RECORDS:
                                        2
 The DB2 Access Plan:
               Access Flan:.00000001 Blk:1 Pln:1 Mth:0 Typ:Mix:0 RidTno:1 Table:SYSIBM. SYSPACKSTMTCorr:ASYSIBM. DSNKSX01Ixo:Y Mtch:1 SortC|N UJOG:Mrg:-1 Adg:-1 Aid:-1 Jdg:-1 Jid:-1 Cid:Typ:T Enc:CCSID S|M|D:00 Coll:SQLPAPref:S Lock:IS App:Pgm:ANLPRER Ver:
  Qry: 100000001 Blk:
                                                                                                           Mix: 0 Rid:
                                                                                                                                            Bif:
 Acc: I
  Index: SYSIBM
                                                                                                     1 SortC|N UJOG: NNNN NNNN
 Jon:
                                                                                                                                   -1 Par:
  Rng:
 Grp:
  The DB2 Statistics used:

        Ncol:
        18 Rowz:
        404 CARD:
        21

        Npag:
        14830 Pctp:
        90 COMP:
        0 Encd:
        E Tsty:
        Part:

        Nidx:
        1 Pgsz:
        4 Ntab:
        8 Nact:
        15543 Sgsz:
        4 Tclo:

        Lkpt:
        N Maxr:
        255 Lksz:
        A Crat:
        0.000000 Cled:
        N Cing:
        Y Ixsz:
        4096

        Iclo:
        N Leaf:
        -1 1key:
        -1 Fkey
        -1 Fkey

                                                                                                                                              212216
                                                                                                                                                0
                                                                                                         15543 Sgsz: 4 Tclo: N
                                           -1 1key:
5 Iunq: U Ityp: 2 Piec:
                Leaf: -1
-1 Kcol: 5 Iu
0 Iver: 0 Nmqt:
                                                                                                                                                      -1
  Nlvl:
                                                                                                               0 Log: Y
                                                      0 Ailn:
                                                                                0 Arln:
                                                                                                     404
  Tver:
                            0 Pools: BP0
                                                                                Volt: N Ipad: - Dpsi:
  Dsiz:
                                                               BP0
                                                                                                                                      0 Irel: Y
  Proc MS:
                              7240 Proc SU:
                                                                160886 Category: A Reason: NORMAL
 The DB2 Access Plan:
                                           1 Pln:
  Qry: 100000001 Blk:
                                                              2 Mth: 1 Typ:
                                                                                                           Mix:
                                                                                                                         0 Rid:
                                                                                                                                            Bif:
               Tho: 2 Table: SYSIBM . SYSPACKDEP Corr: B

SYSIBM . DSNKDX01 Ixo: Y Mtch: 2 SortC|N UJOG: NNNN NNNN

Mrg: -1 Adg: -1 Aid: -1 Jdg: -1 Jid: -1 Cid: -1 Nid: -1 Par:

Typ: T Enc: CCSID S|M|D: 0 0 0 Coll: SQLPA

Prof: Lock: TS App: Pgm: ANL PPEP Ver:
  Acc: I
  Index: SYSIBM
 Jon:
  Rng:
                                          Lock: IS App:
 Grp:
                            Pref:
                                                                                         Pgm: ANLPRER Ver:

      Tty: T Loc:
      Ncol: 10 Rowz: 97 CARD:
      1

      Npag:
      292 Pctp: 90 COMP:
      0 Encd: E Tsty:
      Part:

      Nidx:
      3 Pgsz:
      4 Ntab:
      8 Nact:
      15543 Sgsz:
      4 Tclo:

      Lkpt: N
      Maxr:
      255 Lksz:
      A Crat:
      0.000000 Cled:
      N Cing: Y Ixsz:
      4096

      Iclo: N
      Leaf:
      -1 1key:
      -1 Ekey:
      -1 Ekey:

  The DB2 Statistics used:
                                                                                                                                                11829
                                                                                                                                                 0
                                                                                                         15543 Sgsz: 4 Tclo: N
                Leaf: -1 1key:
-1 Kcol: 4 Iunq: D Ityp: 2 Piec:
0 Iver: 0 Nmqt: 0 Ailn: 0 Arln
                                                                                                                                                      -1
                                                                                                               0 Log: Y
  Nlvl:
                                                                                0 Arln:
                                                                                                       97
  Tver:
  Dsiz:
                             0 Pools: BP0
                                                               BP0
                                                                                Volt: N Ipad: - Dpsi:
                                                                                                                                      0 Irel: Y
                                                                160886 Category: A Reason: NORMAL
 Proc MS:
                              7240 Proc SU:
```



Migrating SQL Performance Analyzer from one Db2 version to another Db2 version

After you have migrated your Db2 subsystem, you must configure SQL Performance Analyzer for the migrated Db2 subsystem. **Related concepts** Customizing SOL PA After you install Db2 SQL Performance Analyzer (SQL PA) by following the installation instructions in the Program Directory, you must run Tools Customizer to specify the variables for each Db2 subsystem and to customize the configuration and user parameters.

Migrating SQL Performance Analyzer to Db2 10

To migrate SQL Performance Analyzer to Db2 10, complete the following steps.

About this task



Attention: Enabling-new-function mode (ENFM) is not supported.

In Db2 10, EXPLAIN tables that are in pre-Version 8 format are no longer supported. EXPLAIN tables that are in Version 8 or Version 9 format and EBCDIC-encoded EXPLAIN tables are deprecated.

For more information about the restrictions for EXPLAIN tables in Db2 10 and about jobs for completing this task, see the DB2 10 for z/OS Installation and Migration Guide.

Procedure

- 1. Convert your EXPLAIN tables to UNICODE encoding. If you converted your EXPLAIN tables when you migrated your Db2 subsystems to Db2 10, skip this step.
 - a) Edit and submit the *high-level-qualifier*.SDSNSAMP(DSNTIJXA) job. This job changes the format of the EXPLAIN tables to the format of the current Db2 version.
 - b) Edit and submit the *high-level-qualifier*.SDSNSAMP(DSNTIJXB) job. This job generates SQL to create the new objects in UNICODE, rename the existing EXPLAIN tables, and insert the data from these tables into the new tables.
 - c) Run the generated SQL from SPUFI or edit the *high-level-qualifier*.SDSNSAMP(DSNTIXJC) job. This job uses the Db2 cross-loader control statements that are generated in the DSNTIJXB job.
- 2. Use Tools Customizer to customize SQL Performance Analyzer for Db2 10.
 - a) Edit the Db2 parameters for the migrated subsystems.
 - b) Regenerate the customization jobs.
 - c) Submit only the jobs that are customized from the ANLDROPD template, the ANLCRE8I template, the ANLCRE8D template, and the ANLBNDGR template as identified in the TEMPLATE column on the **Finish Product Customization** panel.

Important: Submitting the job to drop the Db2 objects will delete all EXPLAIN tables in the SQL Performance Analyzer database, and you will lose SQL Performance Analyzer history. If you need the existing data in the EXPLAIN tables, save the data before you submit the job, and restore the data after the objects are created.

Migrating SQL Performance Analyzer to Db2 11 or Db2 12

To migrate SQL Performance Analyzer to Db2 11 or Db2 12, complete the following steps.

About this task



Attention: Enabling-new-function mode (ENFM) is not supported.

Procedure

Use Tools Customizer to customize SQL Performance Analyzer for Db2 11 or Db2 12.

a) Edit the Db2 parameters for the migrated subsystems. It is important to ensure that "Drop database when dropping Db2 objects" is set to NO. If not, everything in the database will be dropped when the customized job from ANLDROPD is run.

- b) Regenerate the customization jobs.
- c) Submit only the jobs that are customized from the ANLDROPD template, the ANLCRE8I template, the ANLCRE8D template, and the ANLBNDGR template as identified in the TEMPLATE column on the **Finish Product Customization** panel.

Important: Submitting the job to drop the Db2 objects will delete all EXPLAIN tables in the SQL Performance Analyzer database, and you will lose SQL Performance Analyzer history. If you need the existing data in the EXPLAIN tables, save the data before you submit the job, and restore the data after the objects are created.

Chapter 3. Running SQL PA

You can run SQL PA as a batch job or interactively under TSO using ISPF panels.

In either case, processing is similar and the reporting options are the same. SQL PA also runs under QMF as an interceptor, providing costing information for SQL statements in flight, prior to running. It can be invoked from any program with Db2 access by calling the SQL PA stored procedure or from any browse or edit session by using the ANLCSPA macro.

SQL PA can analyze any Db2 runtime environment—users run SQL PA with a specific configuration (the target host system) and application scenario. These different environments are described by the catalog on the system where processing takes place. Catalog statistics are set to represent production volumes for a full production system while it is still under development. SQL PA users can forecast performance of SQL under CICS, IMS, batch or any other attach facility, on any real or virtual target host machine.

Topics:

- "Using SQL PA in batch" on page 87
- "Running SQL PA using the ISPF interface " on page 92
- "Using SQL PA in QMF" on page 117
- "Using the SQL PA stored procedure" on page 119
- "Plan Table report levels" on page 121
- <u>"EEEPATH table" on page 123</u>
- "SQL data file: user_ID.ANLEEE.SQL" on page 123
- "VIADRDA remote connect" on page 123

Using SQL PA in batch

In batch, SQL PA is run as a single step. Input parameter files, SQL and work data sets, and reports, are defined in this information.

Topics:

- "SQL PA Batch JCL" on page 87
- "Comparing old and new packages in batch" on page 88
- "Testing current access paths for packages in batch" on page 89
- "Generating JCL for batch processing" on page 91
- "Required ddnames" on page 91
- "Using debug batch JCL" on page 92
- "Sort Query Limits Report batch JCL" on page 92

SQL PA Batch JCL

You can use this sample JCL for batch processing of your SQL PA job.

The sample JCL in the following figure shows the allocations necessary to run Db2 SQL Performance Analyzer under a sample user ID of TDT690. It includes an optional first step that deletes information left after a previous run.

This sample JCL is not intended to be used with generic IDs. To use generic IDs in batch, use the BATCH command to generate the JCL. If generic IDs are used incorrectly in batch, the IDs might not be properly freed in the REGISTRY. See <u>"Resetting the Db2 SQL Performance Analyzer REGISTRY" on page 213</u> if no more generic IDs are available.

//* SQL PA -- SQL PEFORMANCE ANALYZER * //***** //* //* PROGRAM PROPERTY OF IBM CORPORATION PRODUCT NUMBER 5697-W51 //* (C) COPYRIGHT 2000-2015 BY IBM (C) 1993-2005 BY IMSI ALL RIGHTS RESERVED WORLDWIDE //* SQL PA IS AN IBM LICENSED PROGRAM //* //́★ ALL DB2 RELEASES USE PROGRAM 'ANLSQLPA' //★ //* STEP 0: DELETE THE PREVIOUS RUN'S PERMANENT REPORTS, IF NECESSARY //* //ANLSTEP0 EXEC PGM=IEFBR14 //GOAWAY1 DD DSN=&SYSUID..ANLREP.RPT,DISP=(MOD,DELETE), SPACE=(TRK, (1,1), RLSE) //GOAWAY2 DD DSN=&SYSUID..QTRACE.RPT,DISP=(MOD,DELETE), SPACE=(TRK, (1,1), RLSE) 11 //GOAWAY3 DD DSN=&SYSUID..QLIMIT.RPT,DISP=(MOD,DELETE), 11 SPACE=(TRK, (1,1), RLSE) //GOAWAY4 DD DSN=&SYSUID..ANLOUT.RPT,DISP=(MOD,DELETE), SPACE=(TRK,(1,1),RLSE) // //* //* //ANLSTEP1 EXEC PGM=ANLSQLPA,REGION=OM 1/* //STEPLIB DD DSN=SYSX.ANL510.SANLLOAD,DISP=SHR SOL PA DD DSN=SYS1.CEE.SCEERUN,DISP=SHR || || 1 F DB2 DD DSN=SYS1.DSN####.SDSNLOAD,DISP=SHR //* //*SNTRACE DD SYSOUT=* TRACE 1/* //ANLCNTL DD DSN=SYSX.ANL510.SANLPARM(SSIDCNTL),DISP=SHR //ANLPARM DD DSN=SYSX.ANL510.SANLPARM(SSIDPARM),DISP=SHR //****** //ANLWORK DD DSN=&&ANLWORK,DISP=(,DELETE,DELETE) SPACE=(CYL, (5,1), RLSE), DCB=(LRECL=80, RECFM=FB, BLKSIZE=4000) 11 //****** //ANLIN DD DSN=SYSX.ANL510.SANLSQL(ANLSTEST), DISP=SHR //****** DD DSN=&SYSUID..ANLOUT.RPT,DISP=(,CATLG,DELETE) //ANLOUT SPACE=(CYL, (1, 1), RLSE), DCB=(LRECL=80, RECFM=FB, BLKSIZE=9040) 11 //* //SYSPRINT DD SYSOUT=*,DCB=LRECL=133 //ANLPRINT DD SYSOUT=*,DCB=LRECL=133 //* //* YOU CAN RECOGNIZE WHICH INPUT SQL PRODUCED THIS PARTICULAR REPORT. //* //* QTRACE IS THE DETAILED TRACE REPORT: WRITTEN WHEN "ALL" REPORTS //* ARE REQUESTED. CONTAINS A MORE DETAILED EXECUTION FORECAST. //* //* QLIMIT IS THE QUERY LIMITS REPORT: IT CONTAINS A FLAG FOR EACH //* LIMIT EXCEEDED, PLUS COST VALUES, ONE LINE PER QUERYNO. //* //* EP DD DSN=&SYSUID..ANLREP.RPT,DISP=(,CATLG,DELETE), SPACE=(CYL,(1,1),RLSE),DCB=(LRECL=132,RECFM=FB,BLKSIZE=23760) //ANLREP //QTRACE DD DSN=&SYSUID..QTRACE.RPT,DISP=(,CATLG,DELETE) SPACE=(CYL, (1,1), RLSE), DCB=(LRECL=120, RECFM=FB, BLKSIZE=23520) 11 //QLIMIT DD DSN=&SYSUID..QLIMIT.RPT,DISP=(,CATLG,DELETE), // //* SPACE=(TRK, (1,1), RLSE), DCB=(LRECL=120, RECFM=FB, BLKSIZE=23520)

Comparing old and new packages in batch

COMPARE compares the access paths of two versions of a bound package.

You can use the sample JCL member, ssidJCMP, in SANLJCL to perform a COMPARE. The output reports only the changed statements.

Input commands

ACTION

Valid values for the ACTION command are COMPARE and END. Each COMPARE that is run must start with ACTION COMPARE and end with ACTION END. Multiple COMPAREs may be run in one JCL execution, but each one must begin and end with the ACTION command. This command is required.

COLLID

The value of the COLLID command indicates the collection id of the current package to be compared. The value must be enclosed in quotes and may contain a wild card character such as "ABC*". If a wild card character is used, both current and previous collection ids must be identical. This command is required.

COLLIDP

The value of the COLLIDP command indicates the collection id of the previous package to be compared. The value must be enclosed in quotes and may contain a wild card character such as "ABC*". If a wild card character is used, both current and previous collection ids must be identical. This command is optional.

CURSQLID

The value of the CURSQLID command indicates the owner of the explain tables that hold the access path information for the current package. The value must be enclosed in quotes. This command is optional and defaults to the current SQLID if not specified.

CURVERS

The value of the CURVERS command indicates the version of the package that is to be used as the current package. The value may be a quoted version id or it may be a relative version number, with 0 as the most recent version, -1 as the next older version, etc. This command is optional and defaults to 0 if not specified.

PKGNAME

The value of the PKGNAME command indicates the name of the package to be compared. The value must be enclosed in quotes and may contain a wild card character such as "ABC*". This command is required.

PREVSQLID

The value of the PREVSQLID command indicates the owner of the explain tables that hold the access path information for the previous package. The value must be enclosed in quotes. This command is optional and defaults to the current SQLID if not specified.

PREVVERS

The value of the PREVVERS command indicates the version of the package that is to be used as the previous package. The value may be a quoted version id or it may be a relative version number, with 0 as the most recent version, -1 as the next older version, etc. This command is optional and defaults to -1 if not specified.

STMTMATCH

The value of the STMTMATCH command indicates the method used to match the statements between package versions. The valid values are 1, Q, and S. A value of 1 matches by table and query type. A value of Q matches by query number. A value of S matches by the sequence of the statements. This command is optional and defaults to 1 if not specified.

Testing current access paths for packages in batch

TEST compares the access paths for a new DBRM or stored procedure instance to the access paths of a previously bound package, using either the package SQL statements from a DBRM library or from the catalog.

TEST requires input from explain table data saved in your explain tables when the package was last bound. You can use the sample JCL member, ssidJTST, in SANLJCL to perform a TEST. The output reports only the changed statements.

There are two popular use cases for TEST:

- First, during a migration from one DB2 level to the next, you can TEST bound packages to detect likely access path and cost changes. Use of wildcards in the PKGNAME should help to process multiple packages in one run.
- Second, before binding a new instance of a package during an application change, you can test just that package to determine if an access path change is likely, and the effect of that change on the performance and cost of package execution.

Input commands

ACTION

Valid values for the ACTION command are TEST and END. Each TEST that is run must start with ACTION TEST and end with ACTION END. Multiple TESTs may be run in one JCL execution, but each one must begin and end with the ACTION command. This command is required.

CURSQLID

The value of the CURSQLID command indicates the owner of the explain tables that hold the access path information for the current package. The value must be enclosed in quotes. This command is optional and defaults to the current SQLID if not specified.

DBRMLIB

The value of the DBRMLIB indicates a fully qualified partitioned data set that contains the DBRM to be used to obtain the statements for testing. The value must be enclosed in quotes. This command is only required if FROMCATLG has a value of N.

EXPLTYP

The value of the EXPLTYP command indicates the type of explain that will be used. The choices are D (Dynamic) or X (eXplain only).

- If **D** is entered, the explain is performed using dynamic path selection.
- If **X** is entered, the explain is performed using a bind with the EXPLAIN(ONLY) option. This option uses more system resources, but produces a more accurate access path prediction.

EXPSQLID

The value of the EXPSQLID command indicates the name of SQLID to run EXPLAIN on the specified packages. If Generic ID is required, the value must be "+OFF+". Otherwise the value may be a quoted SQLID. This command is optional and defaults to the current SQLID if not specified.

FROMCATLG

The value of the FROMCATLG command indicates where the statements used for comparing to the package will be obtained. The valid values for the FROMCATLG command are C, N, and Y. A value of N indicates that the statements will be extracted from a DBRM member in the DBRMLIB library. A value of Y indicates that the statements will be extracted from the catalog. A value of C indicates that the statements will be extracted from a DBRM member in the DBRMLIB library if it is found, otherwise the statements will be extracted from the catalog. If a value of N is used, the DBRMLIB command is required. This command is optional and defaults to C if not specified.

PKGNAME

The value of the PKGNAME command indicates the name of the package to be tested. The value must be enclosed in quotes and may contain a wild card character such as "ABC*". This command is required.

STMTMATCH

The value of the STMTMATCH command indicates the method used to match the statements. The valid values are 1, 2, Q, and S. A value of 1 matches by table and query type. A value of 2 matches by table, query type, and statement text. A value of Q matches by query number. A value of S matches by the sequence of the statements. This command is optional and defaults to 1 if not specified.

Generating JCL for batch processing

You can use the BATCH command to generate JCL to be used in batch processing.

Procedure

- 1. From any processing panel off the basic processing panel (these panels include ANLPPKGE, ANLPDBRM, ANLPFILE, ANLPQNO, ANLPQMF, ANLPTCMP, ANLPTTST and ANLPSDQ), enter the **BATCH** command.
- 2. Enter the data set to which you want to save the JCL.

The JCL for running that job is saved in the data set that you specified.

What to do next

Submit the saved JCL to run the job in batch.

Required ddnames

The following list contains descriptions of the roles of the DD statements.

ANLCNTL

Points to a parameter set. These parameters describe the target host configuration, the costs of various resources in that configuration, and the threshold values for excessive use warning message.

ANLIN

Points to the input data set, which is a sequential file, a member of a partitioned data set (PDS), or a DBRM library member. It can be a file containing SQL statements created by Path Checker. It can also point to an entire DBRM library (no member name is specified) if the DBRMKEY parameter is used to identify the member or members to be processed. If the DBRMKEY parameter is set to OFF, only one fully qualified DBRM name can be specified. If more than one DBRM is required, ANLIN*nn* can also be used.

The input source is scanned by ANLSQLPA and all executable SQL statements are extracted from that file's contents. These SQL statements are then submitted to Db2 for the latest Explain, using the current catalog statistics. The tables and indexes chosen on the access path are referenced in the catalog to retrieve the necessary sizing information.

ANLINnn

Points to the input data set, which is a fully qualified DBRM It can be a sequential or a member of a partitioned data set (PDS). It is used along with ANLIN to work with more than one DBRM when those DBRMs cannot be specified by using a single DBRM library with a DBRMKEY to identify the member.

ANLOUT

Contains the actual SQL processed by SQL PA along with a date and time stamp on first record.

ANLPARM

Points to a parameter set. These parameters describe user-specific controls for the application being analyzed, such as the buffer hit ratio, the type attach facility, and the level of reporting by SQL PA.

ANLPRINT

Provides the output data set containing the Cost Summary report.

ANLREP

Provides the output data set for Enhanced Explain report. It is required if the ANLPARM parameter REPORTS is set to REP or DET.

ANLWORK

A temporary work file used when multiple DBRM support is activated by using the DBRMKEY parameter. You can leave the ANLWORK allocation in your batch JCL even if multiple DBRM support is not used.

The DCB parameters for LRECL and RECFM for this file must be set as required by SQL PA.

QLIMIT

Provides the output data set for the QLIMITS (query limits) report.

QTRACE

Provides the output data set for the Detail Trace report. It is required if the ANLPARM parameter REPORTS is set to DET.

STEPLIB

Might be required if the SQL PA modules are not listed in the system link list (LNKLST*xx*) when installed. Also, if the load library for the Db2 subsystem is not on the operating link list for the system, it must be added to STEPLIB for the ANLSQLPA job step. The LE runtime library (SCEERUN) can also require a STEPLIB allocation, as will the DB2 load library, particularly if there are multiple Db2 subsystems present.

SYSPRINT

Provides the output data set containing informational and tracing information.

The data set names for the report files are any names you choose. The DCB parameters for LRECL and RECFM must be formatted as required by SQL PA.

Using debug batch JCL

If you encounter problems with a SQL PA operation, you can document the details to send to IBM support. The *ssid*JBUG job is like the regular batch JCL but contains additional system output to help diagnose problems.

Procedure

1. Reissue SQL PA using the special *ssid*JBUG batch JCL found in the *hiqual*.SANLJCL library.

2. Insert the special diagnostic parameter DBTRACE DMP as the first parameter in the ANLCNTL file, when using debug to document a problem.

This control card does not produce a dump, but rather dumps the content of program storage to help the diagnosis. It also copies all the output report files from a single batch run (such as ANLREP) to SYSOUT.

3. Collect and send the problem data from the SYSOUT directory.

Sort Query Limits Report batch JCL

To sort the Query Limits Report, run SQL PA by using the *ssid*JQST batch JCL that is located in the *hiqual*.SANLJCL library.

About this task

This job is like the regular batch JCL but contains an extra step for sorting.

Running SQL PA using the ISPF interface

Db2 SQL Performance Analyzer can run interactively under TSO using standard ISPF panels.

This option allows you to work through several scenarios online, saving and reviewing previous results, while evaluating a number of alternatives and their costs. SQL PA provides the same level of information online under TSO that is generated by a batch run. When SQL PA is run online, you can reset key parameters on the panel and re-evaluate SQL dynamically. You can also choose from among the target host systems defined by your installation, applying the SQL PA costing against various configurations which might be encountered in production.

Topics:

- <u>"Starting SQL PA using ISPF" on page 93</u>
- "Navigating the SQL PA panels" on page 93

- "Using scrollable fields on Db2 SQL Performance Analyzer panels" on page 94
- "Primary menu panel of Db2 SQL Performance Analyzer" on page 95
- "Defining output reports" on page 96
- "Modifying Db2 SQL Performance Analyzer parameters" on page 96
- "Changing the CSV report options" on page 97
- "Basic processing options" on page 99
- "Comparing old and new packages" on page 99
- "Testing current access paths for packages" on page 101
- "Viewing the last set of generated reports" on page 103
- "ANLCSPA edit macro" on page 103
- "What if? analysis" on page 106
- "Db2 SQL Performance Analyzer statistics migration" on page 112
- "Db2 SQL Performance Analyzer reports menu" on page 114

Starting SQL PA using ISPF

SQL PA using ISPF is invoked by issuing the ANLCANLI command either from standalone TSO, or from ISPF panel 6, the TSO Command Panel.

About this task

For an example of the main selection panel, see Figure 22 on page 95. You can add SQL Performance Analyzer directly to the existing menu of Db2 options, and invoke it as an option from that panel.

ANLCANLI can be invoked with optional parameters:

ANLSSID(ssid)

ANLSSID(*ssid*) can be passed to select the Db2 subsystem that will be used. If no value is passed, the last subsystem used will be selected. If this is the first invocation, the default subsystem that was set up during customization will be used.

ANLDEBUG(YES) and ANLTRACE(YES)

These parameters can be passed to enable different types of tracing. This should only be done under the direction of IBM Support. If these values are not passed, the traces are not enabled.

ANLCANLI can be started in multiple sessions of ISPF by a single user. If multiple sessions are run concurrently, the file prefixes established during customization must be unique for each session. Uniqueness can be established by incorporating the system variable, ZSCREEN, into the prefix. Refer to "Prefix for output reports and work files" information in <u>"Worksheets: Gathering parameter values for Tools Customizer" on page 20</u>.

Figure 20 on page 93 shows the invocation using the ANLCANLI CLIST.

```
----- TSO COMMAND PROCESSOR ------
ENTER TSO COMMAND, CLIST, OR REXX EXEC BELOW:
===> ANLCANLI
```

Figure 20. Invoking ANLCANLI

Navigating the SQL PA panels

All access to SQL PA online begins at the SQL PA main control panel.

From that panel, you can process SQL statements from various sources, evaluate the effect of changing Db2 definitions, and modify the SQL PA configuration. After SQL PA is invoked, you go to the SQL PA Query Limits Report to review the results.

SQLPA510 DB2 SQL Performance Analyzer 5.1.0 10:30							
				DB2 sy DB2 SC	/stem: DBAB)LID : VNDMME(3	
1	Basic DB2 SQL PA P Process SQL stat	Basic DB2 SQL PA Processing Process SQL statements from various sources					
2	Advanced DB2 SQL P Evaluate the eff	A Processing ect of chang:	ing DB2 defin	itions			
3	3 Modify DB2 SQL PA Configuration Change the definition of your SQL PA environment, such as the names of your report data sets						
SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS							
Option F1=HEL F7=UP	===> P F2=SPLIT F8=DOWN	F3=END F9=SWAP	F4=RETURN F10=LEFT	F5=RFIND F11=RIGHT	F6=RCHANGE F12=RETRIEV	=	

Figure 21. Db2 SQL Performance Analyzer welcome panel

After entry into the Db2 SQL Performance Analyzer system under TSO, the welcome panel is displayed, as shown in Figure 21 on page 94. Press Enter to display the SQL PA main control panel.

Using scrollable fields on Db2 SQL Performance Analyzer panels

Db2 SQL Performance Analyzer uses ISPF scrollable fields to accommodate field text that is larger than the viewable area.

About this task

The less than (<) and the greater than (>) symbols denote a scrollable field. A > symbol indicates that the field can be scrolled to the right. A < symbol indicates that the field can be scrolled to the left. Both symbols are displayed when you are in the middle of data and can scroll either left or right.

Procedure

- 1. To scroll through the field, type LEFT or RIGHT in the command field, position the cursor in the field, and press Enter.
- 2. To see the entire contents of the field at once, type EXPAND in the command field, position your cursor in the scrollable field, and press Enter.
- 3. To clear the contents of the field, type ZCLRSFLD in the command field, position your cursor in the scrollable field, and press Enter.

If your level of z/OS does not support the ZCLRSFLD command, you can use the EXPAND command to display the entire contents of the field, and then clear the contents of the field in the window.

4. Optional: Assign your PF keys to be the LEFT, RIGHT, EXPAND, and ZCLRSFLD commands. Using a PF key simulates both typing in the command and pressing Enter.

Example

```
      SQLPA510 ----- Process QMF statement ----- 14:36
Command ===>
      Scroll ===> CSR

      Commands:
      EXPLAIN SQL TABLES
      DB2 system: DSNA
DB2 SQLID : JPUBLIC

      Enter the query identification:
      *

      *Query name.
      .
      .

      Query owner
      .
      .

      Current degree.
      .
      1 (1 or Any)

      (An * indicates a required field.)
      .
```

Primary menu panel of Db2 SQL Performance Analyzer

You begin Db2 SQL Performance Analyzer processing from the Db2 SQL Performance Analyzer **primary menu** panel.

The Db2 SQL Performance Analyzer primary menu panel shown in Figure 22 on page 95. From this panel, you can access panels to perform the following actions:

- · Modify and restore statistics
- · Perform statistics migration
- · Create or drop real or virtual indexes
- Change Db2 system ID
- · Change the current Db2 SQLID
- Change Db2 SQL Performance Analyzer parameters
- · Define report data sets
- · Enter or update batch job card parameters
- · Change the CSV report options

SQLPA510 DB2 SQL Performance Analyzer 5.1.0 11:01					11:01	
					DB2 sy: DB2 SQ	stem: DBAB LID : VNDMMEG
1	Basic Pro	DB2 SQL PA P cess SQL stat	rocessing ements from	various source	es	
2	Advan Eva	ced DB2 SQL P luate the eff	A Processing ect of chang	ing DB2 defin:	itions	
3	Modify Chai the	y DB2 SQL PA nge the defin names of you	Configuratio ition of you r report dat	n r SQL PA envi: a sets	ronment, such	as
Х	X Exit Exit from DB2 SQL PA					
Option F1=HEL F7=UP	===> P	F2=SPLIT F8=DOWN	F3=END F9=SWAP	F4=RETURN F10=LEFT	F5=RFIND F11=RIGHT	F6=RCHANGE F12=RETRIEVE

Figure 22. Db2 SQL Performance Analyzer primary menu panel

Defining output reports

For some output reports, you must specify certain files.

The name of the Cost Summary report file is predefined as *prefix*.ANLCOST.LOG, and is allocated by Db2 SQL Performance Analyzer under the running TSO user ID. The prefix is normally the TSO user ID, but it can be set to a different value.

The Plan Table report is allocated here as ANL510.PLN.

For Enhanced Explain reports, you must specify an OUTPUT EXPLAIN FILE. In batch, this file was allocated with the ANLREP DD statement. In the example in <u>Figure 23 on page 96</u>, the file is named ANL510.EXP under the user ID of the TSO user.

The Detail Trace file is declared as ANL510.DET. The Change report is ANL510.DLT and the CSV report is ANL510.CSV.

The Db2 SQL Performance Analyzer Limits file is also allocated here as ANL510.LIM.

SOLPA510 ------ Report Data Sets ------ 11:04 Command ===> DB2 system: DBAB DB2 SQLID : VNDMMEG Required report data set name information: Cost report: 'VNDMMEG.ANLCOST.LOG' SQL report : 'VNDMMEG.ANLOUT.SQL' Enter the report data set names: *Change report. . . 'VNDMMEG.ANL510.DLT' *CSV report. . . 'VNDMMEG.ANL510.CSV' *Explain report . . 'VNDMMEG.ANL510.EXP' *Plan table report 'VNDMMEG.ANL510.EEE' *Query limit report 'VNDMMEG.ANL510.LIM' *Trace report . . . 'VNDMMEG.ANL510.DET' (An * indicates a required field.) Command ===> 1=HELP F2=SPLIT F3=END F4=RETURN F5=RFIND F6=RCHANGE F9=SWAP F7=UP F10=LEFT F11=RIGHT F8=D0WN F12=RETRIEVE

Figure 23. Db2 SQL Performance Analyzer Output Reports panel

The *prefix*.ANLOUT.SQL file is now allocated and visible as the extract report that contains SQL and the optional objects that are used (tables and indexes). The prefix is normally the TSO user ID, but it can be set to a different value.

Modifying Db2 SQL Performance Analyzer parameters

You can view and modify the current values for all Db2 SQL Performance Analyzer parameters from the **Parameter Data Sets** panel.

About this task

This panel shows the data sets which contain the current values for the user and the system options.

Procedure

- 1. Specify option 3.3 on the main menu. One of two panels will display, depending on the value (YES or NO) assigned to the USRPARM system parameter.
 - If the USRPARM system parameter is set to NO, the following SQL PA Parameter Data Sets panel is displayed.

Figure 24. SQL PA Parameter Data Sets panel when USRPARM set to NO

• If the USRPARM system parameter is set to YES, a different SQL PA Parameter Data Sets panel is displayed (below).

Figure 25. SQL PA Parameter Data Sets panel when USRPARM set to YES

2. To modify these values, enter YES in their respective fields to edit the current parameter settings.

Note: All changes to parameters, except the user-level parameters, will affect all users of SQL PA.

Changing the CSV report options

The CSV report will be generated by the EXPLAIN command if it is enabled. You can view and modify the current values for all CSV parameters from the DB2 SQL Performance Analyzer Change CSV report options panel.

Before you begin

To enable the CSV report, set CSVRPT to YES. The name of the CSV report can be set on the Report Data Sets panel. See "Defining output reports" on page 96 for more information.

About this task

There are two panels used in the setup of the CSV report. The Change CSV report options panel is used to set the values of the CSV parameters. The CSV column selection report panel is used to select the information that will be included in the CSV report. The order in which these panels are displayed depends on whether the CSV report is enabled.

Procedure

1. Specify option 3.6 on the main menu. If the CSV report is not enabled, the Change CSV report options panel is displayed. If the CSV report is enabled, the CSV column selection report panel is displayed.

- 2. If the CSV column selection report panel is displayed, use the OP panel command to display the Change CSV report options panel.
- 3. Modify the values and press Enter to exit the panel.
- 4. If the CSV report is enabled, the CSV column selection report panel is displayed.
- 5. Select the columns to be included in the CSV report by using the SA (Select all) and DA (Deselect all) commands or by selecting the rows with the S line command or the // and //S block selection commands.
- 6. Specify the order of the columns in the CSV report by using the AS (Auto sequence) and RS (Resequence) commands or by manually entering the sequence number in the Seq column.
- 7. Optionally, the display can be sorted by using the ST (Sort by column number) command using the column number and A or D for Ascending or Descending, respectively.
- 8. Optionally, add column header labels to be used when CSVHDRW is set to LBL.
- 9. To exit the panel, use the SV (Save) command to save your changes or use the CN (Cancel) command to discard your changes.

Figure 26 on page 98 show the Change CSV report options panel.

```
SQLPA510 ----- Change CSV report options ----- 15:12

Command ===>
DB2 system: DBAB

DB2 SQLID : VNDMMEG
Enter the CSV report options:

*(CSVRPT)Create CSV report . . . . YES (YES/NO)

*(CSVSEL)Column selection file . . . TAB0

*(CSVHDRW)Column header value . . . COL (+OFF+, COL, LBL)

*(CSVDLIM)Delimiter character . . . COMMA (COMMA/SEMICOL)

*(CSVDCSP)Decimal separator . . . . DOT (COMMA/DOT)
(An * indicates a required field.)
```

Figure 26. Db2 SQL Performance Analyzer Change CSV report options panel

Figure 27 on page 98 show the CSV column selection panel.

SQLPA510 Command ==>	CSV column se	election Row 1 to 13 of 169 Scroll ===> CSR
Commands: SA RS OF SV Line commands (CSVSEL)CSV c	A - Select all DA - Desele 5 - Resequence AS - Auto S - CSV Option ST - Sort I 7 - Save CN - Cance 5 for Sel: S - Select // Column selection file	ect all DB2 system: DBAB sequence is off DB2 SQLID : VNDMMEG by column number l - Select begin //S - Select end
Sel Seq Column	name Table name	CSV column header label
S 1 QBLOCKN S 2 TNAME S 3 PLANNO S 4 SECTNOI S 5 QUERYNO COLLID COLLID EXPLAIN MIXOPSE MIXOPSE ACCESST ACCESST	NO > PLAN_TABLE > PLAN_TABLE > PLAN_TABLE I > PLAN_TABLE D > PLAN_TABLE > PLAN_TABLE > PLAN_TABLE > PLAN_TABLE N_TIME > PLAN_TABLE NLY > PLAN_TABLE EQ > PLAN_TABLE DLS > PLAN_TABLE TYPE > PLAN_TABLE	> > > > > > > > > > > > > > > > > > > > > > > > > > > > > > > > > > > > > >

Figure 27. Db2 SQL Performance Analyzer CSV column selection panel
Basic processing options

You can use the Basic Processing Options panel to choose how you want to input SQL statements from various sources.

About this task

Figure 28 on page 99 shows the options for input SQL statements from the various sources.

```
SQLPA510 ------ Basic Processing Options ----- 14:53
Option ===>
 Options:
                                                         DB2 system: DBAB
                                                         DB2 SQLID : JSMITH
    1
        Process packages from the DB2 system catalog
        Process application DBRM
    2
    3
        Process SQL from a sequential data set or PDS member
    4
        Process SQL from program source
    5
        Process a query number from the plan table
    6
        Process a OMF statement
    7
        Process a compare of package access paths
    8
        Process a test of packages for updated access paths
    9
      Process stabilized dynamic queries
```

Figure 28. SQL Performance Analyzer Basic Processing Options panel

Procedure

- 1. To process packages, select option 1.
- 2. To process the SQL in DBRM members, select option 2.

From the Process DBRM panel, you can select one or more DBRM by leaving the DBRM name blank or by typing a wildcard. If you are selecting multiple DBRM members, a panel displays that allows you to select one or more DBRM members from the list.

- 3. To process SQL from a sequential file or PDS member, select option 3.
- 4. To process SQL from a program source, select option 4.
- 5. To process a query number from the plan table, select option 5.
- 6. To process a QMF statement, select option 6.
- 7. To process a compare of package access paths, select option 7.
- 8. To process a test of packages for updated access paths, select option 8.
- 9. To process stabilized dynamic queries usingDb2 12 or higher, select option 9.

Comparing old and new packages

You can use Db2 SQL Performance Analyzer to compare the access paths of two versions of a bound package. The output reports only the changed statements.

- 1. Specify option 1, Basic DB2 SQL PA Processing, then option 7 to compare package access paths.
- 2. Enter the desired values, then enter COMPARE on the command line.

```
SQLPA51 ------ Process COMPARE of packages ------ 15:04
 Command ===>
                                                                      Scroll===> CSR
 Commands: COMPARE BATCH
                                                                  DB2 system: DBAB
                                                                  DB2 SQLID : JSMITH
  Enter the current package information:
   *Collection ID. : ANLITSTA
                                                                                    >
   *Package name . : ANLITST
                                                                                    >
    Version ID . : 0
SQLID. . . . : JSMITH (of first explain table set )
                                                                                    >
 Enter the previous package information:
    Compare to previous? Y
                                       (N/Y)
    Version ID . . : -1
SQLID. . . . . : JSMITH
                                                                                    >
                                     (of second explain table set)
 Options:
    *Statement Matching. : 1
                                                                  (1, Q, S)
(N,Y,-,+)
    Package cost filter . . . . . . . . . . N
Statement cost filter . . . . . . . . . N
                                                                  (N, Y, -, +)
   (N/Y)
                                                                  (N/Y)
  Show poly packages bound before............(An * indicates a required field.)F1=HELPF2=SPLITF3=ENDF4=RETURNF5=RFINDF7=UPF8=DOWNF9=SWAPF10=LEFTF11=RIGHT
                                                                           F6=RCHANGE
 F1=HELP
                                                                       F12=RETRIEVE
 F7=UP
```

Figure 29. Process COMPARE of packages panel

Options for COMPARE

There are several options on the Process COMPARE of packages panel. These allow you to specify filters and to customize the information returned by a compare.

Options

- **Statement matching**: The value of the STMTMATCH command indicates the method used to match the statements between package versions. The valid values are 1, Q, and S. A value of 1 matches by table and query type. A value of Q matches by query number. A value of S matches by the sequence of the statements. This command is optional and defaults to 1 if not specified.
- **Package cost filter**: Processes packages with a different total cost (all statements in the package). Valid values are
 - N: no filtering
 - Y: TOTAL_COST has changed
 - - (minus sign): TOTAL_COST has decreased
 - + (plus sign): TOTAL_COST has increased

The default is **N**.

- Statement cost filter: Processes statements with a different total cost. Valid values are:
 - N: no filtering
 - Y: TOTAL_COST has changed
 - - (minus sign): TOTAL_COST has decreased
 - + (TOTAL_COST has increased)

The default is **N**.

- **Minimum percent cost change to display**: Minimum package/statement total cost the compare will process. If both Package and Statement cost filters are specified, the threshold value will first be used for Package total cost and then for statements inside the package. If both cost filters are set to **N**, this option has no effect.
- **Show invalid packages**: Determines whether a warning message will be shown for packages flagged as INVALID in SYSIBM.SYSPACKAGE. An invalid package will be skipped, and if this field is set to **Y**, a

warning message will be issued in the report for all invalid packages. For example: ANL0213W Package collid.pkgname is flagged as INVALID by Db2. Rebind is recommended.

- **Show percent change**: This option shows all numeric fields (ACCESS_DEGREE, JOIN_DEGREE, MATCHCOLS, PROCMS, PROCSU and TOTAL_COST) to be shown with an additional percent change display. The default is **N** (no display).
- Show only packages bound before (timestamp): Specify a partial Db2 timestamp in the format yyyymmddHHMMSS to show packages bound packages that were bound before that time. Leave the field blank to select all packages bound before the current time.

Testing current access paths for packages

You can use SQL Performance Analyzer to compare the access paths for a new DBRM or stored procedure instance to the access paths of a previously bound package, using either the package SQL statements from a DBRM library or from the catalog.

About this task

TEST requires input from explain table data saved in your explain tables when the package was last bound. The output reports only the changed statements.

There are two popular use cases for TEST:

- First, during a migration from one Db2 level to the next, you can TEST bound packages to detect likely access path and cost changes. Use of wild cards in the PKGNAME should help to process multiple packages in one run.
- Second, before binding a new instance of a package during an application change, you can test just that package to determine if an access path change is likely, and the effect of that change on the performance and cost of package execution.

- 1. Specify option 1 **Basic DB2 SQL PA Processing**, option 8 Process a test of packages for updated access paths
- 2. Enter the desired values and enter TEST on the command line.



Figure 30. SQL Performance Analyzer Process TEST of packages panel

Options for TEST

There are several options on the Process TEST of packages panel. These allow you to specify filters and to customize the information returned by a test.

Options

- From catalog: Indicates where the statements used for comparing to the package will be obtained. Valid values: C, N, Y.
- EXPLAIN table SQLID: the SQLID to run EXPLAIN on the specified packages.
- Current degree: The degree of parallelism for a dynamic SQL or QMF statement.
- Statement matching: Indicates the method used to match the statements between package versions. The valid values are 1, 2, Q, and S. A value of 1 matches by table and query type. A value of 2 matches by table, query type, and statement text. A value of Q matches by query number. A value of S matches by the sequence of the statements. This command is optional and defaults to 1 if not specified. This is the same as the STMTMATCH command in batch.
- EXPLAIN type: This field has two values:
 - Dynamic (D): The EXPLAIN will use Db2 dynamic path selection.
 - EXPLAIN only (X): The EXPLAIN will use a bind with the EXPLAIN(ONLY) option, which uses more system resources but produces a more accurate access path prediction.
- **Package cost filter**: Processes packages with a different total cost (all statements in the package). Valid values are
 - N: no filtering
 - Y: TOTAL_COST has changed
 - - (minus sign): TOTAL_COST has decreased
 - + (plus sign): TOTAL_COST has increased

The default is **N**.

- Statement cost filter: Processes statements with a different total cost. Valid values are:
 - N: no filtering
 - Y: TOTAL_COST has changed
 - - (minus sign): TOTAL_COST has decreased
 - + (TOTAL_COST has increased)

The default is ${\bf N}.$

- **Minimum percent cost change to display**: Minimum package/statement total cost the compare will process. If both Package and Statement cost filters are specified, the threshold value will first be used for Package total cost and then for statements inside the package. If both cost filters are set to **N**, this option has no effect.
- **Show invalid packages**: Determines whether a warning message will be shown for packages flagged as INVALID in SYSIBM.SYSPACKAGE. An invalid package will be skipped, and if this field is set to **Y**, a warning message will be issued in the report for all invalid packages. For example: ANL0213W Package collid.pkgname is flagged as INVALID by Db2. Rebind is recommended.
- **Show percent change**: This option shows all numeric fields (ACCESS_DEGREE, JOIN_DEGREE, MATCHCOLS, PROCMS, PROCSU and TOTAL_COST) to be shown with an additional percent change display. The default is **N** (no display).
- Show only packages bound before (timestamp): Specify a partial Db2 timestamp in the format yyyymmddHHMMSS to show packages bound packages that were bound before that time. Leave the field blank to select all packages bound before the current time.

For more information on these options, see <u>"Testing current access paths for packages in batch" on page</u> 89, which explains the batch commands corresponding to the settings on this ISPF panel.

Viewing the last set of generated reports

You can use the LASTREPORTS command to view the last set of generated reports without re-running the EXPLAIN command.

Procedure

To view the last set of generated reports, issue the LASTREPORTS command on any of the following panels:

- ANLPPKGE
- ANLPDBRM
- ANLPFILE
- ANLPQNO
- ANLPQMF
- ANLPSDQ

If the last set of reports was generated by running the EXPLAIN command on the ANLPQNO panel, the plan table report is displayed. If the last set of reports was generated by running the EXPLAIN command on a panel other than the ANLPQNO panel, the query limits report is displayed.

ANLCSPA edit macro

In ISPF Edit or Browse mode, Db2 SQL Performance Analyzer allows you to selectively pick a portion of the file for analysis, choosing one or several statements.

Selective analysis uses a special edit macro called ANLCSPA. ANLCSPA uses the letter E in the line number columns to selectively pick a subset of the contents of a file. You can choose to bracket the desired lines between a pair of EE-EE book markers or indicate the number of lines. An entry of E12, for example, selects the next 12 lines. Remember to enter ANLCSPA on the command-line prompt so that ISPF Edit recognizes this special Db2 SQL Performance Analyzer edit macro.

To invoke ANLCSPA from any edit or browse session, the CLIST library must be allocated to SYSPROC. The preferred method to invoke ANLCSPA is through the TSO logon procedure. The CLIST ANLCALLC is provided for you to use if the allocation is necessary after logon.

Tip: Do not mix the formats of the libraries in SYSPROC. If your SYSPROC allocation uses libraries with variable block formatting, select the optional step **Create SANLVCLS** on the **Product Parameters** panel in Tools Customizer. ANLCALLC will allocate the appropriate library, and, if there is a mismatch, a message will be issued and the allocation will not be done.

If the Db2 SQL Performance Analyzer CLIST library is allocated as SYSPROC, you can start Db2 SQL Performance Analyzer from any Edit or Browse session.

If you want to use Db2 SQL Performance Analyzer during a programming session to evaluate an SQL statement, bracket the statements with EE-EE and issue ANLCSPA on the command line.

The ANLCSPA edit macro offers the capability to have SQL statements, embedded in a program or SPUFI source, explained directly from the ISPF/PDF editor. The following languages are supported:

- Assembler
- C/370
- COBOL
- Fortran
- PL/I

The edit macro is invoked under ISPF command ANLCSPA. When editing the source of a program (or SPUFI input), you can specify E as the ISPF/PDF editor line command and then enter ANLCSPA (or another suitable name of a REXX exec) on the command line and ANLCSPA scans the specified range for any SQL statements.

You can use EE on the first and last line, you can use one single E if the statement is on one single line, or you can use En where n is the number of lines to scan. You can even specify E99999 on the first line of the source and ANLCSPA scans the complete source for any SQL statements and then explains the statements that are explainable. The remaining statements display as comments in the Plan Table summary report. If you do not specify E, EE, or En, all the explainable SQL statements in the module are explained.

You can optionally pass the subsystem ID and the HLQ of the Db2 SQL Performance Analyzer installation to the macro. Passing the subsystem ID and the HLQ is helpful if there are several choices where you do not always want to use the default, for example ANLCSPA DSNA, USERS. SQLPA3.

When the edit macro starts, it optionally asks for the language of the source code you are editing. It then gives you the opportunity to specify a table qualifier and the current SQLID (Db2 object qualifier).

Figure 31 on page 104 shows an example of editing a PL/I program. During editing, you need to explain the SQL statement that starts in line number 19 and ends in line number 25.

Enter EE in the sequence number field of the lines indicating the start and end lines. Further, the name of the REXX exec is specified in the command field.

```
EDIT JOHNDOE.PLI.SOURCE(E3ONLINE) - 01.01
Command ===> anlcspa
                                                    Scroll ===> CSR
000100 clostst: procedure options(main reentrant) reorder;
000200
000300
        dcl (number, height)
                                   integer:
000400
        dcl 1 ud,
000500
             3 indikator
000600
                                   char (8)
000700
                                   char (11);
00800
        dcl
000900
          addr
                                   builtin,
001000
                                   builtin.
         high
001100
         translate
                                   builtin,
001200
001300
        dcl plixopt char (26) var init ('ISA(28k), ISAINC(24k), NR') EXT;
001400
        dcl i bin fixed (31,0) init (0) static;
001500
FF
001700
        exec sql include sqlca;
001800
001900
        exec sql
            declare C1 cursor for
002000
002100
            select count(*)
002200
002300
               from sysibm.systables
             where creator = USER
              and name in ('TAB1','MYTAB','EEE_TAB2')
and type = 'T';
002400
002500
EE
002700
        exec sql
```

Figure 31. Edit PL/I program example

The next set of sample figures shows how SQL statements in a given Db2 package are explained while you are using the Db2 Administration Tool. Initially, you specify an owner and collection identification:

DB2 Admin ------ DSN8 System Catalog ----- 12:52 Option ===> k DB2 System: DSN8 DB2 SQL ID: JOHNDOE Object options: AO - Authorization options G - Storage groups D - Databases S - Table spaces P - Plans L - Collections K - Packages T - Tables, views, and aliases M - DBRMs V - Views H - Schemas E - User defined data types A - Aliases Y - Synonyms F - Functions X - Indexes 0 - Stored procedures C - Columns J - Triggers N - Constraints Q - Sequences DS - Database structures DSP - DS with plans and packages Enter standard selection criteria (Using a LIKE operator, criteria not saved): Name===>>Owner===>JOHNDOE% >InD/L/HD/L/H==>ANLPACK>Switch Catalog Copy===>N(N/S/C) And/or other selection criteria (option xC shows you columns for option x) Value ===> Column ===> > Operator ===>

Figure 32. Db2 Administration Tool System Catalog panel – object options

Issue the SQ command to display all SQL statements for the package E61PLAN. The results are shown in Figure 33 on page 105.

Figure 33. Db2 Administration Tool Packages panel

You now start the ANLCSPA macro. Because no range is specified in the sequence number column, all explainable SQL statements are explained:

Figure 34. Db2 Administration Tool Extracted SQL panel

What if? analysis

By using the What if? program you can create scenarios with the optimizer by changing table or index statistics to see if you can influence the access path selection. You can display, modify, save, and restore catalog statistics on both tables and indexes that are most effective in access path selection.

You can define the following criteria for tables:

- Cardinality (number of rows, or CARD)
- Number of pages used by the table (NPAGES)
- Number of active pages (NACTIVE) in the table space
- Percentage of pages used (PCTPAGES)
- Degree of data compression (PCTROWCOMP)

You can define the following criteria for indexes:

- Number of index levels in the tree (NLEVELS)
- Total number of index leaf pages (NLEAF)
- Cardinality of the first column of the index (FIRSTKEYCARD)
- Fully qualified key (FULLKEYCARD)
- Cluster ratio, a 0 1 decimal number that indicates the degree to which rows are clustered together on the same or adjacent pages (CLUSTERRATIO)

Each TSO user has a Db2 SQL Performance Analyzer program-maintained partitioned data set (PDS) that is called *user_ID*.ANL510.SANLSTAT that is built under their TSO user ID. The What If? and catalog migration functions use this PDS to create and manage the statistical processing for backups and restores, temporary parameters, and JCL.

This file can also have a different prefix (like ANLCOST.LOG and ANLOUT.SQL).

Analyzing indexes and modifying statistics

You can analyze indexes and modify statistics to create scenarios with the optimizer to try to influence access path selection. You can determine which catalog statistics on both tables and indexes are most effective in access path selection. You can perform a typical What if? index analysis and statistics modification using this procedure.

Procedure

1. Select option 2, Advanced Db2 SQL PA Processing, on the Figure 22 on page 95.

The Advanced Processing Options panel is displayed, as shown in the following figure:

SQLPA51	0	Ad	dvanced Proc	essing Option	s	11:58
					DB2 sy DB2 SO	stem: DBAB LTD : VNDMMFG
Option	s:					
1	Modify	statistics				
2	Migrat	e statistics				
3	Create	or drop an :	index			
Option F1=HELI F7=UP	===> P	F2=SPLIT F8=DOWN	F3=END F9=SWAP	F4=RETURN F10=LFFT	F5=RFIND F11=RTGHT	F6=RCHANGE F12=RFTRTEVF

Figure 35. Advanced Processing Options panel

2. Select option 1 to display the **Modify Statistics** panel, as shown in the following figure.

SQLPA510	9		Modify Sta	tistics		11:59
					DB2 sy DB2 SC	/stem: DBAB DITD : VNDMMEG
Options	5:					
1	Modif	y table stati	stics			
2	Modif	y index stati	stics			
3	Resto	re statistics				
F1=HELF F7=UP	===>	F2=SPLIT F8=DOWN	F3=END F9=SWAP	F4=RETURN F10=LEFT	F5=RFIND F11=RIGHT	F6=RCHANGE F12=RETRIEVE

Figure 36. Modify Statistics panel

3. To modify the current index statistics, select option 2.

SQLPA510 ------ Modify Index Statistics ------ 12:00 DB2 system: DBAB DB2 SQLID : VNDMMEG DB2 information: DB2 system ID . . .: DBAB Current DB2 version : V10NFM Specify the index to modify statistics: *Index creator . . . : > *Index name : Data set information for saving the old statistics: Data set name . . . : VNDMMEG.ANL510.SANLSTAT *Member name : (member or NOSAVE) (An * indicates a required field.) Command ===> Scroll ===> PAGE F5=RFIND F4=RETURN F2=SPLIT F6=RCHANGE F1=HELP F9=SWAP F3=END F7=UP F8=D0WN F10=LEFT F11=RIGHT F12=RETRIEVE

Figure 37. Modify Index Statistics panel

4. Type the index creator, index name, and member name.

Creating and dropping indexes

You can create and drop real and virtual indexes. Creating virtual indexes and replacing a real index with a virtual index helps to calculate the cost of SQL queries.

About this task

For example, to calculate the cost of SQL queries, you can drop a real index virtually, then create and drop many virtual indexes to simulate a different index configuration. The virtual drop of a real index causes the EXPLAIN command to exclude a virtually dropped real index from the cost calculation of SQL queries, but it does not remove the physical index from the Db2 catalog. Creating a virtual index causes the EXPLAIN command to include a virtual index for the cost calculation, but it does not require processor resources to build a virtual index even when the table size of the index target is large. These virtual index features help to design more efficient indexes without costing too much processor resource.

In order to include the virtual index as a part of the cost calculation of the EXPLAIN SQL command, the same Db2 SQL ID should be set and displayed at the upper right corner of the **Create Index** panel and the SQL PA cost analysis panels. In other words, the table owner of PLAN_TABLE and DSN_VIRTUAL_INDEXES must be the same to include a virtual index in the cost calculation of the EXPLAIN SQL command.

Procedure

1. Select option 2, Advanced DB2 SQL PA Processing, on the Figure 22 on page 95.

The Advanced Processing Options panel is displayed.

2. Select option 3, Create or drop an index, and press Enter.

The Create or Drop Index panel is displayed, as shown in the following figure:

SQLPA510	(Create d	or Drop	Inde>	<			11:58
DB2 informat DB2 system Current DB	ion: ID: 2 version: '	DBAB /10NFM				I	DB2 sy DB2 SQ	stem: DBAB LID : VNDMMEG
Index to be *Index crea *Index name	created or (tor 	dropped:						
*Select an 1 Creat 3 Creat 5 Remove	action: e index e virtual i e virtual c	ndex nanges		2 4	Drop ir Drop ir	ndex ndex vir	tually	
(An * indica	tes a requi:	red fiel	.d.)					
Option ===> F1=HELP F7=UP	F2=SPLIT F8=DOWN	F3=EN F9=SM	ID IAP	F4=F F10=L	RETURN _EFT	F5=RF: F11=RI(IND GHT	F6=RCHANGE F12=RETRIEVE

Figure 38. Create or Drop Index panel

- 3. To create real indexes, complete the following steps:
 - a) On the **Create or Drop Index** panel, specify the name of the index creator and the name of the index to be created.
 - b) Select option 1, Create index, and press Enter.

The **Create Index** panel is displayed, as shown in the following figure:

```
SQLPA510 ------ Create Index ----- 11:58
                                                                               DB2 system: DBAB
                                                                               DB2 SOLID : VNDMMEG
  Index to be created:
     Index creator . . . : index_creator_name
Index name. . . . : index_name
  Index on table:
     Table creator . . . . table_creator_name
     Table name . . . . . table_name
  Enter up to 5 column names in ascending|descending order:
   *A or D. . . A *Index column 1 . . . column_name
A or D. . . A Index column 2 . . .
A or D. . . A Index column 3 . . .
A or D. . . A Index column 3 . . .
A or D. . . A Index column 4 . . .
A or D. . . A Index column 5 . . .
  (An * indicates a required field.)
 Option ===>
  .
F1=HELP
                    F2=SPLIT
                                     F3=END
                                                      F4=RETURN
                                                                        F5=RFIND
                                                                                         F6=RCHANGE
  F7=UP
                                     F9=SWAP
                   F8=DOWN
                                                     F10=LEFT
                                                                      F11=RIGHT
                                                                                        F12=RETRIEVE
```

Figure 39. Create Index panel

- c) Specify the table creator and the name of the table for which the index will be created.
- d) In the **Index column 1** field, specify a name for the column and whether the column will be in ascending or descending order, and press Enter.

The Create Index panel is displayed, as shown in the following figure:

SQLPA510	QLPA510 Create Index				
Index to be Index crea Index name	created: tor:	index_crea index_name	tor_name	DB2 sy DB2 S(ystem: DBAB QLID : VNDMMEG
UNIQUE PADDED CLOSE *PRQTY *SEQTY	NO (Yes NO (Yes NO (Yes 10 20	s or No) s or No) s or No)	CLUSTER INDEX. PARTITIONED WHERE NOT NULL INCLUDE NULL K *STOGROUP *BUFFERPOOL NAM	NO NO NO YES (Ye SYSDEN E BPO	(Yes or No) (Yes or No) (Yes or No) es or No) FLT
	WARNING:	Index is b	uilt, data sets	are created	
(An * indica	tes a require	d field.)			
Option ===> F1=HELP F7=UP	F2=SPLIT F8=DOWN	F3=END F9=SWAP	F4=RETURN F10=LEFT	F5=RFIND F11=RIGHT	F6=RCHANGE F12=RETRIEVE

Figure 40. Create Index panel

e) Specify values for all of the following required fields:

PRQTY

Specify the minimum primary space allocation for a DB2-managed data set.

STOGROUP

Specify the name of a storage group in which to create the index. The storage group must exist at the current server and the privilege set must include SYSADM authority, SYSCTRL authority, or the USE privilege for the storage group.

SEQTY

Specify the minimum secondary space allocation for a DB2-managed data set.

BUFFERPOOL NAME

Specify the buffer pool to use for the index. The name of the buffer pool must identify an activated 4 KB, 8 KB, 16 KB, or 32 KB buffer pool, and the privilege set must include SYSADM or SYSCTRL authority or the USE privilege for the buffer pool.

f) Optional: Specify values for the following optional fields:

UNIQUE

Specify whether the index to be created will be unique.

CLUSTER INDEX

Specify whether the index to be created will be the clustering index for the table.

PADDED

Specify whether the index to be created will be padded.

PARTITIONED

Specify whether the index to be created will be partitioned.

CLOSE

Specify whether the index to be created can be closed when it is not being used.

WHERE NOT NULL

Specify whether NULL values in a unique index can be unique.

INCLUDE NULL KEYS

Specify whether to include NULL key values.

g) Press Enter.

The index is created.

4. To create virtual indexes, complete the following steps:

- a) On the **Create or Drop Index** panel, specify the name of the index creator and the name of the index to be created.
- b) Select option 3, Create virtual index, and press Enter.

The Create Index panel is displayed.

- c) Specify the table creator and the name of the table for which the virtual index will be created.
- d) In the **Index column 1** field, specify a name for the column and whether the column will be in ascending or descending order, and press Enter.

The Create Index panel is displayed, as shown in the following figure:

```
SQLPA510 ------ Create Index ----- 11:58
                                                       DB2 system: DBAB
                                                       DB2 SOLID : VNDMMEG
 Index to be created:
   Index creator . . . : index_creator_name
   Index name. . . . . . : index_name
                                   CLUSTER INDEX. . . . NO
 UNIQUE. . . . . NO
                     (Yes or No)
                                                            (Yes or No)
 PADDED. . . NO
NLEAF. . . . -1
PGSIZE. . . 4
FIRSTKEYCARDF. -1
                     (Yes or No)
                                   PARTITIONED. . . . NO
                                                           (Yes or No)
                                   NLEVELS. .
                                   (4,8,16,32)
                                   FULLKEYCARDF . . . . -1
 (An * indicates a required field.)
Option ===>
             F2=SPLIT
 F1=HELP
                          F3=END
                                      F4=RETURN
                                                 F5=RFIND
                                                              F6=RCHANGE
 F7=UP
             F8=DOWN
                         F9=SWAP
                                    F10=LEFT
                                                 F11=RIGHT
                                                             F12=RETRIEVE
```

Figure 41. Create Index panel

e) Optional: Specify values for the following optional fields:

UNIQUE

Specify whether the index to be created will be unique.

CLUSTER INDEX

Specify whether the index to be created will be the clustering index for the table.

PADDED

Specify whether the index to be created will be padded.

PARTITIONED

Specify whether the index to be created will be partitioned.

NLEAF

Specify the number of active leaf pages in the index.

NLEVELS

Specify the number of levels in the index tree.

PGSIZE

Specify the size in kilobytes of the leaf pages in the index.

CLUSTERRATIOF

Specify the percentage of rows that are in clustering order.

FIRSTKEYCARDF

Specify the number of distinct values of the first key column.

FULLKEYCARDF

Specify the number of distinct values of the key.

f) Press Enter.

The virtual index is created.

5. To drop real indexes, complete the following steps:

- a) On the Create or Drop Index panel, specify the name of the index creator and the name of the index to be dropped.
- b) Select option 2, Drop index, and press Enter.

A confirmation panel is displayed.

c) Confirm that you want to drop the index, or cancel the operation.

- 6. To drop virtual indexes, complete the following steps:
 - a) On the **Create or Drop Index** panel, specify the name of the index creator and the name of the virtual index to be dropped.
 - b) Select option 4, **Drop index virtually**, and press Enter. The specified index is dropped.

Removing virtual changes

Procedure

- 1. On the **Create or Drop Index** panel, specify the name of the index creator and the name of the index to be changed.
- 2. Select option 5 **Remove virtual changes**, and press Enter. The virtual changes are removed.

Restoring statistics for tables and indexes

You can restore previously saved original statistical sets from a table or an index.

Procedure

1. Select option 2, Advanced Db2 SQL PA Processing, from the Figure 22 on page 95.

The Advanced Processing Options panel is displayed, Figure 35 on page 107.

- 2. Select option 1 to display the **Modify Statistics** panel, as shown in Figure 36 on page 107.
- 3. To restore saved statistics for an index or table, select option 3.

Db2 SQL Performance Analyzer statistics migration

By using the statistics migration function, you can manage the process of running the ANLCAT batch program to migrate statistics and use the *ssid*APPL set of JCL in *high_level_qualifier*.SANLJCL to apply them to the system catalog. The statistics migrations panels dynamically build jobs to Collect or Apply statistics under TSO.

Setting up the Apply and Collect jobs

The Collect and Apply processes use the job card information saved in the user parameters during customization.

About this task

If the **(USRPARM) Allow users to have a copy of the user parameters** parameter was selected on the **Product Parameters** panel in Tools Customizer, each user can have his or her own job card. Job cards are saved in the user parameter file. If the user-level user parameters have not been set up before the job cards are updated, the **Parameter Data Sets** panel will be displayed so that the user-level user parameters can be initialized.

- 1. Select option 3, Modify DB2 SQL PA Configuration, from the primary menu panel.
- 2. Select option 5, **Enter or update the batch job card parameters**, to display the **Batch Job Statement Parameters** panel, as shown in the following figure.

```
SQLPA510 ------ Batch Job Statement Parameters ----- 12:06
                                                              DB2 system: DBAB
DB2 SQLID : VNDMMEG
  Edit the standard JOB card information below:
    Jobcard 0 . . //ANLBATCH JOB
    Jobcard 1 . . // USER=JOEUSER,NOTIFY=JOEUSER,MSGCLASS=H,
Jobcard 2 . . // MSGLEVEL=(1,1),REGION=0M,TIME=10
    Jobcard 3 . . /*JOBPARM S=*
  Route this batch job to a different LPAR . (optional /*ROUTE card)
 Command ===>
               F2=SPLIT
                                                                       F6=RCHANGE
                             F3=END
                                           F4=RETURN
                                                         F5=RFIND
  F1=HELP
                             F9=SWAP
  F7=UP
                F8=DOWN
                                          F10=LEFT
                                                        F11=RIGHT
                                                                      F12=RETRIEVE
```

Figure 42. Batch Job Statement Parameters panel

3. Enter modifications to the job card information and press Enter to begin processing the changes.

Collecting catalog statistics

Use the Collect catalog statistics function to specify the collection options and parameters that you want to migrate or apply to the system catalog.

About this task

Recommendation: It is recommended that you collect a set of statistics for your baseline system before applying any changes to the catalog. Because the data is saved in a member, you can always restore those statistics to their original settings after you have completed your analysis.

- 1. Select option 2, Advanced DB2 SQL PA Processing, from the primary menu panel. The Advanced Processing Options panel is displayed.
- 2. Select option 2 to display the Statistics Migration panel, as shown in the Statistics Migration panel.

SQLPA510			Statistics	Migration		14:52
					DB2 s DB2 S	ystem: DBAB QLID : VNDMMEG
Options	s:					
1	Colle Bui	ct statistic ld the param	s eters and ex	ecute the ANLO	CAT program	
2	Apply Upd	statistics ate the cata	log with the	collected sta	atistics	
Option = F1=HELF	===>	F2=SPLIT	F3=END	F4=RETURN	F5=RFIND	F6=RCHANGE
F7=0P		F8=DOMN	F9=SWAP	FI0=LEFI	FII=RIGHI	FIZ=RETRIEVE

Figure 43. Statistics Migration panel

3. Select option 1, **Collect statistics**, to collect statistics by building the input parameters and executing the ANLCAT program.

Applying catalog statistics

Use the Apply catalog statistics function to apply or migrate the collected statistics to the system catalog.

Procedure

To apply catalog statistics that are already collected or that were manually created, use the following steps:

- 1. Select option 2, Advanced DB2 SQL PA Processing, from the primary menu panel. The Advanced Processing Options panel is displayed.
- 2. Select option 2 to display the **Statistics Migration** panel.

```
SQLPA510 ------ Statistics Migration ------ 14:52
                                                       DB2 system: DBAB
DB2 SQLID : VNDMMEG
Options:
   1 Collect statistics
         Build the parameters and execute the ANLCAT program
   2 Apply statistics
         Update the catalog with the collected statistics
Option ===>
                         ⊢3=END
F9=SWAP
             F2=SPLIT
                                     F4=RETURN
F1=HELP
                                                  F5=RFIND
                                                              F6=RCHANGE
                                                 F11=RIGHT
F7=UP
             F8=DOWN
                                     F10=LEFT
                                                              F12=RETRIEVE
```

Figure 44. Statistics Migration panel

3. Select option 2, **Apply statistics**, to apply statistics by updating the Db2 catalog with collected statistics.

Db2 SQL Performance Analyzer reports menu

After SQL costing is complete, Db2 SQL Performance Analyzer displays the Query Limits Report. You can browse the reports created during this run from this panel.

You can sort the Query Limits Report by using the **SORT** command. The optional parameters are the sort column number and the sort order. For example, to sort the **Query No** column in ascending order, the command is **SORT 3 A**.

The Query Limits Report panel is shown in the following figure.

SQLF Con	PA510 nmand =	:=>		Qı	ery Limits	Report	Rc	w 1 to 12 Scroll ===	of 43 ⇒ CSR
Con	nmands:	PR -	PRINT				DB2 DB2 Time	system: DE SQLID : AM e: OE	BAB ILUSER1 B:27
C.	- Cost	Report	F F - Exp	lain F	Report 0 -	Oblock Repo	ort R - F	lan Table	Report
Š	- SQL	Report	T - Trace	Repoi	t W - What	2020011 110p1		100010	
If?	- L			- 1					
				Stmt	CPU	Elapsed	Physical		
S C	CEIQ\$	Error	Query No	Туре	Time	Time	I/0	QUNITS	Cost
			45000						
-		0	15380	SI	0.00100	0.019	4	6	0.0002
-		0	15373	SI	0.00100	0.019	4	6	0.0002
-		Θ	15246	SI	0.00100	0.019	4	6	0.0002
-		Θ	15230	ST	0.00100	0.019	4	6	0.0002
-		Θ	15213	ST	0.00100	0.019	4	6	0.0002
-		Θ	15272	ST	0.00100	0.024	5	3	0.0002
-		Θ	15266	ST	0.00100	0.024	5	3	0.0002
-		0	17356	ST	0.00100	0.019	4	2	0.0002
-		Θ	17321	ST	0.00100	0.019	4	2	0.0002
-		Θ	17282	ST	0.00100	0.019	4	2	0.0002
-		0	17251	ST	0.00100	0.019	4	2	0.0002
-		Θ	17214	ST	0.00100	0.019	4	2	0.0002

Figure 45. Query Limits Report panel

Displaying the Query Block report

To view the Query Block report for a particular SQL statement, issue the "Q" line command from the Query Limits Report panel.

About this task

Each row of the Query Limits report represents an SQL statement that has been processed. Line commands can be used to view different report information or to perform further "What If?" testing for the statement. The "Q" line command will display the query block information for the statement.

- 1. Select the desired query with the "Q" line command and press <Enter>. The Query Block Report panel will be displayed.
- 2. The table and index creators and names are scrollable fields. You can view more information by doing the following:
 - To view more detailed information about the index, select the desired query block with the "I" line command.
 - To view more detailed information about the table, select the desired query block with the "T" line command.
 - To view all of the indexes available for a table, select the desired query block with the "A" line command.
- 3. To sort the Query Block report, use the SORT command with the optional parameters of the sort column number and the sort order.
- 4. To exit the Query Block Report panel, press PF3.

Displaying index and table information from Query Block report

To view the index or table information for a particular query block, issue the A, I, or T line command from the Query Block Report panel.

About this task

Each row of the Query Block report represents a query block of the SQL statement that has been processed. Line commands can be used to view the table information or the index information for the query block.

- The "I" line command will display the index information for the query block.
- The "T" line command will display the table information for the query block.
- The "A" line command will display all available indexes for a table.

Procedure

- 1. To see the *index* information, select the desired query block with the "I" line command, and press <Enter>. The Index Information panel will be displayed.
- 2. The name field as well as the index, table, and index space fields are scrollable. To sort the Index Information, use the SORT command with the optional parameters of the sort column number or name and the sort order.
- 3. Press PF3 to return to the Query Block Report panel.
- 4. To see the *table* information, select the desired query block with the "T" line command, and press <Enter>. The Table Information panel will be displayed.
- 5. The name field as well as the table and table space fields are scrollable. To sort the Table Information, use the SORT command with the optional parameters of the sort column number or name and the sort order.
- 6. Press PF3 to return to the Query Block Report panel.
- 7. To see *all available indexes*, select the desired query block with the "A" line command, and press <Enter>. The All Indexes panel will be displayed.
- 8. Select the desired index with the "I" line command to see the index information, and press <Enter>. The Index Information panel will be displayed.
- 9. Press PF3 twice to return to the Query Block Report panel.

What If? testing from the Query Limits report

To perform What If? testing for a particular SQL statement, issue the W line command from the Query Limits report.

About this task

Each row of the Query Limits report represents an SQL statement that has been processed. Line commands can be used to view different report information or to perform further What If? testing for the statement. The statement can be modified and virtual index modifications can be made during the What If? processing. What If? processing for a statement can be repeated up to 50 times in order to achieve the desired outcome.

Note:

- If any modifications were made using virtual indexes, those changes will remain in effect when the What If? processing is completed. To reverse the effected changes, an extra iteration of What If? processing can be used to reverse the virtual index changes, or virtual index changes can be made by navigating to option 2.3 from the main menu.
- The What If? reports will be deleted when the What If? processing is completed. To keep the What If? reports, rename the desired reports before exiting the What If? processing.

Procedure

- 1. Select the desired query with the W line command. Press Enter.
- 2. The What If? Modifications panel will be displayed. Enter Y in the Create or drop index field to make virtual index changes, or enter N if no virtual index changes are desired. Enter Y in the Modify SQL statement field to modify the query, or enter N to skip the query modification. Press Enter.
- 3. If Y was entered in the Create or drop index field, the Create index panel will be displayed. See Creating and dropping indexes for more information about using virtual indexes. Press PF3 to exit the panel and continue processing. Note: Only virtual index changes are allowed during What If? processing.
- 4. If Y was entered in the Modify SQL statement field, the query will be displayed in an edit session. The original query will be retained at the top in comments as a reference. Modify the query as desired and press PF3 to exit and continue processing.
- 5. The original explain results will be displayed in the Query Limits report with "base" as the value in the Error column. The What If? results will be displayed with 100000001 as the value in the Query No column. To view the new results, select the What If? results with one of the line commands.
- 6. To continue What If? processing, use the WIF panel command to begin another iteration. For each subsequent iteration, the Query No value will be incremented by 1.
- 7. To save the What If? reports, rename the reports before exiting the What If? Query Limits Report panel.
- 8. To exit What If? processing, press PF3 from the What If? Query Limits Report panel.

Using SQL PA in QMF

An optional exit in the QMF Governor allows SQL PA to cost your QMF queries. Using this installation option makes it easy for you to use SQL PA while developing SQL in QMF.

The QMF interface runs in the background. The QMF interface only displays if the SQL statement that you are about to run is likely to exceed the cost limits defined for SQL PA. For queries that exceed the cost limits, the QMF interface displays the cost estimate and gives you the chance to stop before QMF begins processing.

Topics:

- "QMF intercept exit" on page 117
- "Db2 primary options menu" on page 117
- "SQL PA cost summary under QMF" on page 118
- "Run queries that exceed limits in QMF" on page 118
- "QMF batch processing" on page 119

QMF intercept exit

Db2 SQL Performance Analyzer has a special exit used to intercept QMF transactions in flight. The special exit provides you with an immediate cost analysis, before actually running the query. If installed, this option is integrated into your normal QMF processing. To integrate the option, add a call to the SQL PA exit as part of the IBM-supplied QMF Governor's activities.

Db2 primary options menu

You can invoke QMF from a menu or from a QMF CLIST.

You do not do anything special to use SQL Performance Analyzer under QMF. The QMF intercept exit allows SQL Performance Analyzer to take control of a query before it is issued and perform a streamlined cost analysis of the SQL statement.

From the QMF Home panel, users typically create ad hoc queries, composing their SQL in real time.

SQL Performance Analyzer can provide cost and performance information on each SQL statement before Db2 processes it.

SQL PA cost summary under QMF

The QMF Intercept Exit for SQL PA produces the standard Cost Summary Report, along with any warnings that are warranted by the SQL statement.

The three types of costing information displayed for the user to consider are:

- Resource Consumption: CPU time, elapsed time, I/O count
- Relative Cost Valuation: QUNITS (Query Units)
- Financial Valuation: Monetary cost of execution (Charge Back)

The example in the following figure shows this report.

Figure 46. Cost summary under QMF

Run queries that exceed limits in QMF

Based on a customization option, the QMF Intercept Exit can show the cost of every QMF query, or remain invisible, showing the costs only when queries exceed one or more installation limits (CPU time, I/O, elapsed time).

Typically, the choice is to show the costs only when a query exceeds a limit. After presenting the SQL PA Cost Summary report and any warning messages for high usage of resources, the QMF Interceptor Exit asks if you want to cancel the query. If you respond Y (yes), the query is not run. If you respond N (no), processing continues despite the high-cost estimate.

Note: You can configure Db2 SQL Performance Analyzer to prevent queries with excessive cost estimates from running. If a query exceeds a limit, you have no option to run it.

The following figure shows an example of canceling a query with an excessive CPU time estimate.

```
ANL5025W *** WARNING:
ESTIMATE OF
           73.30724 EXCEEDS "CPU TIME" LIMIT OF
                                          10 CPU SECONDS!
QMF ONLINE QUERY WILL REQUIRE 115.09003 SECONDS OF ELAPSED TIME
  DURING WHICH 73.30724 SECONDS OF CPU TIME WILL BE CONSUMED AND
                                                   *
  A TOTAL OF
              784 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK
        1012 ESTIMATED PROCESSING COST $
                                      49.7052 DOLLARS
* OUNITS
                                                   *
*** WOULD YOU LIKE SQL PA TO CANCEL THIS QUERY (Y/N) ?
***
```

Figure 47. Canceling a query under QMF

After cancellation, you return to the QMF Query Panel, and a message indicating that the query was canceled by the Db2 SQL Performance Analyzer governor exit displays on the screen, like the following figure.

*** END ***						
1=Help 7=Backward	2=Run 8=Forward	3=End 9=Form	4=Print 10=Insert	5=Chart 11=Delete	6=Draw 12=Report	
YOUR QUERY WA COMMAND ===>	AS CANCELLED	BY THE SQL	PA GOVERNOR	EXIT.	SCROLL ===> F	PAGE

Figure 48. Query canceled message

If the query is not canceled by you or by the install option, QMF proceeds as normal. The following figure shows the Status Panel with the QMF cost estimate. The cost estimate is relative for QMF it is not the cost estimate for SQL PA.

```
DATABASE STATUS PANEL
|
| Your request is currently being processed by the Database Manager. |
| The relative cost estimate for your request is: 2 |
```

Figure 49. Cost estimate for QMF

Finally, the QMF results panel is displayed, showing the result of the query.

QMF batch processing

You can use batch processing, if your QMF queries are estimated to have long runtimes and you do not want to run them online.

If the QMF Intercept Exit estimates a long runtime and heavy resource usage for your QMF queries, and you would rather not run them online, you can do one of two things after canceling the query:

- 1. Structure the query in a different way, perhaps qualifying with additional predicates that might allow indexed access more efficiently.
- 2. Transform the ad hoc query into a QMF batch job, where it can run in the background while you continue to process other, shorter queries online.

Create batch query jobs, by entering RUN Q.DSQ1EBIN on the command line and then following the QMF instructions. See your QMF user manuals for more information about running QMF batch queries.

Using the SQL PA stored procedure

The SQL PA analysis of SQL statements is available through Db2 stored procedures named ANLPRER or ANLPRCR.

Users from any address space or Db2 Connection can issue a stored procedure call to ANLPRER or ANLPRCR to request the cost of a query before running it. SQL PA returns the costs of the query, along with warnings and return codes indicating if the query runs within the defined limits.

A Db2 stored procedure is a specialized program, or procedure, that is called from any Db2 application program, including Db2 clients in PCs, and applications in CICS, IMS or other regions. The procedure is called like a subroutine or subprogram, accepting host variables as input, and returning host variables and row sets as output.

The SQL PA stored procedure ANLPRCR uses the RRSAF method. An entry for ANLPRCR must be placed in the SYSIBM.SYSROUTINES catalog table and describe the characteristics of the procedure, such as its parameter list. The procedure must be bound as a package on the system before applications can call it.

For more information about Db2 stored procedures, refer to the Db2 publications.

Topics:

"Accessing the SQL PA stored procedure" on page 120

- "ANLSTPR test program" on page 120

Accessing the SQL PA stored procedure

You can access the SQL PA stored procedures using the supplied PL/I sample programs ANLSTPR and ANLSTER.

The source code for all both programs is located in *high_level_qualifier*.SANLSAMP. The execution JCL is also provided in *high_level_qualifier*.SANLJCL.

ANLSTPR test program

You can use the ANLSTPR test program to test your stored procedures.

To start the ANLSTPR test program, run the *ssid*STPR JCL job in *high_level_qualifier*.SANLJCL. The JCL of the test program is shown below.

When Generic ID needs to own EXPLAIN tables, modify the following JCL as stated in the "Note" in this sample (below).

To start the ANLSTPR test program, run the *ssid*STPR JCL job in *high_level_qualifier*.SANLJCL:

```
//JOBNAME JOB (ACCTG), 'PLI RRSAF', CLASS=1, MSGCLASS=X, NOTIFY=USERID
//*DSN9STPR
//*
//* CALLS EXPLAIN-ENHANCED STORED PROCEDURE ANLPRCR (RRSAF)
//* STORED PROCEDURE (ANLPRCR) TEST PROGRAM - ANLSTPR (PL/I) SAMPLE
//*
//*
//*
    LICENSED MATERIALS - PROPERTY OF IBM CORPORATION - 5697-W51
       COPYRIGHT IBM CORPORATION 2000, 2015 ALL RIGHTS RESERVED
//* US GOVERNMENT USERS RESTRICTED RIGHTS - USE, DUPLICATION OR
//* DISCLOSURE RESTRICTED BY GSA ADP SCHEDULE CONTRACT WITH IBM.
//*
//* Note: To use a Generic ID as the PLAN_TABLE owner, add
    '+OFF+' as third parameter of PARM= for ANLSTPR program.
//* '+OFF+' as third parameter of PA
//* Sample : PARM='ssid,JOHNDOE,+OFF
//STEP1 EXEC PGM=ANLSTPR,PARM='ssid,JOHNDOE'
//STEPLIB DD DISP=SHR,
         DSN=SYSX.ANL510.SANLLOAD
11
11
         DD DISP=SHR, DSN=SYS1.CEE.SCEERUN
       DD DISP=SHR, DSN=SYS1.DSN####.SDSNEXIT
//
11
       DD DISP=SHR, DSN=SYS1.DSN####.SDSNLOAD
//DSNTRACE DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//QUERYIN DD DISP=SHR
         DSN=SYSX.ANL510.SANLSQL(ANLSTEST)
```

The QUERYIN DD statement points to the file containing the SQL statements that you want to analyze. If the test program was not bound properly before running, an SQL -805 error code could result.

For each SQL statement in your test input file, the test program ANLSTPR lists the SQL statement, the SQL error code, the Db2 SQL Performance Analyzer warning flags, and the Db2 SQL Performance Analyzer estimates for elapsed time, CPU time, I/O count, QUNITS, and cost, as shown in the following figure.

```
* RRS IDENTIFY RETURNS
                              0
                                         0
                                                     0
* SIGNON USER RETURNS
                              0
                                         0
                                                     0
* CREATE THREAD RETURNS
                                         0
                                                     0
                              0
   EXPLAIN PLAN FOR (LENGTH
                                240)
*
    UPDATE SYSIBM.SYSTABLES
*
    SET NPAGES = -1
    WHERE NPAGES = -1
* ANLPRCR SQLCODE IS
                                    0
* ANLPRCR RETURNS ==> WARNING FLAGS: -----
   ELAPSED: 38.70953 CPU TIME:
                                         0.27816
  I/O COUNT:
                    101
                           QUNITS:
                                              16
  MONETARY:
                    1.18
  ANL CODE:
                      0
                           SQL CODE:
                                                0
* EXPLAIN PLAN FOR (LENGTH
                                320)
   SELECT SUM(NTABLES), AVG(PARTITIONS) FROM SYSIBM.SYSTABLESPACE
WHERE CREATOR <> 'SYSIBM'
*
*
    AND NTABLES > 1 AND
*
    SEGSIZE = 0
* ANLPRCR SQLCODE IS
                                    Θ
* ANLPRCR RETURNS ==> WARNING FLAGS: -----
   ELAPSED: 1.09482 CPU TIME:
                                          0.08612
               18
0.00
                           QUNITS:
  I/O COUNT:
                                               5
  MONETARY:
                    0.20
                           SQL CODE:
  ANL CODE:
                     0
                                                0
   EXPLAIN PLAN FOR (LENGTH 640)
*
    SELECT A.CREATOR, A.NAME, B.COLNAME, B.ORDERING, C.COLSEQ
FROM SYSIBM.SYSINDEXES A, SYSIBM.SYSKEYS B, SYSIBM.SYSFOREIGNKEYS C
*
*
*
    WHERE A.CREATOR = B.IXCREATOR
    AND A.NAME = B.IXNAME
AND A.NAME = C.TBNAME
*
*
    AND A. CREATOR = C. CREATOR
*
          A.COLCOUNT > 1
    AND
*
    ORDER BY C.COLSEQ
* ANLPRCR SQLCODE IS
                                    0
* ANLPRCR RETURNS ==> WARNING FLAGS: YY-YY
  ELAPSED: 5223.66996 CPU TIME: 3403.41581
I/O COUNT: 112 QUNITS: 185853
  MONETARY:
                771.94
                           SQL CODE:
  ANL CODE:
                  Θ
                                               0
* RRS TERMINATE IDENTIFY
                              0
                                         0
                                                     0
* PROGRAM TERMINATION
```

Figure 50. ANLSTPR sample program output

Plan Table report levels

You can use report levels to define the amount of information you want to see in your Plan Table reports.

You can use a summary report to get a general idea of the state of SQL statements in a large plan. Plan Table reporting time might be considerable if a large plan with several hundred SQL statements is being investigated.

To get an idea of the state of the SQL statements in this large plan, it is recommended that you produce only a summary page as the first run. Use the LEVEL statement to produce a summary report, as shown in the following example:

```
Level=Summary
Package=ANL*.ANLITST
```

The one page report example is shown in the following figure, where the access path of each statement is given in one line for each miniplan.

SQL Performance Analyzer Plan Table Report Summary Page _____ ANL Release: 5.1.00 of 2015-09-16 APAR: ANL510 Release:11.1.5DB2 Macro Change Level:V11R1M0Release:7.0.4z/OSSystem id:MVSB DB2 MVS System id: DB2X SQL id: DPGROTH Product Code: 5740-XYR Plan Name: ANLEEE DB2 Curr. SQL id: Package Location: DALLAS-CQ Explain Location: CPHMVS1_DB2X Coded Char. Set Id: 37 Start date: 2005-05-14 Stop date: 2005-05-14 Stop time: 13:17:53 Stop time: 13:17:57 Elapse time: 00:00:04 1 SQL, QNO, PLN, and QMF requests were processed: Level=Summary Summary reports requested 1: Package=ANL*.ANLITST DBRM/Pack Stmt Typ Plan owner: DPGROTH Bind date: 1992-01-18 ANLITST 3 D L Matching index scan (1/2) - data pages scan ANLITST 3 D Additional Sort for ORDER BY ANLITST3 D S* Table space scan - no index will be usedANLITST32 D S* Table space scan - no index will be usedANLITST32 D L Matching index scan (1/2) - data pages scanANLITST32 D Additional Sort for ORDER BY Start time: 13:17:53, Stop time: 13:17:57, Date: 2005-05-14 - - -Program Property of IBM (c) Copyright IBM 1986-2015

Figure 51. Summary page

If this report does not provide sufficient information and you want a fast run time, the next step could be to produce a report that contains the explain and access path information, but excludes index, table, and table space information. Such a report is produced using the following combination of statements:

Level=**NoCatalog** Package=ABC*.LargeOne

Each explainable SQL statement in the plan is a separate page in the report.

For table and table space information, without index information, specify the following statement combination:

Level=**Detail** Package=ABC*.LargeOne,index=**No**

This combination of statements results in a fast response time because the non-indexed system catalog table SYSIBM.SYSKEYS is not accessed.

To retrieve all catalog information, including key distribution and information for all indexes on tables, where the investigated SQL statements perform table space scans, specify the following combination of statements:

```
Level=Detail
Package=ABC*.LargeOne,index=T,level=Keydist
```

EEEPATH table

The EEEPATH table is populated using Plan Table and contains the access plans for Plan Table queries.

This table joins other Plan Table Db2 tables (EEEPLAN, EEEDBRM) as data repositories held and managed by Plan Table. Periodically, Plan Table maintenance deletes old records from all these repositories. To manage your own access plans, you must create a view and maintain your own copy of this table. The table contains the essential PLAN_TABLE and DSN_STATEMNT_TABLE cost items relative to the access path. It is used to generate the old access path cost estimates, and the old access plan itself. Plan Table provides SQL statements from its data sources, such as the Catalog for plan or package statements, and stores the QUERYNO, APPLNAME, and TIMESTAMP to tie the SQL from *user_ID*.ANLEEE.SQL into these saved records.

Db2 SQL Performance Analyzer can take this existing plan data in EEEPATH, perform its access path cost analysis, and use the SQL from *userid*.ANLEEE.SQL to prepare a new cost analysis and plan. SQL PA then prepares a report that shows the OLD plan and cost, the NEW plan and cost, and a comparison of the two. For each user, *authorization_ID*.EEEPATH is the name of this table. Db2 SQL Performance Analyzer uses the USEPLAN parameter to select the *authorization_ID* qualifier.

The QUERYNO is passed with the SQL statements that correspond to the plan records captured in the EEEPATH table. The Application Name and Timestamp help to further identify the proper records for consideration and analysis.

SQL data file: user_ID.ANLEEE.SQL

The user_ID.ANLEEE.SQL data file is the input source for the Explain process.

The SQL extracted from any Plan Table data source (for example, a Plan, Package, QMF, Source Program, or File Input) is passed to Db2 SQL Performance Analyzer in a standard fixed block 80 byte record file, within cc 1-72 (leaving 73-80 for line numbers, if any). It uses a standard name to differentiate it as the Plan Table input source: *user_ID*.ANLEEE.SQL, qualified by the TSO user ID which is the authorization ID that is running the programs. You can set up this file with a different prefix.

VIADRDA remote connect

SQL PA can process a remote copy of the product running under another Db2 subsystem, on the same server or another server, by using the DRDA protocol and the SQL CONNECT statement.

The VIADRDA (package explain location) parameter is equivalent to the Explain location field on the Process packages, the Process DBRMs and the Process SQL panels. The PKGDRDA (package information location) parameter is equivalent to the Package location field on the Process packages panel. If the VIADRDA parameter is specified for DB2 SQL Performance Analyzer, SQL requests for an explain are routed to other copies of the DB2 SQL Performance Analyzer programs on remote systems. If the PKGDRDA parameter is specified for DB2 SQL Performance Analyzer, SQL requests for package information are routed to other copies of the DB2 SQL Performance Analyzer, SQL requests for package information are routed to other copies of the DB2 SQL Performance Analyzer, SQL requests for package information are routed to other copies of the DB2 SQL Performance Analyzer, SQL requests for package information are routed to other copies of the DB2 SQL Performance Analyzer, SQL requests for package information are routed to other copies of the DB2 SQL Performance Analyzer, SQL requests for package information are routed to other copies of the DB2 SQL Performance Analyzer programs on remote systems.

Define the VIADRDA and PKGDRDA parameters in the ANLPARM file or on the Change System Parameters panel under TSO ISPF. Setting the parameter to +OFF+ turns off the feature. The TSO parameter remains set to the default (+OFF+) value unless the TSO user wants to route processing to another Db2 subsystem. These users might experience some delay in response time if the dispatching priority of DDF requests on the remote system is not as high as the local Db2.

VIADRDA and PKGDRDA allow routing of your SQL request to other copies of Db2 SQL Performance Analyzer, while running on a local system. To use TSO to explain a remote package, specify the Package location and Explain location fields on the Process Packages panel. To explain a remote package in batch, set the VIADRDA and PKGDRDA parameters in the user parameter file.

The format of the VIADRDA and PKGDRDA parameters are:

VIADRDA explain-location-name | +OFF+

Where explain-location-name, a 16-character field, is any valid explain location specified in the local SYSIBM.LOCATIONS table. You can leave the parameter in ANLPARMs with a value of +OFF+ to make it non-operational.

Because of the size of the parameter field (16 characters), the comments for this parameter must start in column 31.

REPORTS ALL	SELECT REPORT LEVEL: YES, EXP OR ALL
VIADRDA DALLAS-CQ	ROUTE EXPLAIN REQUEST TO DALCQ SYSTEM
DBRMKEY +OFF+	NO DBRM LIBRARY MEMBERS SELECTED

PKGDRDA package-location-name | +OFF+

Where package-location-name, a 16-character field, is any valid package location specified in the local SYSIBM.LOCATIONS table. You can leave the parameter in ANLPARMs with a value of +OFF+ to make it non-operational. Because of the size of the parameter field (16 characters), the comments for this parameter must start in column 31.

REPORTS ALL	SELECT REPORT LEVEL: YES, EXP OR ALL
PKGDRDA DALLAS-CQ	ROUTE PACKGE INFO REQUEST TO DALCQ SYSTEM
DBRMKEY +OFF+	NO DBRM LIBRARY MEMBERS SELECTED

The remote locations must contain fully installed versions of Db2 SQL Performance Analyzer, including a REGISTRY table and the following explain tables:

- PLAN_TABLE
- DSN_STATEMNT_TABLE
- DSN_FUNCTION_TABLE
- DSN_DETCOST_TABLE
- DSN_PREDICAT_TABLE

Users can run Db2 SQL Performance Analyzer, their SQL, and Db2 subsystem, through the DRDA Connect facilities.

If a local explain of a local package is desired, both VIADRDA and PKGDRDA should be set to +OFF+. If a local explain of a remote package is desired, VIADRDA should be set to +OFF+ and PKGDRDA should be set with the server of the remote package. If a remote explain of a local package is desired, VIADRDA should be set with the server to be used for the explain and PKGDRDA should be set to +OFF+. If a remote explain of a remote package is desired, both VIADRDA and PKGDRDA should be set to +OFF+. If a remote explain of a remote package is desired, both VIADRDA and PKGDRDA should be set to the remote server.

Protecting sensitive user table data

It is important to limit who has access to user data. The DATAGRP parameter enables you to control which users have authority to sensitive user table data.

About this task

SQL PA displays user table information in a few places, such as HIGH2KEY and LOW2KEY. To determine which users have the authority to see user table data, a system parameter, DATAGRP, identifies the user group with the proper authority. The default for the DATAGRP parameter is +0FF+, which disables this feature and allows all users to see user table data.

To limit access to user table data, DATAGRP must be set to the user group for which user table data will be displayed. If a user does not belong to the user group specified by DATAGRP, a value of N/A, blanks, or null will be used as a data value mask:

- When viewing the Plan Table report, the masked value will be displayed in place of user table data for COLVALUE, HIGH2KEY, and LOW2KEY.
- When collecting statistics (option 2.2.1), the masked value will be included in the DML that is created when inserting rows containing COLCARDDATA, COLVALUE, HIGH2KEY, HIGHKEY, HIGHVALUE,

KEYVALUE, LOW2KEY, LOWKEY, and LOWVALUE. Where the DML is updating HIGH2KEY and LOW2KEY values, the updates will be omitted from the generated DML.

Procedure

- 1. Using an existing security application, such as RACF, create a user group for those users who have the authority to view the user table data.
- 2. Protect the EEEAUTH table by not granting another authority other than SELECT to users other than the SQL PA owner. The EEEAUTH table is used by SQL PA and is created during customization with only SELECT authority granted to public.
- 3. Set the DATAGRP value to the name of the user group that is allowed to view the user table data. DATAGRP can be set during customization or it can be updated through SQL PA using Option 3.3, "Change the DB2 SQL parameters", as long as the user has the authority to change the system parameters and has update authority to EEEAUTH.

Note: It is recommended that the DATAGRP value only be changed by the SQL PA owner during customization.

Chapter 4. SQL PA run parameters

There are two sets of parameters used by SQL PA.

One set of parameters is user-oriented and can be found in *hiqual*.SANLPARM member *ssid*PARM.

The second set of parameters describes the configuration of the production system where the application being processed by SQL PA actually runs, and can be found in *hiqual*.SANLPARM member *ssid*CNTL. The *ssid*CNTL parameters describe the configuration of the host system. The CNTL parameters include the type and speed of the CPU processor, the Db2 buffer pool sizes, and other systems-oriented information. Configuration parameters also include the costing guidelines and excessive use limits which trigger warning messages from SQL PA. The best practice is to implement the parameters consistently and uniformly for every user of SQL PA.

You can also use the ISPF interface to modify the SQL PA parameters by using Modify Configuration panel option of Change SQL PA parameters.

Topics:

- "Impact on cost estimates" on page 127
- "Parameter file characteristics" on page 128
- "Specifying ANLPARM user parameters" on page 129
- "Specifying threshold control parameters" on page 129
- "Specifying ANLCNTL configuration parameters and selecting the target host system" on page 130
- "Enabling the predicate analysis report" on page 130
- "Optimizing the table and SQL report processing" on page 131
- "The difference between SANLDATA and SANLPARM" on page 131
- "Qualifying unqualified objects in packages" on page 132
- "Qualifying unqualified objects in DBRMs and SQL files" on page 132

Impact on cost estimates

Most of the parameters specified in the ANLPARM and ANLCNTL parameter sets have some impact on the internal formulas and costing algorithms used by SQL PA.

The following list reviews the impact that specific parameters have on cost estimates.

BUFF08K

Has equivalent effect of BUFFERS on 8 KB page applications.

BUFF16K

Has equivalent effect of BUFFERS on 16 KB page applications.

BUFF32K

Has equivalent effect of BUFFERS on 32 KB page applications.

BUFFERS

Influences the number of pages contained in a prefetch block for user requests from 4 KB buffer pools.

BUFFHIT

Governs the logical-to-physical I/O ratio, allocating a portion of the total logical I/O to disk; impacts the I/O waits, elapsed time, and other calculations.

CONNECT

Imposes processor usage charges for thread management, attach facility and application logic, different for each connection chosen.

CPUCOST

Establishes the base for the monetary cost estimates of CPU time.

DEGREES

Enables the optimizer to consider parallel processing; impacting sequential prefetch, I/O wait, and elapsed time for table scans.

DSGROUP

Influences the processor usage applied to data sharing in a Sysplex environment, and the overall processing capacity of the system.

DYNAMIC

Influences the processor usage for dynamic SQL processing.

IOSCOST

Establishes the base for the monetary cost estimates of physical I/O.

HPOOLRD

Governs read time of physical I/O: are they found in expanded storage, or read from disk? Impacts elapsed times and I/O waits; also influences asynchronous writes back to pool after insert, update, or delete activity.

NEWSTOR, NEWSEEK, NEWROTA, and NEWXFER

Redefine the basis for service times calculated for all physical I/O calls, based on the seek, revolution speed and transfer rate of the new devices. The parameters also impact the wait time estimates and contribute to overall elapsed time.

PRECISE

Causes SQL PA's calculated CPU times and QUNITS to be replaced with estimates from the Db2 optimizer, and only CPU time is considered in SQL PA's costs when PRECISE is YES.

REFRESH

Allows the optimizer to consider materialized query tables (MQTs) that are summaries, aggregates, or reduced row versions of the base tables as access path candidates.

STORAGE

Provides the basis for service times calculated for all physical I/O calls, based on the seek, revolution speed and transfer rate of the devices. This parameter also impacts the wait time estimates and contributes to overall elapsed time.

TIMCOST

Establishes the base for the monetary cost estimates of elapsed time.

VERSION

Selects which set of path lengths are used, with different instruction counts on some functions.

You can obtain many varied results by manipulating the SQL PA parameters. It is important to remember that SQL PA can reflect any existing or proposed configuration description in its parameter set.

Parameter file characteristics

All Db2 SQL Performance Analyzer parameters, when stored in files, have certain characteristics in common.

The common characteristics for SQL PA parameters include the following characteristics:

- Each parameter keyword begins in column 1, is exactly seven characters long, and is followed by a blank.
- You can add comments to any parameter after column 20 except for those parameters whose value can be longer than 10 characters.
- The maximum length of a comment is 60 characters.
- Non-line numbered files are preferred and fixed block 80 byte records are expected.

Specifying ANLPARM user parameters

You can change Db2 SQL Performance Analyzer user parameters to get results that are tailored to your specific requirements.

Before you begin

Create a backup copy of the *hiqual*.SANLPARM member to preserve the original settings.

For more information about the ANLPARM parameters, see "ANLPARM user parameters" on page 230.

Procedure

Use one of the following methods to change the ANLPARM parameters:

- Use ISPF to change the ANLPARM parameters by editing the *high_level_qualifier*.SANLPARM member *ssid*PARM and change the parameter values as necessary.
- Use SQL PA to change the ANLPARM parameters by completing the following steps:
 - a. On the SQL PA main menu, specify option 3, **Modify DB2 SQL PA Configuration**. The **Modify Configuration** panel is displayed.
 - b. Specify option 3, **Change the DB2 SQL PA parameters**. The **Parameter Data Sets** panel is displayed.
 - c. Optional: If (USRPARM) Allow users to have a copy of the user parameters on the Product Parameters panel in Tools Customizer had a value of YES, specify a data set name in the User-level parms field. The user parameters from the data set that is specified in the SQL PA user parms field are copied into the data set that is specified in the User-level parms field. If this data set does not exist, it will be created.
 - d. In the **Edit SQL PA user parms** field, specify YES, and press Enter. The **User Parameters** panel is displayed.
 - e. Modify any parameters, and press Enter.

Specifying threshold control parameters

Use the threshold control parameters to control the levels at which certain SQL Advisor messages are issued.

About this task

You can turn off certain messages and capture only the message ID, excluding the text, using threshold control parameters.

Procedure

1. Determine which of the threshold parameters you want to use for your environment.

For detailed information on each of the parameters, see <u>"ANLPARM user parameters" on page 230</u>.

- 2. Modify one or more of the following threshold parameters by using the ISPF or the ANLPARM file.
 - INLISTS nnn
 - ISCANPG nnnnnnn
 - IXUPDAT nnn
 - JOINTAB nnn
 - MATCOLS n.nn
 - NONIXPG nnnnnnn
 - NOSTATS <u>YES</u> | NO
 - PREDPCT n.nn

- TSCANPG nnnnnnn
- TURNOFF nnnn

Specifying ANLCNTL configuration parameters and selecting the target host system

Db2 SQL Performance Analyzer has the capability to project its anticipated cost and performance estimates on any configuration, including production host machines and hypothetical configurations not yet in place. The ANLCNTL parameters are not normally modified by Db2 SQL Performance Analyzer users.

About this task

The parameters define a target host configuration.

You can run Db2 SQL Performance Analyzer on a test machine and to forecast application performance on a production system.

You can have multiple configurations (PDS members) in batch using the ANLCNTL DD statement to point to the *hiqual*.SANLDATA PDS member, and in TSO by having multiple members in the *hiqual*.SANLPARM PDS.

The ANLCNTL system configuration parameters are stored in a PDS library, allowing you to select a specific target system (PDS member) for the cost analysis of SQL PA.

For SQL PA operating under TSO online, the *hiqual*.SANLPARM library is provided as a place to store the various configurations for different host systems, including ones that exist at your site and sample configurations that you want to test against. This PDS is populated with members for each Db2 host during installation.

In batch mode you can allocate one of these members in the ANLCNTL DD statement, point to your own copy as a flat file, or use SYSIN DD * and define them in-stream.

Procedure

1. Determine which of the configuration parameters you want to use for your environment.

For detailed information about each of the parameters, see <u>"ANLCNTL configuration parameters" on</u> page 226.

- 2. For each member (subsystem, LPAR, or configuration), edit the member for ANLCNTL configuration parameters or use ISPF. To use ISPF to edit the parameters, use the following steps:
 - a) From the primary menu, select Modify Db2 SQL PA Configuration.
 - b) Select Change the Db2 SQL PA parameters.
 - c) Specify Yes on Edit SQL PA system parms.
 - d) Modify and save the new parameter values.

Enabling the predicate analysis report

Enabling this feature is optional, but if it is turned on, the extra predicate analysis report is generated. If it is not activated, the PRECISE ALL parameter does not produce a predicate analysis report.

About this task

The ALL setting provides more precise information in evaluating SQL for Db2, including a Predicate Analysis section in the Explain and Detailed Trace reports.

When PRECISE is set to YES, Db2 SQL Performance Analyzer attempts to replace its path length or instruction count estimates (the CPU time, essentially) with the internal estimates from the Db2 optimizer, drawn from the Explain table DSN_STATEMNT_TABLE.

The QMF Interface for Db2 SQL Performance Analyzer and the Db2 SQL Performance Analyzer stored procedure can also use the optional Explain table (DSN_STATEMNT_TABLE) with the PRECISE YES parameter.

The default for this parameter is YES.

The option of using the optimizer's own internal cost estimate (path length) to process each SQL request is available in all the Db2 SQL Performance Analyzer environments—batch, TSO, QMF, and as a Db2 stored procedure. This ensures the highest available level of accuracy attainable by Db2 SQL Performance Analyzer users from any environment.

Procedure

To generate a predicate analysis report in Db2 use the following procedure.

- 1. Edit the member for ANLCNTL configuration parameters or use ISPF.
 - To use ISPF to edit the PRECISE parameter, use the following steps:
 - a) From the primary menu, select Modify Db2 SQL PA Configuration.
 - b) Select Change the Db2 SQL PA parameters.
 - c) Specify Yes on Edit SQL PA user parms.
 - d) Modify and save the new parameter value.
- 2. Run the *ssid*JSPA job using the PRECISE ALL parameter.

The predicate analysis report is also available under TSO online.

Optimizing the table and SQL report processing

You can optimize table and SQL report processing by bypassing the Explain function.

About this task

To retrieve table report information without having to run EXPLAIN on every statement, set the FASTTBL parameter to YES.

FASTTBL=NO shows information about the shared and exclusive Db2 locks, and the SEL, DEL, INS, UPD, and MER columns reflect query blocks, LIS and LIX reflect Db2 locks, and STM reflects SQL statements (not query blocks).

Procedure

- 1. Review the FASTTBL parameter settings.
 - For detailed information, see <u>"ANLCNTL configuration parameters" on page 226</u>.
- 2. Edit the member for ANLCNTL configuration parameters or use ISPF.
 - To use ISPF to edit the FASTTBL parameter, use the following steps:
 - a) From the primary menu, select Modify Db2 SQL PA Configuration.
 - b) Select Change the Db2 SQL PA parameters.
 - c) Specify Yes on Edit SQL PA system parms.
 - d) Modify and save the new parameter values.

The difference between SANLDATA and SANLPARM

The Db2 SQL Performance Analyzer ISPF interface is intended to use the *hiqual*.SANLPARM library for its parameter values, leaving the *hiqual*.SANLDATA library available for different values that can be used for the batch interface.

Either interface can specify the members from either library. However, there are certain parameter values that are only valid for batch use and are ignored when using ISPF.

When setting up the Db2 SQL Performance Analyzer stored procedure, a copy of the SANLPARM member *ssid*CNTL is recommended. The copy can be allocated to the stored procedure address space so that it does not prevent other users from accessing those parameters.

Qualifying unqualified objects in packages

Occasionally, you need to explain statements from packages that have unqualified objects. Objects are qualified using the qualifier of the package as obtained from the catalog.

Qualifying unqualified objects in DBRMs and SQL files

When explaining statements from DBRMs and SQL files that have unqualified objects, change the qualification using the ISPF interface or the batch interface.

About this task

Occasionally, you need to explain statements from DBRMs and SQL files that have unqualified objects. Objects are qualified differently depending on the interface you are using.

For ISPF interfaces, there is an object qualifier field on the panel. The field is used to qualify the tables, but if it is blank, the SQL ID displayed on the panel is used. Only one value for this field is allowed.

For batch interfaces, the QUALIFY user parameter is what qualifies your unqualified objects. For more information about using the QUALIFY parameter, see "ANLPARM user parameters" on page 230.

- 1. For ISPF interfaces, change the SQL ID by modifying the configuration (option 3 from the primary menu) and then changing the current Db2 SQLID (option 2).
- 2. For batch interfaces, define the QUALIFY parameter in the ANLPARM user parameter file.

Chapter 5. Plan Table report input specifications

You can use Plan Table input commands to set the parameters that SQL PA uses for analysis.

Topics:

- "Explain process" on page 133
- "Configuring table access " on page 133
- "Plan table access" on page 134
- "Input command requirements" on page 134
- "Input commands" on page 135
- "Input command definitions" on page 140
- "Subparameter definitions" on page 142
- "Specifying language and explaining data from another user" on page 148
- "Selected authorization identification" on page 149
- "User authorization table" on page 149
- <u>"User defaults data set" on page 150</u>
- "Transferring complex queries to SQL PA" on page 151

Explain process

The Explain process performs several checks and activities when it runs.

Explain checks to make sure that all necessary tables are accessible.

If all necessary tables are accessible, Explain reads the first input command. The command is checked for validity, and one of the main processes is started.

Each input specification produces a complete report. Each report can consist of several pages.

If LEVEL=SUMmary is in effect, information for the current SQL statement is shown only in the summary page.

At normal completion of Explain, the last page shows a listing of all the input specifications with prefixed report numbers. You can use this page as a table of contents.

Configuring table access

The plan table report makes extensive use of Db2 system catalog tables.

About this task

For a successful plan table report, you must define several settings.

- 1. Define SELECT authorization to the following catalog tables:
 - SYSIBM.SYSDATABASE
 - SYSIBM.SYSTABLESPACE
 - SYSIBM.SYSTABLES
 - SYSIBM.SYSCOLUMNS
 - SYSIBM.SYSINDEXES
 - SYSIBM.SYSFIELDS

- SYSIBM.SYSKEYS
- SYSIBM.SYSSTMT
- SYSIBM.SYSPLAN
- SYSIBM.SYSDBRM
- SYSIBM.SYSPACKLIST
- SYSIBM.SYSPACKAGE
- SYSIBM.SYSPACKSTMT
- SYSIBM.SYSCOLDIST
- 2. To be able to run EXPLAIN on saved QMF queries, SELECT authorization is needed for the following QMF tables:
 - Q.OBJECT_DIRECTORY
 - Q.OBJECT_DATA
- 3. To be able to run EXPLAIN on stabilized dynamic queries, SELECT authorization is needed for SYSIBM.SYSDYNQRY.
- 4. If you do not want the current user's PLAN_TABLE table to be created in the default database, create a PLAN_TABLE table in a specified database and table space before you run a plan table report with an SQL, QMF, or SDQ statement.

Plan table access

The plan table report uses a PLAN_TABLE either prefixed with the authorization ID of the job submitter or belonging to the binder of the package being explained.

For each SQL, QMF, or SDQ command, the plan table report verifies that a PLAN_TABLE exists for the current authorization ID. The plan table report also verifies if a PLAN_TABLE exists if a dynamic EXPLAIN is to be issued for a DML SQL statement in a package.

If you have SELECT authorization to the PLAN_TABLE of another user, you can select EXPLAIN information from this table by specifying the authorization ID in the OWNER parameter of the QNO statement.

If a PLAN_TABLE does not exist and you are executing a dynamic explain using the SQL, QMF, or SDQ command, and you have CREATETAB authorization, a PLAN_TABLE is created in the default database. If the installation has specified a database and table space name in the location view of the EEEWORK table, the dynamically created PLAN_TABLE tables are created in the specified location.

If you are executing a dynamic EXPLAIN, the plan table report determines whether columns need to be added to the table, and adds them.

Input command requirements

The rules you must follow when using the input commands are explained.

To allow the parser to work, your input specifications must adhere to the following rules:

- Start the input command with one of the following statements:
 - Comment
 - Level
 - Package
 - QMF
 - QNO
 - SET
 - SQL
 - SDQ
- Specify only one keyword statement per line.
- Start the keyword statements in position 1, followed immediately by an equal sign (=). Further, the input parameter must immediately follow the equal sign (no embedded blanks). The format is free after the input parameter. You can specify keywords, Package and QMF names, and SQL statements in both upper and lower case.

An SQL statement can span several lines.

- End certain statements with a semicolon (;). If you are using the SQL= keyword, you must end the SQL statement with a semicolon (;) if it is followed by another input command. You can also optionally stop the LEVEL, SET, QNO, QMF, SDQ, and PACKAGE commands with a semicolon (;).
- Generate a Summary report by specifying a level statement, such as LEVEL=SUMmary.

Input commands

The input commands are read from a data set with the ddname INPUT. The following information describes detailed command specifications.

Comment statement



If the first two characters of a command line are hyphens (--), or the first character of a command line is an asterisk (*), the entire command line is regarded as a comment. In the first case, the comment is printed on the Summary Page, but if the first character is an asterisk (*), the comment is not shown in the output.

Level statement



The LEVEL command controls how much information is shown in reports or other output. The command must have one of the following two formats:

```
Level=xxx
LVL=xxx
```

DETail is the default value. For more information about the acceptable values, see <u>"Input command</u> definitions" on page 140.

Note: A LEVEL specification is in effect until a new LEVEL value is specified.

SET statement



The degree of parallelism for a dynamic SQL (or QMF) statement is set by setting the current degree to ANY or 1 (default). The setting is in effect until a new SET CURRENT DEGREE statement is issued.



The setting of a current SQL ID is done using the same statement as in standard SQL DML. If SET CURRENT SQLID=USER is specified, the primary SQL ID is set; otherwise, a valid secondary SQL ID is set. This setting is in effect until a new SET CURRENT SQLID statement is issued.

Note: For compatibility with prior releases of Plan Table, the SQLID= or SID= statement is still supported (SQLID=SecUser).

Current optimization hint		
► SET CURRENT —	HINT = ── 'Hint String' ── ,ID = ── number →	
	— OPTIMIZATION —	
►_;_►		

Db2 hints are passed to the Optimizer through the use of the SET CURRENT statement. In the top part of the Explain report, Explain displays an ID number for the current miniplan:

Opthint: , Hint: , Dir.Row: No , OptHint-ID: 1321

The number 1321 in the following example uniquely identifies a miniplan, where a Db2 hint is given. If you decide to supply a Db2 hint for a given miniplan, you can issue a SET CURRENT HINT command, where the ID-number is supplied together with a hint string, such as:

SET CURRENT HINT = 'Myhint', id=1321

SQL Statement

► SQL= — Explainable SQL Statement (several lines) -:->

To run Explain on an SQL statement, the SQL command must have the following format:

SQL=xxx ;

The statement can span several lines and can include comments (indicated by --). The statement must be ended by a semicolon (;), if it is followed by another Explain statement.

QMF Statement



To run Explain on a QMF query, the QMF command must have the following format, where *xxx* is an optional authorization ID of the query owner. If *xxx* is not specified, the query is assumed to be owned by the job submitter. The name of the QMF SQL query is *yyy*.

QMF=[xxx.]yyyzzz

The valid values for the INDEX subparameter are described in "Subparameter definitions" on page 142.

Package Statement



To run Explain on a package, the PKG command must have one of the following two formats, where a is the combination of the collection ID, the package ID, and the version ID:

```
Package=a[,index=c][,first=d][,last=e][,table=f][,acctype=g]
        [,format=h][,level=i][,force=j][,gen=k][,hostvar=1]
        [,location=m][,topa=n]
PKG=a[,index=c][,first=d][,last=e][,table=f][,acctype=g]
        [,format=h][,level=i][,force=j][,gen=k][,hostvar=1]
        [,location=m][,topa=n]
```

The valid values for the subparameters are described in <u>"Subparameter definitions" on page 142</u>.

Query Number Statement



To run Explain on a query number, the QNO command must have one of the following two formats, where *xxx* is an existing query number in the PLAN_TABLE that is being accessed:

QueryNo=xxx[,owner=yyy][,index=zzz] QNO=xxx[,owner=yyy][,index=zzz]

The OWNER subparameter *yyy* is the authorization ID of the PLAN_TABLE. See <u>"Subparameter</u> definitions" on page 142 for a description of the OWNER subparameter. For additional information see "Selected authorization identification" on page 149.

SDQ Statement



To run the plan table report on a stabilized dynamic query using DB2 12 or higher, the SDQ command must have one of the following formats, where *nnn* is the statement ID:

SDQ=[stabilizationgroupname](nnn)
SDQ=stabilizationgroupname(*)

Valid command examples

The following table displays some examples of valid commands.

Table 14. Valid command examples

Command	Result
QNO=1234,owner=XYZ	QUERYNO. 1234 in XYZ.PLAN_TABLE is processed
LVL=summary	Produces summary information only
QmF=user01.little_query,index=T	QMF query user01.little_query is processed. If table space scan is used, detailed index information for all indexes of the accessed table are shown. Otherwise, only information for the chosen index is shown
level=detail	Back to detailed information
SQL=select A, B, C from USER1 .TABLE3 where A>'abc' and B=78	The specified SQL query is processed

Invalid command examples

Some examples of invalid commands are shown in the following table.

Table 15. Invalid command examples		
Command	Problem	
qno= 2345;	A space after the equals sign (=) is not permitted	
QNO=ABCD	The query number is not numeric	
sql=SELECT * FROM DSN8610.EMP Qmf=ABC.MyQuery	The SQL statement must end with a semicolon (;)	

Input command definitions

The following information defines the input command statements and their valid values.

Comment

If the first two characters of a command line are hyphens (--), or the first character of a command line is an asterisk (*), the entire command line is regarded as a comment. In the first case, the comment is printed on the summary page, but if the first character is an asterisk (*), the comment is not shown in the output.

LEVEL

This function is used to control the level of detail. You can use Level or LVL as the keyword. By default, the report level is detail.

Use the LEVEL=*keyword* parameter to define the level of detail you want in reports. Alternatively, you can specify a LEVEL subparameter on the PLAN-statement to set the report level for the specified plan.

ACCesspath

When this parameter is specified, only the access path, table, and index are shown.

ALLindex

When this parameter is specified, all indexes for the accessed table are shown.

DETail

Any parameter that is not SUMmary, NOKeys, SQLsummary, and NOCatalog is regarded as DETail. Specifying this parameter results in a full detailed report for each SQL statement. DET is the default value for the Level parameter.

KEYdist

When this parameter is specified, the distribution of the 10 most used key values is shown. For all other information, the level is regarded as Detail.

NOCatalog

When this parameter is specified, only EXPLAIN data and the Access Path are shown. It does not produce index, key, table, table space, and bind information.

NOKeys

When this parameter is specified, all key field information is excluded from the report.

SQLsummary

When this parameter is specified, only the SQL statement and Access Path messages are shown.

SUMmary

When this parameter is specified, only the summary page with one line for each SQL statement is produced.

TABular

When this parameter is specified, the SQL statement and a brief report is displayed. Plan table information displays in tabular form followed by statistical information.

CONdensed

When this parameter is specified, the SQL statement and a condensed report is displayed. Plan table information displays in a condensed form that is similar to the tabular form followed by statistical information.

TIYes /TINo

These parameters toggle the printing of the table and index names for each statement in the summary page.

XON / XOFf

These parameters toggle the printing of the explain data listed immediately after the SQL statement being explained.

PACKAGE

When a collection and package name are supplied as input, Explain verifies that the specified package does exist in the catalog table SYSIBM.SYSPACKAGE. If the package does exist, Explain verifies that the package was bound with the EXPLAIN option. If it is not bound with an EXPLAIN option, an EXPLAIN command is dynamically issued for the individual DML SQL statement. The output is stored in the PLAN_TABLE table of the job submitter with a query number of 999,999,998.

The name of the package creator is extracted from the SYSIBM.SYSPACKAGE table.

The EXPLAIN information is selected from the PLAN_TABLE owned by the package owner if the job submitter has select authorization to this table. If not, an error message is posted and Explain continues with the next request.

A user ID that is specified in the PARMS field has no meaning for this function.

If all checks are positive, Explain begins processing the lowest query number in the earliest version within the package. The program proceeds to the next query number and produces a new report for this query number. Processing continues until all statements in the package have been processed.

The run time for this function depends on the number of rows in the accessed PLAN_TABLE. Generally, you create two indexes to your PLAN_TABLE table, if it holds more than 400 to 500 rows. One index is built on the QUERYNO column, the other index is created on PROGNAME, QUERYNO, APPLNAME. Alternatively, you can delete unnecessary rows from your PLAN_TABLE and run RUNSTATS against the table space.

A large part of the run time is spent accessing the non-indexed table SYSIBM.SYSKEYS to provide index information. If this information is not essential, you can suppress access to the table by specifying INDEX=N after the package name.

QMF

When using this function, you are able to EXPLAIN saved or generic QMF queries, providing the queries are written in the SQL language.

You must convert QBE and PROMPTED queries to SQL before they are explained.

Apart from explaining your saved QMF queries, you can EXPLAIN a query created by another user if the query was saved with SHARE=YES.

The QMF query can contain parameters (such as &parm1 and &&parm2). These parameters can also substitute column names in the select list. However, the query must not contain literals and other strings with one or more ampersands (&) enclosed between quotes ('). Also, the query must not contain single quotes in a comment.

If all tests are positive, processing continues the same as with the SQL keyword.

QNO

This function can be used when a dynamic EXPLAIN statement has been run with a given query number.

The EXPLAIN statement can either have been run from DB2I or QMF. Specify the query number as a parameter to the QNO keyword. Use QUERYNO as an alternative to the QNO keyword.

The specified query number might also be a statement number from a Db2 precompile list, if the application was bound with the EXPLAIN(YES) parameter on the BIND statement.

If a user ID was specified in the PARMS field, Explain searches for the query number in the PLAN_TABLE that belongs to this user ID. If no PARMS field was specified, Explain searches for the query number in the PLAN_TABLE of the job submitter.

If the specified query number does not exist in the PLAN_TABLE, a warning is posted and Explain continues with the next command from the input data set.

SDQ

When using this function, you are able to display EXPLAIN information for the specified stabilized dynamic query.

SET

Setting of authorization ID, DEGREE, or hints.

SQL

When using this function, you supply an SQL DML statement as input to Explain. The supplied statement is explained by Explain, using a query number of 999,999,999. If this number exists in the PLAN_TABLE, Explain deletes the rows before processing the SQL statement.

Specify the SQL statement to be investigated in the same way as in DB2I, that is, a free format using several lines with optionally embedded comments (starting with --). If the statement is followed by another Explain statement, it must end with a semicolon (;), otherwise a semicolon is optional.

The EXPLAIN function is always run against the PLAN_TABLE of the job submitter. A user ID that is specified in the PARMS field has no meaning for this function.

After a successful EXPLAIN is run, the rows with the query number 999,999,999 are processed.

Subparameter definitions

The following information defines the subparameters and their valid values.

ACCTYPE

You can specify the ACCTYPE subparameter for the PLAN keyword to control the statements for explanation based on the chosen access path.

All

This parameter is the default value. For each explainable SQL statement in the plan, a report is produced.

Matching

When this subparameter is specified, only explainable SQL statements, where an access path of matching index scan has been selected, are processed. After processing the results are shown in the Explain report.

Nonmatch

When this subparameter is specified, only explainable SQL statements where an access path of non-matching index scan has been selected. After processing the results are shown in the Explain report.

Tabscan

When this subparameter is specified, only explainable SQL statements where an access path of table space scan has been selected. After processing the results are shown in the Explain report.

Hash

When this subparameter is specified, only explainable SQL statements, where an access path of hash access has been selected, are processed. After processing the results are shown in the Explain report.

The following examples show the format and use of this subparameter:

```
PKG=Coll.Pkgnm,AccType=M
Package=Coll.Pkgnm,index=T,first=486,acctype=tablespacescan;
```

In the first example, Explain only selects explain data for SQL statements having matching index scan and access path.

In the second example, only SQL statements with precompiler numbers greater than or equal to 486 and where the access path is table space scan are shown in the Explain report. Because the subparameter INDEX has been set to T, all indexes created for the accessed table are shown in detail.

All is the default value.

ELOC

You can specify the ELOC subparameter for the PACKAGE keyword to determine the location where the package will be explained.

If ELOC is not specified, the name of the last specified location server is used. If ELOC had not been previously specified, the home server location is used.

When a valid server location name is specified in the ELOC subparameter, Explain connects the application process to the specified application server. The location name must appear in the LOCATIONS column of the SYSIBM.LOCATIONS table, or be the name of the home server, that is, the local Db2 subsystem.

The following example shows the format and the use of this subparameter:

```
Package=Coll.Pkgnm,Eloc=CPHMVS1_DB2T
```

FIRST/LAST

You can specify the FIRST or LAST subparameters for the PACKAGE keyword to control the range of SQL statements for explanation within the plan.

If FIRST is not specified, a value of 1 is used. If LAST is not specified, a value of 999,999,999 is used. If FIRST has a higher value than LAST, both subparameters are set to FIRST.

The following examples show the format and the use of these subparameters:

```
PKG=Coll.Pkgnm,First=1234,Last=2345,Index=y
Package=Coll.Pkgnm,index=T,first=486
Package=Coll.Pkgnm,last=4000;
```

In the first example, Explain only explains SQL statements where the QUERYNO (precompiler statement number) is equal to or higher than number 1234, and less than or equal to 2345.

In the second example, only SQL statements with precompiler statement numbers greater than 485 are explained and shown in the report.

In the last example, only statements with query numbers less than or equal to 4000 are included in the report.

FORCE

You can specify the FORCE subparameter for the PACKAGE keyword to control the explaining of statements in a package.

If FORCE is not specified, Explain only explains the statements if only one package conforms to the package specification, or if the total number of SQL statements (including non-explainable) is less than 300.

The FORCE subparameter can have the following values (only the first character is validated):

No

If more than one package is referenced, the statements are only explained if the total number of SQL statements is less than 300.

Yes

When this subparameter is specified, all statements in all packages that conform to the specification are explained.

The following example shows the format and use of this subparameter:

Package=ABC*.DEF*.(*),Force=Yes

The FORCE keyword forces all statements in all versions in any package name starting with DEF in any collection ID starting with ABC to be explained.

No is the default value.

FORMAT

You can specify the FORMAT subparameter for the PACKAGE keyword to control the formatting of the SQL statements in the plan.

Yes

For each explainable SQL statement in the plan, the actual SQL statement is printed in the top part of the report. The SQL statement is formatted so that a new line is started for SQL keywords such as SELECT, INTO, FROM, and WHERE. Subselects, however, are not indented.

No

When this subparameter is specified, the explainable SQL statements are only formatted with respect to keywords like SELECT and UNION. Using this parameter, the SQL statement consumes minimum page space.

The following example shows the format and use of this subparameter:

Package=Coll.Pkgnm,index=T,format=No

The FORMAT keyword compresses the SQL statement as much as possible.

Yes is the default value.

GEN

You can specify the GEN subparameter for the PACKAGE keyword to control the number of versions (or generations) in a package to be explained.

If GEN is not specified, the number of generations is one.

The following example shows the format and use of this subparameter:

Package=MyColl.MyPackage(Version.6),Gen=2

The GEN subparameter results in explaining all SQL statements for Version 6 in addition to the previous version.

HINT

To modify an existing access path in a miniplan, you can specify an optimization hint ID. You must supply a SET CURRENT HINT statement, which updates the OPTHINT column in the PLAN_TABLE table with a specified hint name, for example, MYHINT. For more information, see <u>"Input command definitions" on page 140</u>. In addition, you must specify a unique optimization hint ID number in the statement as shown in the following example:

SET CURRENT HINT = 'MYHINT', id = 4711

Where the number, in this case 4711, is located at the bottom of the top section of the Explain report.

After you submit the Explain statement, perform the following steps:

1. Change the actual access path information. For example, change from a hybrid join to a merge scan join by issuing the following SQL statement:

```
Update Plan_Table
set METHOD = 2
where METHOD = 4
and OPTHINT = 'MYHINT'
```

2. Activate the hint by issuing the following SQL statement:

```
SET CURRENT OPTIMIZATION HINT = 'MYHINT';
```

3. Explain the SQL statement to see if the specified hint takes effect correctly.

By not including the three specified activities in Explain directly, you avoid changing an access path. Changing an access path can have a significant influence on the performance of a miniplan. An access path change must only be applied by SQL programmers who understand the full impact of access paths on the performance of your systems.

HOSTVAR

You can specify the HOSTVAR subparameter for the PACKAGE keyword to control the listing of host variable specifications if the application was precompiled under Db2 Version 2.3 or later.

If HOSTVAR is not specified, Explain does not list any host variable specifications.

The HOSTVAR subparameter can have the following values (only the first character is validated):

No

No host variable definitions are listed in the Explain report.

Yes

When this subparameter is specified, a listing of all (maximum 100) host variables used in the SQL statement are produced. This listing includes the definition type and length.

The following example shows the format and use of this subparameter:

Package=Coll.Pkgnm,Index=Yes,Hostvar=Yes

The HOSTVAR keyword results in a listing of all host variables and their application definitions. The definitions must be comparable to the corresponding column definitions, or the Optimizer might not choose the optimum access path. The host variable report indicates the release number of the Db2 subsystem that was used for precompiling.

No is the default value.

INDEX

You can specify the INDEX subparameter for the PACKAGE, QMF, and QNO keywords to control the level of index information. The valid values for the INDEX subparameter are:



Yes

When Yes is specified and if the Optimizer has selected a matching or non-matching index scan, Explain shows all index information for the selected index including key column information. If the Optimizer has selected a table space scan, Explain shows any available indexes for the accessed table.

Yes is used when CONDENSE or ALLindex is specified.

No

When No is specified, no index information is shown in the Explain report.

Tablespace

When table space is specified, detailed index information for all indexes of the accessed table is shown in the Explain report, if the chosen access path is table space scan. If the access path is an index scan, only information for the chosen index is shown in the report.

Condense

When Condense is specified, information for all indexes used by all tables in the plan or package is shown in one report at the end of the plan or package information. No index information is shown in the individual miniplan reports.

ALLindex

When ALLindex is specified, information for all indexes created for the table is shown in the Explain report. The information is listed after the table details.

The following examples show the format and use of this subparameter:

```
PKG=Coll.Pkgnm,index=N0
QMF=Agent007.Secret query,Index=t
```

Yes is the default value.

LEVEL

You can specify the LEVEL subparameter for the PACKAGE keyword to control the level of information within the current plan being explained. This subparameter has the same values as the LVL command, except XON and XOFF (only the first three characters are validated). Refer to the LEVEL command definition in "Input command definitions" on page 140.

Regardless of the current value set by the LVL statement, the LEVEL subparameter can specify the information level in effect for the plan being explained. When the next v statement is processed, the information level specified by the LVL statement, is the current level.

The following examples show the format and use of this subparameter:

```
Level=Detail
PKG=Coll.Pkgnm,level=NoKeys
Package=Coll.Pkgnm=SalPgm,first=486,acctype=ts,level=summary
```

The global level is set to Detail. In the first example, the Coll.Pkgnm package is explained, but the report does not show key information because the LEVEL subparameter is set to NOKeys.

In the second example the global level is still Detail, but because the LEVEL subparameter is set to SUMMARY, the statements for this package are only shown on the summary page. Only statements with table space scan as the access path and with query numbers greater than 485 are shown.

OWNER

You can specify the OWNER subparameter for the QNO keyword to control the authorization ID of the PLAN_TABLE for the current request.

The following examples show the format and the use of this subparameter:

QNO=1234,Owner=OLFERT,Index=y
QueryNo=9,index=T,owner=USER

In the first example, Explain selects the explain data for query number 1234 in the PLAN_TABLE owned by authorization ID OLFERT. You must have SELECT authorization to the table OLFERT.PLAN_TABLE to perform this function.

In the second example, OWNER is set to USER, which is a special value that is substituted with the authorization ID of the Explain job submitter. This is also true in situations where the PARMS field in the RUN statement has been specified.

PLOC

You can specify the PLOC subparameter for the PACKAGE keyword to determine the location where the package is located.

If PLOC is not specified, the name of the last specified location server is used. If PLOC had not been previously specified, the home server location is used.

When a valid server location name is specified in the PLOC subparameter, Explain connects the application process to the specified application server.

The location name must appear in the LOCATIONS column of the SYSIBM.LOCATIONS table, or be the name of the home server, that is, the local DB2 subsystem.

The following example shows the format and the use of this subparameter:

```
Package=COllid.Pkgnm,Ploc=CPHMVS1_DB2T
```

TABLE

You can specify the TABLE subparameter for the PACKAGE keyword to control the statements for explanation within the package.

If TABLE is not specified, all statements within the package are explained.

If you want only statements accessing a given table explained, the actual table name must be specified.

If you want all statements accessing tables with a given name pattern explained, type a name string that ends in a wildcard. You can end the name string with an asterisk (*) or a percent sign (%).

The following examples show the format and use of the TABLE subparameter:

Package=Coll.Pkgnm,table=table56 Package=Coll.Pkgnm,table=sysibm.systab%,last=1234 Package=Coll.Pkgnm,table=USER.table12,level=NoKeys

In the first example, Explain only explains the statements accessing table *userid*.TABLE56, where *userid* is the SQLID of the current user. All other statements in the plan are not explained.

In the second example, Explain explains all statements with statement numbers less than or equal to 1234, where table creator is SYSIBM and table name starts with SYSTAB, like SYSTABLES and SYSTABLESPACE.

In the third example, Explain only explains the statements accessing table TABLE12, where the creator is current SQLID.

TOPA

You can specify the TOPA subparameter for the PACKAGE keyword to control if all the SQL statements of the package are extracted and later sent to SQL PA.

If TOPA is not specified, Explain does not extract all SQL statements.

The TOPA subparameter can have the following values (only the first character is validated):

Yes

All explainable SQL statements that cost more than 10,000 Service Units (SU) to run, or result in a union of multiple indexes, or result in a star join are extracted and later sent to SQL PA for further investigation.

All

When this subparameter is specified, all SQL statements of the plan or package are extracted and later sent to SQL PA.

No

When this subparameter is specified, no SQL statements are extracted.

The following example shows the format and use of this subparameter:

Package=Coll.Pkgnm,Index=Yes,ToPA=All

All SQL statements are extracted from Package Coll.Pkgnm.

Yes is the default value.

Specifying language and explaining data from another user

You can use the PARMS field to specify whether you want messages displayed in English or not. You can also use the PARMS field to extract EXPLAIN information from a PLAN_TABLE that belongs to another user.

About this task

When specifying the PARMS field that is part of a RUN statement, you can use the following procedure to determine the proper setting.

Procedure

 If an NLS file was allocated to Explain through the DD card TEXT, Explain uses these NLS messages for access path and summary page messages. To overwrite this default, you can specify a parameter (NLS=NO) in the PARMS field of the program execution. For example:

//EEERUN EXEC PGM=ANLEEE,PARMS('/NLS=NO')

The result is that all messages are written in English.

2. If you want to extract EXPLAIN information from a PLAN_TABLE belonging to another user, you can specify the authorization ID of the other user in the PARMS field of the program execution. For example:

//EEERUN EXEC PGM=ANLEEE,PARMS('/USER6')

You must have SELECT authorization to USER6.PLAN_TABLE for this statement to work.

If the specified authorization ID in the PARMS field is different from the authorization ID of the job submitter, all QNO requests are selected from the PLAN_TABLE owned by the user ID specified in the PARMS field, unless the OWNER= subparameter has been specified.

Example

You can specify the two parameters in any order as none, one, or two parameters, as shown in the following examples:

//EEERUN EXEC PGM=ANLEEE,PARMS('/NLS=YES,USER6')
//EEERUN EXEC PGM=ANLEEE,PARMS('/user5,nls=no')
//EEERUN EXEC PGM=ANLEEE,PARMS('/Nls=Yes')
//EEERUN EXEC PGM=ANLEEE

Note: The first character in the PARMS field must be a forward slash (/). The forward slash is a PL/I requirement.

Selected authorization identification

You can specify authorization ID that Explain uses to select the correct PLAN_TABLE.

To select the correct PLAN_TABLE, Explain uses the authorization ID specified as the OWNER=*subparameter* with the QNO keyword as first choice.

If no OWNER=subparameter is specified, Explain uses the authorization ID specified in the PARMS field.

If the OWNER=*subparameter* or the PARMS field were not specified, Explain uses the authorization ID of the job submitter.

The following table shows the selected authorization ID of the PLAN_TABLE for the four main processes:

Table 16. PLAN_TABLE authorization IDs		
Process (=keyword)	Authid of PLAN_TABLE	
SQL	Job submitter	
QMF	Job submitter	
РКС	Package binder	
QNO	1. Owner parameter	
QNO	2. PARMS field	
QNO	3. Job submitter	

Tip: If the authorization exit has specified a secondary authorization ID (current SQLID) or you have issued an SQLID command with a secondary authorization ID, this ID is used instead of the job submitter ID.

User authorization table

The following information describes the columns in the user authorization table.

The user authorization table has the following columns with the given contents:

AUTHID

The authorization ID of the user. If the authorization ID is specified as blanks, this row acts as default values for authorization IDs not found in the authorization table.

NLS

Use of globalization (NLS) data set for writing Explain messages in the national language, if the ddname TEXT points to a data set with messages written in NLS. The default value is N.

PLANINFO

Creation of plan information rows in the plan history and DBRM history tables. If the authorization is given, the Explain plan report contains plan information for the last four plan generations.

The default value is a Y.

FORMATSQL

Formatting of SQL statements, when explaining plans and packages. If this authorization is given, the SQL statement starts a new line for clauses such as FROM, WHERE, AND, and GROUP BY. The default value is Y.

REMOTEXPL

Explanation of remotely bound packages in a distributed environment. If authorization is given, Explain shows explain information for packages found at the server location. The default value is Y.

Insert a row with specific authorizations for a specific user (authorization ID). Each row is time stamped resulting in the use of the most recent authorization row. If more than one row with a given user ID exists, the row with the most recent timestamp is used.

If Explain is not able to find a row with your authorization ID, a default row with a blank user ID is used. If more than one row with a blank user ID exists, the row with the most recent timestamp is used.

The following example shows how to insert an authorization row for user AB12345:

insert	into eeeauth (au value	thid, nls, s ('AB12345',	planinfo, 'N', 'Y',	formatsql, 'Y',	remotexpl) 'N');
	default value	s (''',	'N', 'Y',	'Υ',	'Y');

User defaults data set

Each Explain user can specify their own set of run time default values.

For example, the system default value for ACCTYPE on the plan and package statement is All, but you might want the default value to be Nonmatch. Changing the default value is possible if you specify your preferences in a defaults data set. Specified values are saved in the ANLPARM user parameter file.

The system default values are specified in <u>Chapter 5</u>, "Plan Table report input specifications," on page 133. The specification rules given in that chapter also apply for the user default values specifications.

The following table shows the parameters that you can set to user-defined values.

Table 17. User-defined parameters			
Parameter	Description	Example	System default
ACCTYPE	Show this access type only	Matching	All
DSN8EXP	Use Db2 supplied stored procedure	Yes	No
INDEX	Index information	Tabscan	Yes
HOSTVAR	Show host variables	Yes	No
FORMAT	Formatting of the SQL statement	Yes	Yes
LEVEL	Level of report information	KEYdist	Detail
PLANINF	Level of plan information	Mini	Yes

The default values are specified in a data set for batch jobs, as shown below:

//... //REPORT DD SYSOUT=* //ANLPARM DD DSN=hlq.SANLPARM(ssidPARM),DISP=SHR //DSNTRACE DD SYSOUT=* //...

Alternatively, the user default values may be specified inline in the job JCL as shown in the following example:

//... //REPORT DD SYSOUT=* //ANLPARM DD * ACCTYPE TABSCAN LEVEL ALLINDEX //DSNTRACE DD SYSOUT=* //...

Transferring complex queries to SQL PA

Explain is able to output complex SQL queries to a data set that is used as input to SQL PA.

Procedure

Specify the subparameter TOPA=all to have all SQL statements in a package written to the data set destined for SQL PA.

If you specify TOPA=No, the SQL statement is not written. If you specify the default value of TOPA=Yes, a complex query is written to a data set specified in the DD card named //SQLPA.

The format of the DD card pointing to a fixed blocked 80 bytes data set, could be as shown in the following example:

```
//...
//REPORT DD SYSOUT=*
//SQLPA DD DSN=userid.ANLEEE.SQL,DISP=(,KEEP),
// DCB=(RECF=FB,LRECL=80,BLKSIZE=3120),VOL=SER=...
//DSNTRACE DD SYSOUT=*
//...
```

Results

The output data set with one or more complex queries separated with semicolons are sent directly to SQL PA.

When SQL statements are written to the sequential data set, a number of selected column values from the PLAN_TABLE and DSN_STATEMNT_TABLE tables are copied to a user table named *authid*.EEEPATH. If these tables do not exist, Explain creates them.

Chapter 6. Reporting capabilities in SQL PA

Db2 SQL Performance Analyzer (SQL PA) produces three expanding levels of detail in its performance report set, from the basic Cost Summary, to the Enhanced Explain and, finally, the Detail Trace report.

Each successive report is more detailed than its predecessor, including all the previous information, plus new forecast data for the SQL statements under study. There is also a fourth summary report, SQL PA Limits, which provides a quick eye-catcher for warnings and high costs. These topics examine the contents of each level of reporting, and explain the variables that they contain, and how to relate them to Db2.

Topics:

- "Db2 SQL Performance Analyzer cost summary report" on page 153
- "Db2 SQL Performance Analyzer enhanced explain report" on page 156
- "Db2 SQL Performance Analyzer detail trace report" on page 177
- "QLIMIT (installation limits) report" on page 191
- "Predicate Analysis information" on page 193
- "Host Variables information" on page 195
- "The Plan Table report" on page 197
- "The CSV report" on page 211

Db2 SQL Performance Analyzer cost summary report

The SQL PA cost summary report provides a quick summary of costs.

Each SQL PA operation produces a minimal set of results, which contain costing information about each query examined by SQL PA. In Batch mode, these results are printed to SYSPRINT and appear on your output listing. In TSO mode, the results are written to the standard file *userid*.ANLCOST.LOG, always allocated by the ANLCANLI CLIST during processing. Under QMF, the interceptor program returns the same cost summary report format to the user, for all queries or only the queries that exceed the resource thresholds of the host system. The Db2 SQL Performance Analyzer stored procedure returns the same information in its output parameters.

The Db2 SQL Performance Analyzer cost summary report contains three cost descriptions to accommodate your needs:

- 1. The cost in resource consumption terms (CPU time, I/O, elapsed time)
- 2. The cost in QUNITS (Query Units)
- 3. The cost in monetary terms

Technical users might want CPU time (TCB and SRB) and I/O counts, to fine-tune queries by costs. Other users might prefer a single QUNITS number that sums up the total cost of each query. Or you might be interested in budgets, and prefer to see the costing in monetary terms. All three forms are always available through Db2 SQL Performance Analyzer, so the choice and preference is up to you.

A sample of the Cost Summary Report is shown in the following figure.

Figure 52. SQL PA cost summary report

Topics:

- "Resource consumption estimates" on page 154
- "QUNITS values " on page 155
- "Charge back cost estimates" on page 155
- "Warnings and notifications" on page 155

Resource consumption estimates

The following information describes the resource consumption estimates as shown in the SQL PA cost summary report.

The first line of the report contains the QUERY number and the elapsed time. Db2 SQL Performance Analyzer assigns the QUERY number in the 100,000,000 range. SECONDS OF ELAPSED TIME indicates the Db2 SQL Performance Analyzer estimate for how long the query runs before returning data. Elapsed time, also known as response time, can be critical for online systems. Elapsed time is a computed value for INSERT, UPDATE, and DELETE statements based on the following major contributors:

- CPU processing to be done by application and Db2 address spaces
- The number of I/O requests and their type (prefetch, synchronous, and asynchronous)
- The Db2 SQL Performance Analyzer calculated estimate of I/O length (based on the disk drive speed)
- The processor usage of the attach facility, such as IMS, SPUFI, CAF, or CICS
- Known wait times for data set opens and closes, lock waits, and dynamic binds
- Any other time increments that apply to your SQL query according to Db2 SQL Performance Analyzer

Db2 SQL Performance Analyzer also includes the writing of the pages back out to disk, and logging, processes often accomplished after the query terminates. The Db2 SQL Performance Analyzer elapsed time estimate does not include network time, LAN or VTAM[®], which varies depending upon the baud rate, terminal type, modem specifications, and other network characteristics. Elapsed time is presented in seconds.

The second line of the report contains CPU time as a calculated value. The value is the CPU time necessary to perform all the processing for the query, from the time it is initiated, until it terminates. This processing estimate includes, at a minimum, the CPU consumed by the attach facility (unless NONE is selected for the CONNECT parameter), all the I/O processing, including driving the sequential and list prefetches, synchronous reads, asynchronous writes, logging (if any), the stage 1 and 2 predicate processing by Db2, fetching of the rows and columns, locking, RID list manipulation, sorting, get page calls, binding, plus many other facets of the process. CPU time is based upon privately benchmarked path lengths (instruction counts). The path lengths are derived for the many internal processes of Db2, which are then converted into CPU time using the processing power of the target host system.

The third line of the report contains the I/O Estimate. Only the physical I/O counts are included here, not the logical I/O. The physical I/O requirements can vary, depending on which pages are found in the buffer pool, and which must be physically read in from disk. You can control the percentage of pages found in the buffer pool using the BUFFHIT parameter.

To understand physical and logical I/O, assume that you have a query that scans 64 pages and returns 25 rows every time you run it. Unless you change the data in the table, or your selection criteria, it always scans 64 pages, and retrieve 25 rows. The 64 pages must be read physically from disk, and brought into the buffer pool, where Db2 can logically scan them (using getpage). Those 64 pages are logical I/O requests. Db2 must scan all those pages to produce your 25 rows. Now, assume that your table is being read by other users. Those other users might have left some pages in the buffer pool the last time they processed. You might no longer need to bring 64 pages into the buffer pool to satisfy your query. Maybe this time you only need 32, or next time, 12, or 40. The logical I/O requirements of your query have not changed (it is still 64 pages), but the physical I/O requirements can vary, depending on which pages are found in the buffer pool, and which must be physically read in from disk.

Db2 counts I/O in a rather peculiar way. A synchronous read I/O is a request for a single page, and it counts as one I/O. A sequential or list prefetch I/O might be for 32 pages, or 16, but it also counts as one

I/O. An I/O is defined by the MVS Start I/O instruction (SIO), and it could be issued for a single 4 K page, or for many pages, chained together as a single request. The Db2 SQL Performance Analyzer I/O estimates (and estimates of Db2) do not reflect the number of pages processed. The I/O estimates reflect the number of Start I/O commands issued for physical disk pages. Db2 SQL Performance Analyzer I/O estimates include both reads and writes for tables and indexes, and work files and the Db2 log. For page counts, see the getpage statistic.

QUNITS values

The Query Units, or QUNITS, is a specially derived cost factor developed for users who were not familiar with resource consumption variables. The QUNITS value provides a quick assessment of the relative cost of query statements.

The QUNITS represent the raw processing requirements of each query, influenced by its I/O requirements and its instruction execution. Processing is often the most expensive resource in the system.

QUNITS are based on CPU processing and the CPU time associated with all I/O processing. The higher the QUNITS value, the more processing is required. Lower values mean quicker queries, and less resource consumption.

When comparing two queries, one query might be more CPU intensive, the other more I/O bound, while elapsed time can remain relatively constant. Using the QUNITS value can simplify the choice between which query to use.

As Db2 moves closer to a memory resident database system, the better it is to base your query optimization choices on the QUNITS value.

Charge back cost estimates

Db2 SQL Performance Analyzer parameter values that describe the cost of various system resources for charge back purposes can be defined.

Parameters are used to put a price on CPU time (in hours), elapsed time (terminal connect, also in hours) and I/O counts (charged per 1000). If the parameters are defined, your Db2 SQL Performance Analyzer cost estimate includes a monetary value. Db2 SQL Performance Analyzer can compute the financial cost of any query, based on the resources consumed and the assigned cost of using those resources.

A cost estimate for each query, in the national currency, can quickly build charge back cost awareness. Running a query that costs \$1.22 versus a query costing \$822.00 is an easy decision.

Warnings and notifications

The Db2 SQL Performance Analyzer cost summary report is the warning system. The warnings and notifications as they are shown on the cost summary report alert you when your query exceeds one or more of the limits set for normal resource use.

The installation team chooses and sets high water marks for CPU, I/O, elapsed time, overall monetary costs, and QUNITS. When one of these values is exceeded by the SQL PA cost estimate, Db2 SQL Performance Analyzer issues a warning message. The messages are shown in the following figure.

```
ANL5025W *** WARNING:
ESTIMATE OF
             10.88977 EXCEEDS "CPU TIME" LIMIT OF
                                                       10 CPU SECONDS!
ANL5028W *** WARNING:
                  4105 EXCEEDS "I/O CALL" LIMIT OF
                                                       1000 I/O CALLS!
ESTIMATE OF
ANL5027W *** WARNING:
                822.70 EXCEEDS "MONETARY" LIMIT OF
ESTIMATE OF
                                                       150.00 DOLLARS!
ANL5029W *** WARNING:
ESTIMATE OF
             205.2498 EXCEEDS "ELAPSED TIME" LIMIT OF
                                                          120 SECONDS!
ANL5026W *** WARNING:
ESTIMATE OF
                  1080 EXCEEDS "SERVICE UNIT" LIMIT OF
                                                           500 QUNITS!
```

Figure 53. SQL PA cost summary report warning messages

These messages call your attention to the query, which you can modify to conform to acceptable standards. Set the limits high enough so that they are not easily attained by the average query. Keep values high enough that exception notifications are rarely issued.

Db2 SQL Performance Analyzer enhanced explain report

For the enhanced explain report, Db2 SQL Performance Analyzer takes the output from Db2 Explain in the plan table and expands that information with the sizing statistics obtained from the catalog for the Db2 objects used by the query.

In this report, Db2 SQL Performance Analyzer introduces a more sentence-like structure to the Explain data, saving you the effort of trying to translate the plan table variables.

Db2 SQL Performance Analyzer reports the activities by query number, query block number, plan number, and multiple index operational sequence number, just as they are represented in the plan table by Db2.

The Db2 SQL Performance Analyzer Enhanced Explain Report is written to the file specified by the ANLREP DD card for batch, or the file named by the user as the Explain Report on the Db2 SQL Performance Analyzer Define Report Data Sets Panel under TSO.

This report is not available under the QMF intercept program because it would be too time-consuming to produce. However, long-running QMF queries can always be reexamined under Db2 SQL Performance Analyzer in either TSO or batch mode.

The enhanced explain report includes a recap of the parameters currently in effect, unless DSPPARM is set to NO. These parameter values describe the configuration and assumptions of this Db2 SQL Performance Analyzer run, and are critical factors in the Db2 SQL Performance Analyzer cost assessment and performance forecast.

An example of the parameter portion of the report is shown in Figure 54 on page 158.

5:23:50.338 06-25-2013	3 SQL Per: Enhanced	formance Explain	Analyzer Version 5.1.0 Report Level 51-5100
APAR PI1234	15		
SOL PA Para	ameters		
(ANI PARM)			
VERSTON	V11P1		
DECDEES	1		
DUCCUTT	- -		
	+0FF+		
USEPLAN	+0FF+		
QUALIFY	+0+++		
CONNECT	CAF		
REFRESH	NO		
VIADRDA	+0FF+		
PKGDRDA	+0FF+		
RETCODE	NO		
DELIMIT	OUOTE		
DSN8EXP	ŇŌ		
NI SCODE	+0FF+		
KEEPI AN	YES		
PROCESS	+055+		
NIIMPEDC	VEC		
DEDODTE	DET		
CUOWALT			
SHUWALI	NU		
PRECISE	ALL		
ADVISOR	ALL		
OBJECTS	YES		
QLIMSQL	YES		
NOSTATS	YES		
DSPVARS	YES		
DSPPARM	YES		
JOINTAB	10		
INLISTS	10		
TXUPDAT	5		
TSCANPG	50000		
TSCANPG	1000		
	5000		
MATCOLS	0 50		
DEDECT	0.50		
	0.15		
STURAGE	3390-3		
LEVEL	TAB		
INDEX	NO		
ACCTYPE	ALL		
HOSTVAR	YES		
FORMAT	YES		
PLANINF	YES		
DSPWCOC	YES		
CSVRPT	YES		
CSVSEL	TAB0		
CSVHDRW	COL		
CSVDI TM	COMMA		
CSVDCSP	DOT		

(ANLCNTL)	
SUBVERS	V11NFM
SUBSYST	DSNB
BUFFERS	2000
BUFF08K	500
BUFF16K	250
BUFF32K	100
DYNAMIC	NO
AUTHIDO	ANLUSR51
DBTRACE	OFF
SETPLAN	YES
USRPARM	YES
FASTTBL	YES
DSGROUP	0
CPUTIME	10
COSTING	15
ELAPSED	60
COSTQUN	200
IOCALLS	1000
CPUCOST	500.00
IOSCOST	10.00
TIMCOST	10.00
MONEYIS	DOLLARS
CURRSYM	\$
(ANLPARM)	
DEGREES	1
DBRMKEY	+0FF+
EXPLAIN	NEW
EEECALL	SPA
QUALIFY	
USEPLAN	JSMITH
QUERYNO	ALL
STOPONO	999999999
VERSION	N/A WIII be discovered by code
(ANLUNIL)	N/A will be discovered by esta
SUBVERS	N/A will be discovered by code
SUBSYST	N/A was passed in a program parameter

Figure 54. Parameter recap for enhanced explain report

Topics:

- "Table space scans" on page 158
- "Cluster matching index scans" on page 160
- "Cluster nonmatching index scans" on page 162
- "Random matching index scans" on page 163
- "Random nonmatching index scans" on page 164
- "Multiple index scans" on page 166
- "Joining tables" on page 168
- "Subquery processes" on page 170
- "Sorting results" on page 173
- "Insert, Update, Delete, Select, and Refresh" on page 174

Table space scans

The table space scan information includes the type of table space (simple, segmented, or partitioned), the number of tables in the space, and the fully qualified table name. The size of he table in pages, its record length, number of records, and columns are also provided. Other pertinent information, such as lock size and table space lock mode, close option, segment size, and percent compressed, completes the description of the table space.

This descriptive data is presented for each table accessed, detailing the size, scope, and contents of the tables being accessed by the query. The Db2 SQL Performance Analyzer Cost Summary Report is presented along with the Explain information so all the information is found in a single place. This run

contains the parameter SHOWALT YES, so alternative indexes available to access this table are also presented.

When the optimizer for Db2 chooses a table space scan as the access path, it is ignoring the use of indexes, if any exist on the table, as a means of accessing the data. There are many reasons why you cannot choose an index, and the optimizer has made a valid choice in access path selection.

Table space scans are valuable when columns do not have predicates that participate in these indexes. There is no benefit in using the indexes because they would not eliminate any rows from the search.

Occasionally, the indexed columns might be involved with predicates that are stage 2, or contain subtle incompatibilities, such as being compared to columns or data of the wrong length or type.

Db2 SQL Performance Analyzer explains the table space scan in the report segment shown in the following example. Notice that the query states "FROM L1000," but SQL PA adds the high-level qualifier TDT690 to the table name.

EXPLAIN PLAN SET QUERYNO = 100000001 FOR SELECT * FROM L1000 QUERYNO: 100000001 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> ANL7003I *** GUIDELINE: |Close Yes was specified for the Tablespace and/or the index.. if these are little used this is OK. If high volume access then consider Close No. Extremely relevant pages can be "page fixed" in memory by highly referencing, using Hiperpools (Castout Y/N), putting into a dedicated buffer pool large enough to hold all pages, deploying data sharing with Group Buffer Pool Cache All option, etc. each with associated costs. Close No also increases chances that DBD will remain in EDM Pool for next execution. ANL7006I *** This statement contains a select of all columns in the table. |Typically a select of all columns results in significantly increased processing time to fetch and process each column of each |row. Use select all only when you want to select all columns in a view definition. Improve performance by specifying the name of leach column for the SQL SELECT statement. ANL5008W *** WARNING: This SQL statement contains no predicates and uses no Built in |Functions, so DB2 is selecting the Tablespace Scan access method. |In general, this means poor performance, unless the table is very small: verify the accuracy of catalog statistics (NPAGES, NACTIVE, (CARD), and re-evaluate whether this access path is appropriate. TABLESPACE SCAN CREATOR: TDT690 TABNAME: L1000 TABLESPACE SCAN WAS CHOSEN ON A 4K SEGMENTED TSPACE WITH: 3 TABLES 31680 4K PAGES 0 TABLE CONTAINS A TOTAL OF: VERSION: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES TABLE ROWS: DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N SEQUENTIAL PREFETCH WILL BE EMPLOYED TO ACCESS THIS TABLE

 I

VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLURATIO: 11.6627 FULLKEY CARD:5954 FIRSTKEY CARD:5954TYPE: 2 NLEAF PAGES:155 NLEVELS:2 UNIQUE: D KEY COLS:1 KEY ORDER COLCARD COLUMN NAME ------5954 RIKEY1 1 A ALTERNATIVE INDEX CREATOR: TDT690 IX NAME: L1000R2N VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLURATIO: 70.0010 FULLKEY CARD: 100000 FIRSTKEY CARD: 100000 TYPE: 2 NLEAF PAGES: 359 NLEVELS: 3 UNIQUE: U KEY COLS: 1 KEY ORDER COLCARD COLUMN NAME 1 A 100000 RIKEY2 * QUERING WHICH 26.08726 SECONDS OF CPU TIME WILL BE CONSUMED AND * * A TOTAL OF 6967 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * * QUNITS 199 ESTIMATED PROCESSING COST \$ 14.0652 DOLLARS * * QUNITS ANL5025W *** WARNING: ESTIMATE OF 26.087 EXCEEDS "CPU TIME" LIMIT OF 10 CPU SECONDS! ANL5028W *** WARNING: ESTIMATE OF 6967 EXCEEDS "DISK I/O" LIMIT OF 1000 I/O CALLS! ANL5029W *** WARNING: ESTIMATE OF 95.281 EXCEEDS "ELAPSED TIME" LIMIT OF 60 SECONDS!

Cluster matching index scans

The cluster matching index scan is one of the preferred access paths chosen by the optimizer.

The optimizer uses a clustered index because the data and index are in the same general order at least 80% of the time. Db2 matches one or more columns against that index, qualifying the predicates at the index level. This matching often results in a RID (row identification) list that, when applied against the data table, is shorter and contains fewer pages to be read because the rows follow the same order as the index.

For example, if the predicate being evaluated was WHERE COL1 BETWEEN 10 AND 100 and a clustered index existed on COL1. Most of the rows that contained COL1 values between 10 - 100 would be contiguously placed in the data table, following each other in order, such as 10, 11, 12, 20, 21, 30... 100. If 5 rows fit per page, and 100 rows were returned, then only 20 pages would need to be read to satisfy the query.

Suppose, however, that the index was not clustered. Then there would be no guarantee that the rows would follow each other, or even be on the same page. In fact, it might take up to 100 pages to read in the 100 rows, if the distribution were random.

A cluster matching index scan reads a portion of the index leaf tree, and a portion of the data table pages to satisfy the query. It is one of the best alternatives for table access. If a significant number of data pages are read, sequential prefetch is wanted to retrieve them, and the same is true for index leaf pages.

Detailed information is provided about the clustered index wanted, including its cluster ratio, type, and cardinality. The number of columns which are matched by Db2 for predicates on that index is also included. These are always Stage 1 predicates. The index unique rule is presented as DUPS_OK, UNIQUE or PRIMARY (which also means unique), and CONST-U (unique constraint), NOTNULL (for unique where not null), and PARENT for the non-primary parent keys.

The size of the index, from the number of levels to the number of leaf pages is described, along with the index close option, first and full key cardinality (the number of unique values in the first column of the index, and in all the columns of index), along with other pertinent index descriptive data, including the

index key composition, column cardinality and nonuniform distributions. SHOWALT YES brings the additional information about the alternative indexes that the optimizer could have selected, but did not.

The table on which the index is operating is also described, providing information like the table space scan information. Special notes, such as the IN predicate being used as the matching predicate, provide you with further information about the current access. If the clustered index also has the CLUSTERING flag set (only one does), it is marked as the Insert controlling index, containing the order in which all new rows must be inserted.

The enhanced explain report information for cluster index matching scans is shown in the following example:

EXPLAIN PLAN SET QUERYNO = 100000002 FOR SELECT CIKEY, C2, NIKEY, C4, RIKEY1, C6 FROM TDT690.L1000 WHERE CIKEY IN (50087, 50088, 50089, 50090, 70000) QUERYNO: 100000002 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> THIS INDEXED ACCESS USES THE "IN (LIST)" PREDICATE, WHICH IS NOT A CANDIDATE FOR MULTIPLE INDEX ACCESS. ANL7002I *** GUIDELINE: |This plan step has not selected any Sequential|List Prefetch I/O. |If the SQL processes just a few rows that is OK, but if many rows are involved, you can help promote Sequential Detection by both accessing the data in sequential order (presort?) and by binding |with Release (Deallocate) to avoid resetting counters at Commit. ANL7003I *** GUIDELINE: Close Yes was specified for the Tablespace and/or the index... if these are little used this is OK. If high volume access then |consider Close No. Extremely relevant pages can be "page fixed" in memory by highly referencing, using Hiperpools (Castout Y/N), putting into a dedicated buffer pool large enough to hold all pages, deploying data sharing with Group Buffer Pool Cache All |option, etc. each with associated costs. Close No also increases |chances that DBD will remain in EDM Pool for next execution. CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLURATIO: 99.9995 VERS: TYPE: 2 NLEAF PAGES: 412 NLEVELS 2 10 99340 412 NLEVELS: 3 UNIQUE: D DUPLICATE OK 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 1 OF COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# KEY 1 CIKEY A DECIMAL N 9 Y 99340 10 ALTERNATIVE INDEX CREATOR: TDT690 IX NAME: L1000R1N VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 11.6627 FULLKEY CARD: 5954 FIRSTKEY CARD: 5954 5954 FIRSTKEY CARD:
 FULLKEY CARD:
 5954 FIRSTREY CARD:
 5954

 TYPE: 2 NLEAF PAGES:
 155 NLEVELS:
 2 UNIQUE: D KEY COLS:
 KEY ORDER COLCARD COLUMN NAME 1 A 5954 RTKEY1 ALTERNATIVE INDEX

CREATOR: TDT690 IX NAME: L1000R2N VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 70.0010 FULLKEY CARD: 100000 FIRSTKEY CARD: 100000 359 NLEVELS: 3 UNIQUE: U KEY COLS: 1 TYPE: 2 NLEAF PAGES: KEY ORDER COLCARD COLUMN NAME -----1 A 100000 RIKEY2 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE ANL6020I *** NOTE: |This index is presently designated as allowing "duplicate values" Make sure this is a nonunique index that does not fall into one of the following "unique" categories: explicitly unique, primary key, |non-primary RI parent key, unique where not null, unique column |constraint. Unique indexes have definite advantages whenever they can be declared by the user, so be sure it contains duplicates. INDEXED ACCESS CREATOR: TDT690 TABNAME: L1000 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES S: 100000 COLUMNS: 50 REC LENGTH: 1008 BYTES 0 GB NUMBER OF MOTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES VERSION: TABLE ROWS: DSSIZE: TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N ANL7015I *** GUIDELINE: |Collecting Non-uniform column statistics greatly enhance the DB2 Optimizer accuracy when processing Equals and In (List) preds. * DURING WHICH 0.01803 SECONDS OF CPU TIME WILL BE CONSUMED AND * * A TOTAL OF 6 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * * A TOTAL OF 6 PHYSICAL 1/0 REGELET OUNTES 1 ESTIMATED PROCESSING COST \$ 0.0695 DOLLARS *

Cluster nonmatching index scans

The cluster nonmatching index scan also uses a clustered index, but unlike the cluster matching index scan, the columns are not matched against its specific values.

Rather, the entire index leaf tree structure is read, and the predicate is applied against the column values to arrive at a conclusion. For example, if the predicate were WHERE COL1 NOT LIKE 'ABC', all the values of COL1 must be processed from the index to determine which ones are not like 'ABC'. The nonmatching index scan is used when columns are involved in a summarization built-in function, such as AVG or SUM. The nonmatching index scan is used in a stage 2 predicate process, as shown in the following example.

Notice that in this example no columns are matched against the index. Also note that in this query, the column involved in the Stage 2 arithmetic expression predicate is found in the index, and no others are selected. Only the index is accessed, as noted by SQL PA. Cluster nonmatching index scans make heavy use of sequential prefetch for both index and data pages.

This example also provides a look at the evaluation of the query by SQL Advisor, with informative notes for the user.

```
EXPLAIN PLAN SET QUERYNO = 100000003 FOR
SELECT CIKEY FROM L1000
WHERE CIKEY - 200.50 < 2000 ;
```

QUERYNO: 100000003 QBLOCKNO: 1 PLANNO: 1 MIXSEQ: 0 PROCESS -> CLUSTER NONMATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS:0KEY LEN:6PADDED:-C-ED:YC-ING:YCLUSRATIO:99.9995FULLKEY CARD:99340FIRSTKEY CARD:9934099340TYPE:2NLEAF PAGES:412NLEVELS:3UNIQUE:DKEYCOLS:1 NONE OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 ORDER TYPE DIST LEN NULL COLCARD DIST# COLUMN NAME KEY 99340 10 1 CIKEY A DECIMAL N 9 Y ANL6017I *** NOTE: This query is currently performing a Nonmatching Index Scan, which causes it to read every index Leaf page to process the predicates. Reading the entire leaf tree is time-consuming vs. matching a part of the index. Consider better predicates that allow you to match. ANL6020I *** NOTE: This index is presently designated as allowing "duplicate values" Make sure this is a nonunique index that does not fall into one of the following "unique" categories: explicitly unique, primary key, non-primary RI parent key, unique where not null, unique column constraint. Unique indexes have definite advantages whereever they have deplaced by the user one because it contains duplicated |can be declared by the user, so be sure it contains duplicates. THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE. |ANL6025I *** NOTE: |The Composite (final) Filter Factor indicates that more than 15% of the rows will be returned: predicates not very restrictive ANL6026I *** NOTE: Arithmetic Expressions that are not exclusively using integers will be Stage 2 and not matching indexable. Move to Select list if possible, or do the math in your application later. SEQUENTIAL PREFETCH WILL BE EMPLOYED DURING THIS INDEX ONLY ACCESS

Random matching index scans

The random matching index scan is like the cluster matching index scan with two major differences.

The differences include the following characteristics:

- The rows in the data table are not clustered together in the order of the index, but rather scattered randomly throughout the pages of the table.
- The list prefetch is used to retrieve the data pages. The RIDs are sorted into the proper order of appearance (table page order), so that the list prefetch I/O is sequential, if not contiguous, in nature. An example of the information for a random matching index scan is shown in the following information.

Db2 SQL Performance Analyzer provides the same basic information for a random index as it does for a clustered index and, if there is table access, the form and structure of the table is also shown.

EXPLAIN PLAN SET QUERYNO = 100000004 FOR SELECT * FROM L1000 WHERE RIKEY2 = 38; QUERYNO: 100000004 QBLOCKNO: 1 PLANNO: 1 MIXOPSEO: 0 PROCESS -> RANDOM MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000R2N VERS:0KEYLEN:6PADDED:-C-ED:NC-ING:NCLUSRATIO:70.0010FULLKEYCARD:100000FIRSTKEYCARD:100000TYPE:2NLEAFPAGES:359NLEVELS:3UNIQUE:UKEYCOLS:1 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LENGTH NULL COLCARD DIST# -----1 RIKEY2 A DECIMAL N 9 Y 100000 0 ALTERNATIVE INDEX CREATOR: TDT690 IX NAME: L1000CIN VERS:0KEYLEN:6PADDED:-C-ED:YC-ING:YCLUSRATIO:99.9955FULLKEYCARD:99340FIRSTKEYCARD:99340TYPE:DNLEAFPAGES:412NLEVELS:3UNIQUE:DKEYCOLS:1 KEY ORDER COLCARD COLUMN NAME 99340 CIKEY 1 A ALTERNATIVE INDEX CREATOR: TDT690 IX NAME: L1000R1N VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 11.6627 FULLKEY CARD: 5954 FIRSTKEY CARD: 5954 TYPE: 2 NLEAF PAGES: 155 NLEVELS: 2 UNIQUE: D KEY COLS: 1 KEY ORDER COLCARD COLUMN NAME 1 A . 5954 RIKEY1 MATCHING IX SCAN CREATOR: TDT690 TABNAME: L1000 THIS INDEXED ACCESS IS APPLIED TO 32K SEGMENTED SPACE WITH: 3 TABLES VERSION: 1 TABLE CONTAINS A TOTAL OF: 3335 4K PAGES TABLE ROWS: 100003 COLUMNS: 50 REC LENGTH: 1008 BYTES DSSIZE: 32 GB NUMBER OF MOTS: 12 LOG: Y AVG ROW LENGTH: 804 BYTES TYPE: M LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROW: 255 BPOOL: BP32K9 ENCODE: E CCSIDS ARE SBCS: 4096 DBCS: 2048 MIXED: 7077 VOLATILE: Y

Random nonmatching index scans

The random nonmatching index scan is seldom chosen as an access path by Db2. It is not efficient to read the entire index leaf tree, and one page per row retrieved (the assumption of the optimizer).

However, certain built-in functions, left unqualified by further WHERE clause predicates, are processed best using a random nonmatching index scan.

It is not uncommon to see the COUNT all rows in the table function processed as a nonmatching index scan against the smallest index on the table that counts all the RIDs that are encountered. The random nonmatching index scan displays the complete set of index statistics and includes the table characteristics, if the query requires rows from the data table. The query in the example does not.

The following example uses the COUNT(*) clause:

EXPLAIN PLAN SET QUERYNO = 100000005 FOR SELECT COUNT (*) FROM L1000 QUERYNO: 100000005 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> ANL7003I *** GUIDELINE: Close Yes was specified for the Tablespace and/or the index... if these are little used this is OK. If high volume access then consider Close No. Extremely relevant pages can be "page fixed" in memory by highly referencing, using Hiperpools (Castout Y/N), putting into a dedicated buffer pool large enough to hold all pages, deploying data sharing with Group Buffer Pool Cache All option, etc. each with associated costs. Close No also increases chances that DBD will remain in EDM Pool for pout overwiter chances that DBD will remain in EDM Pool for next execution. RANDOM NONMATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000R1N VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 11.6627 FULLKEY CARD: 5954 FIRSTKEY CARD: 5954 TYPE: 2 NLEAF PAGES: 155 NLEVELS: 2 UNIQUE: D KEY COLS: 1 NONE OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# -----1 RIKEY1 A DECIMAL N 9 Y 5954 10 ALTERNATIVE INDEX CREATOR: TDT690 IX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD: 99340 FIRSTKEY CARD: 99340 TYPE: 2 NLEAF PAGES: 412 NLEVELS: 3 UNIQUE: D KEY COLS: 1 KEY ORDER COLCARD COLUMN NAME 1 A 99340 CIKEY ALTERNATIVE INDEX CREATOR: TDT690 IX NAME: L1000R2N 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 70.0010 VERS: FULLKEY CARD:100000FIRSTKEY CARD:100000TYPE: 2NLEAF PAGES:359NLEVELS:3UNIQUE: UKEY COLS:1 KEY ORDER COLCARD COLUMN NAME 100000 RIKEY2 1 A ANL6020I *** NOTE: This index is presently designated as allowing "duplicate values" Make sure this is a nonunique index that does not fall into one of |the following "unique" categories: explicitly unique, primary key, |non-primary RI parent key, unique where not null, unique column |constraint. Unique indexes have definite advantages whenever they |

Multiple index scans

Sometimes Db2 chooses to process a query using multiple index scans, building a composite RID list by ANDing and ORing RIDs which match each predicate, until one final RID list is created. The list is then used to access the data table pages using list prefetch.

For multiple index scans, several points are worth noting. The MIXSEQ (multiple index operational sequence) number identifies each step within a PLANNO (plan number) that is taken during the multiple index access. Each index is a matching scan, done as index only. The same index can appear several times in the sequence. If the predicates are linked by AND, the intersection of the two RID lists is used because a row satisfying the predicate clause must exist in both lists. If the predicates are joined by OR, the union of the two RID lists is used because a row satisfying one of the predicates could appear either on RID list A or RID list B. SQL PA explains each of the steps in this process, including the final data access using List Prefetch.

The following example shows a sample report for a multiple index scan. Three different indexes were used for the Multiple Index access query in the sample. In step 1 the CIN index was accessed and, in step 2, the R1N index. These are intersected in step 3 (MI) to create a composite list made up of qualifying RIDs. Then the R2N index is accessed in step 4, and its RIDs are unioned (MU) with the composite RID list in step 5. Finally, data was accessed using list prefetch. Db2 Explain identifies this action as step 0 (MIXSEQ = 0), but it is performed last, not first.

EXPLAIN PLAN SET QUERYNO = 100000006 FOR SELECT * FROM TDT690.L1000 WHERE CIKEY BETWEEN 1001 AND 1100 AND RIKEY1 < 10000 OR RIKEY2 BETWEEN 52 AND 1500 1 MIXOPSEQ: QUERYNO: 100000006 QBLOCKNO: 1 PLANNO: 0 PROCESS -> A MULTIPLE INDEX OPERATION HAS BEEN REQUESTED TO ACCESS THIS TABLE: QUERYNO: 100000006 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 1 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD: 99340 FIRSTKEY CARD: 99340 TYPE: 2 NLEAF PAGES: 412 NLEVELS: 3 UNIQUE: D DUPLICATE OK 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# KEY -----1 CIKEY A DECIMAL N 9 Y 99340 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE QUERYNO: 100000006 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 2

RANDOM MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000R1N VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 11.6627 FULLKEY CARD: 5954 FIRSTKEY CARD: 5954 TYPE: 2 NLEAF PAGES: 155 NLEVELS: 2 UNIQUE: D DUPLICATE OK 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 ORDER TYPE DIST LEN NULL COLCARD DIST# COLUMN NAME KEY A DECIMAL N 9 Y 5954 10 1 RIKEY1 THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE SEQUENTIAL PREFETCH WILL BE EMPLOYED DURING THIS INDEX ONLY ACCESS _____ STEP 3. MULTIPLE INDEX INTERSECTION WAS PERFORMED (MI). _____ QUERYNO: 100000006 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 4 PROCESS -> RANDOM MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000P2N VERS:0KEYLEN:6PADDED:NC-ED:NC-ING:NCLUSRATIO:70.0010FULLKEYCARD:100000FIRSTKEYCARD:100000TYPE:2NLEAFPAGES:359NLEVELS:3UNIQUE:UDECLAREUNIQUE: 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: BPOOL: BP10 ANL6052I *** NOTE: This index is specified as "Not Padded", allowing storage of a varying length index key. Padded indexes use blanks to fill out their fixed length keys and are not eligible for Index Only scan. |"Not Padded" indexes do not blank fill CHAR and VARCHAR columns, allowing greater flexibility and better use of storage, packing more entries into a single Leaf page. Index Only access allowed. +----KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# -----1 RIKEY2 A DECIMAL N 9 Y 100000 0 THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE _____ STEP 5. MULTIPLE INDEX UNION WAS PERFORMED (MU). _____ MULTIPLE INDEX OPERATIONS ARE NOW COMPLETE: DATA ACCESS PERFORMED. QUERYNO: 100000006 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 5 PROCESS -> * A TOTAL OF * QUITS 4 QUITS

PROCESS ->

Joining tables

Db2 SQL Performance Analyzer provides information about multi-table accesses, or joins. Information that SQL PA can ascertain about the join sequence is presented in the report. The report included information about the inner and outer table selection, and the type of join processing.

The merge scan join method is also explained by Db2 SQL Performance Analyzer.

The optimizer does much analysis before deciding the best join method, the processing order of the tables, and the inner and outer tables in the joining process. SQL PA describes the chosen approach, pointing out the access methods used to retrieve rows from each table in the process, and the inner and outer table order and pairing.

In the following example, the nested loop join method is shown:

EXPLAIN PLAN SET QUERYNO = 100000007 FOR SELECT A.CIKEY, A.C2, A.NIKEY, A.C4, A.RIKEY1, B.C6,B.C7,B.C8,B.C9,B.RIKEY2 FROM TDT690.S100 A, TDT690.L1000 B WHERE (A.RIKEY2 BETWEEN 20 AND 270) AND (A DIVEY2 - P.DIVEY2) AND (A.RIKEY2 = B.RIKEY2)QUERYNO: 100000007 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> RANDOM MATCH TX SCAN IX CREATOR: TDT690 INDEX NAME: L1000P2N 0 KEY LEN: 6 PADDED: N C-ED: N C-ING: N CLUSRATIO: 70.0010 VERS FULLKEY CARD:100000FIRSTKEY CARD:100000TYPE:2NLEAF PAGES:359NLEVELS:3UNIQUE:UUDECLAREUNIQUE: 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 ANL6052I *** NOTE: |This index is specified as "Not Padded", allowing storage of a varying length index key. Padded indexes use blanks to fill out their fixed length keys and are not eligible for Index Only scan. "Not Padded" indexes do not blank fill CHAR and VARCHAR columns, |allowing greater flexibility and better use of storage, packing |more entries into a single Leaf page. Index Only access allowed. COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# KFY 1 RIKEY2 A DECIMAL N 9 Y 100000 0 INDEXED ACCESS CREATOR: TDT690 TABNAME: L1000 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS: 100000 COLUMNS: 50 REC LENGTH: 1008 BYTES DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N THE CODDEL ATTON NAME ASSOCIATED WITH THIS TABLE TS: B THE CORRELATION NAME ASSOCIATED WITH THIS TABLE IS: B LIST PREFETCH (SKIP SEQ.) WILL BE EMPLOYED TO ACCESS THIS TABLE A JOIN OF 2 TABLES HAS BEEN DETECTED. THIS WAS THE FIRST TABLE ACCESS. QUERYNO: 100000007 QBLOCKNO: 1 PLANNO: 2 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: S100RI2N

FULLKEY CARD:1000FIRSTKEY CARD:1000TYPE: 2NLEAF PAGES:5NLEVELS:2UNIQUE: DDUPLICATE OK1OF1COLUMNS ARE MATCHEDCLOSE:YLOCK MODE:ISBP00L: BP10 ORDER TYPE DIST LEN NULL COLCARD DIST# KFY COLUMN NAME -----A DECIMAL N 9 Y 1000 1 RIKEY2 INDEXED ACCESS CREATOR: TDT690 TABNAME: S100 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS: 1000 COLUMNS: 10 REC LENGTH: 108 BYTES DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 106 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 87% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N THE COMPELATION NAME ASSOCIATED WITH THIS TABLE TS: A THE CORRELATION NAME ASSOCIATED WITH THIS TABLE IS: A THIS TABLE IS JOINED WITH THE PRIOR TABLE VIA THE "NESTED LOOP" METHOD. THIS IS THE INNER TABLE IN THE NESTED LOOP JOIN. * QUERY 100000007 WILL REQUIRE 103.91500 SECONDS OF ELAPSED TIME * * DURING WHICH 68.15169 SECONDS OF CPU TIME WILL BE CONSUMED AND * * A TOTAL OF 7 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * * QUNITS 3722 ESTIMATED PROCESSING COST \$ 15.5035 DOLLARS * ANL5025W *** WARNING: ESTIMATE OF 68.152 EXCEEDS "CPU TIME" LIMIT OF 10 CPU SECONDS! ANL5026W *** WARNING: ESTIMATE OF 3722 EXCEEDS "SERVICE UNIT" LIMIT OF 200 QUNITS ! ANL5027W *** WARNING: 15.50 EXCEEDS "MONETARY" LIMIT OF ESTIMATE OF 15.00 DOLLARS ! ANL5029W *** WARNING: ESTIMATE OF 103.915 EXCEEDS "ELAPSED TIME" LIMIT OF 60 SECONDS! In the following example, the hybrid join method is shown: EXPLAIN PLAN SET QUERYNO = 100000008 FOR SELECT A.CIKEY, A.C2, A.NIKEY, A.C4, A.RIKEY1, B.C6,B.C7,B.C8,B.C9,B.RIKEY2 FROM TDT690.S100 A. TDT690.L1000 B WHERE (A.CIKEY BETWEEN 505 AND 550) AND (A.CIKEY = B.CIKEY) QUERYNO: 100000008 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: S100CIN VERS:0KEYLEN:6PADDED:-C-ED:NC-ING:NCLUSRATIO:100.0000FULLKEYCARD:839FIRSTKEYCARD:839TYPE:2NLEAFPAGES:4NLEVELS:2UNIQUE:DDUPLICATE0K10F1COLUMNSAREMATCHEDCLOSE:YLOCKMODE:ISBP10 ORDER TYPE DIST LEN NULL COLCARD DIST# KEY COLUMN NAME -----1 CIKEY A DECIMAL N 9 Y 839 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE INDEXED ACCESS CREATOR: TDT690 TABNAME: S100 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES

VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 85.1000

VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS:1000COLUMNS:10 REC LENGTH:108 BYTESDSSIZE:0 GBNUMBER OF MQTS:0 LOG: Y AVG ROW LENGTH:106 BYTESTYPE:TLOCK SIZE:A TS LOCK MODE:ISLOCK PART:N CLOSE TABLE: YPAGES WITH ROWS:87%PCT COMPRESSED:0% MAX ROWS:255 BPOOL:BP11ENCODE:ECCSIDS ARE SBCS:833 DBCS:834 MIXED:933 VOLATILE:N THE CORRELATION NAME ASSOCIATED WITH THIS TABLE IS: A A JOIN OF 2 TABLES HAS BEEN DETECTED. THIS WAS THE FIRST TABLE ACCESS. _____ QUERYNO: 100000008 QBLOCKNO: 1 PLANNO: 2 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD: 99340 FIRSTKEY CARD: 99340 TYPE: 2 NLEAF PAGES: 412 NLEVELS: 3 UNIQUE: D DUPLICATE OK 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# 1 CTKEY A DECIMAL N 9 Y 99340 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE INDEXED ACCESS CREATOR: TDT690 TABNAME: L1000 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES INDEXED ACCESS IS APPLIED ID A 4K SEGMENTED ISFACE WITH. 5 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS: 1000000 COLUMNS: 50 REC LENGTH: 1008 BYTES DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N THE CORRELATION NAME ASSOCIATED WITH THIS TABLE IS: B LIST PREFETCH (SKIP SEQ.) WILL BE EMPLOYED TO ACCESS THIS TABLE THIS TABLE IS JOINED WITH THE PRIOR TABLE VIA THE "HYBRID JOIN" METHOD. THIS IS THE INNER TABLE IN THE HYBRID JOIN, ACCESSED BY ITS RID LIST. QUERY 100000008 WILL REQUIRE 98.27025 SECONDS OF ELAPSED TIME * DURING WHICH 65.73150 SECONDS OF CPU TIME WILL BE CONSUMED AND * A TOTAL OF 8 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * * * A TOTAL OF 8 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK QUNITS 3590 ESTIMATED PROCESSING COST \$ 14.9600 DOLLARS * * ANL5025W *** WARNING: ESTIMATE OF 65.731 EXCEEDS "CPU TIME" LIMIT OF 10 CPU SECONDS! ANL5026W *** WARNING: ESTIMATE OF 3590 EXCEEDS "SERVICE UNIT" LIMIT OF 200 QUNITS ! ANL5029W *** WARNING: ESTIMATE OF 98.270 EXCEEDS "ELAPSED TIME" LIMIT OF 60 SECONDS!

Subquery processes

Db2 subquery processing is like join processing, except that the queries are coded as table accesses, one within another, rather than collectively from a set of tables.

Subqueries are correlated using a *correlation name* to associate columns from one query with tables from another query. A correlated subquery results in a repeated step wise run of the inner and outer queries, looking for matches.
For a noncorrelated subquery, the inner query is issued once, and the outer query scans the results of the inner query for its matches.

Db2 SQL Performance Analyzer presents information from Explain for both subquery types.

The following example shows a correlated subquery (query 9) and a noncorrelated query (query 10). In the correlated subquery, SQL PA specified that correlation names were found (a correlated subquery block). In the noncorrelated subquery, installation specified limits were exceeded and appropriate messages are displayed. For each query block, the tables accesses are depicted, along with any chosen indexed paths and other pertinent information, such as index-only processing.

EXPLAIN PLAN SET QUERYNO = 100000009 FOR SELECT * FROM TDT690.S100 X WHERE CIKEY BETWEEN 1005 AND 1500 AND EXISTS (SELECT * FROM TDT690.L1000 WHERE CIKEY = X.CIKEY) QUERYNO: 100000009 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: S100CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 100.0000 FULLKEY CARD:839 FIRSTKEY CARD:839TYPE: 2 NLEAF PAGES:4 NLEVELS: 2 UNIQUE: D DUPLICATE OK1 OF 1 COLUMNS ARE MATCHEDCLOSE: Y LOCK MODE: IS BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# -----1 CIKEY A DECIMAL N 9 Y 839 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE INDEXED ACCESS CREATOR: TDT690 TABNAME: S100 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES 0 TABLE CONTAINS A TOTAL OF: VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS: 1000 COLUMNS: 10 REC LENGTH: 108 BYTES DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 106 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 87% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N THE CODEL ATTON NAME ASSOCIATED WITH THIS TABLE IS: Y 31680 4K PAGES VERSION: THE CORRELATION NAME ASSOCIATED WITH THIS TABLE IS: X A SUBQUERY HAS BEEN DETECTED. THIS IS QUERY BLOCK NUMBER 2 QUERYNO: 100000009 QBLOCKNO: 2 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD:99340 FIRSTKEY CARD:99340TYPE: 2 NLEAF PAGES:412 NLEVELS: 3 UNIQUE: D DUPLICATE OK 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# 9 Y 99340 1 CIKEY A DECIMAL N 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE THIS IS A CORRELATED SUBQUERY BLOCK.

ANL5025W *** WARNING: ESTIMATE OF 80.356 EXCEEDS "CPU TIME" LIMIT OF 10 CPU SECONDS! ANL5026W *** WARNING: ESTIMATE OF 4389 EXCEEDS "SERVICE UNIT" LIMIT OF 200 QUNITS ! ANL5027W *** WARNING: ESTIMATE OF 18.30 EXCEEDS "MONETARY" LIMIT OF 15.00 DOLLARS ! ANL5029W *** WARNING: ESTIMATE OF 125.047 EXCEEDS "ELAPSED TIME" LIMIT OF 60 SECONDS! EXPLAIN PLAN SET QUERYNO = 100000010 FOR SELECT * FROM TDT690.S100 WHERE CIKEY IN (SELECT CIKEY FROM TDT690.L1000 WHERE CIKEY BETWEEN 1005 AND 1500) QUERYNO: 100000010 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: S100CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: N C-ING: N CLUSRATIO: 100.0000 FULLKEY CARD:839 FIRSTKEY CARD:839TYPE: 2 NLEAF PAGES:4 NLEVELS: 2 UNIQUE: D DUPLICATE OK1 OF1 COLUMNS ARE MATCHEDCLOSE: Y LOCK MODE: IS BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# -----A DECIMAL N 1 CIKEY 9 Y 839 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE INDEXED ACCESS CREATOR: TDT690 TABNAME: S100 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS: 1000 COLUMNS: 10 REC LENGTH: 108 BYTES DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 106 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 87% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N A SUBOUERY HAS BEEN DETECTED. THIS IS QUERY BLOCK NUMBER 1 _____ QUERYNO: 100000010 QBLOCKNO: 2 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD:99340FIRSTKEY CARD:99340TYPE: 2NLEAF PAGES:412NLEVELS:31OF1COLUMNS ARE MATCHEDCLOSE:YLOCK1OF1COLUMNS ARE MATCHEDCLOSE:YLOCKMODE:IS FULLKEY CARD: TYPE: 2 NLEAF PAGES: KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# A DECIMAL N 9 Y 99340 10 1 CIKEY THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE

THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE * QUERY 100000010 WILL REQUIRE 807.29139 SECONDS OF ELAPSED TIME * * DURING WHICH 557.57389 SECONDS OF CPU TIME WILL BE CONSUMED AND * * A TOTAL OF 338 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * QUNITS 30448 ESTIMATED PROCESSING COST \$ 129.5278 DOLLARS * ANL5025W *** WARNING: 557.574 EXCEEDS "CPU TIME" LIMIT OF 10 CPU SECONDS! ESTIMATE OF ANL5026W *** WARNING: ESTIMATE OF 30448 EXCEEDS "SERVICE UNIT" LIMIT OF 200 QUNITS ! ANL5027W *** WARNING: ESTIMATE OF 129.53 EXCEEDS "MONETARY" LIMIT OF 15.00 DOLLARS ! ANL5029W *** WARNING: ESTIMATE OF 807.291 EXCEEDS "ELAPSED TIME" LIMIT OF 60 SECONDS!

Sorting results

Db2 SQL Performance Analyzer provides information about sorting in its Enhanced Explain Reports, including sorts done on behalf of joins, sorts requested by the user in ORDER BY or GROUP BY clauses, and sorts for uniqueness caused by the DISTINCT or UNION keywords.

The following example shows SQL PA generated information for sorts. Sorts caused by UNION processing or a JOIN of multiple tables might produce several additional messages:

A "UNION" SORT FOR UNIQUENESS HAS BEEN DETECTED A "JOIN" SORT IS ALSO REQUIRED A SORT WILL BE PERFORMED ON THE RESULTS EXPLAIN PLAN SET QUERYNO = 100000011 FOR SELECT * FROM TDT690.L1000 WHERE CIKEY BETWEEN 501 AND 510 UNION SELECT * FROM TDT690.L1000 WHERE CIKEY BETWEEN 501 AND 510 QUERYNO: 100000011 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD:99340FIRSTKEY CARD:99340TYPE: 2NLEAF PAGES:412NLEVELS:3UNIQUE: DDUPLICATE OK 99340 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# A DECIMAL N 9 Y 99340 10 1 CIKEY THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE INDEXED ACCESS CREATOR: TDT690 TABNAME: L1000 4K SEGMENTED TSPACE WITH: INDEXED ACCESS IS APPLIED TO A 3 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS: 100000 COLUMNS: 50 REC LENGTH: 1008 BYTES DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N 31680 4K PAGES

ANL6038I *** NOTE: |Building an index with the same ordering of sort key columns |will negate the sort requested here... consider as warranted. ANL7022I *** GUIDELINE: Avoid referencing Varchar columns in the Sort, as these will be padded to maximum size in the sort work records. Long sort records over 4000 bytes cause a "Tag" sort, which requires even |more overhead to park records in a file and read them afterward. QUERYNO: 100000011 QBLOCKNO: 2 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD:99340FIRSTKEY CARD:99340TYPE: 2NLEAF PAGES:412NLEVELS:3UNIQUE: DDUPLICATE OK1OF1COLUMNS ARE MATCHEDCLOSE:YLOCK MODE:ISBP00L: BP10 COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# KEY 1 CIKEY A DECIMAL N 9 Y 99340 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE INDEXED ACCESS CREATOR: TDT690 TABNAME: L1000 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES TABLE ROWS: 100000 COLUMNS: 50 REC LENGTH: 1008 BYTES DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N 31680 4K PAGES VERSION: O TABLE CONTAINS A TOTAL OF: A "UNION" SORT FOR UNIQUENESS HAS BEEN DETECTED. QUERY 100000011 WILL REQUIRE 39.55082 SECONDS OF ELAPSED TIME * DURING WHICH 34.58685 SECONDS OF CPU TIME WILL BE CONSUMED AND * A TOTAL OF 4 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * * * A TOTAL OF 4 PHYSICAL I/O REQUESTS WIL QUNITS 1889 ESTIMATED PROCESSING COST \$ 7.8358 DOLLARS * ANL5025W *** WARNING: 34.587 EXCEEDS "CPU TIME" LIMIT OF 10 CPU SECONDS! ESTIMATE OF ANL5026W *** WARNING: ESTIMATE OF 1889 EXCEEDS "SERVICE UNIT" LIMIT OF 200 QUNITS !

Insert, Update, Delete, Select, and Refresh

SQL Performance Analyzer reports on INSERT, UPDATE, DELETE, SELECT, and REFRESH statements.

Users might want direct I/U/D processing against the tables, or to use "where current of cursor" techniques. Both forms of requests are handled in the Enhanced Explain reports. Other information, such as the number of columns updated, is also provided.

The Db2 SQL Performance Analyzer Enhanced Explain Report contains descriptions of processing under Db2, with the emphasis on what steps Db2 is taking, and in what order. SQL PA provides the statistics

accompanying the tables and indexes used for access. Other notations describe query transformation from subquery to join, subquery to simple predicate, or join to subquery access, by direct rowid.

Referential integrity relationships are also noted in the explain report for INSERT, UPDATE, and DELETE statements, notifying you of the additional processing caused by the relationships.

The following example shows Insert (query 12) processing.

EXPLAIN PLAN SET QUERYNO = 100000012 FOR INSERT INTO TDT690.L1000 VALUES (100100, 'AAAAA', 'BBBBBB', 'CCCCC', 'DDDDD', 'EEEEE', 'FFFFF 'GGGG', 'HHHHH', 'IIIII', 100100, 'JJJJJ', 'KKKKK', 'LLLLL','MMMM', 'NNNNN', '00000', 'PPPP', 'QQQQQ', 'RRRRR', 100100, 'SSSS','TTTTT', 'UUUUU', 'VVVV', 'WWWWW', 'XXXXX', 'YYYY', 'ZZZZ', 'AAAAA', 100100, 'BBBB', 'CCCCC', 'DDDDD', 'EEEE', 'FFFF', 'GGGGGG', 'HHHH', 'IIII', 'JJJJJ', 'KKKKK', 'LLLL', 'MMMM', 'NNNNN', '0000', 100100, 'PPPPP', 'QQQQQ', 'RRRR', 'SSSS') 'FFFFF'. QUERYNO: 100000012 QBLOCKNO: 1 PLANNO: 0 MIXOPSEQ: 0 PROCESS -> INSERT VIA CLUSTERING INDEX CREATOR: TDT690 TABNAME: L1000 VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD:99340FIRSTKEY CARD:99340TYPE: 2NLEAF PAGES:412NLEVELS: 3UNIQUE: DDUPLICATE OK FULLKEY CARD: 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 ANL7012I *** GUIDELINE: |Make sure that you have defined adequate Free Space parms in this | index structure, and in the data table, to keep rows in the | |clustered order and avoid fragmentation. Both Freepage and Pctfree| |should be deployed, and Runstats periodically run to update stats.| INSERT VIA IDX CREATOR: TDT690 TABNAME: L1000 INSERT PROCESSING APPLIED TO A 4K SEGMENTED TSPACE WITH 3 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGE TABLE ROWS: 100000 COLUMNS: 50 REC LENGTH: 1008 BYTE DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTE TYPE: T LOCK SIZE: A TS LOCK MODE: IX LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: 1 VERSION: 0 TABLE CONTAINS A TOTAL OF: 31680 4K PAGES 50 REC LENGTH: 1008 BYTES AVG ROW LENGTH: 1006 BYTES 933 VOLATILE: N ANL5025W *** WARNING: 25.637 EXCEEDS "CPU TIME" LIMIT OF ESTIMATE OF 10 CPU SECONDS! ANL5026W *** WARNING: 1400 EXCEEDS "SERVICE UNIT" LIMIT OF ESTIMATE OF 200 QUNITS ! The following example shows Update (query 13) processing. EXPLAIN PLAN SET OUERYNO = 100000013 FOR UPDATE TDT690.L1000 SET C2 = 'NEW11',

SET C2 = 'NEW11', C7 = 7654321 WHERE CIKEY = 100100 QUERYNO: 100000013 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> UPDATE OPERATION HAS BEEN REQUESTED TO CHANGE THE VALUE OF 2 COLUMNS

ANL7001I *** GUIDELINE: Make sure that your frequently updated columns are being kept together in the table row: DB2 logs from the first to last changed byte. Also, put Varchar at the end of the row: DB2 logs from the first changed byte to end of row. Length changes cause the entire | |row to log, as length field is kept with the row header. Rows that |are compressed are variable length & logged in compressed format. | CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD: 99340 FIRSTKEY CARD: 99340 TYPE: 2 NLEAF PAGES: 412 NLEVELS: 3 UNIQUE: D DUPLICATE OK 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IX BPOOL: BP10 ORDER TYPE DIST LEN NULL COLCARD DIST# KFY COLUMN NAME 1 CIKEY A DECIMAL N 9 Y 99340 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE |ANL6020I *** NOTE: This index is presently designated as allowing "duplicate values" Make sure this is a nonunique index that does not fall into one of the following "unique" categories: explicitly unique, primary key, non-primary RI parent key, unique where not null, unique column constraint. Unique indexes have definite advantages whenever they can be declared by the user, so be sure it contains duplicates. THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE UPDATE VIA IDX CREATOR: TDT690 TABNAME: L1000 UPDATE PROCESSING APPLIED TO A 4K SEGMENTED TSPACE WITH 3 TABLES VERSION: 0 TABLE CONTAINS A TOTAL OF: 316 TABLE ROWS: 100000 COLUMNS: 50 REC LENGTH: DSSIZE: 0 GB NUMBER OF MQTS: 0 LOG: Y AVG ROW LENGTH: 31680 4K PAGES 50 REC LENGTH: 1008 BYTES AVG ROW LENGTH: 1006 BYTES DSSIZE: 0 GB NUMBER OF MOTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTES TYPE: T LOCK SIZE: A TS LOCK MODE: IX LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N * DURING WHICH 46.63751 SECONDS OF CPU TIME WILL BE CONSUMED AND * * A TOTAL OF 20 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * * QUNITS 2547 ESTIMATED PROCESSING COST \$ 10.7093 DOLLARS * ANL5025W *** WARNING: ESTIMATE OF 46.638 EXCEEDS "CPU TIME" LIMIT OF 10 CPU SECONDS! ANL5026W *** WARNING: ESTIMATE OF 2547 EXCEEDS "SERVICE UNIT" LIMIT OF 200 QUNITS ! The following example shows Delete (query 14) processing.

EXPLAIN PLAN SET QUERYNO = 100000014 FOR DELETE FROM TDT690.L1000 WHERE CIKEY = 100100 OR RIKEY2 = 100300 QUERYNO: 100000014 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> A MULTIPLE INDEX OPERATION HAS BEEN REQUESTED TO ACCESS THIS TABLE: QUERYNO: 100000014 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 1 PROCESS ->

RANDOM MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000P2N VERS: 0 KEY LEN: 6 PADDED: N C-ED: N C-ING: N CLUSRATIO: 70.0010 FULLKEY CARD: 100000 FIRSTKEY CARD: 100000 TYPE: 2 NLEAF PAGES: 359 NLEVELS: 3 UNIQUE: U DECLARE UNIQ 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IX BPOOL: BP10 ANL6052I *** NOTE: This index is specified as "Not Padded", allowing storage of a varying length index key. Padded indexes use blanks to fill out their fixed length keys and are not eligible for Index Only scan. Not Padded" indexes do not blank fill CHAR and VARCHAR columns, allowing greater flexibility and better use of storage, packing more entries into a single Leaf page. Index Only access allowed. KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# -----1 RIKEY2 A DECIMAL N 9 Y 100000 0 THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE QUERYNO: 100000014 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 2 PROCESS -> CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD: 99340 FIRSTKEY CARD: 99340 TYPE: 2 NLEAF PAGES: 412 NLEVELS: 3 UNIQUE: D DUPLICATE OK 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IX BPOOL: BP10 KEY COLUMN NAME ORDER TYPE DIST LEN NULL COLCARD DIST# 1 CIKEY A DECIMAL N 9 Y 99340 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE THIS IS AN "INDEX ONLY" ACCESS: NO DATA PAGES ARE READ FROM THE TABLE STEP 3. MULTIPLE INDEX UNION WAS PERFORMED (MU). MULTIPLE INDEX OPERATIONS ARE NOW COMPLETE: DATA ACCESS PERFORMED. QUERYNO: 100000014 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 3 PROCESS ->

Db2 SQL Performance Analyzer detail trace report

The Db2 SQL Performance Analyzer detail trace report, which is the most detailed, contains hypothetical trace information covering virtually every facet of processing.

The information describes each query step as though it were issued, and shows the expected usage of all system resources. Db2 logical entities are isolated, with the processing broken down into small, finite pieces. The individual components contributing to overall response time is shown, allowing you to fine-tune queries at the most granular level possible.

For example, Db2 SQL Performance Analyzer shows the predicted path lengths in instructions for fetching rows, or performing sequential prefetch I/O calls, or logging records updated by the query. SQL PA also predicts how long Db2 spends scanning the rows. The elapsed (response) time is broken down into its components, from waiting for data sets to open and close, to waiting for synchronous (or prefetch) I/O to complete.

The report is written to the file allocated by the QTRACE DD statement in batch and to the file designated as the trace report in the SQL PA Define Report Data Sets panel.

Db2 SQL Performance Analyzer shows its predictions in typical Db2 terms, such as forecast numbers for getpages, synchronous reads, sequential and list prefetches, and the CPU time users might anticipate in the Db2 SMF records.

Topics:

- "Forecasting query processing" on page 178
- "List of the Db2 SQL Performance Analyzer install parameters" on page 179
- "Processing steps" on page 180
- "I/O estimates" on page 182
- "Path lengths and CPU time" on page 183
- "Physical and logical I/O" on page 185
- "Elapsed wait time" on page 186
- "Sample Db2 SQL Performance Analyzer detail trace report" on page 187
- "Sorting results" on page 173
- "Insert, Update, Delete, Select, and Refresh" on page 174

Forecasting query processing

The entire SQL PA cost summary estimate, along with the forecasted elapsed time, CPU time, and I/O are determined by mapping the access path chosen by Db2 against the current picture of the tables and indexes, as represented by the catalog.

Using known physical attributes for CPU processing speeds, disk device service times, and thousands of individual path lengths observed using private benchmarks of Db2, the query execution is analyzed.

Db2 SQL Performance Analyzer parses the SQL, and detects the type of query. SQL PA analyzes as much as possible for the cost estimation. The cost estimation is based on all the small and large details that influence performance, including the calculations on a WHERE clause, built-in functions, the use of certain predicates, join table orders, and RID list counts.

Db2 SQL Performance Analyzer standard Cost Summary format:

Db2 optimizer's Cost Summary format (PRECISE YES):

The key statistical variables presented in the SQL PA Detail Trace Report are mapped to their Db2 counterparts in the sample Detail Trace Report.

List of the Db2 SQL Performance Analyzer install parameters

The Detail Trace report lists all of the Db2 SQL Performance Analyzer installation parameters at the top of the report.

Unless DSPPARM is set to NO, the beginning of a Detail Trace report lists the key parameter values used by Db2 SQL Performance Analyzer for a run, as shown in the following example:

15:23:50.338 06-25-2013	SQL Performance Analyzer Execution Forecast: Detail Trace	Version 5.1.0 Level 51-5100 APAR PI12345
SQL PA Parameters (ANLPARM) VERSION V11R1 DEGREES 1 BUFFHIT 0 DBRMKEY +OFF+ QUALIFY +OFF+ QUALIFY +OFF+ CONNECT CAF REFRESH NO VIADRDA +OFF+ PKGDRDA +OFF+ RETCODE NO DELIMIT QUOTE DSN8EXP NO NLSCODE +OFF+ KEEPIAN VES	SQL PA Parameters	APAR PI12345
PROCESS +0FF+ NUMBERS YES REPORTS DET SHOWALT NO PRECISE ALL ADVISOR ALL OBJECTS YES QLIMSQL YES NOSTATS YES DSPVARS YES DSPVARS YES DSPVARS YES DSPPARM YES JOINTAB 10 INLISTS 10 IXUPDAT 5 TSCANPG 50000 ISCANPG 1000 NONIXPG 5000 MATCOLS 0.50 PREDPCT 0.15 STORAGE 3390-3 LEVEL TAB		
INDEX NO ACCTYPE ALL HOSTVAR YES FORMAT YES PLANINF YES DSPWCOC YES CSVRPT YES CSVRPT YES CSVSEL TAB0 CSVHDRW COL CSVDLIM COMMA CSVDCSP DOT (ANLCNTL) SUBVERS V11NFM SUBSYST DSNB		
BUFFERS 2000 BUFF08K 500 BUFF16K 250 BUFF32K 100 DYNAMIC NO AUTHIDO ANLUSR51 DBTRACE OFF SETPLAN YES USRPARM YES FASTTBL YES DSGROUP 0 CPUTIME 10 COSTING 15 ELAPSED 60		

COSTOUN	200
TOCALLS	1000
CDUCOST	
TOCCOST	
TUSCUST	10.00
TIMCOST	10.00
MONEYIS	DOLLARS
CURRSYM	\$
(ANLPARM)	
DEGREES	1
DBRMKEY	+0FF+
EXPLAIN	NEW
EEECALL	SPA
OUAL TEY	
USEPI AN	JSMTTH
STOPONO	00000000
	N/A will be discovered by eads
	N/A WIII DE dISCOVEIEd Dy Code
(ANLUNIL)	N/A 177 be diserved by sede
SUBVERS	N/A WILL be discovered by code
SUBSYST	N/A was passed in a program parameter

Processing steps

Each major step in the Db2 process noted in the Enhanced Explain Report is expanded upon in the Detail Trace Report. The SQL PA Detail Trace Report presents an estimate of how many rows and columns are processed by the statement, along with the percent of the table processed.

These process estimate numbers are derived by SQL PA as the Db2 optimizer would derive them, and are based on WHERE clause predicates and their filter factors. At times, they are different from what the user experiences because the optimizer uses predefined filters for many of the predicates, and defaults for others.

The OPTIMIZE FOR *n* ROWS clause and FETCH FIRST *n* ROWS ONLY are recognized by Db2 SQL Performance Analyzer as an attempt to establish a more precise estimate based on rows returned. When Db2 SQL Performance Analyzer detects this clause, it does not use the optimizer emulation calculations, but starts with your actual row estimate for greater accuracy. Give SQL PA a realistic estimate of the actual number of rows returned by a query for the best results. From this estimate, SQL PA calculates many things, including the number of pages needed to hold those rows, the I/O required for those pages, the processing impacts.

You can use the internal estimates of the optimizer for service units and CPU time using the PRECISE parameter (see <u>"ANLPARM user parameters" on page 230</u>). However, the estimates of the optimizer might not be as accurate as a row estimate that you provide.

The banner for each SQL statement contains QUERYNO (query number), QBLOCKNO (query block number), PLANNO (plan number) and MIXSEQ (multiple index operational sequence) from the Db2 Explain facility which identify each step in the process.

The block is marked **PROCESS ->** and contains the following estimates:

ROWS PROCESSED

Db2 SQL Performance Analyzer estimate of rows returned, based on predicate filters or OPTIMIZE FOR n ROWS clause or FETCH FIRST n ROWS clause.

PERCENT TABLE PROCESSED

Db2 SQL Performance Analyzer calculated percentage of the table accessed by query (always computed).

COLS PROCESSED

A count of columns from SELECT list, or from catalog if all rows selected.

BOOLEAN FILTER FACTORED or BOOLEAN FILTER FROM Db2|OPT

Db2 SQL Performance Analyzer final filter factor from predicate analysis (or from CPU time as supplied by optimizer).

DATA PAGES READ

Estimate of total table page reads required by this PLAN/MIXSEQ step.

INDEX LEAF PAGES READ

Db2 SQL Performance Analyzer estimate of index leaf page reads; it does not include root or nonleaf pages.

Each major step in the Db2 process noted in the Enhanced Explain Report is expanded upon in the Detail Trace Report.

For each step, there is a summary which includes the enhanced explain information, and estimates for many key variables and path lengths in the Db2 SQL Performance Analyzer evaluation process. These estimates are computed by SQL PA. The estimates are based on the DB2-explained plan, the catalog statistics for the indexes and tables used in this step of the plan, optimizer algorithm knowledge, and SQL PA parser intelligence data. The step information in the Detail Trace Report contains information like the following example:

SELECT * FROM TDT690.L1000 WHERE CIKEY = 50086 QUERYNO: 100000001 QBLOCKNO: 1 PLANNO: 1 MIXOPSEQ: 0 PROCESS -> ANL7002I *** GUIDELINE: This plan step has not selected any Sequential|List Prefetch I/O. |If the SQL processes just a few rows that is OK, but if many rows are involved, you can help promote Sequential Detection by both |accessing the data in sequential order (presort?) and by binding |with Release (Deallocate) to avoid resetting counters at Commit. ANL7003I *** GUIDELINE: Close Yes was specified for the Tablespace and/or the index... if these are little used this is OK. If high volume access then consider Close No. Extremely relevant pages can be "page fixed" in memory by highly referencing, using Hiperpools (Castout Y/N), putting into a dedicated buffer pool large enough to hold all pages, deploying data sharing with Group Buffer Pool Cache All option, etc. each with associated costs. Close No also increases chances that DBD will remain in EDM Pool for next execution. -----|ANI 7006T *** This statement contains a select of all columns in the table. Typically a select of all columns results in significantly increased processing time to fetch and process each column of each row. Use select all only when you want to select all columns in a view definition. Improve performance by specifying the name of |each column for the SQL SELECT statement. CLUSTER MATCH IX SCAN IX CREATOR: TDT690 INDEX NAME: L1000CIN VERS: 0 KEY LEN: 6 PADDED: - C-ED: Y C-ING: Y CLUSRATIO: 99.9995 FULLKEY CARD: 99340 FIRSTKEY CARD: 99340 TYPE: 2 NLEAF PAGES: 412 NLEVELS: 3 UNIQUE: D DUPLICATE OK 1 OF 1 COLUMNS ARE MATCHED CLOSE: Y LOCK MODE: IS BPOOL: BP10 ORDER TYPE DIST LEN NULL COLCARD DIST# KEY COLUMN NAME A DECIMAL N 9 Y 99340 1 CIKEY 10 THIS IS THE CLUSTERING ("INSERT & LOAD ORDER") INDEX FOR THIS TABLE ANL6020I *** NOTE: This index is presently designated as allowing "duplicate values" |Make sure this is a nonunique index that does not fall into one of |the following "unique" categories: explicitly unique, primary key, |non-primary RI parent key, unique where not null, unique column

constraint. Unique indexes have definite advantages whenever they |can be declared by the user, so be sure it contains duplicates. INDEXED ACCESS CREATOR: TDT690 TABNAME: L1000 INDEXED ACCESS IS APPLIED TO A 4K SEGMENTED TSPACE WITH: 3 TABLES 0 TABLE CONTAINS A TOTAL OF: INS A TOTAL OF: 31680 4K PAGES 100000 COLUMNS: 50 REC LENGTH: 1008 BYTES VERSION: TABLE ROWS: DSSIZE: 0 GB NUMBER OF MOTS: 0 LOG: Y AVG ROW LENGTH: 1006 BYTE TYPE: T LOCK SIZE: A TS LOCK MODE: IS LOCK PART: N CLOSE TABLE: Y PAGES WITH ROWS: 99% PCT COMPRESSED: 0% MAX ROWS: 255 BPOOL: BP11 1006 BYTES ENCODE: E CCSIDS ARE SBCS: 833 DBCS: 834 MIXED: 933 VOLATILE: N ANL7015I *** GUIDELINE: |Collecting Non-uniform column statistics greatly enhance the DB2 Optimizer accuracy when processing Equals and In (List) preds. ROWSPROCESSED =10COLSPROCESSED =50DATAPAGESREAD=3 PERCENT TABLE PROCESSED = 0.000100000 BOOLEAN FILTER FROM OPT = 0.000100000 INDEXED LEAF PAGES READ = 1 PREFETCH I/OS = ASYNC WRT I/OS = GET PAGE CALLS 6 0 TABLE =3 TNDFX =3 TABLE = 0 INDEX = 0 0 TABLE = 0 INDEX = 0 GET PAGE CALLS = 38 SYSIO = 32 LOGIO = 0 0.044100 PREFETCH = 0.000000 ASYNC WRT = SYNC READ = 0.000000 GET PAGES = 0.093100 SYSTEMS = 0.258450 LOG WRITE = 0.000000 DECOMPRESS= 0.000000 COMPRESS = 0.000000 HIPERPOOL = 0.000000 FETCH ROW = 0.044500 LOCK/ETC = 0.042000 PROCESSES = 0.066795 1 CLASS 1 = PREDICATE = 0.548945 OTHER 0/H = 0.846400 QUERYNO: 100000001 QBLOCKNO: 1 PLANNO: 1 MIXOPSEO: SUMMARY -> THE TOTAL CUMULATIVE PATH LENGTH FOR THIS QUERY IS RESULTING IN A TOTAL CPU TIME OF 0.01311 SECOND 1.395345M INS. 0.01311 SECONDS CONSUMED OVERALL. 0.00516 SECONDS IN SMF 101 RECORD. DB2 WILL PUT CLASS 1 CPU TIME OF 0.00795 SECONDS IN SMF 100 RECORD. DB2 SHOWS ADDITIONAL CPU TIME OF ESTIMATED TOTAL LOGICAL I/O CALLS = 6 (EXCLUDING SYSTEM) AND ESTIMATED TOTAL PHYSICAL I/O CALLS = 6 WITH HIT RATIO = 1.000 6 WITH HIT RATIO = 1.000. WAIT TIME FOR SYNC READ I/O = 0.13831 PREFETCH I/O = WAIT TIME ON ASYNC WRITE I/O = 0.00000 TOTAL IWAIT = PREFETCH I/O =0.00000 0.13831 WAIT TIME FOR VSAM OPEN/CLOSE MACROS, BINDING AND LOCKING = 1.79730 QUERY 100000001 WILL REQUIRE 1.94872 SECONDS OF ELAPSED TIME * * DURING WHICH0.01311SECONDS OF CPU TIME WILL BE CONSUMED AND *A TOTAL OF6 PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK *QUNITS1 ESTIMATED PROCESSING COST \$0.0683 DOLLARS * * *

I/O estimates

The Db2 SQL Performance Analyzer detail trace report includes I/O estimate information. The total number of data pages and index leaf pages read is a function of the predicate filters and the SQL PA estimate of the percent of the table and index processed to retrieve the necessary number of rows.

Some of these pages are read through synchronous reads (one page at a time) or by sequential prefetch (both table and index) or list prefetch (table only). Thus, the number of pages does not necessarily equal the number of I/Os. An asynchronous sequential or list prefetch can read in 8, 16, 32, or more pages with a single I/O request.

Db2 SQL Performance Analyzer separates the individual I/O estimates in the SQL PA Detail Trace Report as follows:

SYNC READ I/Os

Db2 SQL Performance Analyzer estimate of the total number of single, synchronous read I/Os issued; separate counts are provided for TABLE and INDEX.

PREFETCH I/Os

Db2 SQL Performance Analyzer estimate of the total number of multiple page, asynchronous read I/Os issued, including both sequential and list prefetches; TABLE and INDEX counts are again provided.

ASYNC WRT I/Os

Db2 SQL Performance Analyzer estimate of the asynchronous write I/Os caused by INSERT, UPDATE, or DELETE statements; the number of pages written back to disk with updated values for TABLE and INDEX pages.

GET PAGE CALLS

Fully loaded estimate of getpage calls, which includes all the logical reads for tables and indexes, and the system I/O (see SYSIO).

SYSIO

Estimate of internal getpages caused by actions such as dynamic binds, as appropriate for query.

LOGIO

Db2 SQL Performance Analyzer estimate of Db2 log writes in support of insert, update, and delete transactions, if any.

The logical I/O numbers do not necessarily equal the total number of physical I/O calls caused by this step in the query. Consider that the logical I/O requests are uniform for each query. If a query needs to read 25 data pages and 6 index pages, then it requires the same page counts (getpages) every time, unless the predicates or data composition change. Whether those pages are already found in the buffer pool, or need to brought in from disk, determines the buffer hit ratio. Physical I/O consists of those pages which must be read from disk, and not the total getpages issued. The parameter BUFFHIT, reflects this physical to logical I/O ratio, and all three values are shown later in the report.

Path lengths and CPU time

The Detail Trace Report includes the path lengths, or instruction counts, for various processes during the processing of each step.

Db2 SQL Performance Analyzer obtains the information through a series of benchmarks that capture the individual costs of each Db2 process, and are like most analytical queuing and discrete simulation models used for DB2-related capacity planning. The path lengths refer to the number of instructions needed to accomplish a specific task within Db2, and are the results of thousands of benchmarked measurements on all possible Db2 access paths and scenarios.

The path lengths are presented at a granular level, to provide you with an incremental-cost view of each query processed. In the report, you are able to see where SQL PA estimates the bulk of processing time is to be spent. All path lengths are presented in millions of instructions, so a value of 0.950750 represents 950,750 instructions, and a value of 23.500250 indicates that 23,500,250 instructions are issued.

Db2 SQL Performance Analyzer is not a modeling tool, but it can provide input about individual transaction resource consumption to a modeling tool. SQL PA provides a complete, end-to-end picture of individual query processing. These are the building blocks of the capacity model, which then summarize the individual queries, combining them into workloads at various arrival rates. These workloads are then used to project the overall application performance and capacity of a Db2 system.

Db2 SQL Performance Analyzer does not predict performance of specific application volumes or within specific workload mixes. Application performance is a job reserved for the queue and simulation model tools. The role of the model is to evaluate the performance of the entire application, given a specific transaction mix, while considering the rest of the workloads processing on the complex.

The role of Db2 SQL Performance Analyzer is to evaluate the performance of each individual query. SQL PA does predict individual transaction performance, and evaluates the relative cost of using one method to access data compared with another. With the performance information provided by SQL PA, you can determine individual query performance assessment and perform fine-tuning of your queries.

Path length-related values shown in the report include the following categories:

PROCESSES

Estimate of the primary processing path length, including Stage 1 and Stage 2 predicate scans, index and data page scans, built-in functions, calculations, and similar processes for the access path chosen.

CLASS 1

Db2 SQL Performance Analyzer estimate of those processes charged to class 1 CPU time, including PROCESSES, GET PAGE, FETCH ROW, SYNC READ, SYSTEMS, DECOMPR, LOCK/ETC, and the Read part of HIPERPOOL.

OTHER O/H

Db2 SQL Performance Analyzer estimate of those processes charged to Db2 processor usage, including the Attach facility, PREFETCH, LOG WRITE, COMPRESS, and ASYNCWRT.

SYNC READ

Estimate for the cost of issuing all single page, synchronous read I/O requests.

PREFETCH

Db2 SQL Performance Analyzer estimate for the cost of issuing all multi-page sequential and list prefetch I/O requests.

ASYNC WRT

Db2 SQL Performance Analyzer estimate for the cost of issuing all data and index page writes to disk, after inserts, updates, and deletes.

DECOMPRESS

The path length to decode rows for processing.

COMPRESS

The path length to encode rows before write.

HIPERPOOL

The total path length for hiperpool activity, including both Reads and Writes of updated pages.

GET PAGES

Db2 SQL Performance Analyzer estimate for the cost of all getpage logical I/O requests issued by this step.

SYSTEMS

General processor usage initiated by the system for the SQL statement, including open and close table, index and cursor, and commits.

LOG WRITE

The cost estimate for any log records that are written because of inserts, updates, and deletes by this step of the query.

FETCH ROW

Estimate for the path length caused by fetching the rows and columns for this step.

LOCK/ETC

Db2 SQL Performance Analyzer category for the odd path lengths, in which locking, HAVING clauses, descending sort and other such path length costs are attributed.

PREDICATE

The number of predicates detected on the WHERE clause by the parser.

The two summary fields, **CLASS 1** and **OTHER O/H**, are totals of other fields which appear in this section.

CLASS 1 = PROCESSES + SYNC READ + GET PAGES + SYSTEMS + FETCH ROWS + LOCK/ETC + HIPERPOOL (Reads) + DECOMPRESS OTHER O/H = PREFETCH + ASYNC WRT + LOG WRITE + COMPRESS + HIPERPOOL (Writes)

All numbers are given in millions of instructions.

TOTAL CUMULATIVE PATH LENGTH

A running total of the entire path length for all plan or multiple index operational sequence (MIXSEQ) steps in the process.

To create a model for estimating a transaction that will be run repeatedly in a bundle, do not multiply the result of one statement by the number of times that the statement will be run. Since the connection overhead is only needed once, multiplying the running total for one will overstate the results. Instead, remove the processor usage before you multiply, and add the processor usage to the result after you multiply, if the transactions can be bundled.

The next section of the report, following the cumulative path length, converts these path lengths into CPU time, based on the description of the CPU processor specified in the installation parameters.

The following example shows the cumulative path length and CPU time estimates, along with I/O and wait time values.

QUERYNO: 100000001 SUMMARY -> 1 PLANNO: 1 MIXOPSEQ: QBLOCKNO: 0 THE TOTAL CUMULATIVE PATH LENGTH FOR THIS QUERY IS 1.395345M INS. RESULTING IN A TOTAL CPU TIME OF DB2 WILL PUT CLASS 1 CPU TIME OF DB2 SHOWS ADDITIONAL CPU TIME OF 0.01311 SECONDS CONSUMED OVERALL. 0.00516 SECONDS IN SMF 101 RECORD. 0.00795 SECONDS IN SMF 100 RECORD. ESTIMATED TOTAL LOGICAL I/O CALLS = ESTIMATED TOTAL PHYSICAL I/O CALLS = 6 (EXCLUDING SYSTEM) AND 6 WITH HIT RATIO = 1.000. 0.13831 WAIT TIME FOR SYNC READ I/O = PREFETCH I/0 =0.00000 WAIT TIME ON ASYNC WRITE I/O = 0.00000 TOTAL IWAIT = 0.13831 WAIT TIME FOR VSAM OPEN/CLOSE MACROS, BINDING AND LOCKING = 1.79730 -*-*-*-*-*

Figure 55. Process totals

The total cumulative path length for this query contains the rolling cumulative total of all instructions issued, including the CLASS 1, and OTHER O/H path lengths assessed for each step. The value in this field is typically a cumulative total, except in the following two cases—multiple index operations and subqueries. In these two situations, the value shows the individual path length for the multiple index operation step or the subquery block, providing you with a greater insight into the incremental costs of the query.

Resulting in a total CPU time of xx seconds consumed overall converts those instruction counts into CPU time, using the processor speed determined by the configuration parameters. Regardless of whether the path length is cumulative or individual (MIX, SBQRY), the CPU times calculated in this total are always *cumulative*.

Db2 puts class 1 CPU time of xx seconds in SMF 101 record shows the part of the CPU time that SQL PA estimates shows up on a class 1 accounting report; that is, charged to the user. This class 1 estimate is part of the total CPU time just above it on the trace report, and corresponds to the total CLASS 1 path length.

Db2 shows additional CPU time of xx seconds in SMF 100 record shows the CPU time that SQL PA estimates are collected system wide and put in statistics record; that is, absorbed by the Db2 address spaces. Additional CPU time corresponds to CPU time consumed by the OTHER O/H path length, and is the other part of the total CPU time two lines above it.

Physical and logical I/O

SQL PA reports estimates of the logical I/O requirements of a query and the physical I/O consequences of those requests.

There is a substantial difference between the logical I/O requirements of a query and the physical I/O consequences of those requests. SQL PA reports both estimates, governed by the BUFFHIT parameter, which specifies the percentage of time that a getpage is satisfied by a page found in the buffer pool. This is important for determining overall elapsed time and query response because the wait time for various

physical I/O is a significant contributor to the elapsed time estimate. Similarly, HPOOLRD influences the Physical I/O elapsed time—the more pages read from hiperpool, the less time to read.

Estimated total logical I/O calls

The anticipated I/O requirements of the query. The total is based on the filter factors derived from the predicates, and the number of index and data table pages SQL PA has forecasted to be read to resolve the query. This number should not change between processing of identical queries. Logical I/O includes all synchronous reads, sequential and list prefetch requests, asynchronous writes, logging, sort work reads and writes, and those special scans caused by subqueries and joins.

Estimated total physical I/O calls are what is left of the logical I/O calls after applying the BUFFHIT parameter to them. SQL PA expects these I/O to require disk access, and these pages must be brought into the buffer pool before scanning. This number can vary between processing of identical queries, as the number of pages found in the buffer pool could vary with usage by many users. Some of these pages can also be retrieved from hiperpool (HPOOLRD) rather than disk.

With hit ratio

Describes the physical I/O factor for this run or Logical I/O * (1 - BUFFHIT) = Physical I/O. The elapsed time estimate includes physical I/O wait time. SQL PA computes elapsed time with the knowledge of which pages are read from disk as opposed to those pages found in the buffer pool. Thus, elapsed times predicted by SQL PA vary with the BUFFHIT parameter value.

Elapsed wait time

The wait time for physical I/O is one of the components of elapsed time. The Detail Trace Report tracks separate wait time estimates for the following major I/O categories.

The BUFFHIT, DEGREES, STORAGE, VERSION, and CONNECT user parameters, with the system configuration description parameters in ANLCNTL influence the outcome of Db2 SQL Performance Analyzer elapsed time estimation.

Similarly, defining your own DASD devices with the NEWSTOR, NEWSEEK, NEWROTA, and NEWXFER parameters can dramatically change the elapsed time estimates.

Wait time for sync read I/O

Shows the calculated amount of time to process the estimated number of synchronous reads, using the disk characteristics discussed in <u>anluc_install_parms.dita</u>. Specifically, the time to read a 4 KB, 8 KB, 16 KB, or 32 KB page is calculated and shown. The wait time varies with different disk devices specified as STORAGE, because they operate at varying speeds.

Wait time for prefetch I/O

Shows the calculated amount of time to process the estimated number of I/Os for sequential and list prefetch activities. The physical disk characteristics are applied to the number of I/O estimated by Db2 SQL Performance Analyzer. Hiperpool reads are considered here, and any parallel I/O opportunities, and their effect on elapsed time.

Wait time for async write I/O

Shows wait time for actually updating the data and index pages on disk, typically a process performed after commit (and often after user termination). The values are normally not seen on a class 1 report. However, they are an important part of the workload for inserts, updates, and deletes. The values are ascertained by Db2 SQL Performance Analyzer. Hiperpool writes are also considered here.

Total IWAIT time

Shows an aggregate summation of the Wait times for Sync Read, Prefetch and Async Write, after applying the BUFFHIT hit ratio. It determines the anticipated wait time only for I/O that is physically taking place, and where it is happening (hiperpool or disk).

Wait time for VSAM open/close, binding, locking

Collects the remaining wait events and totals them into a final number which also contributes to the overall elapsed time estimate of Db2 SQL Performance Analyzer. If table spaces and indexes are created with CLOSE YES option, they must often wait for VSAM to physically open the data sets which contain them. Opening data sets takes time, accounted for by Db2 SQL Performance Analyzer.

Similarly, the bind process for dynamically issued SQL, such as SPUFI or QMF, and the impact of lock requests determined from the complete locking strategy is shown.

Elapsed Time

Wait time for the entire query, the **Elapsed Time** depicted in the cost summary block, is a function of the wait times for all physical synchronous read I/O, list and sequential prefetch I/O, asynchronous write I/O, binding, locking, opening and closing data sets and sorting, plus the CPU processing requirements of all steps in the access path.

Sample Db2 SQL Performance Analyzer detail trace report

Detail trace report

ANL3098I *** CONFIGURATION NOTE: The overall rating for this system is totmips Million Instructions/Second Each individual engine will operate at miprate Million Instructions/Second ANL3036I *** The assumed Overhead for Connection Type of attach has been estimated in Millions of Instructions for these BATCH SQL Stmts: Thread Mgmt = thrdpl Attach = atchpl Appl = applpl ANL3037I *** Estimated device DASD Service Times are: Sync Read 4K Page = syncrd Prefetch block of 32 4K Pages = prefrd Sync Read 8K Page = sync08 Prefetch block of 16 8K Pages = pref08 Sync Read 16K Page = sync16Prefetch block of 8 16K Pages = pref16Sync Read 32K Page = sync32Prefetch block of 4 32K Pages = pref32Async Wrt 4K Pages = asyn4kAsync Write of 16K Page Block = asyn16Async Wrt 8K Pages = asyn8kAsync Write of 32K Page Block = asyn32 _____ ROWSPROCESSED =norowsPERCENTTABLEPROCESSED =pctrowCOLSPROCESSED =colsBOOLEANFILTERFROM OPT =boleanDATAPAGESREAD=datapgINDEXEDLEAFPAGESREAD =leaves syncio TABLE = seqprf TABLE = asywrt TABLE = getpag SYSIO = SYNC READ I/OS = tabrio INDEX = idxrio PREFETCH I/OS = ASYNC WRT I/OS = INDEX = tabprf idxprf INDEX = tabasv idxasy GET PAGE CALLS = sysio LOGIO = logio GEI PAGES = syncpl DECOMPRESS= decod FETCH ROW -PREFETCH = prefpl ASYNC WRT = SYSTEMS = syspl LOG WRITE = COMPRESS = encode HIPERPOOL = ASYNC WRT = asynpl LOG WRITE = logpl decode hpool ftchpl LOCK/and so on = oddpl PROCESSES = procpl PREDICATE = npred CLASS 1 = cls1pl OTHER O/H = othrpl THE TOTAL CUMULATIVE PATH LENGTH FOR THIS QUERY IS cumpl M INS. RESULTING IN A TOTAL CPU TIME OF cputim SECONDS CONSUMED OVERALL. DB2 WILL PUT CLASS 1 CPU TIME OF cl1tim SECONDS IN SMF 101 RECORD. clltim SECONDS IN SMF 101 RECORD. db2tim SECONDS IN SMF 100 RECORD. DB2 SHOWS ADDITIONAL CPU TIME OF ESTIMATED TOTAL LOGICAL I/O CALLS = logcio (EXCLUDING SYSTEM) AND ESTIMATED TOTAL PHYSICAL I/O CALLS = physio WITH HIT RATIO = buffhit WAIT TIME FOR SYNC READ I/O = wait1 PREFETCH I/O = wait2 WAIT TIME ON ASYNC WRITE I/O = wait3 TOTAL IWAIT = waitim WAIT TIME FOR VSAM OPEN/CLOSE MACROS, BINDING AND LOCKING = waitd2 QUERY 100000001 WILL REQUIRE elapse SECONDS OF ELAPSED TIME * DURING WHICH cputim SECONDS OF CPU TIME WILL BE CONSUMED AND * A TOTAL OF physic PHYSICAL I/O REQUESTS WILL BE ISSUED TO DISK * * * QUNITS qunits ESTIMATED PROCESSING COST \$ money

The following list is an explanation of all variables estimated in Db2 SQL Performance Analyzer Detail Trace Report:

totmips

The total MIPS rating for the entire processor complex

miprate

The rated millions of instructions per second (MIPS) for a single processor engine

thrdpl

Path length associated with thread management, in millions of instructions

atchpl

The attach facility processor usage, in millions of instructions

applpl

Estimated application program path length, in millions of instructions

device

Type of disk device holding most/all of the database accessed

syncrd

Time to read one 4 KB page from the disk through a synchronous read I/O

sync08

Time to read one 8 KB page from the disk through a synchronous read I/O

sync16

Time to read one 16 KB page from the disk through a synchronous read I/O

sync32

Time to read one 32 KB page from the disk through a synchronous read I/O

prefrd

Time to read a block of "n" 4 KB pages from the disk through sequential or list prefetch I/O

pref08

Time to read a block of "n" 8 KB pages from the disk through sequential prefetch I/O

pref16

Time to read a block of "n" 16 KB pages from the disk through sequential prefetch I/O

pref32

Time to read a block of "n" 32 KB pages from the disk through sequential prefetch I/O

asyn4k

Time to write a block of 4 KB pages to disk through asynchronous write I/O

asyn8k

Time to write a block of 8 KB pages to disk through asynchronous write I/O

asyn16

Time to write a block of 16 KB pages to disk through asynchronous write I/O

asyn32

Time to write a block of 32 KB pages to disk through asynchronous write I/O

norows

Estimated number of rows returned by this query

cols

Estimated number of columns fetched by this query

pctrow

The percentage of rows processed from the table (always given)

bolean

The estimated Boolean factor for the query (only sometimes given)

datapg

Estimated number of data pages to be read from the table

leaves

Estimated number of index leaf pages to be read from the index

syncio

Total number of synchronous read I/Os issued for this step

tabrio

Portion of SYNCIO read from the table

idxrio

Portion of SYNCIO read from the index

seqprf

Total number of sequential and list prefetch I/Os issued for this step

tabprf

Portion of SEQPRF read from the table

idxprf

Portion of SEQPRF read from the index

asywrt

Total number of asynchronous writes issued for this step

tabasy

Portion of ASYWRT written to the table

idxasy

Portion of ASYWRT written to the index

getpag

Total number of Getpages estimated, including dynamic binds

sysio

Estimated system-oriented I/O, for catalog access, binding, and so on.

logio

Estimated log activity for writing updated records

procpl

Path length for most processes in access path, in millions of instructions

cls1pl

Path length attributable to class 1 activities, in millions of instructions

othrpl

Path length absorbed by Db2, in millions of instructions

syncpl

Path length to issue synchronous read I/Os, in millions of instructions

prefpl

Path length to issue sequential and list prefetch I/Os, in millions of instructions

asynpl

Path length to issue asynchronous write I/Os, in millions of instructions

gtpgpl

Path length to issue Getpage requests, in millions of instructions

syspl

Path length accounting for various system activities, in millions of instructions

logpl

Path length to issue log writes, in millions of instructions

ftchpl

Path length to fetch rows and columns, in millions of instructions

oddpl

Path length for application-specific activity, such as cursor manipulation

npred

Number of predicates identified by Db2 SQL Performance Analyzer in SQL statement

decode

Path length to extract rows, using hardware assist or software

encode

Path length to compress data rows before writing out to disk

hpool

Path length for Hiperpool accesses (reads and writes)

cls1pl

PROCPL + GTPGPL + SYNCPL + FTCHPL + SYSPL + ODDPL + DECODE + HPOOL (Reads), essentially all the class 1 processing

othrpl

PREFPL + LOGPL + ASYNPL + ENCODE + HPOOL (Writes) and THRDPL + ATCHPL + APPLPL on first block, representing all the processing absorbed by Db2 on your behalf

totpl

CLS1PL + OTHRPL, represented as a cumulative total

cumpl

TOTPL, total path length as a cumulative total, except where multiple index and subquery are concerned, where this field reflects individual step totals

cputim

TOTPL / MIPRATE (also on cost summary report)

cl1tim

CLS1PL / MIPRATE

db2tim

OTHRPL / MIPRATE

logcio

SYNCIO + SEQPRF + ASYWRT + LOGIO + SORTIO + SQRYIO + JOINIO; essentially the count of all I/Os issued

buffhit

The percentage of I/O satisfied in the buffer pool, negating the need for physical I/O to disk

physio

LOCGIO * (1 - BUFFHIT) (also on cost summary report)

wait1

Synchronous read I/O wait time

wait2

Sequential and list prefetch I/O wait time

wait3

Asynchronous write I/O wait time

waitd2

Other wait time for activities such as binding, locking, and open/close

waitc1

CL1TIM + DB2TIM, or wait time for class 1 activities

waitim

(WAIT1 + WAIT2 + WAIT3) * (1 - BUFFIT)

elapse

WAITIM + WAITC1 + WAITD2 (also on cost summary report)

qunits

A numeric value representation of the cost of the SQL statement, weighted towards the CPU processing cost, and including the I/O drive CPU time and Db2 CPU time (also on Cost Summary Report)

money

The final cost calculated, in monetary terms, based on parameters which price out the CPU, I/O, and elapsed times for the SQL statement (also on Cost Summary Report)

QLIMIT (installation limits) report

The QLIMIT report, which summarizes the entire run into a single record per query is available for SQL PA Batch and TSO users.

You can use the QLIMIT report for audit trail analysis. The QLIMIT report provides a summary to highlight irregular or inefficient queries.

The QLIMIT DD card allocates the report in batch runs while the Query Limits report on the SQL PA Output Data Sets panels do so under TSO. Sample JCL that shows the proper definition of this report is located in the *hiqual*.SANLJCL library, in members *ssid*JBUG, *ssid*JPA, and *ssid*JQST.

Use the output file for your analysis, because it provides a one line cost synopsis of each query evaluated with warning flags. The output file includes the SQL CODES for most statements that SQL PA was not able to evaluate (for example, illegal use of parameter markers or table was not found). When DBRMs are processed, the member name where the statement belongs is populated. The QLIMIT report contains the following information:

5 Warning Flags

Set to Y or N to indicate whether each query has exceeded SQL PA limits for CPU time, elapsed time, physical I/O count, Qunits, or monetary costs, in that order. If there is an error during the processing of a query, the flags are set to E and you can interrogate the next field.

Lines that represent SQL statements and costs are marked with a hyphen (-) if they do not exceed the CPU, I/O, elapsed time, query units, or monetary costs, to make the Y stand out more prominently.

Return Code

Set to 0 if successful, or returns the error code encountered during query processing. This code corresponds to the typical SQL PA error message format: ANL*xxxx*, where *xxxx* is this four-digit return code.

Query Number

Value assigned by SQL PA during processing, to cross reference the query in the SQL PA Summary Cost, Explain, and Detail Trace reports.

Туре

Identifies the statement type. The possible statement type values are shown in the following table.

Table 18. Statement type values						
Value	Туре					
ST	SELECT					
IS	INSERT					
UE	UPDATE					
DE	DELETE					
SD	SELECT with FOR UPDATE OF					
СВ	Correlated subselect or fullselect					
NB	Non-correlated subselect or fullselect					
ТХ	Table expression					
TR	WHEN clause on CREATE TRIGGER					
UN	UNION					
UA	UNION ALL					

CPU Time

Db2 SQL Performance Analyzer CPU time estimate for this query.

Elapsed Time

Db2 SQL Performance Analyzer elapsed time estimate for this query.

I/O Count

Db2 SQL Performance Analyzer physical I/O count estimate for query, after applying the buffer hit ratio.

QUNITs

Db2 SQL Performance Analyzer Qunits estimate for this query.

Monetary Cost

Db2 SQL Performance Analyzer Cost estimate for this query, in the local currency.

Member name

Name for the DBRM members.

Some users have automated their analysis of SQL PA output and might want to scan the QLIMIT output file through programs and take action on overruns. The file format is shown in the following figure.

|CLIM|ELIM|ILIM|QLIM|MLIM|RETCODE|QUERYN0|TYPE|CPUTIME|ELAPSED|IOCOUNT|QUNITS|COSTS|MEM | A1 | A1 | A1 | A1 | A1 | A1 X I4 X I10 X I4 X F11,5 X F11,3 X I10 X I9 XF13,4|XA8

```
where A1 is 1 alphabetic character
X is a blank character
In is Integer of n Digits
Fn,m is Fixed Decimal of n Digits with m decimal places
```

Figure 56. Installation limits file format

A sample QLIMIT file is shown in the following figure.

10:05: 05-23	35.99 -2009	3		SQL Performance Query Limits	e Analyzer Report		Versio Level	n 5.1.0 51-5100	
CEIO\$	Error	Ouervno	Type	CPU Time	Elapsed	Phys I/O	Ounits	Monetary Cost	Member Reason
	0	100005112	UE	0.14755	1,224	18	33	0.2039	
	õ	100012852	ŬĒ	0.01402	1.608	19	4	0.1964	
	Ō	100012907	ŬE	0.01402	1.608	19	4	0.1964	
	0	100005055	ST	0.01296	1.089	18	3	0.1848	
	0	100005217	UR	0.00056	0.460	1	1	0.0114	
	0	100007639	ST	0.00021	0.069	3	1	0.0302	
	0	100007650	ST	0.00021	0.069	3	1	0.0302	
	0	100009083	ST	0.00042	0.119	5	1	0.0504	
	Θ	100009089	ST	0.00042	0.119	5	1	0.0504	
	0	100009100	ST	0.00030	0.119	5	1	0.0504	
	0	100009106	ST	0.00030	0.119	5	1	0.0504	
	0	100009112	ST	0.00026	0.095	4	1	0.0403	
	0	100009118	ST	0.00026	0.095	4	1	0.0403	
	0	100009147	ST	0.00030	0.119	5	1	0.0504	
	0	100009155	ST	0.00030	0.119	5	1	0.0504	
	0	100009162	ST	0.00043	0.096	4	1	0.0403	
	0	100009169	ST	0.00043	0.096	4	1	0.0403	
	0	100009179	ST	0.00029	0.119	5	1	0.0504	
	0	100009186	ST	0.00029	0.119	5	1	0.0504	
	0	100009193	ST	0.00030	0.050	2	1	0.0202	
	0	100009200	ST	0.00030	0.050	2	1	0.0202	
	0	100010502	ST	0.00025	0.092	4	1	0.0403	
	0	100010510	ST	0.00025	0.092	4	1	0.0403	
	0	100010590	ST	0.00030	0.116	5	1	0.0504	
	0	100010604	ST	0.00030	0.116	5	1	0.0504	
	0	100010654	ST	0.00028	0.092	4	1	0.0403	
	0	100010662	ST	0.00028	0.092	4	1	0.0403	
	0	100010747	ST	0.00028	0.116	5	1	0.0504	
	0	100010755	ST	0.00028	0.116	5	1	0.0504	
	0	100010925	ST	0.00024	0.092	4	1	0.0403	
	0	100010934	ST	0.00024	0.092	4	1	0.0403	

Figure 57. Sample QLIMIT file

A header is provided on line 1 of the file to help remind users of the order of contents. SQL PA reports on the limits set during installation, which are used as warning flags for all Db2 SQL Performance Analyzer runs. The QLIMIT file is always created whenever the Explain or Detail Trace reports are requested in either TSO or batch modes.

You can sort the QLIMIT report from the ISPF interface by entering **SORT** n A/D where n is the column name and A or D indicates ascending or descending order. For example, **SORT 3 D** would sort the report in descending order based on the third column from the left, which is Queryno.

Predicate Analysis information

You can add a special predicate analysis report section to the Enhanced Explain and Detail Trace reports.

When operating with the parameter PRECISE ALL specified, SQL PA adds a special predicate analysis report section to both the Enhanced Explain and Detail Trace reports that are created during batch and TSO operations.

The predicate analysis is embedded like the performance notations of the SQL Advisor. The predicate analysis provides information about each predicate being processed, including those predicates generated by the Db2 optimizer for predicate transitive closure. It is no longer necessary to guess whether a predicate is Stage 1 or Stage 2 or what filter factor was calculated: the predicate analysis presents each predicate in a format to help you understand how the optimizer evaluated their predicates and deployed them for access path selection. Figure 58 on page 194 is an example of a Predicate Analysis report for a Cartesian join process.

SOL PA ANALYSIS FOR QUERYNO 100000001 SELECT COUNT(*) FROM SYSIBM.SYSPACKSTMT A, SYSIBM.SYSPACKDEP B WHERE A.LOCATION = ? AND B.DLOCATION = ? AND B.DCOLLID = ? AND A.COLLID <> ? THIS IS A CARTESIAN JOIN, ALL ROWS TO ALL ROWS, NO JOIN PREDICATE. QUERYNO: 100000001 PREDICATE ANALYSIS 1 PREDNO: 1 FILTER: 1.0000000 TYPE: AND **OBLKNO:** JOIN PRED? N ŠTAGE 1? Y BOOLEAN TERM? Y INDEX KEYFIELD? N REDUNDANT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? N LITERALS: LEFT SIDE --> TABNO: 0 BLOCKNO: 0 PREDNO: 0 RIGHT SIDE -> TABNO: 0 BLOCKNO: 0 PREDNO: 5 PSUEDO TEXT: (((A.LOCATION=(EXPR) AND B.DLOCATION=(EXPR)) AND B.DCOLLID=(EXPR)) AND A.COLLID<>(EXPR)) OBL KNO · 1 PREDNO. 2 FILTER: 0.0400000 TYPE: EQUAL JOTN PRED? N ŠTAGE 1? Y BOOLEAN TERM? Y INDEX KEYFIELD? Y REDUNDÅNT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? N LITERALS: HV LEFT SIDE --> LOCATION TABNO: 1 BLOCKNO: 0 PREDNO: 0 RIGHT SIDE -> VALUE TABNO: 0 BLOCKNO: 0 PREDNO: Θ PSUEDO TEXT: A.LOCATION=(EXPR) QBLKNO: 1 PREDNO: 3 FILTER: 0.0400000 TYPE: EQUAL JOIN PRED? N STAGE 1? Y BOOLEAN TERM? Y INDEX KEYFIELD? Y REDUNDANT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? N LITERALS: HV LEFT SIDE --> DLOCATION TABNO: 2 BLOCKNO: 0 PREDNO: 0 RIGHT SIDE -> VALUE TABNO: 0 BLOCKNO: 0 PREDNO: 0 PSUEDO TEXT: B.DLOCATION=(EXPR) QBLKNO: 1 PREDNO: 4 FILTER: 0.0400000 TYPE: EQUAL JOIN PRED? N STAGE 1? Y BOOLEAN TERM? Y INDEX KEYFIELD? Y REDUNDANT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? N LITERALS: HV LEFT SIDE --> DCOLLID TABNO: 2 BLOCKNO: 0 PREDNO: 0 RIGHT SIDE -> VALUE TABNO: 0 BLOCKNO: 0 PREDNO: 0 PSUEDO TEXT B.DCOLLID=(EXPR) QBLKNO: 1 PREDNO: 5 FILTER: 0.9600000 TYPE: EQUAL JOIN PRED? N STAGE 1? Y BOOLEAN TERM? Y INDEX KEYFIELD? N REDUNDANT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? Y LITERALS: HV LEFT SIDE --> COLLID TABNO: 1 BLOCKNO: 0 PREDNO: RIGHT SIDE -> VALUE TABNO: 0 BLOCKNO: 0 PREDNO: 0 **PSUEDO TEXT:** A.COLLID<>(EXPR)



In Figure 58 on page 194, four predicates are ANDed together to access two tables from the catalog. But because there are no common join columns, this join is a Cartesian product join. You can see the format of each query in the PSUEDO TEXT, and the TYPE of predicate (in this case EQUAL), which largely determines its characteristics, such as STAGE 1 and BOOLEAN TERM. The computation of the FILTER factor shows the percentage of table rows that are filtered by this predicate and INDEX KEYFIELD, if Y, indicates that the column is part of an index key. FOR NEGATION, set to Y for Predicate 5, indicates that the predicate is a NOT (in this case <>). LITERALS HV indicates that host variables were used in the predicate and the left and right side, table numbers, and predicate numbers (AND or OR) provide additional information about each predicate in the statement.

Figure 59 on page 195 provides another look at Predicate Analysis, for an OR of two Range predicates.

SQL PA ANALYSIS FOR QUERYNO 100000002 SELECT * FROM TDT690.L1000 WHERE CIKEY < 10 OR CIKEY > 99999 QUERYNO: 100000002 PREDICATE ANALYSIS OBLKNO: 1 PREDNO: 1 FILTER: 0.0199000 TYPE: OR JOIN PRED? N STAGE 1? Y BOOLEAN TERM? Y INDEX KEYFIELD? N REDUNDANT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? N LITERALS: LEFT SIDE --> TABNO: 0 BLOCKNO: 0 PREDNO: 2 RIGHT SIDE -> TABNO: 0 BLOCKNO: 0 PREDNO: 3 PSUEDO TEXT: (L1000.CIKEY<1 OR L1000.CIKEY>9) QBLKNO: 1 PREDNO: 2 FILTER: 0.0100000 TYPE: RANGE JOIN PRED? N STAGE 1? Y BOOLEAN TERM? N INDEX KEYFIELD? N REDUNDANT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? N LITERALS: 1 LEFT SIDE --> CIKEY TABNO: 1 BLOCKNO: 0 PREDNO: 0 TABNO: 0 BLOCKNO: 0 PREDNO: 0 RIGHT SIDE -> VALUE PSUEDO TEXT: L1000.CIKEY<1 3 FILTER: 0.0100000 TYPE: RANGE OBLKNO: 1 PREDNO: JOIN PRED? N STAGE 1? Y BOOLEAN TERM? N INDEX KEYFIELD? N REDUNDANT? N AFTER JOIN? N ADDED BY PTC? N FOR NEGATION? N LITERALS: 9 TABNO: 1 BLOCKNO: 0 PREDNO: TABNO: 0 BLOCKNO: 0 PREDNO: LEFT SIDE --> CIKEY 0 RIGHT SIDE -> VALUE 0 PSUEDO TEXT: L1000.CIKEY>9

Figure 59. Predicate analysis report: OR

The information in the Predicate Analysis report gives SQL PA users an advantage in understanding how each of the predicates are evaluated by Db2. The final composite filter factor for AND predicates is not currently computed (it is always 1.0) but is accurately assessed for OR predicates.

This information is automatically included as part of the Enhanced Explain and Detail Trace reports when PRECISE ALL is chosen.

Host Variables information

When a DBRM is processed as input, SQL Performance Analyzer extracts host variable information from the module. The extracted information displays in both the enhanced explain (ANLREP) and detail trace (QTRACE) reports, immediately following the Predicate Analysis report (if selected).

If there are no host variables, no report is provided for that SQL statement. The host variables are numbered in the order of their appearance in the statement. Key attributes, such as the data type, length, precision and scale (for decimal), input or output, along with the full name of each host variable are displayed. A sample Host Variables report is shown in the following figure.

OUERYNO: 100004702 HOST VARIABLES USED HV 1 DCLIDX.NAME USAGE: OUTPUT DATA TYPE: 448 VAR CHARACTER CCSID: 0 DATA LENGTH: 128 (PRECISION: 0 SCALE: 0 PACKED DECIMAL ONLY) HV 2 DCLIDX.CREATOR USAGE: OUTPUT DATA TYPE: 448 VAR CHARACTER CCSID: 0 DATA LENGTH: 128 (PRECISION: 0 SCALE: 0 PACKED DECIMAL ONLY) HV 3 DCLIDX.TBNAME USAGE: OUTPUT DATA TYPE: 448 VAR CHARACTER CCSID: 0 DATA LENGTH: 128 (PRECISION: 0 SCALE: 0 PACKED DECIMAL ONLY) 4 DCLIDX.TBCREATOR USAGE: OUTPUT DATA TYPE: 448 VAR CHARACTER CCSID: DATA LENGTH: 128 (PRECISION: 0 SCALE: 0 PACKED DECIMAL ONLY) 5 DCLIDX.UNIQUERULE HV USAGE: OUTPUT DATA TYPE: 452 CHARACTER CCSID: 0 DATA LENGTH: 1 (PRECISION: 0 SCALE: 0 PACKED DECIMAL ONLY) HV 6 DCLIDX.COLCOUNT USAGE: OUTPUT DATA TYPE: 500 SMALL INTEGER CCSID: 0 DATA LENGTH: 2 (PRECISION: 0 SCALE: 0 PACKED DECIMAL ONLY) 7 DCLIDX.CLUSTERING HV USAGE: OUTPUT DATA TYPE: 452 CHARACTER CCSID: 0 1 (PRECISION: 0 SCALE: 0 PACKED DECIMAL ONLY) DATA LENGTH:

Figure 60. Host Variables report

Checking host variables

When using host variables, the type and length of each table column is compared to the type and length of the host variable used with the column to ensure they match. However there are some guidelines and limitations in how host variables are checked.

Host variable checking only applies when the host variable information is available, which is when processing packages and DBRMs. Keep this in mind:

- To enable host variable checking for packages, set the HOSTVAR user parameter to YES.
- To enable host variable checking for DBRMs, set the PRECISE user parameter to ALL.

Host variables are checked in the following expressions:

- Host variables used in predicates in the WHERE clause will be checked.
- Host variables that are not part of a more complex expression will be checked. Only the following formats will be checked:

columnname <oper> :hostvariable

and

:hostvariable <oper> columnname

and

columnname BETWEEN :hostvariable1 AND :hostvariable2

where $\langle \text{oper} \rangle$ can be LIKE or any valid combination of \neg , =, \langle , and \rangle .

When checking each host variable, there are three possible outcomes:

1. The host variable is checked against the column and they match each other. No message is printed.

- 2. The host variable is checked against the column and they do not match each other. The ANL1077W message is displayed.
- 3. The host variable is not checked against the column. The ANL1079I message is displayed.

The Plan Table report

An Plan Table report can consist of several pages.

The individual pages are numbered as *x*-*y*, where *x* denotes the report number and *y* is the page number within the report.

If the keyword used is QNO or PKG, the query or statement number is shown at the top of the report. If the keyword is PKG, the DBRM and plan names followed by the actual SQL statement are also shown.

If the QMF keyword is used, the associated SQL statement is shown in the same form in which it was saved in QMF.

If the SQL keyword is used, the specified SQL statement is shown in the same format as you stated.

Each report lists the EXPLAIN-supplied information. This part of the report is a formatted listing of all the data taken from a row in the PLAN_TABLE table at the specific server location. For a description of data produced by the EXPLAIN function, see *Db2 UDB for OS/390 and z/OS V8 Administration Guide*.

The next section of the report shows the access path chosen by Db2.

If an index is used, that is, an access path other than a table space scan, index information is included in the report. This part of the report contains information about the following types of index-related data:

- Full key cardinality
- First key cardinality
- Key range
- Space allocated
- Number of pages used
- Number of levels in the index
- Size of subpage
- Uniqueness of the index
- Close and erase rules
- Buffer pool name
- Cluster ratio
- Clustering
- · Prefetch factor
- I/O factor
- Index Type
- Piecesize

If an index is used, the next section of the report lists all the key columns in the index. If a matching index scan is used in the access path, the report indicates the columns that are used in the index scan. Used columns are indicated by an arrow (<===) in the Key Used column. The number of arrows corresponds to the contents of the matching columns field in the EXPLAIN data.

If LEVEL=KEYdist has been specified and RUNSTATS has produced key distribution information for the first column in the index in SYSIBM.SYSCOLDIST, a section in the report contains the distribution of the 10 (or fewer) most frequently used key values.

The next part of the report shows information about the accessed table and its corresponding table space. The information included in this part is:

Name of table

- Number of rows and columns in the table
- Row length
- Name of Edit and Validation Procedures
- Name of table space and data base
- Name of default storage
- Creator of table and table space
- Number of tables in the table space
- · Size of the page in the table space
- Number of active pages in the table space
- Number of pages with rows
- · Percent pages used in the active part
- Number of table space partitions
- Bufferpool for table space
- Status of the table
- Status of table space
- ID for data base, table space, and table
- Close and erase rules
- Auditing rule
- Segment size
- Lock rule
- · Percentage of rows compressed
- Row level locking

If table space scan was selected as the access path method, Plan Table prints a list of all available indexes on the subject table. This list includes the column names on which the indexes are built (a maximum of 12 columns, or the number of names that fit on the print line is listed).

If a query number was specified using the QNO (or QUERYNO) keyword for a miniplan created by the BIND process, only the last section of the report is shown. This section lists the following bind information:

- · Bind date and time
- Isolation level
- Acquire and release points
- Plan validity
- Plan binder and owner
- Validation time
- Plan size
- Cache size
- Number of plan and system entries
- Number of SQL statements in plan
- Name of current server
- Degree of parallelism
- Disconnect option
- SQL rules

This part of the report is only produced with the first SQL statement of the plan.

If you are authorized to store and list plan information, the report shows the items in the previous list for the last four generations of the plan. DBRM and package information as outlined in the following list is also included in the report:

- DBRM/Package name
- Precompile date and time
- Source language
- Number of SQL statements
- Characterset
- Use of comma
- Use of Decimal(31)
- Source type (DBRM or Package)
- Db2 release when precompiled

After normal completion, the last page of the output shows a listing of all the input commands. Each command is prefixed with the report number. For each SQL, QUERYNO, and QMF statement, the access path is listed next to the input request. For each Plan or Package statement, the access path is listed for every SQL statement in the plan. To the right of the access path information, the report page number (*x-y*) is shown, unless the Level is Summary. In this way, you can use the summary page as a table of contents.

The top of the summary page contains a listing of the Db2 environment. This listing includes information such as:

- Level of Plan Table, Db2, and MVS
- Db2 subsystem ID
- Location name
- MVS system ID
- Job name (if batch)
- · Batch or TSO
- Start and stop time
- Run time

For each SQL statement in the plan or package, the summary page shows if the application module is a DBRM (D) or package (P) (see a in the following figure), if a List Prefetch (L) or Sequential Prefetch (S) scan was used (see b in the figure), and if a table space scan (*) or non-matching index scan (+) was selected by Db2 (see c in the figure).



Figure 61. Application module summary

Topics:

- <u>"Package reports" on page 200</u>
- "Remote packages reports" on page 202
- "Explain report review" on page 203
- "Explain tabular report reference" on page 205

Package reports

There are a variety of package reports available.

When a plan is bound, you can include a package list that consists of wildcard characters, for example COLLID3.* or *.*. However, using wildcard characters can result in a plan pointing at thousands of packages with an even greater number of explainable SQL statements.

To prevent packages with SQL statements from being explained unnecessarily, Plan Table looks for generic package lists in the plan being explained. If the collection ID or the package name in a package list was specified as an asterisk (*), Plan Table determines the number of packages that belong to the package list specification.

If more than 100 packages belong to the package list specification, a report with all the package names is produced, but no SQL statements in these packages are explained. The collection IDs are not listed. If any DBRMs belong to the plan, SQL statements in these DBRMs are explained.

The plan TESTPLAN was bound with a package list of *.*, a plan statement therefore results in the report shown in the following figure.

2005-04-30 *** Explained EXPLAIN Information *** Report 1 Information for Plan: TESTPLAN *** Package Report *** Page 1-001									
Package Location: DALLAS-CQ Explain Location: CPHMVS1_DB2X									
ANL164I Co Us	ANL164I Collection specification *.* has more than 100 packages Use PACKAGE=collid.name(version) to get a detailed listing								
DB2ASTM I DSNQVCDA I DSNQVIND I DSNQVPAS I DSNQVFBP I E23SERV E CURRENA (CURREN1 (CURREN1 (CURREN9 A CURRENE (DB2ADMP I SQLUD1A0 S P152FA3 I	DMSQC DSNQVCOL DSNQVPLN DSNQVPLN DSNQVSQL DSNQVTBS E23SERV CURRENB CURRENB CURREN2 ANLECH CURRENE DB2ASTM SQLUE1A0 JFI	DSNESM68 DSNQVDAS DSNQVKAS DSNQVFA DSNQVUAS E23MAIN CURRENC CURREN3 E23MAIN ANLECH DB2ASTM SQLUF1A0 DB2PGM2	DSNESM68 DSNQVDBR DSNQVKGE DSNQVKGE DSNQVSTO DSNQVVOL E23PLAN CURREND CURREN4 E23MAIN LONGVER DB2ASTM SQLUG1A0 LONGVER	DSNQVALI DSNQVDBS DSNQVKGL DSNQVKAL DSNQVTAB E31MAIN E23SERV CURRENE CURRENE CURREN5 E23PLAN UFI DB2ASTM UFICMDS	DSNQVAUD DSNQVFOR DSNQVKGV DSNQVKGV DSNQVTAS E31SERV E23MAIN CURRENF CURRENF CURREN6 E23SERV DB2ASTM DB2ASTM DB2PGM2	DSNQVAUT DSNQVIEW DSNQVOBG DSNQVRIL DSNQVTBA E31SERV E23PLAN CURRENT CURRENT CURRENT LONGVER DB2ASTM P152FA1	DSNQVCAS DSNQVINA DSNQVOBJ DSNQVROO DSNQVTBC E23MAIN E23SERV CURREN0 CURREN8 NEGZER0 DB2ASTM DB2ASTM P152FA2		

Figure 62. TESTPLAN package report

If more than 10 but less than 100 packages belong to the package list specification, a report with all the package names is produced, but no SQL statements in these packages are explained. This report shows the collection IDs, the creator and owner names, the version identifications, and the dates of precompilations. If any DBRMs belong to the plan, SQL statements in these DBRMs are explained.

The plan LARGPLAN was bound with a package list of E23PACK.*, a plan statement therefore results in the report shown in the following figure.

2005-04-30*** Explained EXPLAIN Information ***Report 1Information for Plan: LARGPLAN*** Package Report ***Page 1-004								
Package L Explain L	Package Location: DALLAS-CQ Explain Location: CPHMVS1_DB2X							
ANL164I	ANL164I Collection specification E23PACK.* has more than 10 packages Use PACKAGE=collid.name(version) to get a detailed listing							
Package	Collection Id.	Creator	Owner	Exp	PC-Date	Version		
E23MAIN E23MAIN E23MAIN	E23PACK E23PACK E23PACK E23PACK	DPGROTH DPGROTH DPGROTH	DPGROTH DPGROTH DPGROTH	Yes Yes Yes	1999-12-20 1999-12-19 1999-12-19	1999-12-20-14.49. 1999-12-19-20.12. 1999-12-19-16.31.		
•••	•••	• • •	• • •	• • •	•••			
E23MAIN E23MAIN E23MAIN E23MAIN	E23PACK E23PACK E23PACK E23PACK	DPGROTH DPGROTH DPGROTH DPGROTH	DPGROTH DPGROTH DPGROTH DPGROTH	Yes Yes Yes Yes	 1999-12-04 1999-12-03 1999-12-02	1999-12-04-19.16. 1999-12-03-08.28. ANL420.5		
E23MAIN	E23PACK	DPGROTH	DPGROTH	Yes	1999-11-23	ANL420.4		
E23MAIN	E23PACK	DPGROTH	DPGROTH	Yes	1999-11-13	ANL420.2		
E23PLAN	E23PACK	DPGROTH	DPGROTH	Yes	1999-12-04			
E23SERV	E23PACK	DPGROTH	DPGROTH	Yes	1999-12-03	1999-12-03-08.12.		
E23SERV	E23PACK	DPGROTH	DPGROTH	Yes	1999-12-02	ANL420.5		
E23SERV	E23PACK	DPGROTH	DPGROTH	Yes	1999-11-25	ANL420.4		
LONGVER	E23PACK	DPGROTH	DPGROTH	No	1999-11-06	THIS_IS_A_VERY_LO		
NEGZERO	E23PACK	DPGROTH	DPGROTH	Yes	1987-10-27			
SQLUF1A0	E23PACK	UK11241	UK11241	NO	0001-01-01			
UFICMDS	E23PACK	ISIJE2	ISIJE2	NO	0001-01-01			

Figure 63. LARGPLAN package report

When a package is being explained, you might specify the collection ID, the package name, or both as generic names using an asterisk (*) as a wildcard character. For example, you might specify:

Package=e23*.* Package=e23pack.* Package=e23pack.e23* Package=e23pack.e23main

If a wildcard character is used, Plan Table counts the total number of SQL statements that belong to the packages (the total number of SQL statements also includes non-explainable statements).

If the number of packages is more than one and the total number of SQL statements in these packages is larger than 300, Plan Table does not explain any SQL statements unless the FORCE=YES subparameter is specified. Instead the mini report shown in the following figure is produced.

```
2005-04-30 *** Explained EXPLAIN Information *** Report 1
Page 1-001
Package Location: DALLAS-CQ
Explain Location: CPHMVS1_DB2X
ANL372W The statement: PKG=E23PACK.E23*
will result in 25 packages
with a total of 2,042 SQL statements (including non-explainable).
If you want to explain that amount of packages, you should specify:
PKG=E23PACK.E23*,FORCE=Yes
```



If the package specification does not contain any wildcard characters, the total number of SQL statements is less than 300, or only one package conforms to the specification, the SQL statements are explained. However, if the package exists in more than one version, only the most recent version is explained. In this case, a report showing all the versions for that package is produced. If you specify Package=e23pack.e23main(-7), gen=3 the report shown in the following figure is produced:

*** Explained EXPLAIN Information *** 2005-04-30 Report 2 Information for Package: E23MAIN Page 2-000 *** Version Report *** Package Location: DALLAS-CO Explain Location: CPHMVS1 DB2X ANL166I Package: E23MAIN in Collection: E23PACK has the following versions Pre-Comp'd Exp Gen Version Identification 1999-12-20 Yes 0 1999-12-20-14.49.03.726556 1999-12-19 Yes -1 1999-12-19-20.12.46.296787

 1999-12-19
 Yes
 -1
 1999-12-19-20.12.46.296787

 1999-12-19
 Yes
 -2
 1999-12-19-16.31.06.369415

 1999-12-19
 Yes
 -3
 1999-12-19-16.15.48.278743

 1999-12-18
 Yes
 -4
 1999-12-18-09.17.18.591934

 1999-12-17
 Yes
 -5
 1999-12-17-09.16.06.101113

 1999-12-17
 Yes
 -6
 1999-12-17-08.44.59.541029

 1999-12-16
 Yes
 -8
 1999-12-16-12.04.03.828613

 1999-12-16
 Yes
 -8
 1999-12-16-12.04.03.828613

 1999-12-15 Yes -9 1999-12-15-11.11.59.937500 1999-12-15 Yes -10 <version id not specified> 1999-12-09 Yes -11 1999-12-09-10.15.10.307455 1999-12-08 Yes -12 1999-12-08-11.10.46.584507 1999-12-08 Yes -13 1999-12-08-10.47.06.119000 1999-12-08 Yes -14 1999-12-08-10.31.08.242248 1999-12-05 Yes -15 1999-12-05-19.01.27.836759 1999-12-04 Yes -16 1999-12-04-19.16.42.852327 1999-12-03 Yes -17 1999-12-03-08.28.25.916853 1999-12-02 Yes -18 ANL420.5 1999-11-13 Yes -19 ANL420.4 1999-11-13 Yes -20 ANL420.2 Start version generation number specified: -7 Number of version generations requested: 3

Figure 65. Package version report

The listed report shows that the package exists in 21 versions, where the version that corresponds to generation number -7 was selected for explanation. The report further shows that you have asked for explanation of three generations (-7, -8, and -9). Instead of specifying a generation identification, the version identification could be specified. You can either specify the version ID in full or in combination with a wildcard character, as shown in the following example:

Package=e23pack.e23main.(1999-12-17-08.22.*),gen=3

A maximum of 100 generations are listed.

Remote packages reports

An application program can consist of DBRMs and packages. The packages are located on the same Db2 subsystem as the plan, but the packages can also be distributed to one or more remote locations.

The following figure shows an application plan named XYZ located at Location_A, which is the home server location. The application plan consists of two DBRMs plus a package list. This package list contains pointers to local and remote packages.

In the example, the package list points to the remote package Package_A, which is located at server Location_C. Package_B is a local package, and Package_C and Package_D are remote packages located at Location_B.

To explain all the static SQL statements for plan XYZ, Plan Table automatically connects to all the locations where the plan has remote packages if the locations were specified at Bind time. In this way, Plan Table reports explain data and catalog data for all static SQL statements in the plan XYZ.

For example, you might want only to explain Package_D at Location_B. By using the location subparameter on the Package statement, you can instruct Plan Table to explain that package at the specified location.



Figure 66. Application plan example

All the explain reports and package reports show the location name for the individual SQL statements and packages.

Explain report review

If an application does not achieve the expected performance, investigate the individual Explain report pages carefully.

Performance analysis is a complex task. It is important to know if the environment where you are running Explain for data analysis is the production environment, or a test environment. The main performance factors to review include the following items, but other factors might be the cause for your performance issues:

Access Path Chosen

Typically, avoid table space scan and non-matching index scans, unless you intend to access all rows in a given table. If one or more indexes exist for the table, try to reconstruct the SQL statement so that a better access path is chosen by Db2. Alternatively, you could consider creating an index.

Also, try to avoid internal sorts. If you are joining, avoid a merge scan join because a sort is involved.

Clustering versus Clustered

If a clustering index was selected by Db2, the Explain report shows if the actual index also is clustered and the clustering ratio is shown. If the clustered value is N (or the cluster ratio is less than 95%), the table space might need reorganization to bring the data rows into clustering sequence again.

Number of matching columns

If a matching index scan is selected by Db2, verify that the number of columns used in the index is what you expect. Locate the <=== arrow under the Key Used heading in the index section. Every

column used in the index search is marked with <===. The access path box and the Summary Page show the ratio between number of matching columns and number of key columns in the index.

Tables per tablespace

In the table space section, locate the heading Tabls/TS. This value gives the number of tables in the table space. Normally, you would expect to have only one table in the table space.

Active pages versus pages with rows

If you are doing table space scans, verify that the number of pages with rows (heading Pages w/Rows in the table section) is about the same size as the value under Pages Active. The value under heading PctPages in the table section needs to be as close to 100% as possible.

If the table space scan takes place for a non-segmented table, and more than 100 pages with no table rows are scanned, and this number of pages is more than 10% of the total table space pages, the number of pages is shown in the access path box.

The following example is the first page of an Explain report where you are authorized to store and list plan information (PLANINFO=YES).

Information for plan: BOOKKEEP Page 4-000									
Package Location: DALLAS-CQ Explain Location: CPHMVS1_DB2X Changes									
Creator: Bind Date: Bind Time: Bound by: Qualifier: Base Size: Avg. Size: Cachesize: Plan Entr: Sys. Entr: SQL Stmts:	DPGROTH 2016-12-30 12:27:01.13 DPGROTH DPGROTH 2,234 3,165 1,024 2 0 11	DPGROTH 2016-08-21 10:31:14.54 DPGROTH 2,521 3,165 1,024 2 0 11	DPGROTH 2016-08-01 20:26:47.11 DPGROTH DPGROTH 4,408 2,936 2,048 2 0 5	DPANL 2016-06-27 14:21:38.32 DPANL DPGROTH 2,096 2,936 1,024 2 0 5	<==== <==== <==== <==== <====				
Validate: Isolation: Valid: Operative:	Bind Cur.Stab. Yes Yes	ind Bind Jr.Stab. Cur.Stab es Yes es Yes		Run Cur.Stab. Yes Yes	<=== <===				
Acquire: Release: Curr.Data: Deferprep: Curr.Serv:	Use Commit Allow Block Yes CPHMVS1_DB2X	Use Commit Allow Block Yes CPHMVS1_DB2X	Allocate Deallocate Inhibit Blk Yes CPHMVS1_DB2X	Use Commit Inhibit Blk Yes CPHMVS1_DB2X	<==== <==== <===				
I/O Dgree: SQL Rules: Disconnet:	Enable Par. DB2 Explicit	Enable Par. DB2 Explicit	Disable Par. DB2 Explicit	Disable Par. Old Plan Explicit	<=== <===				
Stored by: DPGROTH DPGROTH DPANL DPGROTH <=== Stored at: 2016-12-30 2016-08-28 2016-08-09 2016-06-27 <===									
DBRM/Pack PC-Da	DBRM/Pack PC-Date PC-Time Lang. SQLstmt Charset Comma Dec31 Type Rel.								
BKDEBET 2015-03-15 09:26:14 PL/I 6 Alphanum No N/A DBRM 5.1 BKCREDT 2016-10-16 17:23:52 PL/I 5 Alphanum No Yes PCKG 6.1									

Figure 67. Explain report

After explaining an application using Explain where the access path and other factors are satisfactory, save the data for historical purposes. The historical analysis data gives you a base for comparing past and present performance. Save the Explain report together with the application documentation.

The report contains columns with plan data from the last four plan generations. If there is a discrepancy in the column data within a row, a marker (<===) is shown in the rightmost column.

Look for changes in the number of SQL statements, validate, isolation, acquire, and release. A change in any of these areas might influence the performance of the application plan.

The bottom part of the subreport shows data for each DBRM and package in the most recent plan generation. The report states the precompile date and time, the programming language, the number of SQL statements, single or double byte character set, use of comma, use of decimal(31), type of source, and Db2 release indicating when the module was precompiled.

Often an inconsistent definition of host variables compared to the corresponding column definition results in an inefficient access path selection.

In the example shown in the following figure, the access path selected is Table Space Scan even though an index is defined on the only column referenced in the WHERE-clause. By

specifying ...HOSTVAR=Yes,INDEX=T,... on the plan or package statement, Explain shows both the index and the host variable definitions. As the example shows, Db2 has selected Table Space Scan because the column definition is 3 characters (see 1), but the corresponding host variable is defined as 4 characters (see 2). By changing the host variable definition to 3 characters, a matching index scan is selected by the Db2 Optimizer.

```
DECLARE C1 CURSOR WITH HOLD FOR
SELECT DEPTNO, DEPTNAME, LOCATION
FROM DSN8610.DEPT
WHERE DEPTNO = :HOSTVAR_STRUCTURE.DEPARTMENT_NUMBER
ORDER BY DEPTNO
 . . . . . . .
 Table space scan - no index will be used
 Standard sequential PREFETCH will be performed
 Lock mode is Share Lock for the page
 . . . . . . .
Key
                                          Key
                                                                              Key
No.Column NameCol.Type Lng NullCard. Order Low2keyHigh2key1DEPTNOCHAR3No14Asc.c'B01c'I22
                                                                             Used
                                 1
 . . . . . . .
Host var. type Length Ind. Host variable name
                                                              (DB2 rel. 6.1)
Fixed character 4 No HOSTVAR_STRUCTURE.DEPARTMENT_NUMBER
                    2
```

Figure 68. DBRM and Package generation data

Note: If a host variable has a related indicator variable, the Ind. column shows a Yes for the host variable, and the following line in the host variable report shows the indicator variable.

Explain tabular report reference

The Explain report is available in a shortened format that can be specified by setting LEVEL=TABular.

Columns

If LEVEL=TABular has been specified, the report shows information about the accessed table and its corresponding table space. The information includes the following columns:

- Query number
- Block number
- Plan Number

- Creator
- Creator of table and table space
- Table name
- Access type
- Matching columns
- Multiple index operation sequence
- Index only
- Sort unique
- Sort join
- Order by
- Group by
- Merge
- Tablespace lock mode
- Prefetch
- Column function evaluation
- Degree of access
- Access parallel group ID
- Join degree
- Join parallel group ID
- Parallel group ID for the parallel sort of the composite table
- Parallel group ID for the parallel sort of the new table
- Parallelism mode
- Page range
- · When optimize
- Query block type
- Primary access type
- Table type

Sample

1201 1 FIL SQL	12016-08-24 *** Explained EXPLAIN Information *** 1 PI12345 5.1.00 FILE= userid.SQL.INPUT(TSTINPUT) SQL Statement for Query No.100000001							Repor Page	t 1 1-001	
SEL FRC WHE	ECT * M ANLU RE NAM	====== SER0.EI E = 'F:	EEPLAN INDNAME	====== E'	:======					
Bre	eakdown	of EX	PLAIN :	informat	ion fo	r Query	No.100	900001		
=== 0PLA	N_TABL	====== E Data	 :			======			=====	======
Blk No.	Pln A No. T	Mat Mi Col S	IX IPDF eq ORRF	PCPTW TS EMTO LO	SORT k UJOG	C M PID 	SORTI UJOGM I	N Access PID Deg PI	Joi D Deg	.n PID
1	1 I	0 (ЭN	DT 1	S NNNN	N -	NNNNN		-	-
Ту	/pe=SEL	ЕСТ								
==== IX	IX=JOEUSER.X_E3PLN1									
Col Cnt	UCCSH RGDPA	Nleaf	Nlvls	Clu F Rat Key	irst card	Full Keycard	Avg Keyle	n Statstim	е	
2	PNYNN	32	2	96 (53	2252	21	2010-05-	13-11.	08.50
TB=ANLUSER0.EEEPLAN RowLen Col Rows Npages Max Avg Count Statstime -1 -1 -1 0001-01-01-00.00.00 -1 -1 TS=JOEUSER.ANLSPACE Partitions Nactive Statstime -----0 -1 0001-01-01-00.00.00 Key Key Key No. Column Name Col.Type Lng Null Card. OPGMH Low2key High2key Usd --- ---- ---1 NAME VARCHAR 128 No 169 A c'AASFS c'XSR_REGI <== 2 PARM_COUNT SMALLINT 2 No 21 A x'8001 x'8029 Information for ALL available indexes is requested by the user: IX=JOEUSER.X_E3PLN1 IX=JOEUSER.X_E3PLN2 U Col C C Clu First Full Avg R Count G D Nleaf Nlvls Rat Keycard Keycard Keylen Statstime 4 Y Y -1 -1 0 -1 -1 -1 0001-01-00.00.00 1 N N -1 -1 0 -1 -1 -1 0001-01-01-00.00.00 N 4 Y Y Ν Key Key Key No. Column Name Col.Type Lng Null Card. OPGMH Low2key High2key Usd CHAR 8 No FOR CHAR 8 No ---- ------1 Asc -1 Asc -1 Asc -1 Asc -1 Asc 1 NAME N/A N/A 2 CREATOR N/A N/A 2 CREATOR CHAR 8 NO 3 BINDDATE CHAR 6 No 4 BINDTIME CHAR 8 No N/A N/A N/A N/A --------- -----1 NAME CHAR 8 No -1 Asc N/A N/A 1SQL Performance Analyzer Plan Table Report Summary Page OANLRelease:5.1.0ODB2Release:10.1.5MVSRelease:7.1.1ODB2System id:DSNBUserAuth.id:JOEUSERCurr.SQL id:JOEUSERDeclaseDeclaseDeclase of 2013-10-23 APAR: PI12345 of 2013-10-23 APAR. F1123-5 DB2 Macro Change Level: V10R1M0 z/OS System id: SY4A Product Code: 5740-XYR TS0 Plan Name: ANLEEE Package Location: Explain Location: DALLAS-CQ CPHMVS1_DB2X Coded Char. Set Id: 37 OStart date: 2009-08-24 Start time: 10:22:49 Stop time: 10:22:49 Stop date: 2009-08-24 Elapse time: 00:00:00 0 1 SQL, PKG, PLN, QNO, and QMF requests were processed: Page No. Level=TABULAR, Index=Y, Acctype=A, HostVar=Y Format=Y, PlnInfo=Y, SQLPA=Y, SQLIs=079 Default-Options: DSN8EXP=N Options: SET CURRENT DEGREE = '1 Current Degree set to '1' SET CURRENT SQLID = 'JOEUSER ' Current Authorization Id. specified 0 1: QNO=100000001 Matching index scan (1/4) - data pages scan 1-001 -Start time: 10:22:49, Stop time: 10:22:49, Date: 2016-08-24 OProgram Property of IBM --- (c) Copyright IBM 1986-2017

Explain condensed report reference

The Explain report is available in a shortened format that can be specified by setting LEVEL=CONdensed. The condensed report is similar to the tabular report.

Information included in the report

If LEVEL=CONdensed has been specified, the report shows information about the accessed table, its related indexes, corresponding table space, and statistical information for the query. The report includes the following information:

Abbreviation used in the report	Explained information
Blk No.	Block number
Pln No.	Plan Number
AT	Access type
Mat Col	Matching columns
MIX Seq	Multiple index operation sequence
IO	Index only
PR	Page range
DR	
PF	Prefetch
CE	Column function evaluation
PM	Parallelism mode
ТТ	Table type
WO	When optimize
TS Lck	Table space lock mode
SORTC U	Composite table, sort unique
SORTC J	Composite table, sorted for join method
SORTC O	Composite table, sorted for ORDER BY or quantified predicate
SORTC G	Composite table, sorted for GROUP BY
SORTC M	Composite table, merge before join
SORTC PID	Composite table, parallel group ID
SORTN U	New table, sort unique
SORTN J	New table, sorted for join method
SORTN O	New table, sorted for ORDER BY or quantified predicate
SORTN G	New table, sorted for GROUP BY
SORTN M	New table, merge before join
SORTN PID	New table, parallel group ID
Access Deg	Number of parallel operations that are activated by the query
Access PID	Parallel group ID for accessing the new table

Included EXPLAIN information:

Abbreviation used in the report	Explained information
Join Deg	Number of parallel operations that are used in joining the composite table with the new table
Join PID	Parallel group ID for joining the new table with the composite table.
Qblock Type	Query block type
Correl Name	Correlation name
Rows	Number of rows returned by the query
Npages	Number of pages on which rows of the table appear
RowLen Max	Maximum row length returned by the query
RowLen Avg	Average row length returned by the query.
Col Count	Number of columns in the table
Statstime	Date and time the statistics were gathered or updated

Included index information:

Abbreviation used in the report	Explained information
Col Cnt	Number of columns included in the index
UR	Whether the index is unique
CG	Whether the index is specified as the clustering index for the table
CD	Whether the table is actually clustered by the index
SP	Whether the index is sparse
НА	Whether the index is the hash overflow index for a hash table
Nleaf	Number of active leaf pages in the index partition
Nlvls	Number of levels in the index tree
Clu Rat	Percentage of rows that are in clustering order
First Keycard	Number of distinct values of the first key column
Full Keycard	Number of distinct values of the key
Avg Keylen	Average length of keys in the index
Statstime	Date and time the statistics were gathered or updated
Key No.	Numeric position of the key-target in the index
Column Name	Name of the column of the key
Col. Type	Data type of the column of the key
Lng	Length attribute of the column of the key
Null	Whether the key can contain a null value
0	Order of the key (whether the values are in ascending, descending, or random order)
Р	Whether the column is the start or end column for the BUSINESS_TIME period

Abbreviation used in the report	Explained information
G	Indicates the generated attribute of the column
М	
Н	Whether the column is implicitly hidden
Low2key	The second lowest key-value
High2key	The second highest key-value
Key Usd	An indication of whether the key is currently in use for the index

Condensed report sample

12014-09-17 *** Pla	n Table *** 6 4.2.14	Report 1												
1 EXPLAIN Information for Statement	1551 in PCKG: ANLSMP i	Page 1-001 n Plan:												
OSQL statement read from SYSIBM.SYS	STMT:													
DECLARE GETCOL CURSOR FOR SELECT * FROM SYSIBM.SYSCOLUMNS WHERE TBCREATOR = :H AND TBNAME = WITH UR	:н													
Breakdown of EXPLAIN information for	or Query No. 1551													
Powl op Col		Blk Pln	A Ma	t MIX IPDPCP	TW TS	SORTC	SO	RTN Ac	cess Join	Qblock Co	orrel			
Table Count Statstime	Index	No. No.	T Co	l Seq ORRFEM	TO Lck UJ	OGM PI	D UJOG	M PID Deg	PID Deg PI	D Type Na	ame	Rows N	bages	Max Avg
	SYSTEM DSNDCX05	 1 1	т	 2	 T N NN	 NNN -		 N				682046	····· ·	 167
158 41 2014-05-22-19.54.04			-											
Information for ALL available inde	xes is requested by the us	er:												
0 IX=SYSIBM.DSNDCX01 Col UCCSH Clu First Cot RGDPA Nleaf Nlvls Rat Keycard	Full Avg Kevcard Kevlen Statstim	e	Key No.	Column Name	Col.Type	Lng Nu	Ke ull Ca	y rd. OPGMH	Low2kev	High2kev	Key Usd			
3 PNNNN 9006 4 95 98	4 682046 26 2014-05-	22-19.54.04		TBCREATOR	VARCHAR	128 No	0 0 0	984 A	c'%1	c'qWENDY	<==			
			3	NAME	VARCHAR	128 No	o 51	200 A	c'#LOGRBA	c'ts1	<			
0 IX=SYSIBM.DSNDCX05 Col UCCSH Clu First Cnt RGDPA Nleaf Nlvls Rat Keycard	Full Avg Keycard Keylen Statstim	e	Key No.	Column Name	Col.Type	Lng Nu	Ke ull Ca	y rd. OPGMH	Low2key	High2key	Key Usd			
2 DNNNN 4472 3 95 984	4 233671 19 2014-05-	22-19.54.04	1	TBCREATOR	VARCHAR	128 No	 D 	984 A	c'%1	c'qWENDY	<==			
0+	referenced data pages The index has 2 columns d (Cluster ratio: 95%) t: 'A' level is "Uncommitted Read	+ 												
1SQL Performance Analyzer Plan Table	e Report S	ummary Page												
OANLRelease:4.2.14ODB2Release:10.1.5MVSRelease:7.2.1ODB2System id:DSNUserAuth. id:JSMITHCurr. SQL id:JSMITHPackageLocation:DSNCoded Char. Set Id:37OStart date:2014-09-17Stopdate:2014-09-17	of 2014-09-15 APAR: DB2 Macro Change Level: z/OS System id: Product Code: TSO Plan Name: Start time: 09:57:17 Stop time: 09:57:17 Elapse time: 00:00:04	PI24056 V10R1M0 SY4A 5740-XYR ANLEEE												
0 1 SQL, PKG, PLN, QNO, and QMF re	quests were processed:	Page No.												
Default- Level=CON, Inde Options: Format=Y, PluTons: SQLID=JSMITH Curren SET CURRENT DEGREE = '1 0 1: PKG=ANLSMP.ANLSMP.ELCC=,PLOC 0 DBRM/Pack Stmt Typ Pack o COLT: 0 COLT: 0 Vers: J	x=A, Acctype=A, HostVar=Y fo=Y, SQLPA=Y, SQL1s=079 t Authorization Id. specif t Degree_set to '1' ,FORCE=Y wner: JSNITH Bind date: Time: 11.09.38 PCom date: ANLSMP Loc: DSN CONDENSE	ied 2014-09-17 2013-08-22												
ANLSMP 1551P Matchi -Start time: 09:57:17, Stop time: 0 OProgram Property of IBM	ng 1X scan (2/2), data pg 9:57:21, Date: 2014-09-17 (c) Copyright IBM	scan 1-001 1986-2013												

The CSV report

The CSV report can be enabled so that the information can be imported into a spreadsheet or other program that recognizes the comma separated values format.

If the CSV report is enabled, it will be generated according to the current settings for each EXPLAIN that is run. The CSV report can then be downloaded and imported to a spreadsheet.

Chapter 7. Troubleshooting SQL PA

Use this information to diagnose and correct problems that you experience with Db2 SQL Performance Analyzer.

Gathering diagnostic information

Before you report a problem with SQL Performance Analyzer to IBM Software Support, you need to gather the appropriate diagnostic information.

Procedure

Provide the following information for all SQL Performance Analyzer problems:

- A clear description of the problem and the steps that are required to re-create the problem
- All messages that were issued as a result of the problem
- Product release number and the number of the last program temporary fix (PTF) that was installed
- The version of Db2 that you are using and the type and version of the operating system that you are using

Provide additional information based on the type of problem that you experienced:

For online abends, provide the following information:

- A screen shot of the panel that you were using when the abend occurred
- The job log from the TSO session that encountered the abend
- The job log from the server
- · A description of the task that you were doing before the abend occurred

For errors in batch processing, provide the following information:

- The complete job log
- Print output
- · Contents of the any data sets that were used during the processing

Resetting the Db2 SQL Performance Analyzer REGISTRY

Db2 SQL Performance Analyzer includes a cleanup mechanism that resets all the SQL PA entries in the REGISTRY to their original status.

About this task

Occasionally, an entry might have an invalid status of in use in the Db2 SQL Performance Analyzer REGISTRY. The Db2 SQL Performance Analyzer REGISTRY table can be reset if entries are incorrectly marked as in use.

The Db2 SQL Performance Analyzer REGISTRY table contains an entry for each Db2 SQL Performance Analyzer generic plan table, holding an Db2 SQL Performance Analyzer secondary authorization ID, a place for the primary authorization ID of the user (set to ANL99999), an in use flag (Y or N) to help avoid conflicts, and a timestamp indicating last use.

By registering a PLAN_TABLE as in use, Db2 SQL Performance Analyzer gives exclusive control of that table to a specific user, for a portion of the time Db2 SQL Performance Analyzer is processing. The table is released as soon as it is no longer required, before the end of the Db2 SQL Performance Analyzer process.

Tip: To clean up any fragmented or frozen entries in the SQL PA REGISTRY, consider adding *ssid*RSET to your job scheduler so that it runs periodically. Rarely, users might receive an error message that about

there being no unused secondary authorization IDs (ANL1008E).Db2 SQL Performance Analyzer contains code to detect when the REGISTRY is full, and automatically resets any entries that are more than 24 hours old.

Procedure

To run the cleanup mechanism:

- 1. Locate and run *hiqual*.SANLSQL, member *ssid*RSET.
 - *ssid*RSET is designed to run under TSO SPUFI, but you can also run it from a batch job.
- 2. Add *ssid*RSET to your job scheduler so that it runs periodically.

TSO interfacing notes for installers

The following topics contain special TSO interfacing notes for installers.

Topics:

- "Db2 SQL Performance Analyzer variables in the ISPF user profile" on page 214
- "Host variable support from any TSO input medium" on page 214
- "SQL PA supports upper and lower case" on page 214

Db2 SQL Performance Analyzer variables in the ISPF user profile

Db2 SQL Performance Analyzer makes liberal use of ISPF profile variable storage, and keeps the latest values for the TSO interface parameters there. Therefore, the latest values are retained between runs.

Host variable support from any TSO input medium

Db2 SQL Performance Analyzer allows you to specify host variable names in SQL submitted for analysis under TSO (and QMF), in addition to their typical use within application programs, and subsequent DBRMs.

Under normal circumstances, an EXPLAIN of an SQL statement under TSO or QMF would not allow use of application host variables (:Host), preferring the parameter marker (?) in its place. SQL PA adjusts automatically, so that if you want to isolate some SQL from a COBOL program, for example, SQL PA explains it and performs a cost analysis, replacing the :Host variables with the parameter marker automatically as it processes.

SQL PA supports upper and lower case

Db2 SQL Performance Analyzer handles SQL in both upper and lower case. All lower case values get translated to upper case before processing unless the NLSCODE parameter is in effect (DBCS or Mixed Data names). SQL PA's translation to upper case avoids any problems that might occur because of lower case Db2 object names.

Applying user indexes to tables

To improve Plan Table performance, apply user indexes to three system catalog tables.

Procedure

Create the indexes shown in the following table in ASC order.

Table 19. Syste	Table 19. System catalog table indexes			
Index Name	System Catalog	Table Column names		
XUSRDBRM	SYSIBM.SYSDBRM	(PLNAME, NAME)		

Table 19. System catalog table indexes (continued)				
Index Name	System Catalog	Table Column names		
XUSRFLDS	SYSIBM.SYSFIELDS	(TBCREATOR, TBNAME, NAME, FLDPROC)		
XUSRSTMT	SYSIBM.SYSSTMT	(PLCREATOR, PLNAME, NAME, STMTNO, SEQNO)		

Chapter 8. Tools Customizer troubleshooting

Use this information to diagnose and correct problems that you experience with Tools Customizer.

Gathering diagnostic information

Before you report a problem with Tools Customizer to IBM Software Support, you need to gather the appropriate diagnostic information.

Procedure

Provide the following information for all Tools Customizer problems:

- A clear description of the problem and the steps that are required to re-create the problem
- Relevant screen captures
- All messages that were issued as a result of the problem
- Product release number and the number of the last program temporary fix (PTF) that was installed
- The version of Db2 that you are using and the type and version of the operating system that you are using
- The Tools Customizer trace data set
- The Tools Customizer data store data set and the high_level_qualifier.SCCQTENU data set

Determining the trace data set name

You will need to identify the name of the trace data set if you cannot allocate the trace data set, the trace data set runs out of space, or IBM Software Support asks for it.

The name of the trace data set depends on the prefix setting in the TSO profile. To identify the name of the trace data set, you must know the prefix setting.

- If PREFIX is set, the name of the trace data set is *prefix*.CCQ.TRACE, where *prefix* is the TSO prefix that you specified in the profile.
- If NOPREFIX is set, the name of the trace data set is *user_ID*.CCQ.TRACE, where *user_ID* is your TSO user ID.

Chapter 9. How to look up message explanations

You can use several methods to search for messages and codes.

Searching an information center

In the search box that is located in the top left toolbar of any Eclipse help system, such as the IBM Information Management Software for z/OS Solutions Information Center, enter the number of the message that you want to locate. For example, you can enter DFS1065A in the search field.

Use the following tips to improve your message searches:

- You can search for information on codes by entering the code; for example, enter -327.
- Enter the complete or partial message number. You can use the asterisk wildcard character (*) to represent multiple characters, and you can use the question mark wildcard character (?) to represent a single character.

The information center contains the latest message information for all of the Information Management products that are included in the information center.

Searching for messages on the web

You can use any of the popular search engines that are available on the web to search for message explanations. When you type the specific message number or code into the search engine, you are presented with links to the message information in IBM information centers.

Chapter 10. Reference information

This section contains reference information for SQL PA.

Topics:

- "ANLCNTL configuration parameters" on page 226
- "ANLPARM user parameters" on page 230
- "PLAN_TABLE contents" on page 246
- "Migration from IBM Db2 Path Checker to SQL Performance Analyzer" on page 246

Tools Customizer reference

Before you use Tools Customizer, you should understand the Tools Customizer terminology and the data sets that Tools Customizer uses during customization.

Tools Customizer terminology and data sets

Before you use Tools Customizer, you should understand the Tools Customizer terminology and the data sets that Tools Customizer uses during customization.

Tools Customizer terminology

Tools Customizer uses several unique terms that you should be familiar with before you begin to use Tools Customizer.

Products and components

How an IBM Tool is packaged determines whether it is referred to as a product or as a component in the Tools Customizer documentation and interface. An IBM Tool that is ordered as a stand-alone entity (that is, not as part of a solution pack) is referred to as a product. An IBM Tool that is part of a solution pack is referred to as a component. Some IBM Tools are available in both formats; therefore, the same IBM Tool can be referred to as a product or as a component depending on how it is packaged.

Db2 entry

You can customize Db2 SQL Performance AnalyzerDb2 Autonomics Director on one or more Db2 entries. A Db2 entry can be any of the following items:

Db2 subsystem

A distinct instance of a relational database management system (RDBMS) that is not part of a data sharing group. An example of a Db2 subsystem name is DB01.

Db2 group attach name

The name that is used by the TSO/batch attachment, the call attachment facility (CAF), DL/I batch, utilities, and the Resource Recovery Services attachment facility (RRSAF) as a generic attachment name. An example of a group attach name is DSG1.

Db2 data sharing member

A Db2 subsystem that is assigned by the cross-system coupling facility (XCF) to a data sharing group. An example of a Db2 data sharing member name is DB02.

Tools Customizer maintains the following lists of Db2 entries:

Associated list

The list of Db2 entries that are associated with Db2 SQL Performance AnalyzerDb2 Autonomics Director. If the product to be customized requires Db2 entries, you can customize Db2 SQL Performance AnalyzerDb2 Autonomics Director only on Db2 entries that are in the associated list. When you customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, this list is displayed in the DB2 Entries, Associations, and Parameter Status section of the **Customizer Workplace** panel.

You can add and copy Db2 entries to the associated list. When you add or copy Db2 entries to the associated list, the entries are associated with Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Master list

The list of all Db2 entries that are defined but are not associated with Db2 SQL Performance AnalyzerDb2 Autonomics Director. Tools Customizer obtains information about these Db2 entries either from entries that were created manually or from the customizations of other products that were discovered. If you remove a Db2 entry from the associated list, the Db2 entry is added to the master list. When you create a new Db2 entry, it is added to the master list, and when you associate the new entry with Db2 SQL Performance AnalyzerDb2 Autonomics Director, it is removed from the master list and added to the associated list. The master list is displayed on the **Associate a DB2 Entry for Product** panel.

If the associated list does not have the Db2 entries on which you want to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, you can associate existing entries from the master list to the associated list.

You can create new Db2 entries and copy existing entries to the master list.

High-level qualifier

The high-level qualifier is considered to be all of the qualifiers except the lowest level qualifier. A high-level qualifier includes a mid-level qualifier.

Product parameters

Parameters that are specific to Db2 SQL Performance AnalyzerDb2 Autonomics Director. These parameters are defined by Db2 SQL Performance AnalyzerDb2 Autonomics Director and are stored in a data member that is defined by Db2 SQL Performance AnalyzerDb2 Autonomics Director.

LPAR parameters

Parameters on the local LPAR that are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director. These parameters are defined by Tools Customizer and are stored in an LPAR parameter data member.

Db2 parameters

Parameters for a Db2 entry. These parameters are defined by Tools Customizer and are stored in a Db2 parameter data member.

Status type

Product, LPAR, and Db2 entry status type

After you specify the product that you want to customize, the product, the LPAR, and the Db2 entries have a status. The status is partly based on whether required parameters are defined. For some products, LPAR parameters or Db2 parameters might not be required. In these cases, the status is Not Required.

To customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, all of the required parameters must be defined.

If required parameters for the the product parameters, LPAR parameters, or Db2 parameters are not defined, the status of the parameters is Incomplete. Define values for parameters by manually editing them or by generating the customization jobs and specifying values for all of the required parameters that are displayed on the panels.

When values for all of the required parameters are defined, the status is Ready to Customize. Customization jobs can be generated only when all of the required parameters are defined and the status is Ready to Customize or Customized for the product parameters, LPAR parameters, and Db2 parameters for the Db2 entries on which Db2 SQL Performance AnalyzerDb2 Autonomics Director will be customized. The following table shows the meaning of the status types. Each status is defined differently for each type of parameter.

Status	Product	LPAR	Db2 entries
Incomplete	The required product parameters are not defined.	The required parameters are not defined.	The required parameters are not defined.
Discovered	The product parameter definitions were discovered by using the product Discover EXEC.	N/A	N/A
Ready to Customize	The required product, is Ready to Customize or Customized for the LPAR and at least one associated Db2 entry. You can generate the customization jobs.	The required LPAR parameters are defined or LPAR parameters are not required.	The required Db2 parameters are defined or Db2 parameters are not required.
Verify Values	The required product or component parameter values are defined but they either have not been verified or verification is not enabled on the Product or Component Parameters panel.	The required LPAR parameter values are defined, but they either have not been verified or verification is not enabled on the LPAR Parameters panel.	The required Db2 parameter values are defined, but they either have not been verified or verification is not enabled on the Db2 Parameters panel.
Customized	The jobs are customized on the local LPAR.	The jobs are customized for the product or for all of the associated Db2 entries on the local LPAR.	The jobs are customized for the Db2 entry.
Errors in Customization	N/A	N/A	Errors occurred while the customization jobs were being generated.
Not Required	N/A	LPAR parameters are not required.	Db2 parameters are not required.

Table 20. Status types for the product, the LPAR, and the Db2 entries

Related tasks

Creating and associating Db2 entries

You can create new Db2 entries and associate them with Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Copying Db2 entries

You can copy associated and not associated Db2 entries to other Db2 entries or to new Db2 entries.

Removing Db2 entries

You can remove Db2 entries from the associated list.

Data sets that Tools Customizer uses during customization

Tools Customizer uses several unique data sets during the customization process. Familiarize yourself with these data sets before you begin to use Tools Customizer.

Several different data sets are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director with Tools Customizer. These data sets are supplied by Db2 SQL Performance AnalyzerDb2 Autonomics Director, supplied by Tools Customizer, or allocated by Tools Customizer.

Db2 SQL Performance AnalyzerDb2 Autonomics Director provides the following data sets:

Metadata library

Contains the metadata for the product to be customized. Tools Customizer uses the metadata to determine which tasks, steps, and parameters to display on the **Product Parameters** panel, the **LPAR Parameters** panel, and the **DB2 Parameters** panel. This data set also contains the templates that Tools Customizer uses to generate the customization jobs.

The metadata library naming convention is *high_level_qualifier*.SANLDYXDENU, where *high_level_qualifier* is all of the segments of the data set name except the lowest-level qualifier.

You specify the metadata library on the **Specify the Metadata Library** panel. READ access to this data set is required.

Discover EXEC library (CLIST library)

Contains the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC. When you customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, you can use the Discover EXEC to automatically retrieve and store product information, such as parameter values from an already customized product. Tools Customizer saves the discovered information in the data store.

The default name of the data set is the high-level qualifier for the metadata library plus a lowest-level qualifier. For Db2 SQL Performance AnalyzerDb2 Autonomics Director, the lowest-level qualifier is SANLDYXCLSTDENU. You can change the default value on the **Discover Customized Product Information** panel. EXECUTE access to this data set is required.

Tools Customizer provides the following data sets:

Tools Customizer metadata library

Contains the metadata for the Db2 and LPAR parameters that are required to customize Db2 SQL Performance AnalyzerDb2 Autonomics Director. Tools Customizer uses the metadata to determine which parameters to display on the **DB2 Parameters** panel and the **LPAR Parameters** panel. In addition, Tools Customizer uses information in the metadata library to determine whether additional Db2 and LPAR parameters need to be displayed on these panels. As you customize different products, different Db2 and LPAR parameters might need to be defined.

The default name of the data set is DB2TOOL.CCQ110.SCCQDENU. You can change the default value on the **Tools Customizer Settings** panel. READ access to this data set is required.

Tools Customizer table library

Stores information about jobs that are customized. Job information that is stored includes a description of the job, its member name and template name, the SSID, group attach name, and when the job was generated.

The default name of the data set is DB2TOOL.CCQ110.SCCQTENU. WRITE access to this data set is required.

Tools Customizer requires that the following data sets exist during the customization process. If the data sets do not exist, Tools Customizer automatically allocates them.

Discover output data set

Contains the output that is generated when you run the Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC. The Db2 SQL Performance AnalyzerDb2 Autonomics Director Discover EXEC retrieves the metadata and values for the parameters from a previous customization of Db2 SQL Performance AnalyzerDb2 Autonomics Director. The default name of the data set is DB2TOOL.CCQ110.DISCOVER. You can change the default value on the **Tools Customizer Settings** panel or the **Discover Customized Product Information** panel. WRITE access to this data set is required.

Data store data set

Contains product, LPAR, and Db2 parameter values, and Db2 entry associations. Tools Customizer uses this data set to permanently store all information that is acquired about the product, Db2 subsystems or data sharing groups, and LPAR when you customize products on the local LPAR.

The default name of the data set is DB2TOOL.CCQ110.DATASTOR. You can change the default value on the **Tools Customizer Settings** panel. WRITE access to this data set is required.

Customization library

Contains the customization jobs that Tools Customizer generates for Db2 SQL Performance AnalyzerDb2 Autonomics Director.

Tools Customizer checks whether a customization library name was specified for more than one instance of the same version of the same product. If the same customization library name is specified for more than one product of the same version, the CCQD123E message is issued to prevent you from overwriting previously generated customization jobs. Ensure that you specify unique qualifier for the customization library for each instance of the product.

To customize Db2 SQL Performance AnalyzerDb2 Autonomics Director, submit the members of the data set in the order in which they are displayed on the **Finish Product Customization** panel.

The data set naming convention is *hlq*.\$LPAR_name\$.xyzvrm, where:

- *hlq* is the value of the **Customization library qualifier** field on the **Tools Customizer Settings** panel (CCQPSET)
- LPAR_name is the four-character LPAR name
- *xyzvrm* is the three-letter product identifier with the version, release, and modification level

For example, the data set name might be DB2TOOL.PRODUCT.CUST.\$MVS1\$.XYZ410.

WRITE access to this data set is required.

Tools Customizer allocates the data sets for the discover output, the data store, and the customization library with the attributes that are shown in the following table:

Table 21. Data set attributes for allocating the Discover output, data store, and customization library data sets

Data set	Organization	Record format	Record length	Block size	Data set name type
Discover output data set	PO	Variable block	16383	32760	LIBRARY
Data store data set	PO	Variable block	16383	32760	LIBRARY
Product customization library	PO	Fixed block	80	32720	LIBRARY

Restrictions:

• Multiple users cannot simultaneously share the discover output data set, data store data set, Tools Customizer metadata library, and metadata library.

ANLCNTL configuration parameters

The following information describes the ANLCNTL configuration parameters. An example of member ANLCNTL with recommended parameter values is provided after the parameter descriptions.

Descriptions of ANLCNTL parameters

The ANLCNTL parameters are not normally modified by Db2 SQL Performance Analyzer users.

The following parameters define a target host configuration. You can have multiple configurations (PDS members) in batch using the ANLCNTL DD statement to point to the *hiqual*.SANLDATA PDS member, and in TSO by having multiple members in the *hiqual*.SANLPARM PDS. For each member (subsystem, LPAR, or configuration), configure the following parameters of ANLCNTL:

AUTHIDO

The Db2 user ID is used to customize SQL PA. This ID owns the SQL PA objects that are created for the customization and must have SYSADM authority. The default value is ANLUSER0.

BUFF08K nnnnnnn

Set the BUFF08K system parameter to the total number of 8 KB buffers in the 8 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Use the most prominent 8 KB buffer pool here, even if several are used.

The default value is 00000500 buffers.

BUFF16K nnnnnnn

Set the BUFF16K system parameter to the total number of 16 KB buffers in the 16 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Use the most prominent 16 KB buffer pool here, even if several are used.

The BUFF16K default value is 00000250 buffers.

BUFF32K nnnnnnn

Set the BUFF32K system parameter to the total number of 32 KB buffer s in the 32 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Use the most prominent 32 KB buffer pool here, even if several are used.

The default value is 00000100 buffers.

BUFFERS nnnnnnn

Set the BUFFERS system parameter to the total number of 4 KB buffers in the 4 KB buffer pool of the dominant application. This value is used to set the number of pages per block for sequential prefetch and other I/O calculations. Specify the most prominent 4 KB buffer pool here, even if several are used.

The default value is 00002000 buffers.

COMMENT remarks

You can insert a comment anywhere in a parameter file. All remarks are transferred to the output report. To use this parameter, type COMMENT followed by your remarks.

COSTING nnnnnnn

The COSTING system parameter indicates the maximum monetary value, in your currency, that is permitted before the SQL statement is flagged as exceeding the monetary limit. The maximum value is 999999999. A setting of 0 results in no monetary limit.

The default value is 15.

COSTQUN nnnnnnn

The COSTQUN system parameter indicates the maximum number of QUNITS that are permitted before the SQL statement is flagged as exceeding the QUNIT limit. The maximum value is 999999999. A setting of 0 results in no QUNIT limit.

The default is 200.

Reminder: CPUTIME, ELAPSED, IOCALLS, COSTING, and COSTQUN are limits for these resources. SQL PA flags all activity that exceeds the limits set by these parameters. If no parameters are specified, SQL PA assumes that there is no limit for that category.

CPUCOST nnnn.nnn

Set the CPUCOST system parameter to the cost of one hour of CPU time, in the national currency, for this configuration. This parameter is used by SQL PA for charge back and financial cost determination. SQL PA

The default value is 500 per hour for processor time.

CPUTIME nnnnn

The CPUTIME system parameter indicates the maximum CPU time, in seconds, that is permitted before the SQL statement is flagged as exceeding the time limit. The maximum value is 86400, which is equivalent to 24 hours. A setting of 0 results in unlimited CPU time.

The default value is 10.

CURRSYM a

Set the CURRSYM system parameter to the single character symbol that represents the national currency that corresponds to the MONEYIS parameter.

CPUCOST, IOSCOST, and TIMCOST are used to calculate the total monetary value of a transaction, based on your charge back costing guidelines. Setting one or more of these parameters to zero, or leaving out, changes the cost algorithm accordingly. For example, if your installation does not charge for connect time, the costing would reflect only processor and I/O charges.

The default value is the dollar sign, \$.

DATAGRP nnnnnnn

The DATAGRP system parameter indicates the name of the user group authorized to view user table data. If the value of DATAGRP is set to +0FF+, the feature is disabled and all users will be able to view the user table data.

The default value is +0FF+.

DSGROUP nn

The DSGROUP system parameter indicates the number of members in the data sharing group. This parameter is used to adjust the overall processing power of the system to reflect excess processing time caused by data sharing. The result is a more accurate cost assessment.

The default is 0, no data sharing.

DYNAMIC YES | NO

Set the DYNAMIC system parameter to specify whether dynamic statement caching is used. Caching dynamic SQL statements can reduce the excess processing time associated with preparing dynamic SQL for subsequent operations. The values are YES (the system uses Dynamic Statement Caching) or NO.

The default value is NO.

ELAPSED nnnnn

The ELAPSED system parameter indicates the maximum elapsed time, in seconds, that is permitted before the SQL statement is flagged as exceeding the elapsed time limit. The maximum value is 86400 which is equivalent to 24 hours. A setting of 0 results in unlimited elapsed time. For example, a value of 300 would be equivalent to 5 minutes (5 minutes * 60 seconds/minute = 300 seconds).

The default value is 60.

FASTTBL YES | NO

Use this parameter to indicate whether to bypass the explain plan processing. When FASTTBL is NO and the TABLES function is chosen, EXPLAIN is run on the statements. When FASTTBL is YES, and the TABLES function is chosen, processing bypasses the EXPLAIN, and uses the catalog queries to gather the needed information.

IOCALLS nnnnnnn

The IOCALLS system parameter indicates the maximum physical I/O call s that are permitted before the SQL statement is flagged as exceeding the physical I/O limit. The maximum value is 999999999. A setting of 0 results in unlimited physical I/O.

The internal default value is 1000. The maximum value is 99999999.

IOSCOST nnnn.nnn

Set the IOSCOST system parameter to the cost of 1000 I/O calls, in the national currency, for this configuration. This parameter is used by SQL PA for charge back and financial cost determination.

The default value is 10.00 per 1000 I/O.

MONEYIS aaaaaaaa

Set the MONEYIS system parameter to the national currency, in descriptive form, such as DOLLARS, POUNDS, STERLING, DRACHMA, or KRONA.

The default value is DOLLARS.

SETPLAN NOTALLOW | ALLOW | GENERIC

Note: Prior to Version 5.1, the only values accepted were YES and NO, with a default of NO. These values will still be processed if already specified as a parameter in a previous version of SQL PA.

- If the SETPLAN value is set to ALLOW (or YES prior to V5.1), the SETPLAN system parameter authorizes the use of the USEPLAN user parameter to indicate the qualifier of the plan table.
- If SETPLAN is set to NOTALLOW (or NO prior to V5.1), the value of USEPLAN should be +0FF+. The plan table is qualified by a generic ID or the user's TSO ID if no generic ID is available.
- If SETPLAN is set to GENERIC, only a generic ID will be used to qualify the plan table.

The default value for SETPLAN is NOTALLOW. The default value for USEPLAN is +OFF+. That value (or blanks) also negates the effects of USEPLAN, regardless of the SETPLAN setting. SETPLAN is an optional solution intended for sites which restrict PLAN_TABLE access. When possible, use the generic ANLUSER1-n tables for processing.

The parameter is set in the data set identified by the ANLCNTL DD statement in batch and by the system parameters data set identified on the "Parameter Data Sets" panel for TSO. The default USEPLAN value is +0FF+, and it is not activated unless the SETPLAN ALLOW (or YES prior to V5.1) parameter is also included in the system parameters data set member. If you implement this option for all TSO users, then the SANLPARM members, which constitute the DB2 Target Host selections, must be modified to include SETPLAN ALLOW (or YES prior to V5.1).

Note:

1. SETPLAN is an optional solution for those sites that need it. When possible, use the generic ANLUSER1-n tables for their processing.

To comply with your security guidelines for SQL PA generic IDs, you might want to create an ALIAS for the generic ANLUSER1-n tables.

2. The default value for USEPLAN is +0FF+. That value (or blanks) also negates USEPLAN's effects, regardless of the SETPLAN value.

The default value is NOTALLOW.

SUBSYST aaaaaaaa

This parameter specifies the Db2 subsystem name, up to eight characters long, on this machine. Normally, Db2 is called DSN on most processors, but it is possible to run several copies of Db2 on the same machine. If you run more than one copy of Db2, you must define a separate configuration file for each Db2 subsystem. It is not necessary to specify SUBSYST if your Db2 subsystem's name is DSN.

The default value is DSN.

SUBVERS aaaaaaaa

This parameter represents the Db2 version that you are using at your installation. Acceptable values are V8COM, V8NFM, V9COM, and V9NFM. V#COM and V#NFM differentiate between conversion mode and new function mode. This parameter is primarily used in the TSO Target Hosts library's members, *hiqual*.SANLPARM, to allow the ANLALL CLIST a way to select alternate Db2 subsystems or program modules for a single TSO installation that accesses multiple Db2 subsystems at different Db2 levels.

It is also used to select the appropriate catalog access host variables for optimum performance and must always be specified.

TIMCOST nnnn.nnn

Set the TIMCOST system parameter to the cost of one hour of connect time, in the national currency, for this configuration. This parameter is used by SQL PA for charge back and financial cost determination.

The default value is 10 per connect hour.

Note: When PRECISE YES is used, only the processor time provided by the optimizer is used for cost assessment. I/O and elapsed time are free in this case.

USRPARM YES | NO

The USRPARM system parameter authorizes the use of the user-level user parameters.

The default value is NO.

Example of recommended ANLCNTL parameters based on customer usage

The following example shows the ANLCNTL member with the default values and suggested values, based on actual customers, in the COMMENT lines. While the values specified in the comments have been recommended, you must decide what is best for your environment.

```
COMMENT ***** PARAMETERS DISCOVERED DURING INSTALL ******************
COMMENT SUBVERS and SUBSYST are mandatory and cannot be changed
SUBVERS V11NFM
SUBSYST DB2P
COMMENT The bufferpool parameters are not used with PRECISE ALL or YES
COMMENT BUFFERS 2000
COMMENT BUFF08K 500
COMMENT BUFF16K 250
COMMENT BUFF32K 100
COMMENT DYNAMIC YES
COMMENT AUTHIDO is mandatory and cannot be changed
AUTHIDO DB2SYS
COMMENT DATAGRP depends on security choice
DATAGRP +OFF+
COMMENT DBTRACE should be OFF unless directed by IBM L2.
DBTRACE OFF
COMMENT SETPLAN depends on security choice.
SETPLAN ALLOW
COMMENT USRPARM recommended value is YES to allow users to have
COMMENT their own copy of settings.
USRPARM YES
COMMENT FASTTBL recommended value is YES, but does not show
COMMENT locking for the TABLES command.
FASTTBL YES
COMMENT ****** ADJUST THE CPU OVERHEAD IN A DATA SHARING ENV ********
COMMENT DSGROUP not used with PRECISE = ALL or YES.
COMMENT DSGROUP 2
COMMENT ***** SET COST LIMITS USED FOR WARNING MESSAGES *************
COMMENT Thresholds used to trigger flags on the query limits report.
COMMENT Set at your discretion or use a value of 0 to turn off. CPUTIME 10 \,
COSTING 15
```

Chapter 10. Reference information 229

ANLPARM user parameters

The following information describes the ANLPARM user parameters. An example of member ANLPARM with recommended parameter values is provided after the parameter descriptions.

ANLPARM parameter descriptions

When running in batch mode, these parameters are stored in the file specified in the ANLPARM DD statement. They are expected to be modified by most SQL PA users, customized to each particular run. When using TSO, these parameters can be found in the SQL PA user parameter file of the Parameter Data Sets panel. Most of these parameters are used during the online execution, but a few of them are overridden by the values specified on the panels during invocation, such as DBRMKEY, DEGREES, QUALIFY, and USEPLAN.

ADVISOR YES | NO | ALL

The ADVISOR user parameter displays additional information about the performance and design of each query and about the tables and indexes that are accessed by the query. Warnings and Alerts are always displayed, even when ADVISOR is set to NO. Notes and Recommendations are displayed when ADVISOR is set to YES, which can reveal potential performance tuning opportunities. A value of ALL is ideal for those who want to receive guideline messages and confirmation that tasks have concluded successfully.

The SQL Advisor writes to the Explain and Detail Trace files, and is integrated with the TSO and batch reporting structure.

The default value is NO.

BUFFHIT nnn

The BUFFHIT parameter indicates the ratio of physical to logical I/O, expressed as a percentage from 0 to 100. Specify a value for the percentage of pages that are found in the buffer pool (that is, for which disk I/O is not necessary).

The default value is 000, to indicate that 100% of I/O goes to DASD.

COMMENT remarks

You can insert a comment anywhere in either parameter file. All remarks are transferred to the output report. To use this parameter, type COMMENT followed by your remarks.

CONNECT aaaaa

The CONNECT user parameter specifies the expected application connection to DB2 when the SQL is started in production. This value helps SQL PA account for excess processing that is caused by various attach facilities, including thread management and application processing. The acceptable values are:

- IFP (IMS Fast Path)
- WFI (IMS Wait For Input)
- MPR (IMS Message Processing Region)
- BMP (IMS Batch Message Program)
- CICS (Online CICS/VS)
- SPUFI (Online TSO SPUFI)

- QMF (Online TSO QMF)
- DL1 (DL/1 Batch Program)
- CAF (Batch Call Attach Facility)
- DSN (Batch TMP with DSN CLIST)
- RRSAF (Recoverable Resource Manager Services Attach Facility)
- DRDA (remote connections)
- NONE (do not include any attach processor usage in cost)

The default value is CAF.

CSVDCSP COMMA | DOT

The CSVDCSP user parameter determines the decimal point separator. Enter COMMA to use a comma as the decimal point separator. Enter DOT to use a dot as the decimal point separator. If the CSVDLIM user parameter has been set to COMMA, enter DOT.

The default value is DOT.

CSVDLIM COMMA | SEMICOL

The CSVDLIM user parameter determines the CSV delimiter character. Enter COMMA to use a comma as the CSV delimiter character. Enter SEMICOL to use a semicolon as the CSV delimiter character. If the CSVDCSP user parameter has been set to COMMA, enter SEMICOL.

The default value is COMMA.

CSVHDRW COL | LBL | +0FF+

The CSVHDRW user parameter determines the first row of the CSV formatted report. Enter COL to use the column names as the column headers. Enter LBL to use the user defined column labels specified in the column selection file as the column headers. Enter +0FF+ to start with report data in the first row and not have a header row. The default is +0FF+.

CSVRPT NO | YES

The CSVRPT user parameter determines whether the CSV formatted report is generated. Enter YES to generate the CSV formatted report based on the CSV column selection file. Enter NO to not generate the CSV formatted report.

The default value is NO.

CSVSEL aaaa

The CSVSEL user parameter qualifies the CSV column selection file name which has the DB2 subsystem ID as the prefix to show the compatible DB2 version. This file is a list of columns and column headers to be included in the CSV report. The columns will appear in the CSV report from left to right in the order of the column list.

DBRMKEY search-pattern

The DBRMKEY parameter is only applicable when using the batch interface. To process multiple DBRMs using the ISPF interface, specify the search-pattern in the DBRM member name field on the **Process DBRMs** panel.

The DBRMKEY parameter provides a search pattern for selecting multiple DBRMs from the DBRM library pointed to by the ANLIN DD statement in a batch run. You can process up to 200 DBRMs in a single run.

The search pattern can contain up to 8 characters, including the wildcard characters, a percent sign (%) and asterisk (*), as shown in the following examples:

bbbbbbbb

A value of all blanks causes all members in the library to be selected.

*

An asterisk (*) in the first position also causes all DBRMs in the library to be selected.

ABC*

Causes all members whose names start with ABC in the first three characters to be selected. Other character positions in the names are not relevant.

member

Causes the DBRM member whose name is member to be selected from the library.

%%%ABC%%%

Causes members whose names have any character in the % positions and ABC in the third, fourth, and fifth positions to be selected. The length of a member name *must match exactly* the length of the pattern. For example, with DBRMKEY *%%4%%*, the DBRM named ABC4SSS would be selected, but a DBRM named ABC4QQ would not be selected.

+OFF+

This special DBRMKEY value turns off the processing of multiple DBRMs, while allowing the DBRMKEY parameter to remain in the user's parameter list.

When using DBRMKEY, the following four conditions are required to process multiple DBRMs in a batch run of SQL PA:

- 1. The DBRMKEY parameter must be present with a value other than +0FF+.
- 2. An ANLWORK DD statement must be in the batch JCL with DCB=(RECFM=FB, LRECL=80, BLKSIZE=4000) to provide temporary workspace.
- 3. The ANLIN DD statement must allocate the required DBRM library, such as DSN=MY.DBRMLIB, DISP=SHR.
- 4. All DBRMs to be processed must exist in the DBRM library specified in ANLIN.

Note: When processing multiple DBRMs, do not specify a member name in the ANLIN DD statement unless the DBRMKEY value is +0FF+. Otherwise the results are unpredictable.

When not using DBRMKEY, the following three conditions are required to process multiple DBRMs in a batch run of SQL PA:

- 1. If the DBRMKEY parameter is present, it must have a value of +OFF+.
- 2. The ANLIN DD statement must allocate a fully qualified DBRM.
- 3. All other DBRMs to be processed must be specified as fully qualified DBRMs, each using its own ddname of ANLIN00 through ANLIN99, inclusive.

DEGREES ANY | ONE | 1

Set the DEGREES user parameter to specify whether parallel processing is considered during the statement evaluation when the batch interface is used. Set DEGREES to ANY when parallel processing is enabled. Set DEGREES to ONE when parallel processing is not enabled.

ANY

If set to ANY, SQL PA informs the optimizer that parallel processing is an option.

ONE |+ 1

If set to ONE or 1, SQL PA tells Db2 to ignore any parallel cases, even if the query could benefit from parallel processing (a table space scan against a partitioned table, for example).

The default value is 1.

DELIMIT +OFF+| QUOTE

COBOL programs have mutually exclusive precompile options: QUOTESQL or APOSTSQL, which determine whether the SQL uses double quotation marks (") around literal strings instead of single quotation marks or apostrophes ('). Only the COBOL language permits this choice, and most Db2 sites standardize using the single quotation mark during installation. However, if you compile COBOL programs with the double quotation mark delimiter in this environment, Db2 Prepare/Explain rejects the SQL with -206 SQL Code, making it unrecognizable, and thus SQL PA would be unable to provide a cost.

If you choose to use double quotation marks to delimit SQL, and the -206 SQL Code is returned by Db2, you can correct the situation by using the DELIMIT QUOTE parameter. This user parameter instructs SQL PA to transform and temporarily replace double quotation marks with single quotation marks for the Prepare/Explain process, eliminating the -206 error from Db2. This parameter does not change or affect your SQL in any way. Only the processing under c is modified. SQL with single quotation marks are unaffected by this parameter, can be intermixed, and process normally.

The default value is +OFF+.

DSN8EXP YES | NO

Set the DSN8EXP user parameter to specify whether EXPLAINs are run by using the DB2-supplied stored procedure (DSNAEXP) or by embedded SQL. If the value is set to YES, DSNAEXP is used if it is installed and available; otherwise, DSN8EXP is used.

The default setting is NO.

DSPPARM YES | NO

The DSPPARM user parameter determines whether the SQL PA parameters and their values are displayed at the top of the enhanced explain and detail trace reports.

The default setting is YES.

DSPVARS YES | NO

The DSPVARS user parameter determines whether the host variable name s or parameter markers are displayed in SQL statements in the cost reports.

Restriction: DSPVARS is limited to SQL statements with a length less than 32 720.

The default value is NO.

DSPWCOC YES | NO

For DBRMs, specifies whether WHERE CURRENT OF statements are shown in the Query Limits report. The WHERE CURRENT OF clause can falsely cause QUNIT values that exceed limits set for acceptable query statements. Set this parameter to NO, to have WHERE CURRENT OF statements excluded from the Query Limits report.

The default value is YES.

HPOOLRD nnn

Beginning with Version 8, Db2 does not support Hiperpools – if specified, it is ignored.

INLISTS nnn

The INLISTS user parameter specifies the limit of the number of elements that are shown in an IN (LIST) predicate. If the maximum number of elements is exceeded, Db2 SQL Performance Analyzer issues a notification message.

The default value is 10 elements.

ISCANPG nnnnnnn

The ISCANPG user parameter indicates the limit of the number of index leaf pages that are read with a matching index scan. If the maximum number of pages is exceeded, SQL PA issues a notification message.

The default value is 1,000 pages.

IXUPDAT nnn

The IXUPDAT user parameter indicates the limit of the number of indexes that can exist on a table that is being updated. If the maximum number of indexes is exceeded, SQL PA issues a notification message.

The default value is 5 indexes.

JOINTAB nnn

The JOINTAB user parameter indicates the limit of the number of tables that can be joined. If the maximum number of tables is exceeded, SQL PA issues a notification message.

The default value is 10 tables.

KEEPLAN YES | NO

The KEEPLAN user parameter indicates whether plan records that are owned by the value of the USEPLAN parameter are retained when SQL PA runs in batch. SQL PA deletes all plan records at the conclusion of each run, and it deletes all plan records with QUERYNO greater than 100 M on startup. If this parameter is set to YES, the deletion of plan records in your permanent plan tables is prevented. This parameter does not affect online operation, and it does not affect the generic plan tables. The default value is NO.

You are able to access this data until the next SQL PA startup.

Restriction: This parameter operates only in batch mode and applies only to USEPLAN directed or explain tables owned by the current authorization ID. It does not operate for generic PLAN_TABLEs or work under TSO ISPF.

MATCOLS n.nn

The MATCOLS user parameter limits the number of columns in an index that are used in a matching index scan. The parameter value expresses the fraction of columns in the index key. For example, the default value of 0.50 indicated that at least half of the columns in the index key must be used in a matching index scan. If not, SQL PA issues a notification message that indicates that the index is not being used to its fullest potential.

The default value is 0.50 (match half the columns or more).

NEWROTA *n.nnnn*

The NEWROTA user parameter specifies the average rotational delay (half of a revolution, also called latency) in seconds for the definition of a new storage device. This parameter is effective only when the NEWSTOR parameter is also set or when the STORAGE user parameter is set to NEWDSK.

NEWSEEK *n.nnnn*

The NEWSEEK user parameter specifies the average seek time in seconds for the definition of a new storage device. This parameter is effective only when the NEWSTOR parameter is also set or when the STORAGE user parameter is set to NEWDSK.

NEWSTOR *newname*

The NEWSTOR user parameter indicates that you are defining a new storage device. This parameter must be followed with the NEWSEEK, NEWROTO, or NEWXFER parameters to define the new device. Either this parameter or a STORAGE NEWDSK setting indicates that new disk storage is being defined.

NEWXFER *nnnn.n*

The NEWXFER user parameter specifies the average transfer rate of the device over the channel in kilobytes per second. This parameter is effective only when the NEWSTOR parameter is also set or when the STORAGE parameter is set to NEWDSK.

The default value is WDSK.

NLSCODE +0FF+ | KOR | JPN | CHS | CHT | ON

The NLSCODE parameter enables processing SQL that includes DBCS or Mixed Data names, such as that found in the following national languages:

- KOR (Korean)
- JPN (Japanese)
- CHS (Chinese)

If you specify +0FF+, SQL is converted to uppercase before processing, a translation that might alter the DBCS and mixed data names. When one of these national languages is selected, this translation to uppercase does not take place. The parser scans for and identifies shift-out and shift-in characters, which bracket the DBCS and mixed data names and leaves all characters within them unchanged while processing the SQL. This process allows Db2 to find the referenced DBCS and mixed data columns and tables. A value of ON turns off lowercase to uppercase translation.

The default value is +0FF+.

NONIXPG nnnnnnn

The NONIXPG user parameter indicates the limit of the number of index leaf pages that are read with a non-matching index scan. If you exceed the number of pages specified, SQL PA issues a notification message.

The default value is 5,000 pages.

NOSTATS <u>YES</u> | NO

Set the NOSTATS user parameter to YES to be notified when the default was used for the statistics of any table, index, table space, or column because the catalog statistics were not set.

The default value is YES.

NUMBERS YES | NO

The NUMBERS user parameter specifies whether the input file was created with NUMBERS ON.

The default value is YES.

OBJECTS YES | NO

The OBJECTS user parameter determines whether the list of objects (tables and indexes) that are used will be included in the SQL report. If this parameter is set to YES, the list of objects is displayed. If this parameter is set to NO, the list of objects is not displayed.

This list is appended to the *userid*.ANLOUT.SQL file, which also contains a list of the SQL statements that were processed in the run. The ANLOUT DD card allocates this file in batch. To display the contents under TSO, select option 5 on the Reports menu.

The default value is NO.

PKGDRDA package-location-name

This optional parameter routes package information request to another DB2 system. The packagelocation-name (a 16-character field) is any valid location specified in the local SYSIBM.LOCATIONS table. The parameter can be left in ANLPARM with a value of +0FF+ to make it non-operational. Because of the size of the parameter field (16 characters) the comments for this parameter must start in column 31. For example:

REPORTS ALL SELECT REPORT LEVEL: YES, EXP OR ALL PKGDRDA DALLAS-CQ ROUTE PACKAGE INFO REQ TO DALCQ SYSTEM

PRECISE <u>YES</u> | ALL

The PRECISE user parameter provides an extra level of precision to the cost estimate. If this parameter is set to YES, Db2 SQL Performance Analyzer replaces its internal cost estimate and QUNIT result with the values from DB2. It also uses the path length of the optimizer in calculations. If this parameter is set to ALL, the cost estimate includes additional factors and the Predicate Analysis report is generated.

YES

When PRECISE is set to YES, the Db2 SQL Performance Analyzer cost analysis attempts to replace its internal cost estimate with the Db2 optimizer's own processor time estimate, drawn from the optional Explain table DSN_STATEMNT_TABLE and the value for PROCMS, or processor milliseconds. Also included in this table is the PROCSU or Service Units estimate, which replaces the value Db2 SQL Performance Analyzer calculates for QUNITS.

PRECISE YES causes the optimizer's estimate of path length to be used in calculations. However, be aware that this path length is shorter because it only considers the data access portion of the processing. Especially significant are Category B estimates, where the cost estimate is definitely incomplete due to referential integrity, triggers, user-defined functions, and so forth. For more information about the Category B classification, see *Db2 UDB for z/OS V8 Application Programming and SQL Guide*, SC18-7415.

The values returned from PRECISE YES can also be used to govern dynamic queries through the IBM Resource Limit Facility, or RLF. This is accomplished by using the PROCSU (QUNITS) values output from Db2 SQL Performance Analyzer evaluations to populate the RLFASUWARN and RLFASUERR columns in the DSNRLST table. Then, a threshold for warnings and errors are set in

Service Units for the actual running of the dynamic queries evaluated by Db2 SQL Performance Analyzer.

ALL

PRECISE ALL produces Db2 SQL Performance Analyzer cost estimates that include additional factors, extending the cost analysis. With PRECISE ALL, the Predicate Analysis report is generated. This analysis becomes a part of the Enhance Explain and Detailed Trace reports, providing filter factors, predicate processing attributes, and other key insights.

The default value for PRECISE is YES.

PREDPCT n.nn

The PREDPCT user parameter limits the fraction of the number of total rows to be retained after predicate filtering. When this limit is exceeded, a notification message is issued that indicates that the predicates are not restrictive. For example the default value of 0.15 states that if more than 15% of the rows pass filtering, Db2 SQL Performance Analyzer issues a message.

The default value is 0.15, or 15% of all rows.

PROCESS +0FF+ | NOSEQ

Some DBRM modules are created or modified by vendor software products, and sometimes these modifications can combine several packages or plans into one module, or create DBRMs with statement sequence numbers out of sequential order. Db2 SQL Performance Analyzer expects the statement numbers in DBRMs to always be sequential (as the prepared and explained plans are ordered by QUERYNO) and generates ANL2003E Synchronization Errors when this is not the case. To eliminate these errors and process the out-of-sequence DBRM successfully, specify PROCESS NOSEQ to indicate that the DBRM contains statement numbers that are not sequential. When coded, PROCESS NOSEQ causes Db2 SQL Performance Analyzer to assign a unique query numbering scheme that avoids statement sequence errors and handles these DBRMs correctly. The default value is +0FF +.

QLIMSQL YES | NO

Set the QLIMSQL user parameter to YES to display an SQL statement that received a negative SQLCODE in the QLIMITS report.

The default value NO.

QUALIFY qqqqqqqq

The optional QUALIFY user parameter supplies the default high-level name qualifier for all singlelevel table and view names in an SQL statement found in a DBRM or file when running in batch. QUALIFY is used as the CREATOR prefix for single-level names.

For example, if the SQL contained the following statement:

SELECT * FROM L1000 WHERE CIKEY > 25

SQL PA must know the high-level qualifier for the L1000 table, currently unspecified in the SQL. If QUALIFY is set to 'TDT690', SQL PA interprets the query as follows:

SELECT * FROM TDT690.L1000 WHERE CIKEY > 25

If no value is specified, Db2 SQL Performance Analyzer automatically adjusts and defaults to the current user's SQLID.

To qualify unqualified tables when using the ISPF interface, see "Qualifying unqualified objects in packages" on page 132 and "Qualifying unqualified objects in DBRMs and SQL files" on page 132.

Restriction

QUALIFY is limited to SQL statements with a length less than 32720.

Long Qualifiers

High-level qualifiers are limited to 128 characters. To specify a long name as a qualifier, continue the name beyond the normal eight characters allotted for the QUALIFY parameter and continue with another QUALIFY parameter as the next record to finish up the name if necessary. You can also use a plus sign (+) on the second card as a continuation alternative anywhere in columns 1-7, and column 8 is always blank. Db2 SQL Performance Analyzer interprets the first blank encountered after column 8 as the end of the qualifier name.

Specifying alternative EXPLAIN tables

Some installations do not (or cannot) modify the DSN3@ATH authorization exit to add the generic ANLUSER1-*n* authorization IDs which own the reusable explain tables. There are several options available to solve this authorization scheme, including using explain tables under the primary authorization ID, or explicitly granting access to the ANLUSER*x* tables, or using other tables under a secondary authorization ID.

You can also use the USEPLAN *authid* parameter, which specifies the particular high-level qualifier of the explain tables to be used by this run. To activate the parameter, you must also set the SETPLAN parameter to ALLOW (or YES prior to V5.1), which authorizes users to select their own high-level qualifier. If SETPLAN is not ALLOW (or YES), SQL Performance Analyzer ignores the USEPLAN, and nothing changes.

REFRESH ANY | ALL | NO

The REFRESH user parameter allows the optimizer to consider materialized query tables (MQTs) as candidates for access path selection if the value is set to ALL or ANY. If this parameter is set to NO, MQTs are not considered.

The default value is NO, meaning that MQTs cannot be considered by the optimizer during access path selection.

REPORTS LOG | REP | DET

The REPORTS user parameter specifies the level of reporting. Specify one of the following values:

LOG

Turns on cost reporting, providing a summary of query costs, plus any warning messages. This report is always available.

REP

Turns on the enhanced explain reporting.

DET

Provides detailed trace information. Specifying this parameter triggers the Detail Trace Report that contains information about each step of operation, with the SQL PA estimates of Db2 resource consumption at the most granular level.

DET is the default value when you leave this parameter blank.

The QMF Interface Program, ANLGOV1, uses two special additional values for this parameter, MAX or ANY. Also, the stored procedures, ANLPRCR and ANLPRER use a value of STP in this parameter. The SEEMSGS parameter applies only to QMF Interface Program, ANLGOV1, and is invalid in the ANLPARM file.

The default value of the QMF Interface Program is MAX, which is also the minimum level of reporting that you can select under TSO.

RETCODE YES | NO

The RETCODE user parameter governs whether the return code reflects the warnings, error messages, or cost overruns from SQL PA.

NO

A value of NO leaves program return codes unchanged, so they continue to reflect normal processing of SQL PA programs.

YES

A value of YES causes the program return codes to reflect the highest message level achieved within each program. When RETCODE is set to YES, values of 4, 8, 12, or 16 are observed as

program return codes, indicating that warnings, error messages or cost overruns have been generated by the SQL that was examined. This is useful in identifying that further study is warranted when processing jobs with multiple DBRMs, or SQL migrating from Test to Production. Regardless of the RETCODE value set, the highest message level achieved is always displayed in the SYSOUT (or *userid*.ANLCOST.LOG in TSO) for DB2 SQL Performance Analyzer's ANLSQLPA program.

The default value is NO.

RETPLAN YES | NO

Only the explain-capable ANLPRER stored procedure recognizes the RETPLAN parameter. RETPLAN is passed to this procedure by the user parameter list that is embedded in the ANLSTER sample program. In addition to the usual cost estimates and warning flags, when RETPLAN is set to YES, it signals the stored procedure to return Explain plan data and catalog statistics to the calling program. You can use the sample programs as a guide when writing your own stored procedure that calls programs.

The default value is NO.

SHOWALT YES | NO

Set the SHOWALT user parameter to YES to display information about alternate indexes that are not selected by the optimizer in the Enhanced Explain and detail trace reports. The information is presented for access paths of I, N, and R only, and is not presented for multiple index or direct access using row ID. The information includes pertinent catalog statistics about the size and shape of each alternate index, and information about the index key's columns.

The default value is NO.

STORAGE dddddd

The STORAGE user parameter reflects the most prominent DASD storage medium for an application's databases. If your system uses several device types, select the device that holds most of your databases. This value is used for I/O and elapsed time estimates. The acceptable values include 3390, 3390-1, 3390-2, 3390-3, 3390-9, 6390, 6390-3, 6390-5, 6390-9, 7390-1, 7390-2, 7390-3, or 7390-F disk device types, and solid-state devices like the 4305, 4380, 6110, 6110-1, 6110-2, 6110-4, 6680-1, 7900, ICEBRG, STK, EMCX, EMCBAR, EMCELT, RAMAC, RAMAC2, and RAMAC3, ESSE20, ESSF20, ESS800, and ESS80T (turbo). If several device types are on your system, select the device which holds most of your databases.

The STORAGE default value is the 3390-3 disk device.

TSCANPG nnnnnnn

The TSCANPG user parameter indicates the limit of the number of data pages that are read with a table space scan. If the maximum number of pages is exceeded, Db2 SQL Performance Analyzer issues a notification message.

The default value is 50,000 pages.

TURNOFF nnnn BYPASSRC

The TURNOFF user parameter excludes specified messages from the SQL PA output. You must enter each TURNOFF parameter on a separate line. You can specify up to 100 TURNOFF parameters.

Specify the 4-digit numeric message number to turn off the message. For example, TURNOFF 6011 would exclude message ANL6011I from the output.

To ensure that the return code is not set for the same message number that you specify in the TURNOFF parameter, use the following option:

BYPASSRC

This keyword is optional. When you specify the BYPASSRC keyword, the return code is not set for the same specified message number. For example, TURNOFF 6011 BYPASSRC would exclude message ANL6011I from the output, and the return code would not be set.

The BYPASSRC keyword is used for batch processing only.

The TURNOFF parameter and the BYPASSRC keyword are available only for ANL*xnnn* messages, where *x* is 5, 6, or 7, and *nnn* is the last three digits of the message number. However, the TURNOFF parameter is not available for the ANL5025W, ANL5026W, ANL5027W, ANL5028W, and ANL5029W messages. To avoid these message, increase the appropriate limits.

The default is to show all messages.

USEPLAN authid | +OFF+

This optional parameter indicates specific explain tables under this authorization ID. Omit or set the parameter to +0FF+ when this option is not used. The parameter is activated only when the corresponding SETPLAN parameter is set to ALLOW (or YES prior to V5.1) in the ANLCNTL parameters for this run, and it is only used with the batch interface.

When using the ISPF interface, the explain tables used correspond to the current SQL ID as indicated on the panels. This option is also on the TSO panels when the current SQL ID is set to +0FF+, and can be available under QMF (using the running SQLID). If active under QMF, the SQLID is changed to the authorization ID you want by issuing the SET CURRENT SQLID authorization ID statement under QMF.

The default value is +0FF+.

VERSION aaaaa

This parameters indicates the version and release of Db2 that you are using so that SQL PA can select the appropriate path length values. Specify V10R1, V11R1, or V12R1.

VIADRDA explain-location-name

This optional parameter routes explain processing to another Db2 system. The explain-location-name (a 16-character field) is any valid location specified in the local SYSIBM.LOCATIONS table. The parameter can be left in ANLPARM with a value of +0FF+ to make it non-operational. Because of the size of the parameter field (16 characters), the comments for this parameter must start in column 31:

VIADRDA DALLAS-CQ ROUTE QUERIES TO DALCQ SYSTEM

Because of the size of the parameter field (16 characters) the comments for this parameter must start in column 31. For example:

REPORTS ALL SELECT REPORT LEVEL: YES, EXP OR ALL VIADRDA DALLAS-CQ ROUTE EXPLAIN REQUEST TO DALCQ SYSTEM

Examples

The SQL PA user parameters are specified in the file allocated to ANLPARM ddname in batch mode, or taken from the user parameters file on the SQL PA Parameter Data Sets panel. Three examples are provided below.

Example #1 of user parameters

REPORTS REP	THREE LEVELS OF REPORTING: LOGS, REPORTS, DETAILS
VERSION V10R1	THE CURRENT DB2 RELEASE IS V10R1 (ANY MODE)
USEPLAN TDT690	USE DIRECTED PLAN_TABLE UNDER TDT690 (OR '+0FF+')
STORAGE ESS80T	THE DASD FARM IS SHARK ESS MODEL 800 TURBO
BUFFHIT 015	15% OF PAGES ARE FOUND IN BUFFER POOL
CONNECT SPUFI	ATTACH OVERHEAD: CAF DRDA CICS RRSAF ETC.
DBRMKEY ANL*	PROCESS ANY MEMBERS STARTING WITH 'ANL' (OR +0FF+)
QUALIFY TDT690	DEFAULT HIGH LEVEL QUALIFIER => USE YOUR OWN AUTHID
PRECISE YES	USE OPTIMIZER CPU & SERVICE UNIT COSTING YES NO ALL
ADVISOR ALL	WARNINGS/ALERTS + RECS/NOTES=YES, GUIDES/GODD=ALL
SHOWALT YES	SHOW ALTERNATE INDEXES NOT SELECTED BY DB2 OPTIMIZER
SHOWALT YES	SHOW ALTERNATE INDEXES NOT SELECTED BY DB2 OPTIMIZER
RETCODE NO	RETCODE USED FOR MESSAGE LEVEL REPORTING YES NO

Example #2 of user parameters

REPORTS DETTHREE LEVELS OF REPORTING: LOGS, REPORTS, DETAILSSHOWALT NOSHOW ALTERNATE INDEXES NOT SELECTED BY DB2 OPTIMIZERVIADRDA DALLAS-CQROUTE EXPLAIN REQ VIA DRDA CONNECT TO DALLAS SYSTEM CQPKGDRDA +OFF+NO PACKAGE INFO REQ IS ROUTED VIA DRDAVERSION V7R1THE CURRENT RELEASE IS V7R1

STORAGE 3390-3	THE DASD FARM IS 3390-3 DISKS
BUFFHIT 020	20% OF PAGES ARE FOUND IN BUFFER POOL
DEGREES 1	CONSIDER PARALLELISM FOR PARTITIONED TABLESPACES
CONNECT DRDA	ATTACH OVERHEAD: CAF DRDA CICS RRSAF ETC.
DBRMKEY +OFF+	PROCESS ANY MEMBERS STARTING WITH 'ANL' (+OFF+)
PRECISE YES	EXTRA COST OPTIMIZATION FOR PRECISION YES NO ALL
ADVISOR YES	WARNINGS/ALERTS + RECS/NOTES=YES, GUIDES/GOOD=ALL
RETCODE YES	RETCODE USED FOR MESSAGE LEVEL REPORTING YES NO
(Note: no QUALIFY p	parameter in sample #2, so it defaults to CURRENT SQLID)

Example #3: Recommended ANLPARM parameters based on customer usage

The following example shows ANLPARM with the default values and suggested values, based on actual customers, in the COMMENT lines. While the values specified in the comments have been recommended, you must decide what is best for your environment.

COMMENT ***** PARAMETERS DISCOVERED DURING INSTALL ****************** COMMENT VERSION is necessary and cannot be changed. VERSION V11R1 COMMENT DEGRESS only used for manual batch processing. COMMENT If the BATCH command is used from ISPF to generate batch COMMENT jobs, it is unnecessary. COMMENT DEGREES ONE COMMENT BUFFHIT not used with PRECISE = ALL or YES. COMMENT BUFFHIT 0 COMMENT DBRMKEY, USEPLAN and QUALIFY also only used for manual batch COMMENT processing. COMMENT DBRMKEY + OFF+ COMMENT USEPLAN +OFF+ COMMENT CONNECT not with with PRECISE = ALL or YES. COMMENT CONNECT CAF COMMENT REFRESH recommended value is YES. **REFRESH YES** COMMENT VIADRDA and PKGDATA are also only used for manual batch COMMENT processing COMMENT VIADRDA + OFF+ COMMENT PKGDRDA +OFF+ COMMENT RETCODE depends on whether you want use SQL PA in batch COMMENT processing where the results are conditional to running the COMMENT next step. RETCODE NO COMMENT DELIMIT depends on whether you use double quotation marks COMMENT in COBOL that cause SQLCODE -206. DELIMIT +OFF+ COMMENT DSN8EXP recommended value is NO. DSN8EXP NO COMMENT NLSCODE depends on whether you process DBCS. NLSCODE +OFF+ COMMENT KEEPLAN recommended value is YES. **KEEPLAN YES** COMMENT PROCESS depends on whether they have DBRMs with statements COMMENT out of sequence. PROCESS +0FF+ COMMENT NUMBERS depends on your shop standard and whether NUMBERS would COMMENT be on or off. NUMBERS YES COMMENT REPORTS recommended value is DET. REPORTS DET COMMENT SHOWALT recommended value is YES. SHOWALT YES COMMENT PRECISE recommended value is ALL. PRECISE YES COMMENT ADVISOR depends on whether you want to see advice. ADVISOR NO COMMENT OBJECTS recommended value is YES.

OBJECTS YES COMMENT QLIMSQL recommended value is YES. OLIMSOL YES COMMENT NOSTATS recommended value is YES. NOSTATS YES COMMENT Use TURNOFF to turn off messages not relevant for your shop. COMMENT For example, TURNOFF 5042 to turn off alerts for table space COMMENT scans or TURNOFF 5043 to turn off alerts for non-matching COMMENT index scan. COMMENT TURNOFF 5042 COMMENT DSPVARS recommended value is YES. DSPVARS YES COMMENT DSPVARS recommended value is YES. DSPPARM YES COMMENT Set CSVRPT YES if you want to use CSV output. COMMENT CSVRPT NO COMMENT CSV options; not needed for CSVRPT NO. COMMENT CSVSEL TAB0 COMMENT CSVHDRW COL COMMENT CSVDLIM COMMA COMMENT CSVDCSP DOT COMMENT ****** SET LIMITS FOR REPORTING MESSAGES WHEN LIMIT IS EXCEEDED COMMENT Limits to trigger messages are set at their discretion. JOINTAB 10 INLISTS 10 IXUPDAT 5 TSCANPG 50000 ISCANPG 1000 NONIXPG 5000 MATCOLS 0.50 PREDPCT 0.15 COMMENT ***** DEVICE TYPE FOUND IN THE USERS GUIDE APPENDIX C ******* COMMENT STORAGE not used with PRECISE = ALL or YES. COMMENT STORAGE 3390-9 COMMENT Plan table report have several formats available. LEVEL TABULAR COMMENT INDEX recommended value is YES. INDEX YES COMMENT ACCTYPE recommended value is ALL. ACCTYPE ALL COMMENT HOSTVAR recommened value is YES. HOSTVAR YES COMMENT FORMAT recommended value is YES. FORMAT YES COMMENT PLANINF recommened value is YES. PLANINF YES ANLCARD0+ //ANLBATCH JOB 'SQL PA BATCH', ANLCARD1+ // NOTIFY=&SYSUID,MSGCLASS=X, ANLCARD2+ // MSGLEVEL=(1,1),REGION=0M,' ANI CARD3+ /*JOBPARM S=ANY

ANLPARM user parameter settings for QMF

QMF requires some specific settings in the ANLPARM user parameter list.

SUBSYST

SUBSYST must reflect the name of the Db2 subsystem (DSN is the default) for the CAF connections.

REPORTS

If REPORTS is set to MAX, SQL PA only shows costs for those queries that have exceeded one or more limits. MAX is the recommended setting.

If REPORTS is set to ANY, SQL PA shows the costs for any query processed. With the MAX option, SQL PA remains invisible unless the user exceeds a resource threshold.

CONNECT

CONNECT must be set to QMF.

SEEMSGS

If SEEMSGS is set to YES during initial testing or debugging, SQL PA shows informational and error messages encountered during processing. Many of these messages are also displayed by QMF.

If SEEMSGS is set to NO, these messages are not displayed by SQL PA. NO is the recommended setting.

Plan Table table contents

Plan Table depends on a number of Db2 tables.

Topics:

- "The EEEAUTH table" on page 242
- "The EEEWORK table" on page 242
- "The EEEDBRM table" on page 243
- "The EEEHINT table" on page 243
- "The EEEPLAN table" on page 244
- "The EEEPATH table" on page 245

The EEEAUTH table

The EEEAUTH table is used to specify the privileges for each Plan Table user.

If you do not have an entry in the table, default privileges specified in the row with a blank authorization ID are used.

Table 22. EEEAUTH table				
Column name	Data type	Description		
Authid	Char(8) Not Null	Authorization ID of the user		
NLS	Char(1) Not Null	Indicator for National Language (Y/N)		
Planinfo	Char(1) Not Null	Indicator for extended plan info. (Y/N)		
FormatSQL	Char(1) Not Null	Indicator for formatting of SQL (Y/N)		
Timestmp	Timestamp NNWD	Timestamp for creation or change for row		
Remotexpl	Char(1) NNWD	Indicator for use of remote explain (Y/N)		

The EEEWORK table

The EEEWORK table is used internally to find all of the indexes used by a given plan or package if a parameter of INDEX=Condensed has been specified.

The first three columns of the EEEWORK table are also used for defining the database and table space names for the location of the PLAN_TABLE tables that are created.

Table 23. EEEWORK table		
Column name	Data type	Description
Location	Char(16) NNWD	Location name for Db2 subsystem, or blank
Creator	Char(8) Not Null	Authorization ID for row creation
Table 23. EEEWORK table (continued)		
-------------------------------------	------------------	--
Column name	Data type	Description
Tabcreator	Char(8) Not Null	Authorization ID of table owner
Tabname	Varchar(18) NNWD	Table name
Timestamp	Timestamp NNWD	Timestamp for creation or change for row
Туре	Char(1) NNWD	Internal indicator

The EEEDBRM table

The EEEDBRM table is used internally by Plan Table to store DBRM and package information. This information is used to build the Plan Information report.

Data type	Description
char (8) not null	DBRM name
char (8) not null	Plan name
char (8) not null	Plan creator name
char (6) not null	Date for precompilation
char (8) not null	Time for precompilation
integer not null	Number of SQL statements in DBRM
char (1) not null	Programming language
char (1) not null	Character set (Alphanum or Katakana)
char (1) not null	Comma or period
char (1) not null	Decimal 31 used at precompile
char (1) NNWD	Db2 release at precompile
char (1) NNWD	DBRM or Package
char (8) NNWD	Authorization ID
timestamp NNWD	Timestamp for storing row
char(44) NNWD	Full MVS pds name for DBRM
char(18) NNWD	Collection identification
varchar(64) N null	Version identification
	Data typechar (8) not nullchar (8) not nullchar (8) not nullchar (6) not nullchar (6) not nullchar (8) not nullchar (1) NNWDchar (10) NNWDchar

The EEEHINT table

The EEEHINT table is used to save hint IDs for the mini-plans generated by Plan Table The ID identifies each mini-plan by a query number, a user ID, and a timestamp.

Table 25. EEEHINT table		
Column name	Data type	Description
Location	Char(16) Not Null	Location of miniplan
Hint_ID	Integer Not Null	Unique number identifying mini-plan
Creator	Char(8) Not Null	User ID for mini-plan creator

Table 25. EEEHINT table (continued)		
Column name	Data type	Description
Opthint	Char(8) Not Null	Specified optimization hint
Queryno	Integer Not Null	Query number taken from PLAN_TABLE table
Bind_time	Timestamp NNWD	Timestamp for creation of mini-plan row
Progname	Char(8) NNWD	Name of program or package

The EEEPLAN table

The EEEPLAN table is used internally by Plan Table to store plan information for plans explained by Plan Table. The plan information is used to build the Plan Information report.

Table 26. EEEPLAN table		
Column name	Data type	Description
name	char (8) not null	Plan name
creator	char (8) not null	Creator name
binddate	char (6) not null	Date for bind
bindtime	char (8) not null	Time for bind
boundby	char (8) not null	Authorization ID for binder
plsize	integer not null	Size of plan
avgsize	integer not null	Average size of plan sections
sqlstmt	integer not null	Number of SQL statements in plan
validate	char (1) not null	Validity checking
isolation	char (1) not null	Isolation level
valid	char (1) not null	Plan or package valid
operative	char (1) not null	Plan or package operative
acquire	char (1) not null	Acquire option
release	char (1) not null	Release option
qualifier	char (8) not null	Table qualifier
cachesize	smallint not null	Size of cache
storedby	char (8) not null	Authorization ID
timestmp	timestamp NNWD	Timestamp for row creation
plentries	smallint NNWD	Number of plan entries
sysentries	smallint NNWD	Number of system entries
expredicate	char (1) NNWD	Option for CURRENTDATA
deferprep	char (1) NNWD	Option for DEFER
currentserver	char (16) NNWD	Location name
degree	char (3) NNWD	Degree of parallelism
sqlrules	char (1) NNWD	SQL rules

Table 26. EEEPLAN table (continued)		
Column name	Data type	Description
disconnect	char (1) NNWD	Disconnect option

The EEEPATH table

The EEEPATH table is used to save selected PLAN_TABLE and DSN_STATEMNT_TABLE information that later is sent to SQL PA together with the extracted SQL statements.

Table 27. EEEPATH table

Column name	Data type	Description
Column name Queryno Qblockno AppIname Planno Method Creator Tname Accesstype Mathcols Accesscreator Accessname Indexonly Sortn_uniq Sortn_join Sortn_orderby Sortc_uniq Sortc_join Sortc_orderby Sortc_groupby Tslockmode Prefetch Column_fn_eval Mixopseq Access_degree Access_pgroup_id Join_degree Join_pgroup_id Sortn_pgroup_id Parallelism_mode Merge_join_cols Correlation_name	Data type Same definitions as the corresponding columns in PLAN_TABLE.	Description Same descriptions as the corresponding columns in PLAN_TABLE.
Merge_join_cols Correlation_name Page_range Join_type Qblock_type Primary_accesstype Table_type		

Table 27. EEEPATH table (continued)

Column name	Data type	Description
Stmt_type Cost_category Procms Procsu Reason	Same definitions as the corresponding columns in DSN_STATEMNT_TABLE.	Same descriptions as the corresponding columns in DSN_STATEMNT_TABLE.
Mergc Mergn	Same definitions as the corresponding columns in PLAN_TABLE.	Same descriptions as the corresponding columns in PLAN_TABLE.

PLAN_TABLE contents

SQL Performance Analyzer handles a PLAN_TABLE differently depending upon whether a PLAN_TABLE already exists or needs to be created.

If you do not already have a PLAN_TABLE table, one will be created for you. The newly created PLAN_TABLE will be located in the default database, unless a row giving the names of the database and table space was inserted into the EEEWORK table.

The content of the PLAN_TABLE table is dependent on the Db2 version you are using. If you already have a PLAN_TABLE table, SQL Performance Analyzer compares the Db2 level and the columns available in the PLAN_TABLE, and automatically adds any missing columns.

Migration from IBM Db2 Path Checker to SQL Performance Analyzer

Path Checker provided the ability to use explain tables in Db2 that do not follow the normal naming convention. The same functionality can be accomplished by using aliases with SQL Performance Analyzer.

PLAN_TABLEs created with Path Checker having non-standard names

Path Checker provided the capability to name a PLAN_TABLE virtually any name. IBM Db2 does not write access paths to these PLAN_TABLEs because of the non-standard names. Likewise, IBM Db2 cannot retrieve access path hints or overrides from any PLAN_TABLEs having non-standard names.

SQL PA can access these non-standard PLAN_TABLE names using the IBM Db2 ALIAS capability. As an example, let's assume that you created and populated a table with a format like a PLAN_TABLE, and named it 'PRIOR_PLAN_TABLE'. You could obtain the access path information using SQL PA via an ALIAS that uses standard names and a new SCHEMA by issuing these commands:

```
CREATE ALIAS SYSDBA2.PLAN_TABLE FOR SYSDBA.PRIOR_PLAN_TABLE;
CREATE ALIAS SYSDBA2.DSN_STATEMNT_TABLE
FOR SYSDBA.DSN_STATEMNT_TABLE;
```

With an ALIAS in place, you can refer to any appropriately formatted table from SQL PA, and COMPARE, TEST, and EXPLAIN should work with no further customization. On the COMPARE or TEST panel, enter the new SCHEMA as the SQLID to select the non-standard table names.

To set the schema for EXPLAIN, the SETPLAN option controls whether you are allowed to set the PLAN_TABLE owning SQLID. If you are using generic plan tables, selecting an aliased table is not allowed. Otherwise, if SETPLAN is set to ALLOW, you can use option 3.2 to set the SQLID to your new schema. You may then use option 1.2 (EXPLAIN) to populate the aliased plan table from a DBRM.

You must create an ALIAS for both the PLAN_TABLE and the DSN_STATEMNT_TABLE so that cost information may be compared. If you do not have a DSN_STATEMNT_TABLE that matches your

PLAN_TABLE, create an empty DSN_STATEMNT_TABLE. The empty table will impair cost comparison, but will not prevent access path comparison.

Equivalent functionality

The following table shows which SQL PA menu options to use to perform the same functions as various Path Checker commands.

Table 28. Path Checker options and SQL PA menu options		
Path Checker command	SQL PA Menu Options	
COMPARE	On Main Menu, choose Option 1.7 and type "COMPARE".	
TEST	On Main Menu, choose Option 1.8 and type "TEST".	
EXPLAIN DBRM	On Main Menu, choose Option 3.2, then Option 1.2 and type "EXPLAIN".	

Chapter 11. Messages and return codes

These topics describe the error messages and return codes that are issued by Db2 SQL Performance Analyzer.

Db2 SQL Performance Analyzer messages and return codes

Db2 SQL Performance Analyzer messages are grouped by the following:

- Output report messages
- Statistics collector messages
- Db2 SQL Performance Analyzer ISPF panel messages

The warnings, alerts, recommendations, guidelines, notes and good news messages from the Explain and Detailed reports are also listed.

Db2 SQL Performance Analyzer messages are of the form ANLnnnnt, where:

nnnn

is the message number.

t

is a one-character suffix with the value I for informational messages, W for warnings, and E for errors.

Return codes

A return code is set for most Db2 SQL Performance Analyzer messages using the conventions shown in Table 29 on page 249.

Table 29. Return code conventions				
Message ID	Type of Message	ANLSQLPA	ANLQMF	ANLPRCR and ANLPRER
ANLnxxxI	Informational	0	0	0
ANL7 <i>xxx</i> I	Guidelines and Good News	0	n/a	n/a
ANL6 <i>xxx</i> I	Notes [®] and Recommendations	0	n/a	n/a
ANL <i>nxxx</i> I	Default stats, parameters, autofixes	4	4	4
ANLnxxxW	Warning messages	8	8	8
ANL5 <i>xxx</i> W	Alerts and Warnings	12	12	12
ANLnxxxE	Error messages	16	16	16

In the message IDs, n=1, 2, 3, or 4.

Output report messages

The following messages are issued from ANLSQLPA and ANLQMF.

ANL0000I

ANLUSERX.PLAN_TABLE WILL BE USED TO HOUSE EXPLAIN OUTPUT FOR THIS RUN.

Explanation

Informational. Informs you of the plan table name used for evaluations.

None

ANL0100I ACCORDING TO THE DB2-CATALOG, YOU DO NOT HAVE A xxx_TABLE TABLE. A xxx_TABLE TABLE WILL NOW BE CREATED FOR AUTHORIZATION-ID: authid

Explanation

The Explain process was not able to find an *xxx*_TABLE for the current authorization ID. Explain creates an *xxx*_TABLE in the default database (SYSDEFLT), providing that proper authorization exists. The *xxx*_TABLE can be a PLAN_TABLE, a DSN_STATEMNT_TABLE, or a DSN_FUNCTION_TABLE. This message can be followed by message ANL0102I or by message ANL0600E.

User response

None

ANL0102I xxx TABLE NAMED authid.xxx_TABLE CREATED IN THE DEFAULT DATABASE

Explanation

The Explain process has created an *xxx*_TABLE for authorization ID *authid* in the default database (SYSDEFLT). The *xxx*_TABLE can be a PLAN_TABLE, a DSN_STATEMNT_TABLE, or a DSN_FUNCTION_TABLE. This message is preceded by message ANL0100I.

User response

None

ANL0104I COLUMN column name HAS BEEN ADDED TO authid.PLAN_TABLE

Explanation

The Explain process has found that you have specified a dynamic explain request using an existing PLAN_TABLE using an older DB2 version. Explain is upgrading the PLAN_TABLE by adding the appropriate columns to the existing PLAN_TABLE.

User response

None

ANL0106I	COLUMN column name HAS BEEN
	ADDED TO
	authid.DSN_STATEMNT_TABLE

Explanation

The Explain process has found that you have specified a dynamic explain request using an existing DSN_STATEMNT_TABLE using an older DB2 version. Explain is upgrading the DSN_STATEMNT_TABLE by adding the appropriate columns to the existing DSN_STATEMNT_TABLE.

User response

None

ANL0108I	COLUMN column name HAS BEEN
	ADDED TO
	authid.DSN_FUNCTION_TABLE

Explanation

The Explain process has found that you have specified a dynamic explain request using an existing DSN_FUNCTION_TABLE using an older DB2 version. Explain is upgrading the DSN_FUNCTION_TABLE by adding the appropriate columns to the existing DSN_ FUNCTION_TABLE.

User response

None

```
ANL0110I COLUMN column name HAS BEEN
ADDED TO authid.EEEPATH
```

Explanation

The Explain process has found that you have specified a dynamic explain request using an existing EEEPATH using an older SQL PA version. Explain is upgrading the EEEPATH by adding the appropriate columns to the existing EEEPATH.

User response

None

ANL0112I COLUMN column name HAS BEEN ADDED TO authid.DSN DETCOST TABLE

Explanation

The Explain process has found that you have specified a dynamic explain request using an existing DSN_DETCOST_TABLE using an older SQL PA version. Explain is upgrading the DSN_DETCOST_TABLE by adding the appropriate columns to the existing DSN_DETCOST_TABLE.

None

ANL0113I

authid.DSN_DETCOST_TABLE was altered to VOLATILE

Explanation:

Table DSN_DETCOST_TABLE was not defined with the VOLATILE attribute and was altered to VOLATILE.

Table

User response

None

ANL0114I COLUMN column name HAS BEEN ADDED TO authid.DSN PREDICAT TABLE

Explanation

The Explain process has found that you have specified a dynamic explain request using an existing DSN_PREDICAT_TABLE using an older SQL PA version. Explain is upgrading the DSN_PREDICAT_TABLE by adding the appropriate columns to the existing DSN_PREDICAT_TABLE.

User response

None

ANL0115I Table authid.DSN_PREDICAT_TABLE was altered to VOLATILE

Explanation:

Table DSN_PREDICAT_TABLE was not defined with the VOLATILE attribute and was altered to VOLATILE.

User response

None

ANL0116I THE LOCATION FOR xxx_TABLE IS SPECIFIED AS xxxxxx.yyyyy

Explanation

The installation has specified that *xxx*_TABLE tables are created in the database named *xxxxx*, and within that database in the table space named *yyyyyy*. The Explain process attempts to create the *xxx*_TABLE table in that location. The *xxx*_TABLE can be a PLAN_TABLE, a DSN_STATEMNT_TABLE, or a DSN_FUNCTION_TABLE.

User response

None

ANL0118I NO LOCATION SPECIFIED FOR xxx_TABLE. WILL USE DEFAULT DATABASE

Explanation

The installation has not specified a location (database and table space names) in the EEEWORK table for *xxx*_TABLE tables to be created. Consequently, the *xxx*_TABLE table will be created in the DB2 default database. The *xxx*_TABLE can be a PLAN_TABLE, a DSN_STATEMNT_TABLE, or a DSN_FUNCTION_TABLE.

User response

None

ANL0120I	INDEX INFORMATION
	SUPPRESSED BY USER REQUEST

Explanation

You have specified INDEX=N for the PLN, QMF, or QNO keyword. This means that no column information for the chosen index is shown.

User response

None

ANL0130I	NO STATISTICS AVAILABLE -
	EXECUTE RUNSTATS

Explanation

The Explain process has found that no DB2 catalog statistics are available for the current table space. Before starting Explain again, run RUNSTATS against the table space.

User response

None

ANL0136I HINT-ID (xxx) WAS ADDED FOR STATEMENT yyy IN PROGRAM zzz

Explanation

The Explain process has added the specified hint ID to the specified row (statement number *yyy* for program *zzz*) in the PLAN_TABLE table.

User response

None

ANL0140I

NOTE: dbname.tsname CONTAINS xxx ACTIVE PAGES - CONSIDER A USABLE INDEX

Explanation

This message is produced if the DB2 Optimizer has chosen a table space scan for a table located in a table space larger than 10 active pages. If you want the access path to be INDEX=based, create a usable index.

User response

None

ANL0142E TABLE authid.tname NOT FOUND IN SYSIBM.SYSTABLES - SQL-CODE: -xxx

Explanation

The Explain process was not able to find the named table in SYSIBM.SYSTABLES. The table might be a result of a materialization. Otherwise check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

User response

None

ANL0144I ALTER TABLE ... UPDATED TABLE authid.PLAN_TABLE TO V8 FORMAT

Explanation

The Explain process has updated the column definitions for the PLAN_TABLE table, so it reflects the expanded DB2 Version 8 column definitions.

User response

None

ANL0146I ALTER TABLE ... UPDATED TABLE authid.DSN_STATEMNT_TABLE TO V8 FORMAT

Explanation

The Explain process has updated the column definitions for the DSN_STATEMNT_TABLE table, so it reflects the expanded DB2 Version 8 column definitions.

User response

None

ANL0148I ALTER TABLE ... UPDATED TABLE authid.DSN_FUNCTION_TABLE TO V8 FORMAT

Explanation

The Explain process has updated the column definitions for the DSN_FUNCTION_TABLE table, so it reflects the expanded DB2 Version 8 column definitions.

User response

None

ANL0150I	TABLE xxxx.xxxxxx IS AN
	IMMEDIATE TABLE - NO DATA
	AVAILABLE

Explanation

The Explain process has decided that the table being investigated is an immediate temporary table. Explain skips processing the table and continues with other work.

User response

None

ANL0160I AUTHORIZATION TABLE DOES NOT EXIST - WILL USE DEFAULT AUTHORIZATIONS

Explanation

The user authorization table, *binder*.EEEAUTH, could not be found. Default authorization values (messages shown in NLS if available, SQL statements are formatted, plan history is not shown, remote packages explained) will be used. Operation continues.

User response

None

ANL0162I PACKAGE REPORTS NOT REQUESTED

Explanation

You have specified that packages for a specific plan not be explained. The Explain process explains all DBRMs, but no packages. Operation continues.

None

ANL0164I COLLECTION SPECIFICATION

xxxxxxxx.* HAS MORE THAN 10 PACKAGES USE PACKAGE=collid.name.(version) TO GET A DETAILED LISTING

Explanation

You have specified an asterisk (*) as the package name for collection (*xxxxxxx*). The Explain process has found that the collection contains more than 10 packages. The package names are listed. To explain all packages in the collection, set FORCE=YES on the package statement. Operation continues.

User response

None

ANL0166I PACKAGE XXXXXXX IN COLLECTION yyyyyy HAS MORE THAN 5 VERSIONS USE PACKAGE=collid.name.(version) TO GET A DETAILED LISTING

Explanation

You have specified package name (*xxxxxx*) for collection (*yyyyyyyy*). The Explain process has found that the package exists in more than five versions. The version identifications are listed. To explain all versions of the package, set FORCE=YES on the package statement. Operation continues.

User response

None

ANL0168I THE PACKAGE NAME IS SPECIFIED AS A THREE-PART NAME. LOCATION IGNORED.

Explanation

You have specified the package name with a location name in front of the collection and package identifications. The Package (PKG) statement, however, does not support specification of a location name. The location name is ignored, and the operation continues.

User response

None

ANL0170I

DD-NAME FOR INPUT WAS SPECIFIED AS CARD - SHOULD BE INPUT

Explanation

The ddname for the input data set was in the previous "Explain" version named CARD. In the current version of the Explain process, the ddname is INPUT. For compatibility reasons the name CARD can still be used, but change it to INPUT as soon as possible.

User response

None

ANL0172I DD-NAME FOR OUTPUT (REPORT) WAS MISSING - WILL USE SYSTSPRT

Explanation

The ddname for the print data set REPORT was not specified. The Explain process instead uses the ddname SYSTSPRT, when printing the Explain report. Before running again, add the following job control card:

//REPORT DD SYSOUT=*

User response

None

ANL0174I DD-NAME FOR OUTPUT (SYSTSPRT) WAS MISSING

Explanation

The ddname for the print data set SYSTSPRT was not specified. If tracing was enabled, the trace facility is now disabled. Operation continues. Before running again, add the following job control card:

//SYSTSPRT DD SYSOUT=*

User response

None

ANL0176I DD-NAME FOR OUTPUT DATA SET (SQLPA) WAS MISSING

Explanation

The ddname for the output data set SQLPA was not specified. To have the Explain process generate output meant for SQL Performance Analyzer, disable this process. Before running again, add the following job control card:

//SQLPA DD DSN=xxxx,DISP=yyy

User response

None

ANL0180I CONNECTION TO SERVER xxxxx SUCCESSFUL SERVER TYPE IS nnn VER. r REL. p MOD. q

Explanation

An SQL Connect statement was issued by the Explain process to connect to the server location specified by the LOCATION=xxxxx parameter. The connection was established. *nnn* is the type of the server; valid types are: DB2, DB2/VM, DB2/400, DB2/2, and DB2/6000. *r*, *p*, and *q* give the version, release, and modification level of the server system.

User response

None

ANL0182I CONNECT RESET TO xxxxx SUCCESSFUL

Explanation

An SQL Connect Reset statement was issued by the Explain process to reconnect to the server system where Explain is run. The connection was re-established.

User response

None

ANL0184I CURRENT SERVER NAME COULD NOT BE ACQUIRED - WILL USE BLANKS

Explanation

An SQL Set Current[®] Server statement was issued by the Explain process to set the new server, but the call failed. Explain continues running.

User response

None

ANL0200E Invalid input ignored: xxxxxxxx

Explanation:

An unknown keyword, xxxxxxx, was found.

User response:

Specify a valid keyword and reissue the request.

ANL0201E Invalid value ignored: xxxxxxx

Explanation:

The value that follows a keyword was incorrect or incorrectly formatted.

User response:

Specify a valid value and reissue the request.

ANL0202E	Input before action ignored:
	XXXXXXXX

Explanation:

A valid ACTION command, such as ACTION TEST or ACTION COMPARE, must be the first command.

User response:

Specify a valid ACTION command as the first command and reissue the request.

ANL0203E	Previous action incomplete and
	ignored: COMPARE/TEST

Explanation:

The ACTION END command was not found.

User response:

Specify a valid ACTION END command and reissue the request.

ANL0204E	Required COMPARE/TEST
	information is missing: xxxxxxx

Explanation:

The COMPARE or TEST command is missing required information, such as COLLID or PKGNAME.

User response:

Specify the required information and reissue the request.

```
ANLO205E Explain information is missing:
collid.pkg(ver)
```

Explanation:

The specified explain information was not found.

User response:

Verify that the PLAN_TABLE exists, and that the package access paths being reported on exist in the plan table.

```
ANL0206E Error in retrieving string1
information: collid.pkg
```

Explanation:

There was an error in retrieving the specified information. *String1* can be explain, dynamic explain, DBRM/PACKAGE LOCATION, DBRMLIB, DBRMLIB MEMBER, or PACKAGE CATALOG.

Your response varies, depending on the value of *string1*:

- If it is explain information, verify that the PLAN_TABLE exists, and that the package access paths being reported on exist in the plan table.
- If it is DBRM/PACKAGE LOCATION or PACKAGE CATALOG, verify that the DBRM was bound.
- If it is DBRMLIB, verify that the correct DBRMLIB was coded as part of the input commands.
- If it is DBRMLIB MEMBER, verify that the member exists in the specified DBRMLIB.
- If it is a dynamic explain, then the specified package could not be explained. Call IBM Technical Support and provide job output, including output from ANLTRACE DD.

ANL0207E Error in processing COMPARE/ TEST: collid.pkg

Explanation:

This is an internal error.

User response:

Call IBM Technical Support and provide job output, including output from ANLTRACE DD.

ANL0208W Cost information is missing.

Explanation:

The DSN_STATEMNT_TABLE was not created or did not contain the necessary records. This table must be created *before* the DBRM is bound or explained.

User response:

Create the DSN_STATEMNT_TABLE and reissue the request. Contact IBM Technical Support if you need assistance.

ANL0209E	The same explain information
	cannot be used for comparison.

Explanation:

The request attempted to compare the same package version against itself, which is not allowed.

User response:

Change the input to specify two different package versions and reissue the request.

ANL0210W STMTMATCH 2 changed to STMTMATCH 1 for COMPARE.

Explanation:

STMTMATCH 2 cannot be used for action COMPARE, so STMTMATCH 1 is used instead.

User response:

Change STMTMATCH 2 to STMTMATCH 1 when issuing ACTION COMPARE. Try the request again.

ANL0211E Unable to process. Change SETPLAN to ALLOW to use EXPSQLID userid.

Explanation:

The system parameter SETPLAN is not set to ALLOW, so EXPSQLID cannot be used to set the explain SQL ID to *userid*.

User response:

Change SETPLAN to ALLOW to use EXPSQLID userid.

ANL0212E	Unable to process. <i>userid</i> is a
	Generic ID which cannot be used
	with EXPSOLID. When a Generic
	ID is needed, use EXPSQLID +OFF
	+.

Explanation:

A specific generic ID cannot be used with EXPSQLID.

User response:

When a generic ID is needed, use EXPSQLID +OFF+.

ANL0300W INVALID INPUT PARAMETER: FIRST 45 CHARACTERS OF INPUT SPEC.

Explanation

An input record with a keyword different from LVL (or Level), QNO (or Queryno), PLN (or Plan), PKG (or Package), QMF, or SQL has been received by the Explain process. The first 45 characters of the input record are shown. Explain proceeds to the next input record.

User response

None

```
ANL0302W OWNER SUBPARAMETER (authid)
INVALID - PARAMETER IGNORED
```

Explanation

An input record with the QNO keyword has an incorrect specified authorization ID following the OWNER= subparameter. The Explain process ignores the specified authorization ID and continue processing using the PLAN_TABLE of the job submitter or TSO user.

User response

Specify a valid authorization ID for QNO keyword, and issue the request again.

ANL0304W	INDEX SUBPARAMETER
	(indexparm) INVALID - DEFAULT
	VALUE USED

Explanation

An input record with the QNO (or Queryno), PLN (or Plan), PKG (Package), or QMF keyword has an incorrect specified value following the INDEX= subparameter. Valid values are strings starting with Y, N, or T. The Explain function continues processing with a default value of Y.

User response

Specify a valid QNO, PLN, PKG, or QMF value, and issue the request again.

ANL0306W SQL-STATEMENT WAS NOT TERMINATED BY AN ';' -STATEMENT IGNORED

Explanation

The statement following the SQL keyword did not end with a semicolon (;). You can specify the SQL statement on one line or on several lines, but it must always end with semicolon. The statement is ignored and the Explain process proceeds to the next input record.

User response

End the statement with a semicolon, and reissue the request.

ANL0308W INVALID QNO FORMAT: FIRST 50 CHARACTERS OF INPUT SPEC.

Explanation

The parameter following the QNO= or QueryNo= keyword is not numeric or the number does not immediately follow the equal sign (=). The first 50 characters of the statement are shown in the warning. The statement is ignored and the Explain process proceeds to the next input record.

User response

Specify a numeric value for the QNO keyword immediately following the equal sign, and issue the request again.

ANL0310W QUERY NO. xxxx DOES NOT EXIST FOR USER authid - REQUEST IGNORED

Explanation

The specified query number (*xxxx*) following the QNO= or QueryNo= keyword could not be located in the accessed PLAN_TABLE. The statement is ignored and Explain proceeds to the next input record.

User response

Verify that the query number does exist in the table. Also verify that the correct PLAN_TABLE is being accessed. If necessary specify the correct authorization ID in the owner parameter. Specify a valid query number, and reissue the request.

ANL0312W QNO. xxx NOT FOUND IN authid.PLAN_TABLE

Explanation:

The query number (*xxx*) for the stabilized dynamic query was not located in the PLAN_TABLE specified. Processing continues with the next query number requested.

User response:

Specify a valid query number for the stabilized dynamic query for the PLAN_TABLE, and issue the request again.

```
ANL0316W
```

QNO. xxx IN PACKAGE yyyyyyy.zzzzzz NOT IN authid.PLAN_TABLE

Explanation

The Explain process was not able to locate query number (*xxx*) for collection ID (*yyyyyyy*) and package name (*zzzzzz*) in the PLAN_TABLE of the plan binder. Explain will continue by looking for the next query number for collection ID (*yyyyyyy*) and package name (*zzzzzz*).

User response

Specify a valid query number for the specified collection ID and a valid package name for the PLAN_TABLE, and issue the request again.

```
ANL0320W PLAN xxxxxx NOT BOUND WITH
EXPLAIN OPTION - DYNAMIC
EXPLAIN ISSUED
```

Explanation

The plan specified after the PLN= or Plan= keyword was not bound with the Explain option, but the rows in the SYSIBM.SYSSTMT table have been deleted for the given DBRM. The statement is ignored and Explain proceeds to the next input record.

User response

Rebind the plan, and run the job again with the same input record.

ANL0322W

DBRM xxxxxx HAS NO ROWS IN SYSIBM.SYSSTMT - REQUEST IGNORED

Explanation

The plan specified after the PLN= or Plan= keyword was bound with the Explain option, but the rows in the SYSIBM.SYSSTMT table have been deleted for the given DBRM. The statement is ignored and Explain proceeds to the next input record.

User response

Rebind the plan, and run the job again with the same input record.

ANL0324W PLAN_TABLE FOR USER authid DOES NOT EXIST - REQUEST IGNORED

Explanation

The authorization ID given as OWNER subparameter for the QNO keyword does not own a PLAN_TABLE. The statement is ignored and Explain proceeds to the next input record.

User response

Specify a valid authorized ID for the QNO keyword, and issue the request again.

ANL0330W SUBPARAMETER FIRST (xxxx) INVALID - PARAMETER IGNORED

Explanation

An input record with the PLN (or PLAN) keyword has an incorrectly specified value following the FIRST= subparameter. The Explain process ignores the specified value and continues processing using a default value of 1.

User response

Specify a valid value for the PLN keyword, and reissue the request.

ANL0332W SUBPARAMETER LAST (*xxxx*) INVALID - PARAMETER IGNORED

Explanation

An input record with the PLN (or PLAN) keyword has an incorrectly specified value following the LAST= subparameter. The Explain process ignores the specified value and continues processing using a default value of 99999999.

User response

Specify a valid value for the PLN keyword, and reissue the request.

ANL0334W ACCTYPE SUBPARAMETER (xxx) INVALID - DEFAULT VALUE USED

Explanation

An input record with the PLN (or PLAN) keyword has an incorrectly specified value *xxx* following the ACCTYPE= subparameter. The Explain process ignores the specified value and continues processing using a default value of All.

User response

Specify a valid value for the PLN keyword, and reissue the request.

```
ANL0346W LEVEL PARAMETER (string)
INVALID - DEFAULT VALUE USED
(value)
```

Explanation

An input record with the LVL (or LEVEL) keyword has an incorrectly specified value, *string*. The Explain process ignores the specified value and continues processing using the default level value of *value*

User response

Specify a valid value for the LVL keyword and reissue the request.

```
ANL0336W LEVEL SUBPARAMETER (string)
INVALID - CURRENT VALUE USED
(value)
```

Explanation

An input record with the PLN (or PLAN) or PKG (or PACKAGE) keyword has an incorrectly specified value *string* following the LEVEL= subparameter. The Explain process ignores the specified value and continues processing using the current level value of *value*, where *value* is one of the following options:

- DET
- SUM
- NOC
- ACC
- ALL
- KEY
- SQL

- TAB
- CON
- TIN
- TIY
- XOF
- XON

Specify a valid value for the PLN or PKG keyword, and reissue the request.

ANL0338W SUBPARAMETER DBRM (xxx) INVALID - PARAMETER IGNORED

Explanation

An input record with the PLN (or PLAN) keyword has an incorrect specified value *xxx* following the DBRM= subparameter. The Explain process ignores the specified value and continues processing all DBRMs within the plan.

User response

Specify a valid value for the PLN keyword, and reissue the request.

ANL0340W SUBPARAMETER TABLE (xxx) INVALID - PARAMETER IGNORED

Explanation

An input record with the PLN (or PLAN) keyword has an incorrectly specified value *xxx* following the TABLE= subparameter. The Explain process ignores the specified value and continues processing using all tables within the plan.

User response

Specify a valid value for the PLN keyword, and reissue the request.

ANL0342W TABLE NAME xxx DOES NOT EXIST - PARAMETER IGNORED

Explanation

An input record with the PLN (or PLAN) keyword has an incorrectly specified value *xxx* following the TABLE= subparameter. The Explain process ignores the specified value and continues processing using all tables within the plan.

User response

Specify a valid value for the PLN keyword, and reissue the request.

ANL0344W SUBPARAMETER LOCATION (xxx) INVALID - PARAMETER IGNORED

Explanation

An input record with the PLAN or PACKAGE keyword has an incorrectly specified value *xxx* for the LOCATION= subparameter. The Explain process ignores the specified value and continues processing using the last specified location name, or the home server if no location value has been specified.

User response

Specify a valid value for the PLAN or PACKAGE keyword, and issue the request again.

```
ANL0356W INVALID STABILIZED DYNAMIC
QUERY SPECIFICATION: xxxxxxxx
```

Explanation:

An input record with the SDQ keyword has an incorrect specified stabilization group and statement ID combination. The specified value is ignored and processing continues with the next input statement.

User response:

Specify a valid stabilization group and statement ID combination, and issue the request again.

```
ANL0360W PACKAGE STATEMENT IS NOT
SUPPORTED IN DB2 REL. x.y -
REQUEST IGNORED
```

Explanation

An input record with the PKG (or Package) keyword has been read by the Explain process, but Explain is running on a DB2 2.1 or 2.2. subsystem, where the package concept is unknown. The statement is ignored and Explain proceeds to the next input record.

User response

None

```
ANL0362W INVALID COLLECTION AND
PACKAGE SPECIFICATION:
```

Explanation

An input record with the PKG (or Package) keyword has an incorrectly specified collection and package name combination. The Explain process ignores the specified value and continues processing the next input statement.

User response

Specify a valid PKG and collection name combination, and issue the request again.

ANL0364W	INVALID PACKAGE NAME:
	XXXXXXXX

Explanation

An input record with the PKG (or Package) keyword has an incorrectly specified package name. The Explain process ignores the specified value and continues processing the next input statement.

User response

Specify a valid PKG keyword, and issue the request again.

ANL0366W	NO LOCAL PACKAGE LIST FOUND
	FOR PLAN xxxxx

Explanation

The PLENTRIES column in the SYSIBM.SYSPLAN table has indicated that at least one package list is associated with the plan. The Explain process, however, is not able to find a row in SYSIBM.SYSPACKLIST for the specified plan (*xxxxx*). The statement is ignored and Explain proceeds to the

User response

next input statement.

Specify a valid package list for the specified plan, and issue the request again.

ANL0368W LOCATION CANNOT BE DETERMINED UNTIL EXECUTION TIME - PACKAGE IGNORED

Explanation

The LOCATION column in the SYSIBM.SYSPACKLIST table contains an asterisk (*) for the specified plan name and the given sequence number. This means that the Explain process has no way of knowing on what server location the package might be found. The package is ignored and Explain proceeds to the next package in the package list.

User response

Specify a valid LOCATION name for the specified plan and sequence number, and issue the request again.

ANL0370W PACKAGE xxxxxxx.yyyyyyy. (zzzzz) NOT BOUND WITH EXPLAIN OPTION - DYNAMIC EXPLAIN ISSUED

Explanation

The package specified as *xxxxxx.yyyyyy.(zzzzz)* after the PKG= or PACKAGE= keyword was not bound with the EXPLAIN option. Explain is performing a dynamic explain of the statements in the package and shows the result of this dynamic explain.

User response

None

ANL0372W	THE STATEMENT xxxxxxxx WILL
	RESULT IN <i>y</i> PACKAGES WITH A
	TOTAL OF z SQL STATEMENTS
	(INCLUDING NON-EXPLAINABLE).
	IF YOU WANT TO EXPLAIN THAT
	AMOUNT OF PACKAGES, YOU
	SHOULD SPECIFY
	xxxxx.FORCE=YES
	,

Explanation

The Explain process has found that the package statement that you have specified results in an explain of more than one package with a total of more than 300 SQL statements (counting both explainable and non-explainable). The statement is ignored and Explain proceeds to the next input statement.

User response

To explain the number of packages specified, reissue the request specifying the FORCE=YES subparameter.

```
ANL0374W NO PACKAGES FOUND FOR:
```

Explanation

The Explain process was not able to find any packages in SYSIBM.SYSPACKAGE that would satisfy the input specification. The statement is ignored and Explain proceeds to the next input statement.

User response

Specify a valid PACKAGE and issue the request again.

ANL0376W	NO PACKAGES FOUND FOR PLAN:
	<i>ххххххх</i> - PACKAGE: <i>уууууу</i>
	LOCATION: zzzzzz - COLL.ID:
	9999999

Explanation

The Explain process was not able to find any packages in SYSIBM.SYSPACKAGE for plan *xxxxxxx*. The statement is ignored and Explain proceeds to the next input statement.

User response

Specify a valid PACKAGE for the specified plan and issue the request again.

ANL0378W PACKAGE xxxxxx.yyyyyyy.(zzzzzz) HAS NO ROWS IN SYSIBM.SYSPACKSTMT

Explanation

The Explain process was not able to find any rows in SYSIBM.SYSPACKSTMT for the specified package in the specified collection. The statement is ignored and Explain proceeds to the next input statement.

User response

Specify a valid PACKAGE for the specified collection and issue the request again.

ANL0380W QMF QUERY authid.qname DOES NOT EXIST - REQUEST IGNORED

Explanation

A QMF query name specified after the QMF= keyword does not exist. If the Explain process is not the owner of the query, the query name must be prefixed with the authorization ID, for example, *authid.qname*. The statement is ignored and Explain proceeds to the next input statement.

User response

Verify that the query name is spelled correctly, and issue the request again.

ANL0382W QMF OBJECT authid.qname IS A xxxx AND NOT A QUERY -REQUEST IGNORED

Explanation

The QMF object specified as the query name is not a query, but rather a *xxxx*, where *xxxx* is a PROC or a FORM. The statement is ignored and the Explain process continues to the next input statement.

User response

Specify a valid query name, and issue the request again.

ANL0384W QUERY authid.qname IS A xxxx QUERY, MUST BE SQL - REQUEST IGNORED

Explanation

The QMF query specified as the query name is not an SQL query, but rather a *xxxx*, where *xxxx* is QBE or PROMPTED. The statement is ignored and the Explain process continues to the next input statement.

User response

If the QMF version is at least Version 2 Release 3, use the CONVERT command to transform the query into an SQL query.

```
ANL0386W QMF QUERY authid.qname IS
RESTRICTED TO OWNER -
REQUEST IGNORED
```

Explanation

The QMF query specified as the query name can only be explained by its owner. The statement is ignored and the Explain process continues to the next input record.

User response

To explain a QMF query created by a different authorization ID, save the query with the SHARE=YES parameter.

ANL0388W END OF QMF QUERY NOT FOUND -REQUEST IGNORED

Explanation

The QMF statement was found to be larger than 32 K bytes. Either the QMF query is more than 409 lines or the end-of-statement marker was not found. In either case, the Explain process ignores the request and proceeds to the next input record.

User response

Verify that the QMF statement is less then 32 K bytes, less than 409 lines, and that it contains an end-ofstatement marker, and issue the request again.

ANL0390W QMF QUERY xxxx.yyyy IS GREATER THAN 32K - REQUEST IGNORED

Explanation:

The requested QMF query (*xxxx.yyyy*) is longer than the maximum allowable length (32K). The specified query is ignored and processing continues with the next input statement.

User response:

No action is required.

ANL0391W NO STABILIZED STATEMENTS FOUND IN STABILIZATION GROUP

Explanation:

No stabilized statements were found in SYSIBM.SYSDYNQRY for the specified stabilization group. The statement is ignored and processing continues with the next input statement.

User response:

Specify a valid stabilization group, and issue the request again.

ANL0392W STABILIZED STATEMENT (xxxx) IS NOT FOUND

Explanation:

The specified statement ID (*xxxx*) was not found in SYSIBM.SYSDYNQRY. The statement is ignored and processing continues with the next input statement.

User response:

Specify a valid statement ID, and issue the request again.

ANL0393W STABILIZED STATEMENT xxxx IN STABILIZATION GROUP yyyy IS NOT FOUND

Explanation:

The specified statement ID (*xxxx*) with the specified stabilization group (*yyyy*) was not found in SYSIBM.SYSDYNQRY. The statement is ignored and processing continues with the next input statement.

User response:

Specify a valid statement ID for the stabilization group, and issue the request again.

```
ANL0394W INDEX INFORMATION NOT
INSERTED IN EEEWORK TABLE -
SQL-CODE: -xxx
```

Explanation

You requested a condensed index listing by specifying INDEX=Cond on the PLAN or PACKAGE statement. The Explain process was not able to save part of the index information in the EEEWORK table. The index parameter is ignored and Explain continues processing.

User response

None

```
ANL0396W OPEN OF C_WORK FAILED - SQL-
CODE: -xxx
```

Explanation

You requested a condensed index listing by specifying INDEX=Cond on the plan or package statement. The Explain process was not able to select the index information for the specified plan or package in the EEEWORK table. The index parameter is ignored and Explain continues processing.

User response

None

ANL0398W OPEN OF C_WORK2 FAILED - SQL-CODE: -xxx

Explanation

You requested a condensed index listing by specifying INDEX=Cond on the PLAN or PACKAGE statement. The Explain process was not able to select the index information for the specified plan or package in the EEEWORK table. The index parameter is ignored and Explain continues processing.

User response

None

```
ANL0420W LENGTH OF SQL STRING INVALID:
xxxxx - DEFAULT OF 79 USED
```

Explanation

You specified a line length of the printed SQL statement in the report. The length is invalid and has been set to 79 (default value). The Explain process ignores the specified value and continues processing.

User response

Specify a larger valid line length and issue the request again.

ANL0422W FORMAT OF SQL STRING INVALID: xxxxx - DEFAULT OF 79 USED

Explanation

You specified a line length of the printed SQL statement in the report. The format is invalid and the value has been set to 79 (default value). The Explain

process ignores the specified value and continues processing.

User response

Specify a valid line length in the correct format, and issue the request again.

ANL0424W LENGTH OF SQL STRING OUT OF RANGE: xxxxx - DEFAULT OF 79 USED

Explanation

You specified a line length of the printed SQL statement in the report. The value was outside the possible range (40 to 79). The Explain process ignores the specified value and continues processing using the default value of 79.

User response

Specify a line length within the valid range (40 to 79) and reissue the request.

ANL0426W QUERYNO xxxxx IS NOT NUMERIC - REQUEST IGNORED

Explanation

You specified a non-numeric QUERYNO as *xxxxx*. The QUERYNO must be specified as a numeric number. The Explain process ignores the specified value.

User response

Specify a numeric QUERYNO and issue the request again.

ANL0450W INVALID HINT COMMAND:

Explanation

You specified a SET CURRENT OPTIMIZATION HINT command that is in error. The first 50 bytes of the command is displayed. The Explain process ignores the specified value and continues processing using the default value of 79.

User response

Specify a valid SET CURRENT OPTIMIZATION HINT command and issue the request again.

ANL0452W HINT-ID NUMBER NOT FOUND IN EEEHINT TABLE - REQUEST IGNORED

Explanation

The hint ID number specified could not be found in the EEEHINT table. The statement is ignored and the Explain process continues to the next input statement.

User response

Specify a hint ID that is in the EEEHINT table and issue the request again.

ANL0454W HINT-ID WAS NOT ADDED TO THE PLAN_TABLE - REQUEST IGNORED

Explanation

The Explain process tried to update your PLAN_TABLE table with the specified hint ID, but the operation failed. The statement is ignored and Explain proceeds to the next input statement.

User response

None

```
ANL0456W HOST VARIABLE INFORMATION
COULD NOT BE GATHERED
```

Explanation

The Explain process tried to gather host variable information, but the operation failed. The host variable information is ignored and Explain processing continues.

User response

None

```
ANL0458W Package package_name must be bound before it can be used.
```

Explanation:

A package that was bound by using DB2 V9.1 or an earlier release was explained by using a DB2 11 subsystem.

User response:

Rebind the package by using a DB2 11 subsystem.

ANL0500E PREPARE OF xxxx FAILED - SQL-CODE: -yyy

Explanation

A prepare statement failed when attempting to prepare a select statement *xxxx*. A formatted print of the SQLCA and an error message are provided.

Check the non-zero SQL code (-*yyy*), and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0504E DECLARE OF xxxx FAILED - SQLcode: -yyy

Explanation

A declare statement failed when declaring the cursor *xxxx*. A formatted print of the SQLCA and an error message are provided.

User response

Check the non-zero SQL code (-*yyy*), and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0508E OPEN OF C_PLANx FAILED, SQL-CODE: -xxx

Explanation

An open statement failed when attempting to open the cursor C_PLAN*x* for S_DYN*x*, where *x* is A, 1, 2, 3, 4, 5, 6, or 7. A formatted print of the SQLCA and an error message are provided.

User response

Check the non-zero SQL code (-*xxx*), and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0510E RESET OF PRIMARY SQLID (auth.id) FAILED, WILL USE: auth.id

Explanation

The Explain process failed to set the primary SQL ID (primary authorization ID). The program continues using the logon user ID as the authorization ID. A

formatted print of the SQLCA and an error message is provided.

User response

Check the non-zero SQL code (-*xxx*), and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

```
ANL0512E
```

SETTING OF SECONDARY SQLID (auth.id) FAILED, WILL USE: auth.id

Explanation

The Explain process failed to set the secondary SQL ID (secondary authorization ID). The program continues using the primary authorization ID. A formatted print of the SQLCA and an error message is provided. This message is only be produced if running under DB2 Version 2.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0514E CURRENT SQLID COULD NOT BE OBTAINED - WILL USE AUTHORIZATION-ID: auth.id

Explanation

The Explain process failed to select the current SQL ID (secondary authorization ID). The program continues using the primary authorization ID. A formatted print of the SQLCA and an error message is provided. This message is only be produced if running under DB2 Version 2.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0516E

SETTING OF CURRENT DEGREE TO xxx FAILED, WILL USE DEFAULT VALUE

Explanation

The Explain process failed to set the degree of parallelism as specified. The program continues using the default value set by the installation. A formatted print of the SQLCA and an error message is provided. This message is only be produced if running under DB2 Version 2.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0520E	OPEN OF C_KEYS FAILED, SQL-
	CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSKEYS joined with SYSIBM.SYSINDEXES. If the SQL code is -911 or -913, the Explain process waits a short period and then tries to reprocess the current statement in the plan. Refer to message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

User response

Reissue the request when the resource is available.

Severity

8

ANL0522E	FETCH FROM C_KEYS FAILED,
	SQL-CODE: -xxx

Explanation

An error was detected during fetch from SYSIBM.SYSKEYS joined with SYSIBM.SYSINDEXES. If the SQL code is -911 or -913, the Explain process waits a short period and then tries to reprocess the current statement in the plan. Refer to message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

User response

Reissue the request when the resource is available.

Severity

8

ANL0524E OPEN OF C_IXCOL FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSKEYS joined with SYSIBM.SYSINDEXES and SYSIBM.SYSTABLES. If the SQL code is -911 or -913, the Explain process waits a short period and then tries to reprocess the current statement in the plan. Refer to message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

User response

Reissue the request when the resource is available.

Severity

8

ANL0526E

FETCH FROM C_IXCOL FAILED, SQL-CODE: -xxx

Explanation

An error was detected during fetch from SYSIBM.SYSKEYS joined with SYSIBM.SYSINDEXES and SYSIBM.SYSTABLES. If the SQL code is -911 or -913, the Explain process waits a short period and then tries to reprocess the current statement in the plan. Refer to message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

User response

Reissue the request when the resource is available.

Severity

8

ANL0528E OPEN OF C_STMT FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSSTMT. A formatted print of the SQLCA and

an error message is provided. The Explain function stops processing the current request and continues with the next input record.

User response

Check the non-zero SQL code (-xxx) and follow the instructions given in the DB2 Messages and Codes manual.

Severity

8

ANL0530E FETCH FROM C_STMT FAILED, SQL-CODE: -xxx

Explanation

An error was detected during a fetch from SYSIBM.SYSSTMT. A formatted print of the SQLCA and an error message is provided. The Explain function continues processing the current request.

User response

Check the non-zero SQL code (-xxx) and follow the instructions given in the DB2 Messages and Codes manual.

Severity

8

ANL0532E A DEADLOCK OR TIMEOUT HAS **OCCURRED - THE STATEMENT IS** REPROCESSED

Explanation

An SQL code of -911 or -913 was encountered. The Explain function stops processing the current miniplan or statement. A ROLLBACK call is issued to close any open cursors. After a short delay, The Explain function reprocesses the failing statement in the current plan.

User response

Reissue the request when the resource is available.

Severity

8

ANL0534E A DEADLOCK OR TIMEOUT HAS **OCCURRED 3 TIMES WITHIN THIS** PLAN

Explanation

An SOL code of -911 or -913 was encountered three times within processing of the same plan. A formatted print of the SQLCA and the error message is provided. The Explain function stops processing the current plan, and continues with the next input record.

User response

Reissue the request when the resource is available.

Severity

8

EXPLAIN OF A REMOTE OBJECT IS ANL0536E

NOT ALLOWED, SQL-CODE: -xxx

Explanation

An immediate EXPLAIN statement failed because the statement being explained contained a table located in another DB2 system. The EXPLAIN statement can only contain local objects. Explain continues with the next input record.

User response

Check the non-zero SQL code (-xxx) and follow the instructions given in the DB2 Messages and Codes manual.

Severity

8

ANL0538E **EXPLAIN OF SUPPLIED SQL-**STATEMENT FAILED, SQL-CODE: -XXX

Explanation

An immediate EXPLAIN statement failed. A formatted print of the SQLCA and an error message is provided together with the statement to be explained. Explain continues with the next input record.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the DB2 Messages and Codes manual.

Severity

8

ANL0540E

TABLE SPACE *dbname.tsname* NOT FOUND IN SYSIBM.SYSTABLESPACE - SQL-CODE: -*xxx*

Explanation

The Explain process was not able to find the named table space in SYSIBM.SYSTABLESPACE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0544E	COLUMN colname NOT FETCHED
	FROM SYSIBM.SYSCOLUMNS -
	SQL-CODE: -xxx

Explanation

The Explain process was not able to find the named table in SYSIBM.SYSCOLUMNS. If the SQL code is -911 or -913, Explain waits a short period and then tries to reprocess the current statement in the plan. Refer to message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0546E	INDEX authid.ixname NOT FOUND
	IN SYSIBM.SYSINDEXES - SQL-
	CODE: -xxx

Explanation

The Explain process was not able to find the named index in SYSIBM.SYSINDEXES. A formatted print of the SQLCA and an error message is provided.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8	
ANL0548E	PLAN <i>planname</i> NOT FOUND IN SYSIBM.SYSPLAN - SQL-CODE: - <i>xxx</i>

Explanation

The Explain process was not able to find the named plan in SYSIBM.SYSPLAN.

User response

Check the non-zero SQL code (-*xxx*), and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0550E OPEN OF C_KEYDIST FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSFIELDS. If the SQL code is -911 or -913, the Explain process waits for a short time and tries to reprocess the current statement in the plan. See message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

```
8
```

ANL0552E OPEN OF C_COLDIST FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSCOLDIST. If the SQL code is -911 or -913, the Explain process waits a short period and then tries to reprocess the current statement in the plan. Refer to message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0554E DATABASE *dbname* NOT FOUND IN SYSIBM.SYSDATABASE - SQL-CODE: -xxx

Explanation

The Explain process was not able to find the named database in SYSIBM.SYSDATABASE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0560E	OPEN OF C_TSKEYS FAILED, SQL-
	CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSKEYS. If the SQL code is -911 or -913, the Explain process waits a short period and then tries to reprocess the current statement in the plan. Refer to message ANL0532E. For other SQL codes, Explain continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0570E CURRENT APPLICATION ENCODING SCHEME COULD NOT BE FOUND - WILL USE 037 AS DEFAULT

Explanation

An error was detected when the Explain process tried to select the current application encoding scheme. Explain continues operation by using 037 as CCSID. This message is only issued for DB2 Version 8 and later releases.

User response

Specify a valid CCSID and issue the request again.

Severity

```
8
```

ANL0572E CONVERSION FROM UNICODE TO EBCDIC FAILED - RC = xx, REASON = yyy

Explanation

An error was detected during conversion of Unicode to EBCDIC. The Explain process continues operation. This message is only issued for DB2 Version 8 and later releases.

User response

Refer to the explanation for the RC and REASON codes in *z/OS Support for Unicode: Using Conversion Services*.

Severity

8

ANL0580E UNAUTHORIZED ACCESS TO THE Q.OBJECT_DIRECTORY TABLE -REQUEST IGNORED

Explanation

You have specified an input record with the QMF= keyword. DB2 has found that the Explain plan does not have select authorization to the Q.OBJECT_DIRECTORY table and returns an SQL code of -551 or -552. Explain will proceed to the next input record.

User response

Obtain authorization to access the Q.OBJECT_DIRECTORY table and issue the request again.

Severity

8

ANL0582E

UNAUTHORIZED ACCESS TO THE Q.OBJECT_DATA TABLE - REQUEST IGNORED

Explanation

You have specified an input record with the QMF= keyword. DB2 has found that the Explain plan does not have select authorization to the Q.OBJECT_DATA table and returns an SQL code of -551 or -552. Explain will proceed to the next input record.

User response

Obtain authorization to access the Q.OBJECT_DATA table and issue the request again.

Severity

8

ANL0584E ACCESS TO TABLE Q.OBJECT_DIRECTORY FAILED -SQL-CODE: -xxx

Explanation

You have specified an input record with the QMF= keyword. DB2 has found that the Explain plan does not have proper authorization to the Q.OBJECT_DIRECTORY table. A formatted print of the SQLCA and an error message is provided.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual. Explain will proceed to the next input record.

Severity

8

ANL0586E

ACCESS TO TABLE Q.OBJECT_DATA FAILED - SQL-CODE: -xxx

Explanation

You have specified an input record with the QMF= keyword. DB2 has found that the Explain plan does not have proper authorization to the Q.OBJECT_DATA table. A formatted print of the SQLCA and an error message is provided.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual. Explain will proceed to the next input record.

Severity

8

ANL0590E THE LOCATION FOR THE PLAN_TABLE TABLE COULD NOT BE OBTAINED - SQL-CODE: -xxx

Explanation

The installation has specified a location (database and table space name) for all newly created PLAN_TABLE tables. The location information could not be obtained from the table EEEWORK. The PLAN_TABLE table is created in the default database, and the processing continues.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0592E EITHER *dbname* OR *tsname* FOR THE PLAN_TABLE TABLE IS INVALID

Explanation

You have specified an invalid location (database and table space name) for all newly created PLAN_TABLE tables. The Explain process is not able to understand the database name or the table space name (one of the names might be blanks). The processing continues, but the PLAN_TABLE table is created in the default database.

User response

Specify a valid location (dbname or tsname) for the PLAN_TABLE tables and issue the request again.

Severity

8

ANLO600E THE EXPLAIN TABLE authid.PLAN_TABLE WAS NOT CREATED - SQL-CODE: -xxx

Explanation

The Explain process tried to create a PLAN_TABLE for user *authid*, but failed. The reason is probably that the CREATETAB authorization is missing. A formatted print of the SQLCA and an error message is provided. Explain stops processing. This message is preceded by message ANL0100E.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

12

ANL0602E ADDITION OF THREE V3R1 COLUMNS TO authid.PLAN_TABLE FAILED - SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER the three columns PREFETCH, COLUMN_FN_EVAL, and MIXOPSEQ to the existing PLAN_TABLE. One or more of the three ALTER statements failed. Explain continues processing using the pre-V3R1 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0604E ADDITION OF TWO V2R3 COLUMNS TO authid.PLAN_TABLE FAILED - SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER the two columns VERSION and COLLID to the existing PLAN_TABLE. One or more of the two ALTER statements failed. Explain continues processing using the pre-V2R3 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0606E ADDITION OF FOUR V3R1 COLUMNS TO authid.PLAN_TABLE FAILED - SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER the four columns ACCESS_DEGREE, ACCESS_PGROUP_ID, JOIN_DEGREE, and JOIN_PGROUP_ID to the existing PLAN_TABLE. One or more of the four ALTER statements failed. Explain will continue processing using the pre-V3 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0608E ADDITION OF NINE V4R1 COLUMNS TO authid.PLAN_TABLE FAILED - SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER the nine columns SORTC_PGROUP_ID, SORTN_PGROUP_ID, PARALLELISM_MODE, MERGE_JOIN_COLS, CORRELATION_NAME, PAGE_RANGE, JOIN_TYPE, GROUP_MEMBER, and IBM_SERVICE_DATA to the existing PLAN_TABLE. One or more of the nine ALTER statements failed. Explain will continue processing using the pre-V4 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0610E CONNECTION TO DB2 NOT ESTABLISHED - SQL-CODE: -xxx

Explanation

The Explain process could not establish a connection to DB2. This error message is followed by message ANL0612E or message ANL0614E.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0612E VERIFY YOUR EXECUTE AUTHORIZATION TO THIS PLAN PROGRAM IS TERMINATED WITH RETURN CODE: 12

Explanation

A possible reason for this message is that you are missing authorization for the Explain process. This message is preceded by message ANL0610E.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

12

ANL0614E TIMESTAMP DIFFERENCE BETWEEN THE DBRM AND THE LOAD MODULE PROGRAM IS TERMINATED WITH RETURN CODE: 12

Explanation

There is a timestamp conflict between the DBRM and the load module. This message is preceded by message ANL0610E.

User response

If a new load module has been created, make sure that a new BIND is being performed to use the corresponding DBRM. Check the non-zero SQL code (*xxx*), and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

12

ANL0618E PLAN_TABLE IS NOT COMPATIBLE WITH DB2 REL. *x.y* SQL-CODE: *xxx*

Explanation

The format of the PLAN_TABLE is not compatible with the current release of DB2.

User response

Issue an SQL=-statement. This updates the PLAN_TABLE to the same level as your DB2 system. Then reissue the explain request.

Severity

8

ANL0620E UPDATE OF DYNAMICALLY EXPLAINED DATA FAILED, SQL-CODE: -xxx

Explanation

Either the EXPLAIN information did not exist in the catalog, or the user specifically requested a dynamic EXPLAIN. A dynamic EXPLAIN was executed and the subsequent update to the EXPLAIN results resulted in an error and the update failed. Processing continues.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

```
8
```

```
ANL0630E OPEN OF C_AUTH FAILED, SQL-
CODE: -xxx
```

Explanation

An error was detected while opening a cursor for *binder*.EEEAUTH. Default authorization values are used (see message ANL0160I). The processing continues.

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0636E

OPEN OF C_EEEPL FAILED, SQL-CODE: -xxx

Explanation

An error was detected while opening a cursor for *binder*.EEE_ANLPLAN. Printing of plan history information is stopped. The processing continues.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0638E	OPEN OF C_EEED2 FAILED, SQL-		
	CODE: -xxx		

Explanation

An error was detected while opening a cursor for *binder*.EEEDBRM. Printing of DBRM information is stopped. The processing continues.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0640E OPEN OF C_STMTX FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSSTMT. Saving plan information is stopped. The Explain process continues normal processing.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0642E SELECT FROM EEEPLAN FAILED, SQL-CODE: -xxx

Explanation

An error was detected when selecting a row from *binder*.EEEPLAN. Saving plan information is stopped. The processing continues.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

- 8
- ANL0644E OPEN OF C_EEED1 FAILED, SQL-CODE: -xxx

Explanation

An error was detected while opening a cursor for *binder*.EEEDBRM. Saving of DBRM information is stopped. The processing continues.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0646E FETCH FROM C_EEED1 FAILED, SQL-CODE: -xxx

Explanation

An error was detected while opening a cursor for *binder*.EEEDBRM. Saving of DBRM information is stopped. The processing continues.

Check the non-zero SQL code (-*xxx*), and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANLO650E MAX

MAXIMUM HINT ID COULD NOT BE FOUND - SQL-CODE: -xxx

Explanation

The Explain process was not able to find the maximum hint ID number used in the EEEHINT table. Consequently, Explain was not able to assign a hint ID number for the current operation. No hint ID was assigned. Explain will continue with normal processing.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0652E	INSERT IN EEEHINT TABLE
	FAILED - SQL-CODE: -xxx

Explanation

The Explain process was trying to store the current hint ID number in the EEEHINT table, but the operation did not succeed. Explain will continue with normal processing.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0654E INSERT IN EEEPATH TABLE FAILED - SQL-CODE: -xxx

Explanation

The Explain process was trying to store access path information in the EEEPATH table, but the operation

did not succeed. Explain will continue with normal processing.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0660E

ADDITION OF THREE V5R1 COLUMNS TO *authid*.PLAN_TABLE FAILED - SQL-CODE: -*xxx*

Explanation

The Explain process tried to ALTER the three columns WHEN_OPTIMIZE, QBLOCK_TYPE, and BIND_TIME to the existing PLAN_TABLE. One or more of the three ALTER statements failed. Explain will continue processing using the pre-V5 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0662E ADDITION OF THREE V6R1 COLUMNS TO authid.PLAN_TABLE FAILED - SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER the three columns OPTHINT, HINT_USED, and PRIMARY_ACCESSTYPE to the existing PLAN_TABLE. One or more of the three ALTER statements failed. Explain will continue processing using the pre-V6 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0664E

ADDITION OF TWO V7R1 COLUMNS TO *authid*.PLAN_TABLE FAILED - SQL-CODE: -*xxx*

Explanation

The Explain process tried to ALTER the two columns PARENT_QBLOCKNO and TABLE_TYPE to the existing PLAN_TABLE. One or more of the two ALTER statements failed. Explain will continue processing using the pre-V7 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0666E	ADDITION OF nn VxRy COLUMNS
	TO authid.explain_table FAILED -
	SQL-CODE: -xxx

Explanation

The Explain process tried to add *nn* columns to the existing explain table for DB2 Version *x* and Release *y*. One or more of the additions failed. Explain will continue processing using the old format of the explain table.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0680E ALTER TABLE ... COLUMN FAILED FOR authid.PLAN_TABLE - SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER ten columns of the existing PLAN_TABLE. One or more of the column alters failed. Explain will continue processing using the pre-V8 format of the PLAN_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0682E	ALTER TABLE COLUMN FAILED
	FOR
	authid.DSN_STATEMNT_TABLE -
	SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER two columns of the existing DSN_STATEMNT_TABLE. One or more of the column alters failed. Explain will continue processing using the pre-V8 format of the DSN_STATEMNT_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0684E	ALTER TABLE COLUMN FAILED
	FOR
	authid.DSN_FUNCTION_TABLE -
	SQL-CODE: -xxx

Explanation

The Explain process tried to ALTER nine columns of the existing DSN_FUNCTION_TABLE. One or more of the column alters failed. Explain will continue processing using the pre-V8 format of the DSN_FUNCTION_TABLE.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0690E ADDITIONAL xxx STATEMENTS WITH NUMBERS GREATER THAN 32767 WERE FOUND

Explanation

The plan or package explained had SQL statements with statement numbers greater than 32767. These statements are all given a statement number of 0 (zero) in the DB2 SYSSTMT / SYSPACKSTMT catalog tables and can therefore not be related to the individual corresponding rows in the PLAN_TABLE table. The given number of statements are not shown in the Explain report.

User response

Split the application code into smaller source code modules before precompiling takes place.

Severity

8

ANL0704E SELECT FROM SYSIBM.SYSPACKLIST FAILED -SQL-CODE: -xxx

Explanation

An error was detected during selecting from SYSIBM.SYSPACKLIST. The Explain process continues with the next input statement, after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0706E	OPEN OF C_PACKAGE_NAVN
	FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSPACKAGE. The Explain process continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

```
8
```

```
ANL0708E OPEN OF C_PACKAGES FAILED,
SQL-CODE: -xxx
```

Explanation

An error was detected during opening a cursor for SYSIBM.SYSPACKAGE. The Explain process continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0710E OPEN OF C_PACK_STMT FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSPACKSTMT. The Explain process continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

```
ANL0712E FETCH FROM C_PACK_STMT
FAILED, SQL-CODE: -xxx
```

Explanation

An error was detected during fetching a row from SYSIBM.SYSPACKSTMT. The Explain process continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

	_			
		1		
è		1	ĥ	
•			,	

ANL0714E OPEN OF C_SYSPACK_STMT FAILED, SQL-CODE: -xxx

Explanation

An error was detected during opening a cursor for SYSIBM.SYSPACKSTMT. The Explain process continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0716E	OPEN OF C_SDQ_STMT FAILED,
	SQL-CODE: -xxx

Explanation

An error was detected when opening a cursor for SYSIBM.SYSDYNQRY. Processing continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

```
ANL0718E FETCH FROM C_SDQ_STMT
FAILED, SQL-CODE: -xxx
```

Explanation

An error was detected while fetching a row from SYSIBM.SYSDYNQRY. Processing continues with the current request after printing the formatted SQLCA and the error message.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8	
ANL0720E	CURRENT SERVER NAME COULD NOT BE ACQUIRED - SQL-CODE: -
	XXX

Explanation

An error was detected during selecting the CURRENT SERVER special register. The Explain process continues processing using 16 blanks as the current server identification.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

8

ANL0722E CONNECT RESET FAILED - SQL-CODE: -xxx

Explanation

An error was detected during a CONNECT RESET operation. The Explain process continues processing using the current specified server.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

```
8
ANL0724E CONNECT TO xxxxx FAILED -
(NON-CONNECTABLE STATE) -
SQL-CODE: -752
```

Explanation

An error was detected during a CONNECT TO :*location* operation. The Explain process issues a Connect Reset and continues processing on the home server location.

User response

Specify a valid location and issue the request again.

8

ANL0726E CONNECT TO xxxxx FAILED -(SERVER NAME NOT KNOWN) -SQL-CODE: -950

Explanation

An error was detected during a CONNECT TO :*location* operation. The Explain process issues a Connect Reset and continues processing on the home server location.

User response

Specify a valid location and issue the request again.

Severity

8

ANL0728E CONNECT TO xxxxx FAILED - SQL-CODE: -xxx

Explanation

An error was detected during a CONNECT TO :*location* operation. The specified server name is acceptable and the process is in a connectable state. The Explain process issues a Connect Reset and continues processing on the home server location.

User response

Check the non-zero SQL code (-*xxx*) and follow the instructions given in the *DB2 Messages and Codes* manual.

Severity

ANL0730E

8

SERVER IS NOT A DB2 SUBSYSTEM - WILL ISSUE A CONNECT RESET

Explanation

A connect to a remote server location has been established, but Explain has found that the remote server is not a DB2 system. The Explain process issues a CONNECT RESET command to connect to the home server location.

This message is preceded by message ANL0180I that explains the server type. Explain continues processing at the home location.

User response

Verify that the database on the remote server is a DB2 system. If not, specify a valid location, and issue the request again.

Severity

8

ANL0801W PACKAGE xxxxxxx BOUND WITH EXPLAIN OPT - USER REQUESTED DYNAMIC EXPLAIN

Explanation:

The user indicated that a dynamic EXPLAIN should be executed instead of displaying the EXPLAIN information for the package (xxxxxxx) that was created as a result of binding with EXPLAIN(YES).

User response:

None

ANL0802W USER REQUESTED DYNAMIC EXPLAIN FOR xxxxxxxx

Explanation:

The user indicated that a dynamic EXPLAIN should be executed instead of displaying the stabilized EXPLAIN information for the stabilized dynamic query specified (*xxxxxxx*).

User response:

If a dynamic EXPLAIN was expected, no action is required. If you would prefer to view the stabilized EXPLAIN information, enter "S" in the "Type of EXPLAIN" field on the "Process a stabilized dynamic query" panel.

ANL0880E DB2 release level unknown release level assumed to be V8

Explanation

The Explain process was not able to determine the version and release level of the current DB2 subsystem. Explain continues processing assuming that the version and release level is 7.1.

User response

None

Severity

8

ANL0900E	EXPLAIN ABNORMALLY
	TERMINATES WITH RETURN
	CODE: 16

Explanation

The Explain On Error Unit has been activated. A snap dump is being produced. Examine the dump to determine the reason for the error. If the Explain program has failed, a PL/I On Code is printed in the snap dump. Explain stops processing.

User response

Contact the IBM Software Support Center.

Severity

16

ANL1000W PARSING IS INCOMPLETE AND IN DOUBT FOR STATEMENT: nnnnnnnn.PARSING FAILED, BUT PROCESSING CONTINUES. ERRCODE = cccccc.

Explanation

The parser has encountered a sequence of SQL clauses it is unable to handle, possibly due to an incorrect SQL statement combining clauses that are illegal or the parser did not anticipate their use together. The parser gives up and the processing continues, although the cost estimate for this statement might be inaccurate.

User response

Check SQL with EXPLAIN to see if it is valid. If valid, contact IBM Software Support.

ANL1001E A processing error was encountered during SQL statement analysis. ANLOUT output trace file was not defined.

Explanation

One of the files required for processing was not properly allocated. ANLOUT holds the SQL used for reporting and display after the parser checks syntax.

User response

Check for an ANLOUT DD card in batch JCL. For all other environments, contact IBM Software Support.

ANL1002E ALIGNMENT ERROR IN DBRM READ, RELEASE = c, DID NOT POSITION ON RECORD. The record rec is bypassed and the next valid record in the DBRM is searched for.

Explanation

During the processing of DBRM input, the parser has incorrectly read the offsets of blocks within, such that it is no longer properly positioned on a valid control block inside the DBRM. SQL PA supports several versions of DBRM and the recognition is automatic (RELEASE = c indicates the version).

User response

Verify that the DBRM is stable and error free. Then try running it again.

ANL1003E End of DBRM file detected before end of records indicated there. Pgm terminates because the DBRM module is unusable. Check for completeness and rebind if necessary, then try again.

Explanation

A DBRM module indicated several blocks to follow, but the EOF marker was found before the last block indicated might need to rebind DBRM.

User response

Verify that the DBRM is stable and error free. Then try running it again.

ANL1004W Concatenation of multiple DBRMs into ANLIN is not allowed. Only the first DBRM was processed.

Explanation

More than one DBRM data set was concatenated into the ANLIN allocation. Processing stops after the first DBRM.

User response

Allocate subsequent DBRM data sets as an unused ddname from ANLIN00 to ANLIN99 and rerun.

ANL1005E A processing error was encountered during SQL statement analysis. ANLIN input SQL file has not been defined.

Explanation

The input file required for processing was not properly allocated or defined. ANLIN holds the SQL used for analysis and is a sequential or PDS member, or an entire PDS when DBRMKEY parameter is also active.

Check for an ANLIN DD card in batch JCL. For TSO, check Input Data Set Name on the Main panel to make sure it exists. It cannot be an empty member. For all other environments, contact IBM Software Support.

ANL1007I UNABLE TO SET CURRENT SQLID TO SQL PA SECONDARY AUTHID: xxxxxxxx. PGM WILL USE PLAN_TABLE UNDER PRIMARY AUTHID yyyyyyyy INSTEAD. AUTOMATIC CORRECTIVE ACTION TAKEN BY PGM. Processing continues.

Explanation

An SQL error code was encountered, indicating the unavailability of a generic ID. SQL PA automatically reverts to the plan table for the primary authorization ID, if available.

User response

Check the assignment of generic IDs in the REGISTRY table. Ensure that the primary authorization ID owns a PLAN_TABLE.

ANL1008E There are no available generic IDs unused at this time. (Additional information about the cause of the error will be included in the message.)

Explanation

An SQL error code of 100 was encountered during registration of the generic ID when all explain tables were busy during the time SQL PA tried to start. Either there is an insufficient number of generic IDs available for the number of simultaneous users (usually 10), or an error has left all of the generic IDs hung as "in use" erroneously. Normally, SQL PA automatically resets these entries when full.

User response

Assign more generic IDs if fewer than 10 or if all of the generic IDs are in use; otherwise locate and run member *ssid*RSET in *hiqual*.SANLSQL to reset the registry. This member is designed to run under TSO SPUFI.

ANL1009W PGM CANNOT DETERMINE THE COST OF STATEMENT: nnnnnnn, SQLCODE = cccc. ONE OR MORE HOST VARIABLES CANNOT BE CORRECTLY SUBSTITUTED WITH MARKER. PARAMETER MARKERS (?) ARE ILLEGAL IN SOME EXPLAIN REQUESTS. Or, AN ILLEGAL SYMBOL HAS BEEN ENCOUNTERED WITHIN THE SQL. The statement is ignored and processing continues.

Explanation

One of several SQL error codes were encountered: -104, -312 or -314, all indicating an incompatibility with the parameter marker and statement processed. Explain does not allow ? to hold place on both sides of expression, as part of concatenation, and several other instances. SQL PA catches use of ? in the SELECT list, and that is corrected dynamically. The other uses of ? are not allowed. SQL PA does not cost this query, but continues to process others.

User response

Recode the SQL to use allowed predicate structure. Or replace the ? with your guess of the proper values and run again.

ANL1010W	PGM CANNOT DETERMINE THE
	COST OF STATEMENT: nnnnnnnn,
	SQLCODE = cccc. THE PREPARED
	FORM OF THE STATEMENT WAS
	UNACCEPTABLE TO DB2. The
	statement is ignored and
	processing continues.

Explanation

An SQL error code -518 or -084 was encountered, indicating DB2 could not prepare the statement because of its SQL content. SQL PA moves on to the next explainable statement. Sometimes, a -518 is caused by SQL not properly aligned in columns 1-72, in which case SQL PA scans only part of the query.

User response

Check the input file and correct any errors.

ANL1011E SQL PA cannot find an explanation table for *user_ID* because it does not exist or cannot be accessed. The SQL code is *SQL_code*. Processing has stopped.

Explanation

An SQL error code of 219 or -219 was encountered, indicating the absence of the required explanation table.
Create the explanation table and run again.

ANL1012E SQL PA cannot use an explanation table for *user_ID* because the table is not in the correct format. The SQL code is *SQL_CODE*. Processing has stopped.

Explanation

The explanation table was not correctly created or is not at the proper level to support this release of DB2. SQL error codes -221, -200, or 220 can occur.

User response

Correct the explanation table and run again.

ANL1013E PGM TRYING TO USE authid.PLAN_TABLE, SQLCODE = cccc. AN AUTHORIZATION ERROR WAS ENCOUNTERED. PGM DOES NOT HAVE PROPER ACCESS. PGM TERMINATES THIS RUN.

Explanation

An SQL error code -922 was encountered, indicating that SQL PA was not properly installed, and one or more required tables cannot be accessed.

User response

Authorize SQL PA to access the tables. If the error continues, retrace SQL PA steps.

ANL1014W PGM CANNOT DETERMINE THE COST OF STATEMENT: nnnnnnn, SQLCODE = cccc. THE PREPARED STATEMENT IS TOO LONG OR COMPLEX FOR DB2 TO HANDLE. The statement is ignored and processing continues.

Explanation

SQL PA got an SQL error code -101 indicating that DB2 cannot parse the SQL statement, which is too long or too complex. SQL PA continues with other SQL statements in the run.

User response

Correct and run the SQL statement again.

ANL1015W PGM CANNOT DETERMINE THE COST OF STATEMENT: nnnnnn, SQLCODE = cccc. THE PREPARED STATEMENT REFERS TO A REMOTE OBJECT NOT ON THIS SYSTEM. The statement is ignored and processing continues.

Explanation

SQL PA does not do a remote Explain. Therefore, it cannot explain distributed SQL statements that require objects on other systems and catalogs. SQL PA continues to the next statement.

User response

Run again, using VIADRDA to route the SQL to the remote system for processing.

ANL1016E	A fatal SQL error was encountered
	during product registration.
	SQLCODE = ccccc, IOS OPERATION
	= <i>x</i> .

Explanation

SQL PA was unable to register and assign an ANLUSER *n authid* for SQL PA processing. The REGISTRY table might be damaged or inoperable. See additional reason codes.

User response

Start ANLRESET on the ANLUSERO.REGISTRY table, and run again.

The message might also show: **INCOMPATIBILITY BETWEEN DBRM & PROGRAM** IS INDICATED: REBIND ANLSQLPA... only if SQLCODE -818 ERROR OCCURRED DURING DECLARE FOR REGISTRY CURSOR ... if IOS = 1 ERROR OCCURRED DURING SET CURRID TO CURRENT SQLID... if IOS = 2 ERROR OCCURRED DURING OPENING OF REGISTRY CURSOR ... if IOS = 3 ERROR OCCURRED DURING FETCH OF REGISTRY ENTRIES... if IOS = 4 ERROR OCCURRED DURING UPDATING OF REGISTRY TABLE ... if IOS =5 ERROR OCCURRED DURING COMMIT OF REGISTRY UPDATE... if IOS = 6 ERROR OCCURRED DURING SET CURRENT SQLID TO ANLID... if IOS = 7 ERROR OCCURRED USING THE PRIMARY AUTHID PLAN TABLE ... if IOS = 8 ERROR OCCURRED DURING CAF CONNECTION (OPEN OR CLOSE)... if IOS = 9 ERROR OCCURRED DURING RESET REGISTRY ENTRIES > 1 HR OLD... if IOS = 10

ERROR OCCURRED DURING READING OF ANLCNTL PARAMETERS... if IOS = 11 ERROR OCCURRED DURING READING OF ANLPARM PARAMETERS... if IOS = 12 ERROR OCCURRED DURING USE OF SETPLAN AUTHID PLAN_TABLE... if IOS = 20 ERROR OCCURRED DURING ASSIGNMENT OF SETPLAN AUTHID P_T... if IOS = 21 ERROR OCCURRED BECAUSE SETPLAN PLAN TABLE IS NOT USABLE... if IOS = 22

ANL1017W THE INPUT FILE SPECIFIED BY ddname IS TOO LONG... SOME TRUNCATION OR FILLER MAY OCCUR DUE TO RECORD SIZE. Processing continues. COND = cccc.

Explanation

The input to SQL PA was not a DBRM module or a flat, sequential file with LRECL = 80, RECFM = FB. While possible to handle other record sizes, typically 80 is specified for operational use.

User response

None; use 80 character input in future runs.

ANL1019W PGM CANNOT DETERMINE THE COST OF STATEMENT: xxxxxxx, SQLCODE = nnnn LACK AUTHORITY TO PERFORM PREPARE/EXPLAIN ON OBJECTS IN QUERY. The statement is ignored and processing continues.

Explanation

The person starting SQL PA does not have prepare or explain privileges on one or more objects in query. SQLCODE returns -551.

User response

The running authorization ID does not have authority to issue an EXPLAIN on table. Use the SQL statement GRANT SELECT ON TABLE tablename TO authid to allow the user to access the table.

ANL1020E PROCESSING ERROR ENCOUNTERED DURING SQL STATEMENT ANALYSIS. NO EXPLAINABLE SQL STATEMENTS WERE FOUND.

Explanation

The input SQL did not contain a valid SELECT, INSERT, UPDATE, DELETE, or REFRESH statement. Parser syntax failed to find an explainable SQL statement suitable for analysis.

User response

Check SQL for an explainable statement.

ANL1021E PROCESSING ERROR WAS ENCOUNTERED DURING SQL STATEMENT ANALYSIS. MORE THAN 4096 RECORDS FOR THE APAS ARRAY.

Explanation

One of the internal array structures used to hold parser intelligence has overflowed.

User response

Reduce the number of SQL statements to SQL PA and run again.

ANL1022E	PROCESSING ERROR WAS
	ENCOUNTERED DURING SOL
	STATEMENT ANALYSIS. MORE
	THAN maximum number
	RECORDS FOR THE ARELS ARRAY.

Explanation

One of the internal array structures used to hold referential integrity information has overflowed. For ANLSQLPA, the maximum number of records is 4096. For ANLQMF and the stored procedures, the maximum number of records is 512.

User response

Reduce the number of SQL statements to SQL PA and run again.

ANL1023E PROCESSING ERROR WAS ENCOUNTERED DURING SQL STATEMENT ANALYSIS. MORE THAN maximum_number RECORDS FOR THE AEXP ARRAY.

Explanation

One of the internal array structures used to hold access plan data has overflowed. For ANLSQLPA, the maximum number of records is 4096. For ANLQMF and the stored procedures, the maximum number of records is 512.

Reduce the number of SQL statements to SQL PA and run again.

ANL1024E PROCESSING ERROR WAS ENCOUNTERED DURING SQL STATEMENT ANALYSIS. MORE THAN maximum_number RECORDS FOR THE ASTAT ARRAY.

Explanation

One of the internal array structures used to hold catalog statistics has overflowed. For ANLSQLPA, the maximum number of records is 4096. For ANLQMF and the stored procedures, the maximum number of records is 512.

User response

Reduce the number of SQL statements to SQL PA and run again.

ANL1026W SQL PA CANNOT DETERMINE THE COST OF STATEMENT: nnnnnnn, SQLCODE = cccc. THE TABLE NAME IS UNDEFINED OR ELSE REFERENCES A SYNONYM NOT DETECTED. STATEMENT IGNORED AND PROCESS CONTINUES.

Explanation

An SQL error code -204 was encountered, where DB2 was unable to PREPARE the SQL statement, due to its inability to recognize the table name or high-level qualifier; this is a DB2 limitation in certain prepares. SQL PA ignores the statement and continues. This message occurs in QMF version only.

User response

Fix the SQL statement and run again.

ANL1027I UNNECESSARY USE OF SEMICOLON TO END QMF STATEMENT. CHARACTER IGNORED (SQLCODE +98) AND SQL PA CONTINUES.

Explanation

A QMF statement ended with a semicolon (;), which is not necessary under QMF. The Prepare routine issued the +98 error code, and SQL PA removed the semicolon and prepared the statement again automatically.

User response

None

One of the following unnumbered messages could occur in QMF:

• ONE OR MORE LIMITS EXCEEDED BY YOUR QUERY PER SQL PA GOVERNOR EXIT or ONE OR MORE LIMITS EXCEEDED BY YOUR QUERY-ANL HAS TERM I/O ERROR.

Your query exceeded the installation coded limits for CPU, Elapsed Time or I/O estimates. Written to the QMF log. You might be given the option to cancel this query before starting. Term I/O error if DB2 cannot ask you WOULD YOU LIKE SQL PA TO CANCEL THIS QUERY (Y/N)?

• YOUR QUERY HAS BEEN CANCELLED BY THE SQL PA GOVERNOR EXIT.

The SQL PA Governor can prevent queries from running. The installation might choose to enforce this option when installing ANLGOV1 or leave the decision to the user. In either case, run the query in Batch mode.

ANL1028W	ERROR ACCESSING
	ANLn.DSN_STATEMNT_TABLE,
	SQLCODE = ssssss. PGM IGNORES
	TABLE AND CONTINUES
	PROCESSING QUERIES. QUERYNO
	NOT FOUND WAS nnnnn,
	PLANCOUNT = ccccc.

Explanation

SQL PA could not access the optional generic DSN_STATEMNT_TABLE, because it was not created or was damaged (possible SQL return codes are -312, -204 or +100). Users who specify PRECISE YES want to use the Optimizer's Instruction Count (Path Length) estimate, provided by this Explain table. SQL PA ignores the table and moves on, using its own estimate instead. In Version 8 systems, SQL PA attempts to create one before abandoning efforts.

User response

If needed, create DSN_STATEMNT_TABLE and run the query again.

ANL1029E A processing error was encountered during SQL statement analysis. The SYSPRINT output log file was not defined.

One of the files required for processing was not properly allocated. SYSPRINT holds the output messages and Cost Summary analysis for all SQL analyzed.

User response

Check for a SYSPRINT DD card in batch JCL. For all other environments, contact IBM Software Support.

ANL1030E ANLCNTL FILE NOT ALLOCATED.

Explanation

This message is associated with the SQL PA stored procedure. An ANLCNTL DD statement is not defined in the stored procedure address space (DSNSPAS or WLMSPAS) startup JCL. SQL PA is unable to use any installation defined parameters to describe the configuration.

User response

Add an ANLCNTL DD statement identifying the target host configuration file to the DSNSPAS or WLMSPAS procedure in SYS1.PROCLIB.

ANL1031W DB2 OBJECT NOT FOUND (TABLE OR COLUMN NAME) queryno, SQLCODE = cccc.

Explanation

DB2 did not correctly prepare the statement because a table name or similar correlation was used as the high-level qualifier. Although valid in SQL, this does not work under a prepare of SQL statements.

User response

Remove the offending high-level qualifier and try the query again.

ANL1032I PRECISE YES SELECTED BUT USER HAS NO DSN_STATEMNT_TABLE UNDER authid. NORMAL COSTING WILL BE EMPLOYED.

Explanation

You have requested to use the optimizer's CPU and Service Unit estimates with Precise Yes parameter, but the required explain table is not available to obtain the information. SQL PA might attempt to create a DSN_STATEMNT_TABLE. Otherwise, it reverts to *Precise No* costing formulas.

User response

Build generic DSN_STATEMNT_TABLE entries for general use or refer to an existing table using the USEPLAN parameter to identify the *authid*.

ANL1033W DB2 CANNOT PREPARE AN UPDATE OR DELETE WHERE CURRENT OF CSR BECAUSE CURSOR IS UNKNOWN. PGM WILL HANDLE SITUATION AND REPROCESS STATEMENT.

Explanation

An UPDATE or DELETE statement was processed by SQL PA that contained the WHERE CURRENT OF CURSOR clause. Because SQL PA is preparing the statement independently, there is no Declare Cursor to reference. SQL PA automatically handles the situation internally, adjusting for the update or delete in place. SQL code is -504.

User response

None

ANL1034E PROCESSING ERROR ENCOUNTERED DURING SQL STATEMENT ANALYSIS. ANLWORK TEMPORARY WORKFILE WAS NOT DEFINED.

Explanation

One of the files required for processing was not properly allocated. ANLWORK holds the SQL extracted from multiple DBRM members.

User response

Check for an ANLWORK DD card in batch JCL. For all other environments, contact IBM Software Support.

```
ANL1035E UNABLE TO PROCESS MULTIPLE
DBRM MEMBERS BECAUSE ANLIN
PDS COULD NOT BE OPENED
SUCCESSFULLY. RETURN CODE =
rc, REASON = rs, INFO = in.
```

Explanation

The DBRM library named *ddname* (filename) ANLIN could not be opened by the BPAM access routine.

User response

Check the spelling to verify that you have specified the data set name correctly. Attempt to access this data

set using the ISPF Edit or Browse to ensure the directory is intact. If this does not resolve the problem, provide the return code, reason, and info code to the IBM Software Support Center.

ANL1036I The following ddnames are not valid and are ignored when DBRMKEY is active.

Explanation

ANLIN00 - ANLIN99 are only available when processing a DBRM with DBRMKEY set to +OFF+.

User response

To process the ignored ddnames, allocate ANLIN as a fully qualified DBRM, set DBRMKEY to +OFF+, and rerun.

ANL1037I The following ddnames are not valid and are ignored when ANLIN is not a DBRM.

Explanation

ANLIN00 - ANLIN99 are only available when processing a DBRM with DBRMKEY set to +OFF+.

User response

To process the ignored ddnames, allocate ANLIN as a fully qualified DBRM, set DBRMKEY to +OFF+, and rerun.

ANL1038I The following ddnames are not valid and are ignored when ANLIN is in error.

Explanation

ANLIN00 - ANLIN99 are only available when processing a DBRM with DBRMKEY set to +OFF+.

User response

Correct the allocation of ANLIN and rerun.

ANL1039E UNABLE TO PROCESS MULTIPLE DBRM MEMBERS BECAUSE DSERV (LIST PDS DIRECTORY) FAILED. RETURN CODE = rc, REASON = rs, INFO = in.

Explanation

The directory of the DBRM library named *ddname* (filename) ANLIN could not be read by the BPAM access routine.

User response

Check that this directory is not corrupted by using the TSO LISTDS data set ME command, or use ISPF Edit or Browse to ensure the directory is intact. If this does not resolve the problem, provide the return code, reason, and info code to the IBM Software Support.

ANL1040E DIRECTORY MEMBERS EXCEED THE LIMIT OF 2730—ONLY THE FIRST 2730 MEMBERS WILL BE PROCESSED. RETURN CODE = rc, REASON = rs, INFO = in.

Explanation

The DBRMKEY pattern has returned a match on more than 2730 members, the limit for a single run. The first 2730 members are handled.

User response

Restate the DBRMKEY pattern so that less than 2730 members are selected, or run several jobs to pick up all of the required members by modifying the pattern.

ANL1041E	UNABLE TO PROCESS MEMBER
	member-name. THIS NAME WAS
	NOT FOUND IN THE DIRECTORY.
	RETURN CODE = <i>rc</i> . REASON = <i>rs</i> .
	INFO = in.

Explanation

The member name specified for this DBRM library could not be located in the directory by the BPAM access routine.

User response

Check the spelling to verify that you have specified the member name correctly. Attempt to access this data set by using ISPF Edit or Browse to ensure the directory is intact. If this does not resolve the problem, provide the return code, reason, and info code to the IBM Software Support.

ANL1043E DBRM MEMBER memname HAD A READ ERROR AND CANNOT BE PROCESSED. RETURN CODE = rc, REASON = rs, INFO = in.

Explanation

The member name specified for this DBRM library could not be read by the BPAM access routine.

Check the spelling to verify that you have specified the member name correctly. Attempt to access this data set by using ISPF Edit or Browse to ensure the directory is intact. If this does not resolve the problem, provide the return code, reason, and info code to the IBM Software Support.

ANL1044E NO MEMBERS FOUND TO MATCH DBRMKEY PATTERN *pattern*. PROGRAM TERMINATES.

Explanation

The pattern specified, with or without wild cards, for the DBRMKEY parameter did not match any members in the PDS directory. Valid pattern combinations are:

(8 blanks) for all members

*

for all members

memberx

for a specific member

ABC*

for all member names starting with ABC

%%%ABC%%

for all members with any characters in the % positions and the characters A, B, and C in positions 4, 5, and 6

When a pattern uses the % wild card, members are selected only if the member name is the exact length of the pattern.

User response

Specify a more precise pattern in the DBRMKEY parameter and run the job again.

ANL1045W	USER SPECIFIED USEPLAN BUT
	INSTALLATION HAS NOT
	AUTHORIZED OPTION. SQL PA
	WILL SEARCH REGISTRY AND/OR
	CURRENT SQLID FOR
	ALTERNATIVES.

Explanation

You attempted to use USEPLAN *authid*, but the installation has not authorized its use with a corresponding SETPLAN ALLOW (or YES prior to V5.1) in the ANLCNTL parameters. Normal search for generic or user explain tables continues.

User response

Enable this parameter with SETPLAN ALLOW (or YES prior to V5.1).

ANL1046I

USER HAS SPECIFIED USE OF authid.PLAN_TABLE FOR THIS RUN.

Explanation

Informs you of a successful implementation of a USEPLAN *authid* specification that was authorized by the installation with SETPLAN ALLOW (or YES prior to V5.1). The PLAN_TABLE and its owning *authid* are displayed for your information.

User response

None

ANL1047W	UNABLE TO USE
	authid.PLAN TABLE, SQLCODE =
	cccc. SQL PA WILL SEARCH
	REGISTRY AND/OR CURRENT
	SQLID FOR ALTERNATIVES.

Explanation

The PLAN_TABLE that was selected with SETPLAN *authid* was not found and cannot be allocated by the program. The likely SQLCODE is -518. SQL PA uses the standard alternatives starting with the Registry.

User response

Check for, or create, the PLAN_TABLE under the *authid* named in the USEPLAN. Run the job again.

ANL1048W POSSIBLE AUTHORIZATION PROBLEMS WITH PLAN_TABLE USED.

Explanation

When message ANL2012W occurs, and there were actually one or more good statements processed by SQL PA, this message is issued, indicating a likely problem with RACF[®] authorizations in reading and writing the chosen *authid*.PLAN_TABLE.

User response

Check the JES log in batch for ICH408I messages, and their DB2 access privileges on the selected table.

ANL1049W	RESOURCE LIMIT FACILITY HAS
	REJECTED QUERY qqqqqqqqq,
	SQLCODE = cccc. SET RLFASUERR
	AND RLFASUWARN TO HIGHER
	VALUES FOR THIS AUTHID.
	STATEMENT CANNOT BE
	EVALUATED BY SQL PA.
	•

The Resource Limit Facility (RLF) has set CPU constraints too low to process a Prepare or Explain of this SQL statement under this authorization ID.

User response

Modify the values for RLFASUERR and RLFASUWARN as indicated in the DSNRLST table to increase the amount of CPU time available before RLF terminates the Prepare or Explain query with +/- 495 SQLCODE.

ANL1050E A CHAR-NUMERIC CONVERSION ERROR HAS OCCURRED DURING I/O.

Explanation

An invalid format was encountered on a read or write to a file.

User response

Check DCB information for all files allocated. If DCB is current, then contact IBM Software Support.

ANL1051WUNABLE TO CREATE
anl.DSN_STATEMNT_TABLE FOR
PRECISE YES REQUEST. PGM
CONTINUES PROCESSING
WITHOUT THIS INFORMATION.
RETCODE = rrr.

Explanation

SQL PA attempted to create a DSN_STATEMNT_TABLE dynamically because none existed and you requested PRECISE YES. However, it was not able to create the table.

User response

None The program continues processing using normal costing options (Precise No).

ANL1052W UNABLE TO CREATE user.DSN_FUNCTION_TABLE FOR PRECISE YES REQUEST. PGM CONTINUES PROCESSING WITHOUT THIS INFORMATION. RETCODE = rrr.

Explanation

SQL PA attempted to create a DSN_FUNCTION_TABLE dynamically because none existed and you requested PRECISE YES or ALL. However, it was not able to create the table.

User response

None The program continues processing using normal costing options (Precise No).

ANL1053W	UNABLE TO CREATE
	user.DSN_DETCOST_TABLE FOR
	PRECISE ALL REQUEST. PGM
	CONTINUES PROCESSING
	WITHOUT THIS INFORMATION.
	RETCODE = <i>rrr</i> .

Explanation

SQL PA attempted to create a DSN_DETCOST_TABLE dynamically because none existed and you requested PRECISE ALL. However, it was not able to create the table.

User response

None The program continues processing using normal costing options (Precise No).

ANL1054I	PRECISE YES SELECTED BUT
	USER HAS NO
	DSN_FUNCTION_TABLE UNDER
	authid. FUNCTIONS WILL BE
	OMITTED.

Explanation

SQL PA cannot find or create a DSN_FUNCTION_TABLE and you requested PRECISE YES or ALL. Function analysis and reporting are omitted.

User response

None The program continues processing without this additional information.

```
ANL1055I PRECISE ALL SELECTED BUT USER
HAS NO DSN_DETCOST_TABLE
UNDER authid. NORMAL COSTING
WILL BE EMPLOYED.
```

Explanation

SQL PA cannot find or create a DSN_DETCOST_TABLE and you requested PRECISE ALL. Additional precision, analysis, and reporting are omitted.

User response

None The program continues processing without this additional information.

ANL1056W ERROR PROCESSING user.DSN_DETCOST_TABLE, SQLCODE = ccccc. PGM IGNORES

THIS TABLE AND CONTINUES PROCESSING QUERIES. QUERY NOT FOUND WAS: qqqqqqqqqqqqqq PLAN CLOUNT = ppppp.

Explanation

SQL PA did not find an expected entry in DSN_DETCOST_TABLE, yielding SQLCODE ccccc during PRECISE ALL processing.

User response

None The program continues processing using normal costing options (Precise No).

ANL1057W ERROR PROCESSING user.DSN_FUNCTION_TABLE, SQLCODE = ccccc. PGM IGNORES THIS TABLE AND CONTINUES PROCESSING QUERIES. QUERY NOT FOUND WAS: qqqqqqqqqq, PLAN CLOUNT = ppppp.

Explanation

SQL PA did not find an expected entry in DSN_FUNCTION_TABLE, yielding SQLCODE *ccccc* during PRECISE YES|ALL processing. In Version 8, a zparm (SPRMXPL) must be activated to turn on function table writes for built-in functions; however, UDFs and STPs are written to this table, regardless of DB2 release or zparm setting.

User response

None The program continues processing using normal costing options (Precise No).

ANL1058E SQL PA COULD NOT DROP user.DSN_STATEMNT_TABLE, RETURN CODE = ccccc.

Explanation

SQL PA created a DSN_STATEMNT_TABLE for this run but was unable to delete it after processing was complete.

User response

Examine the return code (SQLCODE) and determine the cause of this problem.

ANL1059E SQL PA COULD NOT DROP user.DSN_FUNCTION_TABLE, RETURN CODE = ccccc.

Explanation

SQL PA created a DSN_FUNCTION_TABLE for this run but was unable to delete it after processing was complete.

User response

Examine the return code (SQLCODE) and determine the cause of this problem.

ANL1060E SQL PA COULD NOT DROP user.DSN_DETCOST_TABLE, RETURN CODE = ccccc.

Explanation

SQL PA created a DSN_DETCOST_TABLE for this run but was unable to delete it after processing was complete.

User response

Examine the return code (SQLCODE) and determine the cause of this problem.

ANL1062I TABLEAU BUILD/DROP ON tcode RESULTS IN RC rrrr.

Explanation

Procedure TABLEAU has attempted to either create (build) a required explain table dynamically or drop an explain table that was created dynamically. The associated return code provides the status of that operation (0 is successful). The *tcode* values are:

- CST = DSN_DETCOST_TABLE
- FUN = DSN_FUNCTION_TABLE
- PLN = PLAN_TABLE
- PRD = DSN_PREDICAT_TABLE
- STM = DSN_STATEMNT_TABLE

User response

Examine the return code (SQLCODE), if it is non-zero, and determine the cause of the problem. The program attempts to run without this table unless it is the PLAN_TABLE.

ANL1063E	UNICODE DBRM CONVERSION
	ERROR, RC = <i>rrrrr</i> , REASON =
	cccccc. INPUT CANNOT BE
	PROCESSED. SQL PA TERMINATES
	THIS RUN.

A Version 8 DBRM module could not be converted from Unicode to EBCDIC. The RC and REASON are return code and reason codes from the MVS conversion routine.

User response

The program cannot process this DBRM. Contact IBM Software Support with return and reason codes for analysis.

ANL1064W STRING TRUNCATION HAS OCCURRED FOR MIXED DATA (SBCS, DBCS). SQL PA WILL CONTINUE PROCESSING DATA, COND = cccc.

Explanation

A mixed data variable or field name was truncated by the program because it exceeded the size of the storage area it was being translated into.

User response

The program continues processing. If the SQL is not properly explained, check the spelling of the mixed data name, or substitute a shorter value.

ANL1065W A NON-CRITICAL UNDERFLOW ERROR HAS OCCURRED DURING PROCEDURES. SQL PA WILL CONTINUE PROCESSING DATA.

Explanation

An internal math underflow error has occurred while manipulating internal variables in the program, which does not affect the outcome of path lengths or cost estimates.

User response

None The program continues processing.

ANL1066I PGM CANNOT ESTIMATE COST OF DECLARED GLOBAL TEMPORARY TABLE, SINCE NO DATA EXISTS IN THE CATALOG.

Explanation

The PREPARE failed because DB2 could not find catalog data describing the table, which was a declared global temporary table, carrying the name SESSION.*tablename*. The table is created during SQL operations, and only exists for the life of the application. Thus, DB2 cannot find the permanent table during PREPARE, and issues SQLCODE -204: table not found.

User response

None This message is informational. There are no statistics in the catalog for a declared global temporary table. DB2 cannot PREPARE the statement, and therefore SQL PA cannot evaluate it.

ANL1067W UNABLE TO CREATE user.DSN_PREDICAT_TABLE FOR PRECISE ALL REQUEST. PGM CONTINUES PROCESSING WITHOUT THIS INFORMATION. RETCODE = rrr.

Explanation

SQL PA attempted to create a DSN_PREDICAT_TABLE dynamically because none existed and you requested PRECISE ALL. However, it was not able to create the table.

User response

None The program continues processing using normal costing options (PRECISE NO).

user.DSN_PREDICAT_TABLE, SOLCODE = ccccc. PGM IGNORE	
SOLCODE = ccccc. PGM IGNORE	
	S
THIS TABLE AND CONTINUES	
PROCESSING QUERIES. QUERY	
NOT FOUND WAS: qqqqqqqqqqq	,
PLAN CLOUNT = $ppppp$.	

Explanation

SQL PA did not find an expected entry in DSN_PREDICAT_TABLE, yielding SQLCODE ccccc during PRECISE ALL processing.

User response

None The program continues processing using normal costing options (PRECISE NO).

ANL1069E SQL PA COULD NOT DROP user. DSN_PREDICAT_TABLE. RETURN CODE = rc.

Explanation

SQL PA attempted to drop the DSN_PREDICAT_TABLE which was created dynamically because you requested PRECISE ALL. However, it was not able to drop the table.

Notify your system support and provide them with this message and the full name of the table to delete.

ANL1070W

SIZE ERROR OCCURRED DURING EXPLAIN WHILE STORING A VARIABLE. SQL PA WILL CONTINUE PROCESSING DATA.

Explanation

A variable was stored in a field that was too small to contain its precision.

User response

None Program continues and computations are unaffected.

ANL1071W UNABLE TO CREATE authid.PLAN_TABLE DYNAMICALLY, AND NO TABLE EXISTS. PGM CANNOT CONTINUE WITHOUT PLAN_TABLE FOR OUTPUT. RETCODE = trc.

Explanation

SQL PA found no usable plan table under current *authid*, and no generic plan tables were available. It attempted to create one dynamically but was unable to do so. The value of *trc* is the SQLCODE returned from the CREATE attempt.

User response

Either you have not been granted use of the generic SQL PA table space and database, it does not exist, or you do not have permission to create tables on this system. Check SQLCODE for further assistance. Message ANL1008E follows with further information.

ANL1072E SQL PA COULD NOT DROP authid.PLAN_TABLE. RETCODE = trc.

Explanation

SQL PA dynamically created a plan table under the current *authid* for this operation, but was unable to drop the table at the conclusion of the run. The value of *trc* is the SQLCODE returned from the DROP table attempt.

User response

Either a system or program error has occurred or you do not have permission to drop tables on this system. Check SQLCODE for further assistance.

ANL1073W	"EXPLAIN ANLEEE" BUT NO
	"EEECALL EEEPATH" WAS FOUND.
	EEEPATH INPUT ASSUMED FOR
	THIS RUN. ANLIN MUST BE
	"ANLEEE.SQL".

Explanation

The EXPLAIN ANLEEE parameter was set, but you did not set the corresponding EEECALL EEEPATH parameter. SQL PA assumes that this is what you wanted and the program continues, expecting that data is found in the EEEPATH table and in the ANLEEE.SQL input file.

User response

Code the EEECALL EEEPATH parameter along with EXPLAIN ANLEEE.

ANL1074E "EXPLAIN [EEE|EXT]" WAS SPECIFIED BUT ANLIN IS NOT "ANLEEE.SQL" (EEE).

Explanation

ANLIN file must have been created by EEE to use the [EEE|EXT] option. EXPLAIN NEW is assumed, and SQL PA processing continues.

User response

None

ANL1075W	SQL PA HAS ALTERED THIS
	STATEMENT TO AVOID THE
	SQLCODE -41x. THE REPORTED
	ACCESS PATH ON AN EXPLAIN
	FOR A MODIFIED STATEMENT MAY
	BE INACCURATE. MODIFIED
	STMT:

Explanation

Without SQL PA altering the SQL statement as written, the statement would have produced a SQLCODE -417 or -418 when SQL PA attempted a dynamic explain.

User response

When you view the reports produced by SQL PA, be aware that any access path reported might be inaccurate on statements that have been altered.

ANL1076W

SQL PA ATTEMPTED TO ALTER THE STMT TO AVOID THE SQLCODE -418. THE ALTERATIONS MADE DID NOT ELIMINATE THE PROBLEM. SQLCODE: ',SQLCODE,' IS NOW BEING REPORTED. ALTERED STMT:

Explanation

SQL PA altered the SQL statement as written in an attempt to avoid a SQLCODE -418 during the dynamic explain. However, it was not successful in its attempt.

User response

Recode the SQL to use the allowed predicate structure. Or replace the "?" with your estimate of the proper values and run again.

ANL1078W SQL PA ATTEMPTED TO ALTER THE STMT TO AVOID THE SQLCODE -41x. THE ALTERATIONS MADE DID NOT ELIMINATE THE PROBLEM. SQLCODE: ',SQLCODE,' IS NOW BEING REPORTED. ALTERED STMT:

Explanation

SQL PA altered the SQL statement as written in an attempt to avoid a SQLCODE -417 or -418 during the dynamic explain. However, it was not successful in its attempt.

User response

Recode the SQL to use the allowed predicate structure. Or replace the "?" with your estimate of the proper values and run again.

ANL1079I The host variable and column name could not be matched for the following host variables: host_variable_1, host_variable_2, host_variable_3...host_variable_n

Explanation:

The host variables listed in the message could not be matched to a specific column name. This message displays for your information only.

User response

No action is required.

Recode the SQL to use the allowed predicate structure. Or replace the "?" with your estimate of the proper values and run again.

ANL1080E

A processing error was encountered during SQL statement analysis. More than the maximum number of records were encountered.

Explanation:

Processing has stopped during SQL statement analysis because the maximum number of internal table records has been reached. Some SQL statements use more internal table records. An example is an INSERT statement.

User response

Reduce the number of SQL statements to SQL PA and run again.

ANL1081E	authid.PLAN_TABLE is selected for
	this run

Explanation

SQL PA selected *authid* based upon the value of SETPLAN.

User response

None

ANL2000I See the full message text below.

Explanation

Informational message which shows how many records were read from PLAN_TABLE and input from SQL file (ANLIN and ANLIN*nn*, where *nn* is 00-99). Additionally, it shows the number of different tables, table spaces, indexes, and relations actually encountered during processing. The catalog is only accessed once for each entity marked STORED, so the total of all stored values is equivalent to the number of catalog reads for this set of SQL. Also, some nonrepeating accesses are categorized under NOT STORED, and can occur depending on options set (like SHOWALT).

User response

None

Full message text:

TOTAL NUMBER OF PLAN RECORDS PROCESSED WAS *nnn* TOTAL NUMBER OF PARSED RECORDS SCANNED WAS *nnn* TOTAL NUMBER OF ERRORS ENCOUNTERED IN RUN *nnn* CATALOG READS – STORED:

```
TABLES = n TABSPACES = m
INDEXES = l ROUTINES = u
IDXKEYS = k COLUMNS = c
CAT READS – NOT STORED:
RELATIONS = r
COLDIST = d ALT INDEX = a
```

ANL2001E FATAL SQL ERROR ENCOUNTERED DURING EXPLAIN/CATALOG ACCESS. DB2 SQLCODE = cccc. DB2 OPERATION: objname, @ EXPLAIN REC: nn, @ PARSER REC: mm.

Explanation

An SQL error occurred during processing. The current DB2 object being accessed is indicated, and where the processing error occurred in the plan table and parser input files.

User response

Check plan table and input files for errors. If the error continues, contact IBM Software Support.

ANL2002E PROCESSING ERROR ENCOUNTERED DURING EXPLAIN/CATALOG ACCESS. ERROR CODE IS: cccc. DB2 OPERATION: objname, @ EXPLAIN REC: nn, @ PARSER REC: mm.

Explanation

Note: Depending upon the error, this message can also indicate a Possible Cause for the condition and suggest hints for resolution:

Error Code

Possible Cause/Resolution

22

Record length is wrong size. Check DCB information.

42

Transmission error occurred reading an input file, due to missing file DD, no member name for PDS, and so on.

83

DCB information incorrect - check LRECL, RECFM, BLKSZ.

84

No DD statement for required data set – missing DD.

612 | 613

Internal conversion error from character to numeric, or Steplib to DSNLOAD or SCEERUN are required.

1040

Output file used all available space – define larger.

39**20**

Program out of storage – run with larger region size.

8094

Protection Exception – check authorization ID/ RACF permissions or Input data set or PDS member not found for *ddname*.

8097

Data Check on input – DBRM was expected by program.

9050

Input data set or PDS member was not found for *ddname* or internal PL/I abend.

An internal error occurred during processing. Save the error code, DB2 object, and position in the plan table and save the parser input file for DB2 Support.

User response

Check plan table and input files for errors. Explore possible causes if reported. If the error continues, contact IBM Software Support.

```
ANL2003E SYNCHRONIZATION PROBLEM
DETECTED WHILE MATCHING
RECORDS. EXPLAIN: qryno,
qblkno, pcount PARSER: qryno,
qblkno, parcnt.
```

Explanation

The plan table and parser input file are out of synchronization due to an error in parsing or a transformation of the query undetected by SQL PA. The current query number and block number of each are displayed, and record count and position in each file.

User response

If the DBRM statement numbers are out of sequence, then reprocess with the PROCESS NOSEQ parameter set. Check plan table and input files for errors. If the error continues, note EXPLAIN and PARSER record information and contact IBM Software Support.

ANL2004I ALIAS ENCOUNTERED FOR QUERY: mmmmmmmm QBLOCK: nnn. aliasname USED AS ALIAS FOR: tablename.

Informational. An ALIAS was uncovered. The correct table name was substituted for the catalog statistics.

User response

None

ANL2005I VIEW MATERIALIZATION FOR QUERY: mmmmmmmm QBLOCK: nnn. VIEW NAME USED IS: vname.

Explanation

Informational. This query materializes a view, as indicated by view name.

User response

None

ANL2006I NO TABLE STATS FOR QUERY mmmmmmmmm QBLOCK nnn. DEFAULTS USED.

Explanation

Informational. No statistics were found in the catalog for the table referenced by this query block.

User response

None

ANL2007I NO TSPACE STATS FOR QUERY mmmmmmmmm QBLOCK nnn. DEFAULTS USED.

Explanation

Informational. No statistics were found in the catalog for table space referenced by this query block.

User response

None

ANL2008I NO INDEX STATS FOR QUERY mmmmmmmmm QBLOCK nnn. DEFAULTS USED.

Explanation

Informational. No statistics were found in catalog for the index referenced by this query block.

User response

None

ANL2009I

OPTIMIZER HAS TRANSFORMED QUERY: mmmmmmmmm. WAS A SUBQUERY... NOW A JOIN, FOR YOUR INFORMATION.

Explanation

Informational. SQL PA has detected an optimizer transformation from a subquery to a join.

User response

None

ANL2010I

OPTIMIZER HAS TRANSFORMED QUERY: mmmmmmmmm. WAS A JOIN... NOW A SUBQUERY, FOR YOUR INFORMATION.

Explanation

Informational. SQL PA has detected an optimizer transformation from a join to a subquery.

User response

None

ANL2011I OPTIMIZER HAS TRANSFORMED QUERY: mmmmmmmmm. WAS A SUBQUERY... NOW A PRED, FOR YOUR INFORMATION.

Explanation

Informational. SQL PA has detected a transformation from a subquery to a simple predicate.

User response

None

ANL2012W NO VALID PLANS WERE GENERATED BY INPUT SQL. (CHECK CATALOG FOR OBJECTS EXISTENCE & QUERY SYNTAX).

Explanation

SQL PA was unable to Prepare or Explain any of the SQL input, indicating that the objects referred to in the query did not exist in the catalog, or the SQL had syntax errors or could not be prepared.

User response

Check for these conditions, correct, and run again.

ANL2013I

GLOBAL TEMPORARY TABLE ON QUERY: mmmmmmmmm QBLOCK: nnn. GLOBAL NAME USED IS: tempname.

Explanation

Informational. This query materializes a global temporary table, as indicated by table name *tempname*.

User response

None

ANL2014I AUXILIARY TABLE (LOBs) ON QUERY: mmmmmmmm QBLOCK: nnn. AUX TABLE NAME USED IS: tempname.

Explanation

Informational. This query uses an auxiliary table to hold LOBs, as indicated by table name.

User response

None

ANL2015I NO KEY STATS FOR QUERY: nnnnnn BLOCK: qqqq INDEX KEY NOT FOUND: xxx.yyyyyyy.

Explanation

Informational. Statistics are unavailable during processing.

User response

None

ANL2016I NO COLUMN STATS FOR QUERY: nnnnnn BLOCK: qqqq COLUMN NOT FOUND: cccccccccc.

Explanation

Informational. Statistics are unavailable during processing.

User response

None

ANL2017I NO DISTRIBUTION STATS FOR QUERY: nnnnnn BLOCK: qqqq COLUMN NOT FOUND: ccccccccccc.

Explanation

Informational. Statistics are unavailable during processing.

User response

None

ANL2018I NO ROOM FOR CHOSEN INDEX, QUERY: qqqqqqqq BLOCK: bbb MAX 1024 EXCEEDED.

Explanation

The internal array size for storing information about the indexes chosen for access paths has been exceeded. There is room for up to 1024 indexes.

User response

Reduce the amount of SQL being processed in a single run of SQL PA.

ANL2019I NO ROOM FOR ALTERNATE INDEX, QUERY: qqqqqqqq BLOCK: bbb MAX 1024 EXCEEDED.

Explanation

The internal array size for storing information about the other indexes not chosen for access paths has been exceeded. There is room for up to 1024 indexes.

User response

Reduce the amount of SQL being processed in a single run of SQL PA.

ANL2020I NO ROOM FOR ALT IDX KEYS, QUERY: qqqqqqqq BLOCK: bbb MAX 4096 EXCEEDED.

Explanation

The internal array size for storing information about the alternative index keys has been exceeded. There is room for up to 4096 index key entries.

User response

Reduce the amount of SQL being processed in a single run of SQL PA.

ANL2021W The VERSION is set to *vvvv*, but *yyyy* was detected and used. Update the VERSION parameter.

The VERSION parameter was set for one version of DB2 (*vvvv*), but SQL PA detected and used a different version (*yyyy*) for analysis.

User response

Correct the VERSION parameter to reflect the current operating release.

ANL2022I NO TABLE STATS FOR QUERY: qqqqqqqq BLOCK: bbb JOIN WORK TABLE: explan.tname.

Explanation

The DSNWFQB*nn* temporary join work file was used in this plan step.

User response

None This message avoids the redundant *no stats for table* message.

ANL2023I NO TABLE STATS FOR QUERY: qqqqqqqq BLOCK: bbb TABLE FUNCTION OR INTERMEDIATE WORK FILE USED (t).

Explanation

A table function or intermediate work file was used in this plan step. The table code *t* is: F = Table Function, Q (intermediate memory table), or W (work file).

User response

None This message avoids the redundant *no stats for table* message.

ANL2024I NO TABLE STATS FOR QUERY: qqqqqqqq BLOCK: bbb GLOBAL TEMPORARY TABLE OR VIEW MATERIALIZATION.

Explanation

A global temporary table (SESSION.*tname*) was referenced, or a view was materialized to access view data.

User response

None This message avoids the redundant *no stats for table* message.

ANL2025I NO INDEX STATS FOR QUERY: qqqqqqqq BLOCK: bbb SPARSE INDEX USED.

Explanation

A sparse index (temporary, partial keys) was detected in use for this query. Sparse indexes are present in some non-correlated IN (Subquery) predicates and in some star joins.

User response

None This message avoids the redundant *no stats for index* message.

ANL2026I NO TABLE STATS FOR QUERY: queryno BLOCK: qblockno BUFFERS IN MEMORY USED TO PROCESS THIS STATEMENT.

Explanation

The access path used buffered data in memory for this materialized table access, created from a related SQL operation. No I/O was necessary to read the table pages, which were already stored in the buffer pool.

User response

None This message is informational. There are no stats in the catalog for a virtual buffered table.

ANL2027I	NO TABLE STATS FOR QUERY:
	queryno BLOCK: qblockno
	COMMON TABLE EXPRESSION OR
	RECURSIVE COMMON TABLE
	EXPRESSION WAS USED.

Explanation

SQL of the form *WITH table-expression AS table name, SELECT...* was used to define a table and then access it in subsequent SQL. This type of SQL is referred to as a common table expression, and it allows you to define an instant table or view of the data, and then use it, within a single SQL statement.

User response

None This message is informational. There are no stats in the catalog for an instantly defined table.

ANL2029E	ERROR OCCURRED ACCESSING
	EEEPATH TABLE, SQLCODE =
	sqlcode. OLD PLAN AND
	COMPARISON TO NEW PLAN ARE
	NOT AVAILABLE. PROGRAM
	CONTINUES WITH NEXT QUERY.

A processing error occurred while accessing the EEEPATH table.

User response

Check the SQLCODE and correct the problem, or run ANLEEE maintenance to clean up the EEEPATH table and re-populate the data.

ANL2030I NO HISTOGRAM DISTRIBUTION STATS FOR QUERY: nnnnnn BLOCK: qqqq COLUMN NOT FOUND: cccccccccc.

Explanation

Informational statistics were unavailable during processing.

User response

None

ANL3000E UNABLE TO ACCESS STATISTICS ARRAY: ASTAT. SQL PA TERMINATES WITH UNRECOVERABLE DATA ERROR.

Explanation

An array used to hold catalog statistics has been corrupted during processing.

User response

Run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3001E UNABLE TO ACCESS PARSER ARRAY: APAS. SQL PA TERMINATES WITH UNRECOVERABLE DATA ERROR.

Explanation

An array used to hold parser intelligence was corrupted during processing.

User response

Run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3002E UNABLE TO ACCESS THE EXPLAIN ARRAY: AEXP. SQL PA

Explanation

An array used to hold explain plan data has been corrupted during processing.

User response

Run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3003W UNABLE TO OPEN REPORTS FILE ANLREP. SQL PA WILL NOT PROVIDE ENHANCED EXPLAIN REPORT.

Explanation

The output file identified by ANLREP DD is not allocated or is unusable. The enhanced explain report cannot be created.

User response

Correct the problem with ANLREP DD and run again.

ANL3004W	UNABLE TO OPEN REPORTS FILE
	QTRACE. SQL PA WILL NOT
	PROVIDE A DETAILED TRACE
	REPORT.

Explanation

The output file identified by QTRACE DD is not allocated or is unusable. The detailed trace report cannot be created.

User response

Correct the problem with QTRACE DD and run again.

ANL3005W VERSION *vvvv* IS NOT SUPPORTED BY THIS PROGRAM: *yyyy* ASSUMED.

Explanation

The version selected by VERSION parameter (*vvvv*) is not supported by this release of the product. The default version (*yyyy*) is substituted.

User response

None SQL PA supports three versions of DB2 simultaneously, the current and two back levels.

ANL3006W

DEVICE *ddddd* IS NOT SUPPORTED BY THIS PROGRAM: 3390-2 ASSUMED.

Explanation

An unknown device type was entered in the user parameter file. SQL PA uses 3390-3 device.

User response

Correct the STORAGE parameter in the user parameter file. Optionally, define your own device using the NEWSTOR parameter and related parameters.

ANL3007E UNABLE TO ACCESS PARSER DATA ANLPAS. SQL PA TERMINATES WITH AN UNRECOVERABLE ERROR.

Explanation

An array used to hold parser intelligence cannot be accessed.

User response

Run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3008E UNABLE TO ACCESS STATISTICS IN ANLSEP. SQL PA TERMINATES WITH AN UNRECOVERABLE ERROR.

Explanation

An array used to hold access plan data cannot be accessed.

User response

Run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3009E SYNCHRONIZATION PROBLEM BETWEEN DATA INPUTS: EXPLAIN QUERY IS queryno qblkno planno mixopseq PARSER QUERY IS queryno qblkno RELATION QUERY IS queryno qblkno SQL PA PROGRAM TERMINATES WITH UNRECOVERABLE ERROR.

Explanation

The cost estimation process finds inconsistencies between several inputs representing the PLAN_TABLE, parser intelligence, and relationships. SQL PA cannot continue. *Queryno* is the current query number and *qblkno* is the current query block for he respective input sources. *Planno* and *Mixopseq* are plan step and multiple index steps respectively from the access plan.

User response

Run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3010E UNABLE TO OPEN STATEMENTS FILE ANLOUT. SQL PA TERMINATES WITH UNRECOVERABLE FILE ERROR.

Explanation

The file identified in the ANLOUT DD statement cannot be opened.

User response

Correct errors in the DD statement and run again.

ANL3011E UNABLE TO READ STATEMENTS FILE ANLOUT. SQL PA TERMINATES WITH UNRECOVERABLE FILE ERROR.

Explanation

The file identified by ANLOUT DD is not allocated, is unusable, or is corrupted. This file holds the SQL statements for report presentation.

User response

Correct the problem with ANLOUT DD and run again. If the problem persists, contact IBM Software Support.

ANL3012W

UNABLE TO CREATE REPORTS FILE QLIMIT. SQL PA WILL NOT PROVIDE A QUERY LIMITS REPORT.

Explanation

The output file identified by QLIMIT DD is not allocated or is unusable. The query limits report cannot be created.

Correct the problem with QLIMIT DD and run again.

ANL3013W UNABLE TO ACCESS RELATIONS ARRAY: ARELS. SQL PA WILL NOT SHOW RI INFORMATION FOR THIS RUN.

Explanation

An array used to hold referential integrity data cannot be accessed.

User response

The program runs without providing RI relationship data. If the problem persists, run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3014W	UNABLE TO ACCESS RELATIONS
	IN ANLREL. SQL PA WILL NOT
	DISPLAY ANY FURTHER RI INFO.

Explanation

An array used to hold referential integrity data cannot be accessed during the program.

User response

The program runs without providing RI relationship data from this point forward. If the problem persists, run ANLDEBUG JCL with diagnostics turned on (add a DBTRACE DMP parameter at the top of the ANLCNTL file) and contact IBM Software Support.

ANL3015W AN UNKNOWN PARAMETER WAS READ FROM THE ANLPARM FILE. THE KEYWORD SCANNED WAS: *xxxxxx*. THIS PARM WAS IGNORED.

Explanation

A parameter read from ANLPARM is not recognized as valid. SQL PA ignores the parameter.

User response

Correct the parameter spelling and run again.

ANL3016W AN UNKNOWN PARAMETER WAS READ FROM THE ANLCNTL FILE. THE KEYWORD SCANNED WAS: xxxxxx. THIS PARM WAS IGNORED.

Explanation

A parameter read from ANLCNTL is not recognized as valid. SQL PA ignores the parameter.

User response

Correct the parameter spelling and run again.

ANL3017I NO STATISTIC FOR CARDF ON TABLE authid.tablename OPTIMIZER DEFAULT USED FOR QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually -1) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object.

User response

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3018I	NO STATISTIC FOR NPAGES F ON
	TABLE authid.tablename
	OPTIMIZER DEFAULT USED FOR
	QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually -1) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object. Depending on the level of DB2, the statistic is NPAGES or NPAGESF.

User response

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3019I NO STATISTIC FOR PCTPAGES ON TABLE authid.tablename OPTIMIZER DEFAULT USED FOR QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually -1) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object.

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3020I NO STATISTIC FOR NACTIVE|F ON TABLE authid.tablename OPTIMIZER DEFAULT USED FOR QUERYNO nnnnnnn.

Explanation

The DB2 catalog contained a default (usually 0) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object. Depending on the level of DB2, the statistic is NACTIVE or NACTIVEF.

User response

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3021I NO STATISTIC FOR NLEAF ON INDEX authid.indexname OPTIMIZER DEFAULT USED FOR QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually -1) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object.

User response

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3022I	NO STATISTIC FOR NLEVELS ON
	INDEX authid.indexname
	OPTIMIZER DEFAULT USED FOR
	QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually -1) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object.

User response

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3023I	NO STATISTIC FOR
	FIRSTKEYCARDF ON INDEX

authid.indexname OPTIMIZER DEFAULT USED FOR QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually -1) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object.

User response

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3024I NO STATISTIC FOR FULLKEYCARDF ON INDEX authid.indexname OPTIMIZER DEFAULT USED FOR QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually -1) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object.

User response

None Optionally, run RUNSTATS or populate the catalog with the desired statistic, and run again.

ANL3025I NO STATISTIC FOR CLUSTERRATIO|F ON INDEX authid.indexname OPTIMIZER DEFAULT USED FOR QUERYNO nnnnnnnn.

Explanation

The DB2 catalog contained a default (usually 0) for this statistic for the table *authid.tablename* referred to by query *nnnnnnn*. SQL PA uses the same values that the DB2 optimizer uses internally for this object. Depending on the level of DB2, the statistic is CLUSTERRATIO or CLUSTERRATIOF.

User response

None Optionally, run RUNSTATS or populate the catalog with the statistics you want, and run again.

ANL3026W NO STATISTICS FOUND IN CATALOG FOR ONE OR MORE VARIABLES. OPTIMIZER DEFAULTS WERE USED WHERE MISSING VALUES FOUND.

Follow-up note in report files (ANLREP, QTRACE) indicating that some statistics are missing.

User response

None

ANL3027I

QUERYNO mmmmm WAS TRANSFORMED BY THE DB2 OPTIMIZER.

Explanation

Informational note indicating a transformation was performed by the optimizer—the Plan does not match the original SQL. The costing is done for the new plan.

User response

None

ANL3028E THERE WERE NO STATISTICS COLLECTED FROM CATALOG FOR INPUT SQL. PGM CANNOT DETERMINE COST - LACKING CRITICAL DATA. MOST LIKELY, ANLSQLPA WAS UNABLE TO PROCESS INPUT.

Explanation

This message indicates that the objects haven't been created yet. No statistics exist in the catalog for the objects referred to in the SQL. Defaults would have been be used if the objects exist, but RUNSTATS was not run.

User response

Create the objects, run RUNSTATS or populate the catalog with the statistics you want, and run the job again.

ANL3029W THE EXPLAIN FOR QUERY mmmmmmmm IS NOT VALID. PGM CANNOT PROVIDE A COST ESTIMATE. STATEMENT IGNORED.

Explanation

No valid records where written to the plan table for this query. SQL PA cannot process the statement, and moves on to the next one.

User response

Correct any errors in the query and run again.

ANL3030E

CATALOG STATISTICS FOR THIS SQL HAVE CAUSED DIVISION BY ZERO. CHECK STATISTICS AND CORRECT, THEN RE-RUN PGM.

Explanation

One of the catalog statistics used by SQL PA for this statement has caused a division by zero in the path length calculations.

User response

Check the catalog statistics for the objects used, and run again.

ANL3031E OVERFLOW ERROR DETECTED DURING CALCULATIONS. CATALOG STATISTICS ARE A LIKELY CAUSE. CHECK AND RE-RUN PGM.

Explanation

One of the catalog statistics used by SQL PA for this statement has caused a computational exception in the path length calculations.

User response

Check the catalog statistics for the objects used, and run again.

ANL3032E UNDERFLOW ERROR DETECTED DURING CALCULATIONS. CATALOG STATISTICS ARE A LIKELY CAUSE. CHECK AND RE-RUN PGM.

Explanation

One of the catalog statistics used by SQL PA for this statement has caused a computational exception in the path length calculations.

User response

Check the catalog statistics for the objects used, and run again.

ANL3033E	FIXEDOVERFLOW ERROR
	DETECTED DURING
	CALCULATIONS. CATALOG
	STATISTICS ARE A LIKELY CAUSE.
	CHECK AND RE-RUN PGM.

One of the catalog statistics used by SQL PA for this statement has caused a computational exception in the path length calculations.

User response

Check the catalog statistics for the objects used, and run again.

ANL3034E SIZE ERROR DETECTED DURING CALCULATIONS. CATALOG STATISTICS ARE A LIKELY CAUSE. CHECK AND RE-RUN PGM.

Explanation

One of the catalog statistics used by SQL PA for this statement has caused a computational exception in the path length calculations.

User response

Check the catalog statistics for the objects used, and run again.

ANL3035I THE ASSUMED OVERHEAD FOR CONNECTION TYPE OF connectiontype HAS BEEN ESTIMATED IN MILLIONS OF INSTRUCTIONS FOR THESE ONLINE SQL STMTS: THREAD MGMT = ttttttttt ATTACH = aaaaaaaa APPL = pppppppp.

Explanation

Informational. SQL PA reports on the processor usage estimates used for the attach facility and related components.

User response

None

ANL3036I THE ASSUMED OVERHEAD FOR CONNECTION TYPE OF connectiontype HAS BEEN ESTIMATED IN MILLIONS OF INSTRUCTIONS FOR THESE BATCH SQL STMTS: THREAD MGMT = tttttttttt ATTACH = aaaaaaaa APPL = pppppppp.

Explanation

Informational. SQL PA reports on the processor usage estimates used for the attach facility and related components.

User response

None

ANL3037I ESTIMATED storage DASD SERVICE TIMES ARE: SYNC READ **4K PAGE = syncrd PREFETCH** BLOCK OF nn 4K PAGES = prefrd SYNC READ 8K PAGE = sync8 **PREFETCH BLOCK OF** nn 8K PAGES = pref8 SYNC READ 16K **PAGE = sync16 PREFETCH BLOCK** OF nn 16K PAGES = pref16 SYNC READ 32K PAGE = sync32 PREFETCH BLOCK OF nn 32K PAGES = pref32 ASYNC WRT 4K **PAGE = async4 ASYNC WRITE OF 16K PAGE BLOCK = asyn16 ASYNC** WRT 8K PAGE = async8 ASYNC WRITE OF 32K PAGE BLOCK = asyn32

Explanation

Informational. SQL PA computes the average DASD service times for all buffer pool page sizes based on the disk device types and specifications.

User response

None

ANL3038I USER HAS DEFINED VALUES FOR THE *newdasd* STORAGE SUBSYSTEM.

Explanation

Informational. Informs you that your own DASD device type has been defined and used for evaluation.

User response

None

ANL3039I SHOWALT ACTIVE, BUT NO ALTERNATE INDEXES AVAILABLE FOR THIS ACCESS.

Explanation

Informational. User-specified SHOWALT parameter, but this table does not have any alternative indexes to display.

User response

None

ANL3040W

UNABLE TO OPEN PARAMETER FILE parmfile

Explanation

The parameter file identified by parmfile DD is not allocated or unusable. Default values are used for the corresponding parameters. The following values are the possible values for parmfile:

ANLCNTL

SQL PA system parameters

ANLPARM

SQL PA user parameters

DEFAULT

Plan table report parameters

User response

Verify and correct the parameter files specified in the ANLPPARM panel of the **Change the DB2 SQL PA parameters** option when modifying your configuration.

ANL3041I NO EXPLAINABLE STATEMENTS FOUND IN *FILENAME*.

Explanation:

The specified location did not contain any explainable statements.

ANL3042I PROCESSING SQL STATEMENTS FROM FILENAME.

Explanation:

SQL PA is processing the statements from the specified location.

ANL3099E A TERMINAL PROCESSING ERROR HAS OCCURRED. PLI ERROR CODE =pppppp SQL PA ERROR CODE =cccccc RETURN CODE =rr.

Explanation

Note: Depending upon the error, this message can also indicate a Possible Cause for the condition and suggest hints for resolution:

Error Code

Possible Cause/Resolution

22

Record length is wrong size. Check DCB information.

42

Transmission error occurred reading an input file, due to missing file DD, no member name for PDS, and so on.

83

DCB information incorrect - check LRECL, RECFM, BLKSZ.

84

No DD statement for required data set – missing DD.

34**0**

Variable size error - check catalog stats, RUNSTATS.

612 | 613

Internal conversion error from character to numeric or Steplib to DSNLOAD or SCEERUN can be required.

1040

Output file used all available space – define larger.

39**20**

Program out of storage – run with larger region size.

8094

Protection Exception – check authorization ID/ RACF permissions.

8097

Data Check on input – DBRM was expected by program.

9050

Input data set or PDS member was not found for ANLIN or internal PL/I abend.

A catastrophic error has occurred during ANLSQLPA processing that was not explained by any other messages.

User response

Note the error and return codes and explore possible causes, if reported. Save any system dump. Contact IBM Software Support.

ANL4001I SQLPA HAS CONNECTED TO SYSTEM AT location LEVEL version.

Explanation

Informational. You have requested to process this SQL at the remote *location* by including the VIADRDA and the PKGDRDA parameters in ANLPARMs, and SQL PA has successfully connected through the DRDA Connect statement to that system. The current version of DB2 running at that location is also returned as *version* (for example, DSN08011).

User response

None

ANL4002I

CONNECTION TO SYSTEM AT *location* HAS BEEN RELEASED.

Explanation

Informational. Your connection to the *location* has been released by a RELEASE statement, prior to terminating the local DB2 thread and session.

User response

None

ANL4003I

THE CAF OPEN RETURN CODE IS ccc FOR plan-name ON ssid.

Explanation

Informational. SQL PA uses the call attach facility to connect to DB2, and this shows the return code from the OPEN call (*cccc*), the plan name (*plan-name*) and the DB2 subsystem ID (*ssid*) that you are trying to connect to.

User response

If there are problems with the CAF Open, then check the SUBSYST parameter for correct DB2 subsystem name, and also see if it is started and available for access.

ANL4004I THE CAF CLOSE RETCODE IS cccc FOR plan-name ON ssid.

Explanation

Informational. SQL PA uses the call attach facility to connect to DB2, and this shows the return code from the CLOSE call (*cccc*), the plan name (*plan-name*) and the DB2 subsystem ID (*ssid*) that you are trying to disconnect from.

User response

If there are problems with the CAF CLOSE, check the SUBSYST parameter for the correct DB2 subsystem name, and see if it is started and available for access.

ANL4005W CONNECTION TO SYSTEM AT loc COULD NOT BE ESTABLISHED SQLPA WILL ATTEMPT TO PROCESS SQL ON THIS SUBSYSTEM (ssid). SQLCODE RETURNED TO CALLER WAS sqlcode.

Explanation

Warning. The program attempted to CONNECT through DRDA to the system at *loc* but was unable to do so. Possible causes might be improper binding of the Remote Package, improper binding of the local plan (not including the correct PKLIST) or the DDF address space of the Remote facility might be down (this would generate a -30080 SQLCODE on the call). SQL PA attempts to run the SQL on the local processor, under the subsystem ID *ssid*. The SQLCODE returned by the error is also displayed.

User response

As appropriate.

ANL4006E	CONNECTION TO SYSTEM location
	COULD NOT BE RELEASED.
	SQLCODE = sglcode.

Explanation

Error. After connecting through DRDA and processing SQL on the Remote server, the local program was unable to RELEASE the DRDA connection. This generally does not affect the outcome of processing (all work is done at this point and files stay intact) but investigate the cause of the error. Refer to SQLCODE for reasons.

User response

As appropriate.

S013 MVS ABEND - SYSTEM 013.

Explanation

When running SQL PA in batch mode, it is possible to encounter this error if the input data set exists but the member name does not. For example, a valid DBRM library was used for ANLIN or ANLIN*nn* (where *nn* is 00-99), but the member name specified did not exist. This condition also results in JOB NOT RUN - JCL ERROR message. Under MVS, the member name is not checked until run time, therefore, the S013 abend.

User response

Check the spelling on the appropriate DD statement and run again.

U4091 LE ABEND.

Unexpected condition occurred during LE processing with SQL PA. Possibly, a new release of LE was installed with other than default values for ERRCOUNT.

User response

Add override parameters to ANLBATCH JCL:

//ANLSTEP1 EXEC PGM=ANLSQLPA,PARM='ER(0)/'
//ANLSTEP2 EXEC PGM=ANLSQLPA,PARM='ER(0)/'

Warnings and alerts

In general, the following messages alert you to special processing by SQL PA or warn you about characteristics of your query that might affect its performance. **Warnings** and **Alerts** are always available and are not affected by the ADVISOR parameter in the user parameters file (ANLPARM).

Where the message is self-explanatory, no additional explanation is given.

ANL5005W	*** ALERT: THE PERCENTAGE OF ROWS ESTIMATED TO BE READ WITH THE ACCESS PATH OF		PERFORMANCE OF THIS IN PREDICATE.
	MULTIPLE INDEXES + LIST	User respons	e
	PREFETCH IS APPROACHING 25% OF TOTAL TABLE ROWS. SOON,	As appropriate.	
	DB2 WILL CHOOSE TABLESPACE SCAN INSTEAD OF MATCHING THE MULTI-INDEXES. RUNSTATS IS RECOMMENDED, INCLUDING THE COLLECTION OF NON- UNIFORM STATISTICS. THIS SHOULD CULMINATE IN BETTER STATS, AND ALLOW DB2 TO CONTINUE TO SELECT THIS PATH.	ANL5008W	*** WARNING: THIS SQL STATEMENT CONTAINS NO PREDICATES AND USES NO BUILT- IN FUNCTIONS, SO DB2 IS SELECTING THE TABLESPACE SCAN ACCESS METHOD. IN GENERAL, THIS MEANS POOR PERFORMANCE, UNLESS THE TABLE IS VERY SMALL: VERIFY THE ACCURACY OF CATALOG
User respons	se		STATISTICS (NPAGES, NACTIVE,
As appropriate.			WHETHER THIS ACCESS PATH IS
ANL5006W	*** ALERT: THIS QUERY USES AN		APPROPRIATE.
	IN LIST PREDICATE APPLIED AGAINST A RANDOM INDEX, WHICH CANNOT USE LIST	User response	
	PREFETCH TO ACCESS THE DATA	As appropriate.	
	PAGES. THUS, PERFORMANCE WILL SUFFER AS ONLY SYNCHRONOUS READ I/OS WILL BE USED TO READ THE DATA PAGES. IN LIST WORKS MUCH BETTER WITH A CLUSTERED INDEX. LIST PREFETCH ALLOWED IN V5R2+.	ANL5009W	*** WARNING: THIS NONSEGMENTED TABLESPACE ACTUALLY CONTAINS MULTIPLE TABLES, MEANING THAT YOU ARE SCANNING UNNECESSARY PAGES THAT DO NOT HOLD ROWS FOR YOUR TABLE, INCLUDING PAGES IN THE TABLESPACE THAT BELONG TO OTHER TABLES NOT
			BEING ACCESSED HERE.
		User response	
ANLOUU/W	LIST PREDICATE CONTAINS A	As appropriate.	
	SIGNIFICANT NUMBER OF ELEMENTS: EACH ENTRY ON THE LIST IS A SEPARATE INDEX SCAN. THEREFORE, THE LONGER THE LIST, THE WORSE THE	ANL5010W	*** ALERT: THIS QUERY IS ACCESSING THE TABLE VIA A TABLESPACE SCAN AGAINST A TABLESPACE THAT IS

NONSEGMENTED. EVEN THOUGH THIS IS JUST A SIMPLE TABLESPACE (ONLY ONE TABLE) YOU ARE NEEDLESSLY READING EXTRA PAGES BECAUSE DB2 MUST SCAN EVERY ACTIVE PAGE IN THE SPACE TO FIND ROWS.

User response

As appropriate.

ANL5012W *** ALERT: THIS PARTITIONED TABLESPACE IS NOT BEING ACCESSED USING ANY FORM OF PARALLELISM. POSSIBLE REASONS INCLUDE: INSUFFICIENT PARALLEL BUFFERS DEFINED IN POOL, OR DEGREE ANY NOT SPECIFIED ON THE BIND. ELAPSED TIME WOULD BE CONSIDERABLY ENHANCED BY CHOOSING A PARALLEL PROCESSING OPTION ON THIS PARTITIONED TABLESPACE.

User response

As appropriate.

ANL5013W *** ALERT: THE CURSOR WAS CONSIDERED AMBIGUOUS AT BIND TIME, MEANING THAT DB2 HAS NOT DECIDED HOW MANY DEGREES OF PARALLELISM WILL BE AVAILABLE AT RUN TIME. THE MAXIMUM NUMBER OF PARALLEL DEGREES IS EQUAL TO THE NUMBER OF PARTITIONS. PROGRAM MAY CHOOSE A CONSERVATIVE VALUE.

User response

As appropriate.

ANL5016W *** WARNING: THIS TABLE IS ACCESSED BY ROWID DIRECTLY. HOWEVER, IF THE PRIMARY ACCESS PATH SHOULD FAIL, THE FALLBACK ACCESS PLAN IS A TABLESPACE SCAN. YOU SHOULD HAVE AN INDEXED BACKUP PLAN IN CASE THE ROWID IS CHANGED BY REORG, ETC. TO SAFEGUARD PERFORMANCE.

User response

As appropriate.

ANL5017W ***** ALERT: AT PRESENT, THE** PERCENTAGE OF ROWS **ESTIMATED TO BE READ WITH** THIS CLUSTERED INDEX VIA SEQUENTIAL PREFETCH IS **APPROACHING 50% OF TABLE ROWS. SOON, DB2 WILL CHOOSE** TABLESPACE SCAN INSTEAD OF MATCHING THE CLUSTERED INDEX. RUNSTATS IS **RECOMMENDED, INCLUDING COLLECTION OF NON-UNIFORM** STATISTICS. THIS SHOULD CULMINATE IN BETTER STATS. AND ALLOW DB2 TO CONTINUE TO SELECT THIS PATH.

User response

As appropriate.

ANL5018W ******* ALERT: THIS CLUSTERING (INSERT) INDEX HAS A LOW **CLUSTER RATIO (BELOW 80)** AND/OR IS NOT WELL CLUSTERED (CLUSTERED FLAG = N), MEANING THAT ALL INSERTS TRY TO FOLLOW AN UNCLUSTERED PATTERN, WHICH IS NOT **HELPFUL. YOU SHOULD REORGANIZE YOUR TABLE AND** INDEX. AND RE-RUN RUNSTATS TO UPDATE THESE STATISTICS. THIS SHOULD RESULT IN BETTER ACCESS PATHS AND PERFORMANCE IN GENERAL.

User response

As appropriate.

ANL5019W *** ALERT: PRESENTLY, THE PERCENTAGE OF ROWS ESTIMATED TO BE READ WITH THIS RANDOM INDEX VIA LIST PREFETCH IS APPROACHING 25% OF TOTAL TABLE ROWS. SOON, DB2 WILL CHOOSE TABLESPACE SCAN INSTEAD OF MATCHING THE RANDOM INDEX. RUNSTATS IS RECOMMENDED, INCLUDING THE COLLECTION OF NON-UNIFORM STATISTICS. THIS SHOULD CULMINATE IN BETTER

STATS, AND ALLOW DB2 TO CONTINUE TO SELECT THIS PATH.

User response

As appropriate.

ANL5020W *** ALERT: A LOCKSIZE OF ROW HAS BEEN EMPLOYED WITH A SELECT STATEMENT. BE AWARE THAT THIS CAUSES MORE LOCKING THAN WITH LOCKSIZE PAGE, AND IF THERE IS NOT A CONSIDERABLE UPDATE REQUIREMENT, THIS LOCKSIZE SHOULD NOT BE EMPLOYED. EACH ROW LOCK COSTS AS MUCH AS A PAGE LOCK, AND COUNTS TOWARDS THE MAXIMUM NUMBER OF LOCKS HELD, ETC.

User response

As appropriate.

ANL5021W *** WARNING: THIS IS A TYPE 1 INDEX STRUCTURE, WHICH HAS MORE THAN 1 SUBPAGE DEFINED. INSERTS TO THIS INDEX WILL CAUSE LOGICAL PAGE SPLITS, AS WELL AS PHYSICAL SPLITS. CONVERSION TO TYPE 2 INDEX IS SERIOUSLY RECOMMENDED, OR AT LEAST REFORMAT USING SUBPAGES OF 1 (4096 BYTES).

User response

As appropriate.

ANL5022W	*** ALERT: A LOCKSIZE OF
	TABLESPACE IS NOT
	RECOMMENDED WHEN YOU
	EMPLOY DB2-ENFORCED
	REFERENTIAL INTEGRITY.
	CONSIDER CHANGING LOCKSIZE

User response

As appropriate.

ANL5023W	*** ALERT: THIS MERGE SCAN
	JOIN IS NOT ELIGIBLE FOR
	EITHER CPU OR SYSPLEX
	PARALLELISM, SIMPLY BECAUSE
	IT JOINS ON MORE THAN ONE
	COLUMN.

User response

As appropriate.

*** ALERT: THIS HYBRID JOIN IS
NOT ELIGIBLE FOR MULTIPLE CPU
OR SYSPLEX PARALLELISM
SIMPLY BECAUSE IT SORTS THE
INNER TABLE RID LIST.

User response

As appropriate.

ANL5025W	*** WARNING: ESTIMATE OF:
	cputim EXCEEDS CPU TIME LIMIT
	OF MAXCPU SECONDS!

User response

As appropriate.

ANL5026W	*** WARNING: ESTIMATE OF:
	qunits EXCEEDS SERVICE UNIT
	LIMIT OF maxqun QUNITS!

User response

As appropriate.

ANL5027W	*** WARNING: ESTIMATE OF: cost
	EXCEEDS MONETARY LIMIT OF
	maxcst money!

User response

As appropriate.

```
ANL5028W *** WARNING: ESTIMATE OF:
physio EXCEEDS I/O CALL LIMIT
OF iolim I/O CALLS!
```

User response

As appropriate.

ANL5029W *** WARNING: ESTIMATE OF: elapse EXCEEDS ELAPSED LIMIT OF maxelap SECONDS!

User response

ANLSUSUW	*** WARNING: THE USE OF
	TRIGGERS ON THIS STATEMENT
	WILL CAUSE ADDITIONAL
	PROCESSING, BEYOND THE COST
	ESTIMATES CALCULATED: BE
	AWARE.

As appropriate.

ANL5031W	*** WARNING: USER DEFINED
	FUNCTIONS (UDFS) ARE INVOKED
	DURING THIS STATEMENT
	EXECUTION, WHICH WILL ADD
	ADDITIONAL OVERHEAD TO THE
	FINAL COST.

User response

As appropriate.

ANL5032W *** ALERT: LARGE OBJECTS ARE BEING REFERENCED IN THIS SQL STATEMENT FROM AN AUXILIARY TABLE. MATERIALIZING LOBS CAN BE A COSTLY PROCESS.

User response

As appropriate.

ANL5033W *** ALERT: THIS IS A CORRELATED SUBQUERY WHICH IS INVOKING A TABLESPACE SCAN TO ACCESS DATA. CORRELATED SUBQUERIES ARE ALWAYS A STAGE 2 PREDICATE PROCESS. THEREFORE, CONSIDER REWRITING THIS SUBQUERY AS EITHER NON-CORRELATED OR AS A JOIN TO IMPROVE PERFORMANCE.

User response

As appropriate.

ANL5034W	*** ALERT: A MASS DELETE IS
	BEING PERFORMED AGAINST A
	NONSEGMENTED TABLESPACE.
	EACH ROW MUST BE
	INDIVIDUALLY DELETED AND
	LOGGED. UNLESS THE
	TABLESPACE PARTICIPATES IN
	RI, IT SHOULD BE CONVERTED TO
	A SEGMENTED TS TO ALLOW
	MASS DELETE TO CHANGE AND
	LOG ONLY THE SPACE MAP PAGES.

User response

As appropriate.

ANL5035W	*** WARNING: A DATA
	PARTITIONED SECONDARY INDEX
	(DPSI) IS BEING USED AS THE

CLUSTERING INDEX FOR THIS TABLE. THIS COULD CAUSE ACCESS PATHS TO JUMP TO MULTIPLE PARTITIONS WHEN ACCESSING ROWS, LEADING TO AN EXPOSURE FOR SEQUENTIAL ACCESS. KEEP AN EYE ON NEAROFFPOSF AND FAROFFPOSF STATISTICS IN SYSINDEXPART.

User response

As appropriate.

ANL5036W	*** ALERT: THIS INDEX IS THE
	PARTITIONING INDEX FOR THE
	TABLE, BUT IT IS NOT
	DESIGNATED AS THE
	CLUSTERING INDEX. THIS MEANS
	THAT ANOTHER INDEX HAS BEEN
	CHOSEN FOR INSERT/LOAD
	ORDERING, WHICH COULD
	POSSIBLY WORSEN
	PERFORMANCE WHEN
	SEQUENTIALLY INSERTING DATA,
	BY JUMPING ACROSS MULTIPLE
	PARTITIONS.

User response

As appropriate.

ANL5037W	*** ALERT: REFRESH MQT TABLE
	PERFORMED ON A
	NONSEGMENTED TABLE SPACE.
	REFRESH INCLUDES A DELETE OF
	ALL ROWS – IF TABLE SPACE IS
	SEGMENTED THIS IS A MASS
	DELETE, JUST UPDATING THE
	SPACE MAP PAGES. HOWEVER,
	YOUR TABLE SPACE WILL CAUSE A
	LOG WRITE FOR EACH ROW
	DELETED – VERY COSTLY!
	CONVERT THE TABLE SPACE TO
	SEGMENTED FOR BETTER
	PERFORMANCE

User response

ANL5038W	*** WARNING: THIS BASE TABLE
	HAS MQTS ASSOCIATED WITH IT,
	BUT THEY ARE NOT BEING
	CONSIDERED BY THE OPTIMIZER
	FOR AUTOMATIC QUERY REWRITE
	BECAUSE THE CURRENT REFRESH
	AGE IS SET TO 0. SET THE USER
	PARAMETER REFRESH ANY

As appropriate.

ANL5039W ***** WARNING: MULTIPLE** DISTINCT SORTS HAVE BEEN **DETECTED FOR THIS STATEMENT. ALTHOUGH V8+ ALLOWS LIBERAL USE OF THE DISTINCT KEYWORD,** THIS SQL WILL ACTUALLY **EXECUTE MULTIPLE SORTS,** WHICH IS NOT DESIRABLE FOR **GOOD PERFORMANCE.**

User response

As appropriate.

ANL5040W ***** ALERT: THIS TABLE HAS BEEN** DESIGNATED AS VOLATILE, SUGGESTING THAT DB2 SHOULD ALWAYS USE AN INDEX ON IT. HOWEVER, TABLE SPACE SCAN IS **BEING SELECTED HERE. CHECK** YOUR SQL CAREFULLY AND DETERMINE IF USING AN INDEXED COLUMN WILL IMPROVE YOUR ACCESS.

User response

As appropriate.

ANL5041W	*** ALERT: THIS MATERIALIZED QUERY TABLE (MQT) IS NOT	User response: As appropriate.	
lloor roonon	BEING ACCESSED VIA AN INDEX SCAN, BECAUSE NO INDEXES EXIST ON THE MQT. BUILDING ONE OR MORE INDEXES ON THE MQT WILL RESULT IN BETTER PERFORMANCE.	ANL5047W User response: As appropriate.	*** Alert: Global optimization of this subquery has transformed the SQL statement and materialized a virtual table. Run time might be extended.
User respons	be	ANL5048W	*** Alert: XMLEXISTS is a stage 2
As appropriate.			predicate unless an XML values
ANL5042W	*** ALERT: SITE REQUIRES NOTIFICATION WHEN ANY TABLESPACE SCAN IS EMPLOYED.		employed. Performance decreases when XML is used. The XMLEXISTS predicate is not a candidate for index screening.
User response	Se la	llear rasponsor	8.
As appropriate.		As appropriate.	
ANL5043W	*** ALERT: SITE REQUIRES NOTIFICATION WHEN ANY	ANL5049W	*** Alert: PRECISE NO will be deprecated and changed to

NONMATCH IX SCAN IS EMPLOYED.

User response

As appropriate.

ANL5044W	*** ALERT: SITE REQUIRES
	NOTIFICATION WHEN ALL TS
	PARTITIONS ARE SCANNED.

User response

As appropriate.

ANL5045W	*** ALERT: THIS TABLE SPACE
	HAS THE NOT LOGGED ATTRIBUTE.
	UNDO AND REDO LOGGING FOR
	THE TABLE SPACE AND ALL
	INDEXES FOR TABLES IN THE
	TABLE SPACE IS SUPPRESSED.
	LOGGING IS ALSO SUPPRESSED
	FOR THE AUXILIARY INDEXES
	FOR ALL AUXILIARY TABLES
	ASSOCIATED WITH TABLES IN
	THE TABLE SPACE.

User response

As appropriate.	
ANL5046W	*** Alert: Using the DECFLOAT CAST function results in a stage 2 predicate. Predicate transitive closure and indexing are not allowed.
User response: As appropriate.	
ANL5047W	*** Alert: Global optimization of this subquery has transformed the SQL statement and materialized a virtual table. Run time might be extended.
User response: As appropriate.	
ANL5048W	*** Alert: XMLEXISTS is a stage 2 predicate unless an XML values index or a NODEID index is employed. Performance decreases when XML is used. The XMLEXISTS predicate is not a candidate for index screening.
User response:	

PRECISE YES in DB2 V9 and later releases. When you use the PRECISE parameter, specify PRECISE YES or PRECISE ALL. predicates, or aggregate DISTINCT functions.

	*** Alert: DX access is not a candidate for any type of parallelism.	As appropriate.	
User response: As appropriate.		ANL5055W *** WARNING: THE HOST VARIABLE DEFINITION DOES MATCH THE COLUMN DEFINIT columnname IS columntype(length), hostvaria IS bytype(length)	*** WARNING: THE HOST VARIABLE DEFINITION DOES NOT
ANL5050W			MATCH THE COLUMN DEFINITION columnname IS columntype(length), hostvariable IS hytype(length)
User response:			15 hviype(length)
As appropriate.		Explanation:	
ANL5051W	*** Alert: XML is not allowed with SELECT DISTINCT, GROUP BY, or ORDER BY clauses, subselects	The <i>length</i> is onl have more than VARCHAR.	ly displayed for those types that can one length, such as CHAR or
	that are not UNION ALL, quantified BETWEEN, DISTINCT, IN, or LIKE	User response: As appropriate.	

User response:

Notes and recommendations

The following messages make general comments and recommendations on improving the performance of your queries. **Notes** and **Recommendations** are shown in reports if the ADVISOR parameter is set to YES or ALL in the user parameters file (ANLPARM).

These messages are self-explanatory; no additional explanation is given.

ANL6001I **QUERIES. SEQUENTIAL *** NOTE: PERFORMING** DETECTION MAY INTERVENE **UPDATES ON TABLES WHICH ARE** LATER. **HEAVILY INDEXED CAN HAVE AN** IMPACT ON YOUR PERFORMANCE. WHEN TOO MANY INDEXES EXIST User response **ON A TABLE AND THE COLUMNS** As appropriate. SET ARE INDEXED, YOU CAN **INCUR A LARGE OVERHEAD FOR** ANL6003I ***** NOTE: THIS QUERY HAS INDEX MAINTENANCE. EACH REQUESTED TO USE SELECTIVE INDEXED COLUMN REQUIRES PARTITION LOCKING BUT THE TWO CHANGES TO EACH** LOCK SIZE IS TABLESPACE, **AFFECTED INDEX: ONE TO DELETE** WHICH WILL NEGATE THIS THE OLD ENTRY, AND ANOTHER FEATURE. SELECTIVE PARTITION TO INSERT THE NEW VALUE. DO LOCKING ALSO DOES NOT WORK NOT OVER-INDEX FREQUENTLY WITH TYPE 1 INDEX STRUCTURES **UPDATED TABLES. OR ACQUIRE (ALLOCATE) BIND** PARAMETERS. FOR THIS QUERY **User response** TO USE SELECTIVE PARTITION LOCKS, SET LOCK SIZE TO PAGE As appropriate. OR TABLE. *** NOTE, THIS OUEDVILLAS

ANL60021	INTENTIONALLY TURNED OFF THE USE OF PREFETCH BY SPECIFYING A LOW VALUE ON	User response As appropriate.	
	THE OPTIMIZE FOR N ROWS CLAUSE. THIS PREVENTS SELECTION OF NOT ONLY SEQUENTIAL OR LIST PREFETCH, BUT ALSO MULTI-INDEX ACCESS. IT ALSO DISCOURAGES BLOCK FETCH MODE FOR REMOTE CLIENT	ANL6004I	*** NOTE: THIS QUERY CONTAINS EITHER A MIN OR MAX BUILT-IN COLUMN FUNCTION AND HAS THE POTENTIAL TO CHOOSE AN I1 ACCESS METHOD, READING ONLY THE INDEX LEAF PAGE, BUT HAS NOT DONE SO. CONSIDER

BUILDING INDEX THAT IS ASCENDING FOR MIN, OR DESCENDING FOR MAX, ON THE COLUMN(S) USED IN THE FUNCTION. ALL OTHER PREDS MUST BE STAGE 1 AS WELL.

User response

As appropriate.

***** NOTE: THIS STATEMENT** ANL6005I **CONTAINS A BUILT-IN COLUMN** FUNCTION; HOWEVER, IT IS NOT **BEING PROCESSED AT EITHER STAGE 1 RETRIEVAL OR SORT** TIME. IN GENERAL, THIS MEANS **POOR PERFORMANCE, BECAUSE** THE COLUMN(S) GET EVALUATED **ONLY AT THE END OF STAGE 2 PROCESSING. THIS MIGHT BE DUE TO MULTIPLE COLUMNS USED, GROUP BY, NOT ALL STAGE** 1 PREDS, ETC.

User response

As appropriate.

ANL6006I ***** NOTE: THIS STATEMENT IS ACCESSING TABLE VIA A TABLESPACE SCAN. BUT NOT USING PREFETCH TO ACCESS THE** PAGES. VERIFY THAT THE STATISTICS IN THE CATALOG ARE **ACCURATE -- USUALLY A SMALL** VALUE FOR NPAGES AND / OR NACTIVE WILL CAUSE THIS SITUATION.

User response

As appropriate.

ANL6010I ***** NOTE: THIS QUERY COULD ONLY CHOOSE I/O PARALLELISM,** ANL6007I ***** RECOMMENDATION: HERE IS DESPITE BEING IN A MORE** SOME ADVICE: CONVERT THE PARALLEL ENVIRONMENT. **TABLESPACE TO A SEGMENTED POSSIBLE REASONS ARE: DESIGN. THIS ALLOWS BETTER CORRELATED SUBQUERY, MERGE** SPACE MANAGEMENT SCANNING **SCAN JOIN ON MORE THAN 1 ONLY ROWS THAT BELONG TO THE COLUMN, UPDATEABLE CURSOR** SINGLE TABLE REQUIRED AND **OR CURSOR WITH HOLD, HYBRID** ALSO PROMOTES CLUSTERING, JOIN WITH SORTN_JOIN YES, **ALLOWS LOCKSIZE TABLE, MASS** UNION, LACK OF CPU ENGINES, **DELETE, SPACE AVAILABILITY** OR ACCESS VIA TYPE 1 INDEX. AFTER A DROP TABLE WITHOUT **REWRITE RECOMMENDED. REORG, AND MUCH BETTER PERFORMANCE OVERALL.**

User response

As appropriate.

ANL6008I ***** RECOMMENDATION: IT IS POSSIBLE THAT THIS** TABLESPACE SCAN MIGHT BE CAUSED BY THE ORING OF PREDICATES. WHEN OR DOES NOT CHOOSE MULTIPLE INDEX SCAN OR NON-MATCHING INDEX SCAN FOR STAGE 1 PREDICATES. IT MIGHT BE WORTH REWRITING THIS QUERY AS THE UNION OF **SEVERAL STAGE 1 PREDS INSTEAD. CONSIDER THIS REWRITE TO AVOID THE TSCAN.**

User response

As appropriate.

***** RECOMMENDATION: THIS** ANL6009I TABLE APPARENTLY CONTAINS SOME VERY SMALL ROWS. BEING **UNDER 24 BYTES IN SIZE, WHICH COULD ALLOW MORE THAN 127 ROWS TO BE STORED ON SINGLE** PAGES: THIS COULD RESULT IN BETTER PERFORMANCE VIA LESS **I/O. YOU MIGHT ALTER COMPRESS YES TO FORMAT PAGE** FOR UP TO 255 ROWS, AND USE DATA COMPRESSION OR ALTER **MAXROWS TO SPECIFY UP TO 255 ROWS PER PAGE. REORG THE** TABLE, IN EITHER CASE, FOR **CHANGES TO TAKE EFFECT.**

User response

As appropriate.

ANL6011I	*** NOTE: THIS QUERY COULD	A
	ONLY CHOOSE SINGLE SYSTEM CP	
	PARALLELISM, DESPITE BEING	
	ELIGIBLE FOR SYSPLEX	
	PARALLELISM AS MEMBER OF A	
	DATA SHARING GROUP. POSSIBLE	
	REASONS ARE: CORRELATED	
	SUBQUERY, INDEX PROBES, IN	
	LISTS, MERGE SCAN JOIN ON	
	MORE THAN 1 COLUMN,	
	UPDATEABLE CURSOR OR	
	CURSOR WITH HOLD, HYBRID	
	JOIN WITH SORTN_JOIN YES,	I
	UNION, RR OR RS ISOLATION,	
	LIST PREFETCH WITH RANDOM	Α
	OR MULTI-INDEX, EXISTS, OUTER	7
	JOINS, AMBIGUOUS CURSOR	
	WITH CURRENTDATA YES,	
	MATERIALIZED VIEWS OR	
	GLOBAL TEMPORARY TABLES, OR	
	SYSTEM IS NOT A DESIGNATED	
	PARALLEL COORDINATOR.	
	INVESTIGATE CAUSE AND	
	REWRITE IF NEEDED.	

User response

As appropriate.

ANL6014I *** NOTE: MASS DELETES (ALL ROWS) FROM SEGMENTED TABLESPACES CAN BE NEGATED BY USE OF REFERENTIAL INTEGRITY, VALIDATION PROCS, OR THE REMOTE RECOVERY DATA FACILITY (RRDF) DATA CAPTURE CHANGES OPTION...MAKE SURE THAT YOU DO NOT FALL INTO ONE OF THESE EXCEPTION CATEGORIES.

User response

As appropriate.

ANL6015I *** NOTE: THE SEGMENTED TABLE IS MORE THAN 32 PAGES IN SIZE, YET THE SEGSIZE IS BELOW 32 PAGES. IN A MULTITABLE TABLESPACE, THE FINAL SEQUENTIAL PREFETCH COULD INADVERTENTLY READ PAGES IN THE NEXT TABLES SEGMENT BUT YOU CAN AVOID THIS SITUATION BY USING A SEGSIZE OF 32 OR 64.

User response

As appropriate.

ANL6012I	*** RECOMMENDATION: BIG
	TABLESPACES DEFINED BY
	DSSIZE OR LARGE SHOULD BE
	OPENED WITH CLOSE YES, AS
	EACH DATA AND INDEX
	PARTITION COUNT TOWARDS THE
	MAXIMUM NUMBER OF DATA SETS
	THAT CAN BE OPEN AT ONE TIME
	(10-32K THROUGH V7, OR 100K
	IN V8), WHICH MAY BE
	SUPERSEDED BY AN EVEN LOWER
	DSMAX PARM VALUE IN DB2.

User response

As appropriate.

ANL6013I *** RECOMMENDATION: THIS AMBIGUOUS CURSOR DOES NOT SEEM TO BE CAUSED BY HOST VARIABLES. THEREFORE, IT IS LIKELY THAT YOU CAN IMPROVE YOUR PERFORMANCE AND ELIMINATE AMBIGUITY BY CODING FOR FETCH ONLY ON THIS SELECT STMT.

User response

As appropriate.

ANL6016I *** NOTE: PRESENTLY, THIS QUERY IS ONLY MATCHING SOME OF THE INDEXED COLUMNS AVAILABLE WITH THIS INDEX. MAXIMIZING THE USE OF PREDICATES AGAINST THESE INDEX COLUMNS WILL IMPROVE PERFORMANCE, EITHER BY MATCHING OR SCREENING ROWS AND THEREFORE REDUCING DATA PAGE I/O REQUIREMENTS.

User response

ANL6017I	*** NOTE: THIS QUERY IS
	CURRENTLY PERFORMING A
	NONMATCHING INDEX SCAN,
	WHICH CAUSES IT TO READ
	EVERY INDEX LEAF PAGE TO
	PROCESS THE PREDICATES.
	READING THE ENTIRE LEAF TREE
	IS TIME-CONSUMING VS.

MATCHING A PART OF THE INDEX. CONSIDER BETTER PREDICATES THAT ALLOW YOU TO MATCH.

User response

As appropriate.

ANL6018I *** RECOMMENDATION: THIS NONMATCHING INDEX SCAN ACCESS PATH MIGHT BE CAUSED BY THE ORING OF PREDICATES. WHEN OR DOES NOT CHOOSE MULTIPLE INDEX SCAN OR NON-MATCHING INDEX SCAN FOR STAGE 1 PREDICATES, IT MIGHT BE WORTH REWRITING THIS QUERY AS THE UNION OF SEVERAL STAGE 1 PREDS INSTEAD. CONSIDER THIS REWRITE TO AVOID THE TSCAN.

User response

As appropriate.

ANL6019I *** RECOMMENDATION: OLD INDEX TYPE FOUND: TYPE 1 INDEXES SHOULD BE CONVERTED TO TYPE 2 TO ELIMINATE INDEX LOCKING ON INSERTS, UPDATES AND DELETES. THIS INDEX SHOULD BE CONVERTED TO TYPE 2 AS SOON AS POSSIBLE.

User response

As appropriate.

ANL6020I ***** NOTE: THIS INDEX IS** PRESENTLY DESIGNATED AS ALLOWING DUPLICATE VALUES. MAKE SURE THIS IS A NONUNIQUE INDEX THAT DOES NOT FALL INTO ONE OF THE **FOLLOWING UNIQUE CATEGORIES: EXPLICITLY** UNIQUE, PRIMARY KEY, NONPRIMARY RI PARENT KEY, UNIQUE WHERE NOT NULL, UNIQUE COLUMN CONSTRAINT. UNIQUE INDEXES HAVE DEFINITE ADVANTAGES WHERE EVER THEY CAN BE DECLARED BY THE USER. SO BE SURE IT CONTAINS DUPLICATES.

User response

As appropriate.

ANL6021I *** NOTE: THIS TABLESPACE IS DESIGNATED COMPRESSED, BUT YOUR INDEX ACCESS IS RANDOM...THE LIKELIHOOD OF FINDING MANY ROWS PER PAGE TO SATISFY YOUR QUERY IS REMOTE. THUS, DATA COMPRESSION IS NOT REDUCING YOUR I/O SIGNIFICANTLY (OK FOR STORAGE REDUCTION). PREFER CLUSTER INDEX.

User response

As appropriate.

ANL6022I *** RECOMMENDATION: ANY TYPE 1 NON-PARTITIONED SECONDARY INDEXES BUILT UPON PARTITIONED TABLESPACES DO NOT ALLOW CONCURRENT ACCESS, AND MAY DEFEAT PARALLEL EFFORTS. CONVERT THIS INDEX TO TYPE 2, TO TAKE ADVANTAGE OF LOGICAL PARTITIONING AVAILABLE VIA THE EXTRA RID BYTE, WHICH HOLDS PART NO.

User response

As appropriate.

ANL6023I	*** RECOMMENDATION: YOUR
	LARGER NON-PARTITIONED
	SECONDARY INDEXES BUILT ON
	PARTITIONED TABLESPACES CAN
	BE PHYSICALLY BROKEN UP AND
	DISTRIBUTED ACROSS MANY
	DISK DRIVES USING THE
	PIECESIZE PARAMETER TO
	DESIGNATE THE SIZE OF EACH
	SLICE OF THE INDEX TO BE
	STORED ON A DIFFERENT DATA
	SET.

User response

ANL6024I	*** NOTE: STAGE 2 PREDICATE(S)
	HAVE BEEN DETECTED IN
	STATEMENTTHESE DO NOT
	PERFORM AS WELL AS STAGE 1
	AND CONSUME EXTRA CPU
	PROCESSING.

As appropriate.

ANL6025I *** NOTE: THE COMPOSITE (FINAL) FILTER FACTOR INDICATES THAT A LARGE NUMBER OF ROWS WILL BE RETURNED.

User response

As appropriate.

ANL6026I *** NOTE: ARITHMETIC EXPRESSIONS THAT ARE NOT EXCLUSIVELY USING INTEGERS WILL BE STAGE 2 AND NOT MATCHING INDEXABLE. MOVE TO SELECT LIST IF POSSIBLE, OR DO THE MATH IN YOUR APPLICATION LATER.

User response

As appropriate.

ANL6027I *** NOTE: THE REFERENTIAL DELETE RULE OF CASCADE ON THE PARENT TABLE WILL FORCE CASCADED DELETES TO THIS DEPENDENT TABLE AS WELL.

User response

As appropriate.

ANL6028I *** NOTE: THE REFERENTIAL DELETE RULE OF SET NULL ON THE PARENT TABLE WILL FORCE CASCADED UPDATES TO THIS DEPENDENT TABLE AS WELL.

User response

As appropriate.

ANL6029I *** NOTE: THE REFERENTIAL DELETE RULE OF RESTRICT WILL NOT ALLOW ROWS TO BE DELETED FROM THIS TABLE IF ANY DEPENDENT ROWS EXIST.

User response

As appropriate.

ANL6030I *** NOTE: THE REFERENTIAL DELETE RULE OF NO ACTION WILL ONLY ALLOW SELF-REFERENCING

User response

As appropriate.

ANL6031I	*** NOTE: THE REFERENTIAL
	DELETE RULE OF CASCADE WILL
	FORCE CORRESPONDING ROWS
	TO BE DELETED FROM THIS TABLE
	IF ANY DEPENDENT ROWS EXIST.

User response

As appropriate.

ANL6032I	*** NOTE: THE REFERENTIAL
	DELETE RULE OF SET NULL WILL
	FORCE CORRESPONDING ROWS
	TO BE UPDATED IN THIS TABLE IF
	ANY DEPENDENT ROWS EXIST.

User response

As appropriate.

ANL6033I *** NOTE: EITHER THE NESTED LOOP OR MERGE SCAN JOIN METHODS ARE THE ONLY POSSIBILITIES WHEN LEFT OR RIGHT OUTER JOIN HAS BEEN REQUESTED.

User response

As appropriate.

ANL6034I	*** NOTE: ONLY THE NESTED
	LOOP JOIN METHOD SUPPORTS
	THE USE OF NON-EQUIJOIN
	PREDICATES (T1.C1 <= T2.C2). IF
	YOU ARE USING THESE
	PREDICATES TO JOIN TABLES
	TOGETHER, YOU ARE LIMITING
	DB2 TO SELECTING ONLY
	NLMERGE SCAN AND HYBRID
	REQUIRE EQUIJOIN PREDS (T1.C1
	= T2.C2).

User response

As appropriate.

ANL6035I	*** NOTE: THE MERGE SCAN JOIN
	METHOD IS ALWAYS EMPLOYED
	WHENEVER A FULL OUTER JOIN
	HAS BEEN REQUESTED.

DELETES TO THIS TABLE, BUT NOT TO DEPENDENTS.

As appropriate.

ANL6036I	*** NOTE: THIS IS A SUBQUERY
	PROCESS APPLIED TO A
	PARTITIONED TABLE. NO
	PARALLEL OPERATIONS ARE
	POSSIBLE BECAUSE IT INCLUDES
	THE EXISTS PREDICATE, WHICH
	NEGATES THE USE OF
	PARALLELISM.

User response

As appropriate.

ANL6037I *** NOTE: THIS IS A NONCORRELATED SUBQUERY WHICH IS USING THE IN PREDICATE. NO PARALLEL OPERATIONS WERE CHOSEN ON THIS PARTITIONED TABLESPACE: THEREFORE, IT IS POSSIBLE THAT DB2 IS BUILDING A SPARSE INDEX ON THEN INNER QUERY COLUMN, TO SPEED UP ACCESS TO THE INNER TABLE.

User response

As appropriate.

ANL6038I *** NOTE: THIS IS A CORRELATED SUBQUERY PROCESSED UPON A PARTITIONED TABLE. NO PARALLEL OPERATIONS ARE POSSIBLE WITH CORRELATED SUBQUERIES. THEREFORE, CONSIDER REWRITING AS NON-CORRELATED, OR AS A JOIN, SO THAT YOU CAN TAKE ADVANTAGE OF PARALLEL OPERATIONS.

User response

As appropriate.

ANL6039I *** NOTE: SCROLLABLE CURSORS DO NOT SUPPORT PARALLELISM AND CANNOT BE ACCESSED VIA PRIVATE PROTOCOL (DB2 MVS-MVS). IN ADDITION, BE AWARE THAT BUILT-IN FUNCTIONS DO NOT RE-EVALUATE (AVG STAYS CONSTANT) BUT ARITHMETIC EXPRESSIONS WILL BE RE-EVALUATED FOR EACH ROW FETCHED. DB2 USES DECLARED

User response

As appropriate.

ANL6040I *** NOTE: THIS STATEMENT WAS DECLARED WITH AN INSENSITIVE SCROLLABLE CURSOR. THE SIZE, THE ORDER OF ROWS AND THE VALUES FOR EACH ROW OF RESULT TABLE DO NOT CHANGE AFTER THE CURSOR IS OPENED. THE RESULT TABLE IS READ ONLY, AND THE CURSOR CANNOT BE USED FOR UPDATE | DELETE.

User response

As appropriate.

ANL6041I *** NOTE: THIS STATEMENT WAS DECLARED WITH A SENSITIVE STATIC SCROLLABLE CURSOR. POSITIONED UPDATES AND DELETES WILL BE VISIBLE IN THE RESULT TABLE AND CHANGES MADE TO THE UNDERLYING TABLE BY OTHER CURSORS MAY OR MAY NOT BE VISIBLE, DEPENDING UPON WHETHER FETCH IS SENSITIVE OR INSENSITIVE TO CHANGES. INSERT TO UNDERLYING TABLE IS NOT VISIBLE.

User response

As appropriate.

ANL6042I *** NOTE: DYNAMIC STATEMENT CACHING IS ASSUMED FOR THIS SET OF ESTIMATES.

User response

As appropriate.

ANL6043I *** NOTE: IMPROVEMENTS TO OUTER JOIN PROCESSING IN THIS AND SUBSEQUENT DB2 RELEASES INCLUDE ABILITY TO USE PREDICATE TRANSITIVE CLOSURE AND APPLY PREDICATES FROM THE WHERE CLAUSE, PLUS OTHER STREAMLINING.

As appropriate.

ANL6044I *** NOTE: CONNECTION VIA DRDA IS INDICATED, AND OPTIMIZE FOR N ROWS IS ALSO DEPLOYED. A MINIMUM NUMBER OF ROWS MAY BE RETURNED (16) PER FETCH. AND DIMENSIONS, NO CORRELATED SUBQUERY CROSSES DIMENSIONS, NO OUTER JOINS ARE LEFT AFTER QUERY REWRITE, AND DATA TYPE AND LENGTHS ARE THE SAME FOR ALL THE JOIN PREDICATES.

User response

As appropriate.

ANL6052I

User response

As appropriate.

ANL6045I *** NOTE: THIS INDEX IS THE PRIMARY KEY: YOU MUST DROP EITHER THE KEY OR THE REFERENTIAL CONSTRAINT BEFORE DROPPING THIS INDEX.

User response

As appropriate.

ANL6046I	*** NOTE: IN (LIST) PREDICATE
	IS BEING PROCESSED IN
	PARALLEL, RESULTING IN
	IMPROVED ELAPSED TIME AND
	PERFORMANCE.

User response

As appropriate.

ANL6047I *** NOTE: THIS NC SUBQUERY IS BEING PROCESSED USING AN INDEX ON THE OUTER COLUMN. FIRST, INNER QUERY IS EXECUTED AND SORTED INTO THE OUTER COLUMN ORDER. THEN, FOR EACH VALUE IN THE INNER TABLE, THE OUTER INDEX IS SCANNED FOR MATCHES, REVERSING USUAL NONCORRELATED ORDER.

User response

As appropriate.

ANL6051I *** NOTE: ONCE A STAR JOIN QUALIFIES, NO OTHER JOIN METHOD IS CONSIDERED. QUALIFICATIONS INCLUDE A SIGNIFICANTLY LARGER FACT TABLE (25X DEFAULT), AT LEAST DIMENSION TABLES, ALL EQUI-JOIN BOOLEAN TERM PREDICATES BETWEEN THE FACT

***	* NOTE: THIS INDEX IS
SPI	ECIFIED AS NOT PADDED,
ALI	OWING STORAGE OF A
VAI	RYING LENGTH INDEX KEY.
PAI	DDED INDEXES USE BLANKS
то	FILL OUT THEIR FIXED LENGTH
KE	YS AND ARE NOT ELIGIBLE FOR
IND	DEX ONLY. NOT PADDED
IND	DEXES DO NOT BLANK FILL
CH	AR/VARCHAR COLUMNS
ALI	OWING GREATER FLEXIBILITY
AN	D BETTER USE OF STORAGE,
PAG	CKING MORE ENTRIES INTO A
SIN	IGLE LEAF PAGE. INDEX ONLY
AC	CESS IS PERMITTED ON NOT
PAI	DDED INDEXES.

User response

As appropriate.

ANL6053I	*** NOTE: THIS INDEX IS
	DEFINED AS A DATA
	PARTITIONED SECONDARY INDEX
	(DPSI) FOR THE TABLE. A DPSI
	CONTAINS ORDERED SETS OF
	KEYS ON COLUMNS WITHIN EACH
	PARTITION AND MAY IMPROVE
	PERFORMANCE WHEN ACCESS IS
	COMBINED WITH PARTITIONED
	KEYS. DPSI IS A TABLE-BASED
	PARTITIONING SCHEME

User response

As appropriate.

ANL6054I *** NOTE: THIS INDEX IS DEFINED AS THE PRIMARY PARTITIONING INDEX FOR THE TABLE. THE TRADITIONAL INDEX-BASED PARTITIONING SCHEME IS CONTROLLED BY THIS INDEX.

User response

ANL6055I	*** RECOMMENDATION: THIS MQT IS NOT CREATED IN A SEGMENTED TABLE SPACE. ATTEMPTS TO REFRESH THIS TABLE WILL CAUSE LARGE		APPLIES TO TRUE OPPOSITE INDEXES THAT INCLUDES REVERSAL OF ASC AND DESC ORDER.
	OVERHEAD FROM A MASS DELETE	User response	e
	REPOPULATION. CONVERT TO A	As appropriate.	
	SEGMENTED TABLE SPACE	ANL6059I	*** RECOMMENDATION: CHANGE
User respons	e		THE SEGSIZE OF THIS TABLE SPACE TO 32 PAGES, MOST
As appropriate	-		TABLES BETWEEN 28 AND 127
			PAGES WILL BENEFIT FROM THIS
ANL6056I	*** NOTE: THIS STATEMENT WAS		CHANGE BY MAXIMIZING PREEFTCH PERFORMANCE, DO
	SCROLLABLE CURSOR, WHICH		THIS UNLESS TABLE SPACE
	ALLOWS SCROLLING DIRECTLY		CONTAINS MANY OTHER SMALLER
	ON THE BASE TABLE AND IS		TABLES.
	SENSITIVE TO ALL COMMITTED		_
	USERS DO NOT HAVE TO COMMIT	User response	e
	THEIR CHANGES TO SEE THEM	As appropriate.	
	WITH DYNAMIC SCROLLING	ANL6060I	*** RECOMMENDATION: CHANGE
llear rochang			THE SEGSIZE OF THIS TABLE
User respons	e		SPACE TO 64 PAGES. ANY TABLE THAT IS AT I FAST 128 PAGES
As appropriate.			WILL BENEFIT FROM THIS
ANL6057I	*** NOTE: INDEXED ACCESS ON A		CHANGE BY MAXIMIZING
	SCROLLABLE CURSOR CAN AVOID		PREFETCH PERFORMANCE. DO
	ORDER BY SORT WHEN THE KEY IS AN EXACT MATCH OR EXACT		CONTAINS MANY OTHER TABLES
	OPPOSITE (FOR BACKWARD		128 PAGES IN SIZE.
	SCAN). THUS INDEX ON C1 ASC,		
	C2 DESC WILL AVOID ORDER BY C1 ASC, C2 DESC FORWARD SCAN, AND EMPLOY BACKWARD	User response	
		As appropriate.	
	SCAN TO AVOID ORDER BY C1		
	DESC, C2 ASC. HOWEVER, SORTS	ANL60611	INDEX IS USED TO ACCESS TABLE.
	MUST BE PERFORMED FOR ORDER		BUT LIMITED PAGE RANGE SCAN
	BY C1 DESC, C2 DESC OR ORDER BY C1 ASC C2 ASC AS THEY ARE		WAS NOT PERMITTED. TO
	NOT EXACT MATCHES OR		IMPROVE CHANCES OF PRUNING
	OPPOSITES.		SOME PARTITIONS FROM SCAN
			REOPT(ONCE) TO FORCE
User respons	e		REEVALUATION OF HOST
As appropriate.			VARIABLES.
ANL6058I	*** RECOMMENDATION:	User response	e
	COLUMNS IN REVERSE ORDER	As appropriate.	
	CAN NOW BE DELETED AS DB2	ANL60621	*** NOTE: PARALLEL DATA
	DOES NOT REQUIRE THEM FOR ORDER BY OR SEQUENTIAL		ACCESS IS BEING PERFORMED,

BUT PARALLEL SORTING WAS

PARALLEL SORTING, THE TOTAL

NOT SELECTED. TO ENABLE

SCANS: INDEXES CAN NOW BE

SCANNED BACKWARDS. THIS

RECOMMENDATION ONLY
SORT DATA SIZE MUST BE > 2 MB (OR 500 PAGES), AND/OR THE SORT DATA PER PARALLEL DEGREE MUST BE > 100KB (OR 25 PAGES). IF YOUR DATA EXCEEDS THESE SIZES THEN ENSURE THAT THE ZPARM QWP40PSE IS ENABLED.

***** NOTE: THIS STATEMENT**

CONTAINS A MULTI-ROW FETCH.

IN WHICH ROWS ARE GROUPED IN A SET AND OPERATIONS CAN **BE PERFORMED ON THE CURRENT ROW SET, OR THE USER CAN** FETCH THE NEXT SET.

WITH THIS MATCHING INDEX SCAN OPERATION.

User response

As appropriate.

ANL6068I ***** NOTE: A SIGNIFICANT** NUMBER OF INDEX LEAF PAGES (SITE DEPENDENT) ARE READ WITH THIS NON-MATCHING **INDEX SCAN OPERATION.**

User response

As appropriate.

ANL6069I	*** NOTE: BUILDING AN INDEX
	WITH THE SAME ORDERING OF
	SORT KEY COLUMNS WILL
	NEGATE THE SORT REQUESTED
	HERECONSIDER AS
	WARRANTED.

User response

User response

As appropriate. ANL6063I

As appropriate.

ANL6064I	*** NOTE: A MULTI-ROW INSERT
	HAS BEEN DETECTED WITH THE
	ATOMIC ATTRIBUTE: IN THE
	EVENT OF ANY FAILURE, ALL
	CHANGES WILL BE ROLLED BACK.

User response

As appropriate.

ANL60651 ***** NOTE: A MULTI-ROW INSERT** HAS BEEN DETECTED WITHOUT THE ATOMIC ATTRIBUTE: ANY **INSERTED ROWS WILL BE COMMITTED, EVEN IF SOME INSERTS FAIL.**

User response

As appropriate.

ANL6066I

***** NOTE: A SIGNIFICANT** NUMBER OF DATA PAGES (SITE **DEPENDENT) ARE READ WITH** THIS TABLESPACE SCAN **OPERATION.**

User response

As appropriate.

ANL6067I	*** NOTE: A SIGNIFICANT
	NUMBER OF INDEX LEAF PAGES
	(SITE DEPENDENT) ARE READ

User response

As appropriate.

ANL6070I	*** NOTE: THE TABLE APPEND
	FLAG IS SET. THEREFORE ALL
	INSERTS WILL BE DONE AT THE
	END OF THIS TABLE SPACE.

User response

As appropriate.

ANL6071I *** NOTE: THIS IS A PARTITION-**BY-GROWTH, WHICH HAS THE BENEFITS OF ADDING** PARTITIONS AS REQUIRED, PLUS THE INTERNAL FORMAT OF A SEGMENTED TABLESPACE.

User response

As appropriate.

ANL6072I ***** NOTE: THIS TABLE SPACE** CONTAINS OBJECTS THAT ARE **INVOLVED IN A CLONE RELATIONSHIP.**

User response

As appropriate.

ANL6073I ***** NOTE: THIS IS A UNIVERSAL TABLE SPACE (UTS), WHICH HAS** THE BENEFITS OF USING THE

INTERNAL FORMAT OF A
SEGMENTED TABLE SPACE (MASS
DELETES ON SPACE MAP ONLY,
BETTER SPACE MANAGEMENT,
ETC.).

As appropriate.

ANI 6074T

NL6074I	*** NOTE: REFERENTIAL
	INTEGRITY PRIMARY AND
	FOREIGN KEYS CANNOT BE
	CREATED AS "INDEX ON
	EXPRESSION" TYPE.

User response

As appropriate.

ANL6075I	*** NOTE: INDEX ON
	EXPRESSION WAS USED AS THE
	ACCESS PATH FOR THIS QUERY
	BLOCK.

User response

As appropriate.

ANL6076I ***** NOTE: SELECT FROM UPDATE OR DELETE WILL NOT USE ANY** PARALLELISM.

User response

As appropriate.

Good news and guidelines

The following messages are intended to guide newer programmers toward good query coding. Good News and Guidelines are active if the ADVISOR parameter is set to ALL in the user parameters file (ANLPARM).

The messages are self-explanatory; no additional explanations are shown.

ANL7001I	*** GUIDELINE: MAKE SURE THAT YOUR FREQUENTLY UPDATED COLUMNS ARE BEING KEPT TOGETHER IN THE TABLE		LENGTH & LOGGED IN COMPRESSED FORMAT.
	ROW: DB2 LOGS FROM THE FIRST TO LAST CHANGED BYTE. ALSO,	User respons As appropriate.	e
	ROW: DB2 LOGS FROM THE FIRST CHANGED BYTE TO END OF ROW. LENGTH CHANGES CAUSE THE ENTIRE ROW TO LOG, AS LENGTH FIELD IS KEPT WITH THE ROW HEADER. ROWS THAT ARE COMPRESSED ARE VARIABLE	ANL7002I	*** GUIDELINE: THIS PLAN STEP HAS NOT SELECTED ANY SEQUENTIAL LIST PREFETCH I/O. IF THE SQL PROCESSES JUST A FEW ROWS THAT IS OK, BUT IF MANY ROWS ARE INVOLVED, YOU CAN HELP PROMOTE SEQUENTIAL DETECTION BY BOTH ACCESSING

***** NOTE: FETCH FIRST N ROWS** ANL6077I LIMITS THE ANSWER SET **PRODUCED BY THIS QUERY. ORDER BY LIST MAY NOT BE COMPLETE QUALIFIED ROWS** LIST.

User response

As appropriate.

ANL6080I ***** NOTE: SKIP LOCKED ROWS CLAUSE MAY RESULT IN QUERY PROCESSING LESS ROWS THAN ACTUALLY QUALIFIED IF ROWS OR PAGES ARE LOCKED WHEN** READ.

User response

As appropriate.

ANL60831 *** RECOMMENDATION: The package or plan program-name was last bound by a version of DB2 that is earlier than Version 9. DB2 will either bind the package or issue an error, depending on the value of the DB2 AUTOBIND option.

THE DATA IN SEQUENTIAL ORDER (PRESORT?) AND BY BINDING WITH RELEASE (DEALLOCATE) TO AVOID RESETTING COUNTERS AT COMMIT.

User response

As appropriate.

ANL7003I ***** GUIDELINE: CLOSE YES WAS** SPECIFIED FOR THE TABLESPACE AND/OR THE INDEX...IF THESE ARE LITTLE USED THIS IS OK. IF **HIGH VOLUME ACCESS THEN CONSIDER CLOSE NO. EXTREMELY RELEVANT PAGES CAN BE PAGE FIXED IN MEMORY BY HIGHLY REFERENCING, USING HIPERPOOLS (THROUGH V7),** PUTTING INTO A DEDICATED **BUFFER POOL LARGE ENOUGH TO** HOLD ALL PAGES, DEPLOYING DATA SHARING WITH GROUP **BUFFER POOL CACHE ALL OPTION, ETC. EACH WITH ASSOCIATED COSTS. CLOSE NO** ALSO INCREASES CHANCES THAT **DBD WILL REMAIN IN EDM POOL** FOR NEXT EXECUTION.

User response

As appropriate.

ANL7004I *** GOOD NEWS: THIS QUERY IS ACCESSING A TABLE AND/OR INDEX IN PARALLEL EMPLOYING A LIMITED PAGE RANGE SCAN. THIS MEANS THAT NOT ALL PARTITIONS ARE NECESSARY TO RESOLVE THE QUERY. INSTEAD, DB2 WILL FIND THE ANSWER SET IN ONLY A SUBSET OF THE PARTITIONS OF THE TABLE AND/OR INDEX.

User response

As appropriate.

ANL7005I *** GOOD NEWS: THIS SQL FEATURES A UNIQUE INDEX BEING USED TO ACCESS THE TABLE THAT CAN AVOID DISTINCT SORTS WITH GROUP BY AND COUNT DISTINCT C1. ALSO, UNIQUE INDEX WILL AVOID 4K IN MEMORY TABLE ON CORRELATED SUBQUERY, CONSIDERS SPECIAL ACCESS PATH WHEN USED WITH = PREDS, AND MAY CONVERT SUBQUERY TO JOIN ON IN, =ANY OR =SOME PREDICATES.

User response

As appropriate.

ANL7006I *** This statement contains a select of all columns in the table. Typically a select of all columns results in significantly increased processing time to fetch and process each column of each row. Use select all only when you want to select all columns in a view definition. Improve performance by specifying the name of each column for the SQL SELECT statement.

Explanation:

Using the SELECT ALL statement results in slower query performance when compared to queries that specify the column names.

User response

Modify your queries to specify the column names in the SELECT statements.

ANL7007I *** GUIDELINE: YOU MAY WISH TO CONSIDER CONVERTING THIS TABLESPACE TO PARTITIONED DESIGN. THE TOTAL NUMBER OF PAGES IS SIGNIFICANT ENOUGH TO BENEFIT FROM PARALLEL PROCESSING TO REDUCE OVERALL ELAPSED TIME.

User response

As appropriate.

ANL7008I *** GUIDELINE: ATTEMPT TO PARTITION YOUR TABLESPACES EVENLY, AND MAKE ENOUGH PARTITIONS TO EXPLOIT THE MAXIMUM NUMBER OF PROCESSORS ONLINE.

User response

As appropriate.

ANL7009I	*** GUIDELINE: ON COMPOSITE
	(MULTI-COLUMN) INDEXES,
	FAVOR MATCHING INDEX SCAN

OF AS MANY COLUMNS AS POSSIBLE BY USING EQUALS (=), RANGE (>, <, =>, <=, BETWEEN, LIKE), IN (LIST) AND NULL PREDICATES, WHERE A RANGE PRED IS THE LAST MATCHING PREDICATE. APPLY LEFT TO RIGHT AGAINST INDEX COLUMNS, SKIPPING NO COLUMNS IN THE L-R SEQUENCE. DB2 MAY SCREEN REMAINING STAGE 1 PREDS AGAINST THE RID LIST BEFORE DATA ACCESS.

User response

As appropriate.

ANL7010I ***** GUIDELINE: ON COMPOSITE** (MULTI-COLUMN) INDEXES, COLLECT CORRELATED KEY STATS **ON COLUMN PAIRINGS, AS WELL** AS CARDINALITY STATS, LIKE 2NDKEYCARD, 3RDKEYCARD, ETC. WITH RUNSTATS. THESE ADDITIONAL STATISTICS WILL **ALLOW DB2 TO MAKE BETTER** FILTER ESTIMATES FOR EQUALITY AND RANGE AND IN LIST **PREDICATES, THEREBY** SELECTING BETTER INDEXES AND MORE REALISTIC ESTIMATES FOR **ROWS RETURNED FROM THE** QUERY (QCARD).

User response

As appropriate.

ANL7011I *** GUIDELINE: THIS OLD TYPE 1 INDEX FORMAT SHOULD BE CONVERTED INTO TYPE 2 TO ELIMINATE INDEX LOCKING. TYPE 1 INDEXES DO NOT ALLOW: LOCKING AT THE ROW LEVEL, UR ISOLATION, CP OR SYSPLEX PARALLELISM, LOGICAL PARTITIONING OF NONPARTITIONED SECONDARY INDEXES, ETC.

User response

As appropriate.

ANL7012I *** GUIDELINE: MAKE SURE THAT YOU HAVE DEFINED ADEQUATE FREE SPACE PARMS IN THIS INDEX STRUCTURE, AS WELL AS IN THE DATA TABLE, TO KEEP ROWS IN THE CLUSTERED ORDER AND AVOID FRAGMENTATION. BOTH FREEPAGE AND PCTFREE SHOULD BE DEPLOYED, AND RUNSTATS PERIODICALLY RUN TO UPDATE STATS.

User response

As appropriate.

ANL7013I ***** GUIDELINE: DO NOT USE** ABSOLUTE CURSOR POSITIONING WHEN YOU ARE UPDATING ONLY A SINGLE ROW, UNLESS YOU HAVE SELECTED THE ROW TO **GUARANTEE THAT NO CHANGES** WILL OCCUR BETWEEN THE SELECT AND UPDATE. ELSE, IT IS MORE TIME CONSUMING FOR DB2 **TO POSITION THE CURSOR RATHER THAN TO DO A DIRECT** UPDATE ON THE ROW. ALSO, **CONSIDER KEEP UPDATE LOCKS AS A MEANS OF SERIALIZING** UPDATES AND COMPLETING THEM QUICKLY.

User response

As appropriate.

ANL7014I *** GUIDELINE: DO NOT USE ABSOLUTE CURSOR POSITIONING WHEN YOU ARE DELETING ONLY A SINGLE ROW, UNLESS YOU HAVE SELECTED THE ROW TO GUARANTEE THAT NO CHANGES WILL OCCUR BETWEEN THE SELECT AND DELETE. ELSE, IT IS MORE TIME CONSUMING FOR DB2 TO POSITION THE CURSOR RATHER THAN TO DO A DIRECT DELETE ON THE ROW.

User response

As appropriate.

ANL7015I *** GUIDELINE: COLLECTING NON-UNIFORM COLUMN STATISTICS GREATLY ENHANCE THE DB2 OPTIMIZER ACCURACY WHEN PROCESSING EQUALS AND IN (LIST) PREDS.

As appropriate.

ANL7016I *** GUIDELINE: CONSIDER BIND OPTION REOPT (VARS) REOPT(ONCE) IN V8+ TO HELP ENHANCE PERFORMANCE OF POORLY RUNNING SQL USING HOST VARIABLES, REOPTIMIZING AT RUNTIME.

User response

As appropriate.

ANL7017I *** GUIDELINE: THIS PROCESS JOINS A SIGNIFICANT NUMBER OF TABLES TOGETHER: THE EXPOSURE TO PERFORMANCE PROBLEMS IS INCREASED. IN GENERAL, YOU SHOULD TRY TO KEEP TO A REASONABLE NUMBER FOR OPTIMUM QUERY PERFORMANCE. LIMIT IS 15 TABLES FOR NORMAL JOIN, (UP TO 225 in V8+), 225 FOR STAR JOIN.

User response

As appropriate.

ANL7020I *** GOOD NEWS: THIS SUBQUERY HAS BEEN AUTOMATICALLY TRANSFORMED BY DB2 INTO A SIMPLE PREDICATE, ELIMINATING THE SUBQUERY PROCESSING OVERHEAD.

User response

As appropriate.

ANL7021I	*** GOOD NEWS: THIS
	CORRELATED SUBQUERY HAS A
	UNIQUE INDEX ON ONE BOTH
	TABLES AND THEREFORE WILL
	NOT HAVE TO BUILD A 4K IN
	MEMORY TABLE TO STORE
	RESULTS OF PREVIOUS
	CORRELATIONS. BEING UNIQUE,
	THE CHANCES OF FINDING A
	MATCH ARE NIL, SO DB2 AVOIDS
	THE UNNECESSARY OVERHEAD.

User response

As appropriate.

ANL7018I *** GUIDELINE: IF YOU WISH TO REVERSE THE NESTED LOOP JOIN TABLE ORDER, USE SOME REDUNDANT PREDICATES (TN.CN = TN.CN) ON THE TABLE YOU WOULD LIKE TO BECOME THE OUTER TABLE IN THE PROCESS (REVERSE THIS LOGIC FOR MERGE SCAN JOINS, APPLYING THEM TO THE INNER TABLE INSTEAD).

User response

As appropriate.

ANL7022I *** GUIDELINE: AVOID REFERENCING VARCHAR COLUMNS IN THE SORT, AS THESE WILL BE PADDED TO MAXIMUM SIZE IN THE SORT WORK RECORDS. LONG SORT RECORDS OVER 4000 BYTES CAUSE A TAG SORT, WHICH REQUIRES EVEN MORE OVERHEAD TO PARK RECORDS IN A FILE AND READ THEM AFTERWARD.

User response

As appropriate.

ANL7019I*** GUIDELINE: TO SELECT
NESTED LOOP OR HYBRID JOIN
INSTEAD OF THE MERGE SCAN,
YOU CAN TRY THE OPTIMIZE FOR
N ROWS CLAUSE TO MAKE THE
DB2 OPTIMIZER THINK® THAT
THESE TABLES ARE SMALLER,
AND THE RESULTING ANSWER
SET IS ALSO SMALL. MS IS LIKELY
SELECTED WHEN MORE ROWS
WILL RESULT, OR WHEN NO

User response

As appropriate.

ANL7023I *** GOOD NEWS: EQUI-JOIN PREDICATES OF UNEQUAL LENGTH MAY BE HANDLED AT STAGE 1: DB2 PADS OUT CHAR AND VARCHAR COLUMN TYPES TO ACCOMPLISH THE JOIN.

As appropriate.

ANL7024I	*** GOOD NEWS: INDEX
	SCREENING IS AVAILABLE FOR
	LIST PREFETCH PROCESSING,
	WHICH ALLOWS RANDOM
	INDEXES TO SCREEN
	ADDITIONAL PREDICATES AFTER
	MATCHING INDEX SCAN.

User response

As appropriate.

ANL7025I *** GOOD NEWS: ORDER BY SORTS NO LONGER NEED TO SPECIFY SORT KEY COLUMNS IN THE SELECT LIST, BEGINNING IN V5R2+. HOWEVER, THE RESTRICTION STILL APPLIES WITH ORDER BY USED IN CONJUNCTION WITH UNIONS, OR GROUP BY|DIFSTINCT SORTS AND MIN, MAX OR SUM BUILT-IN FUNCTIONS.

User response

As appropriate.

ANL7026I ***** GUIDELINE: THIS IS A** MATERIALIZED QUERY TABLE (MQT) WHICH CONTAINS SUMMARY DATA. MOTS CAN BE **USER OR SYSTEM MAINTAINED – BOTH METHODS CAN USE THE REFRESH TABLE STATEMENT, BUT** USER-MAINTAINED ALSO ALLOW **INSERT, UPDATE, DELETE AND** LOAD UTILITY. CURRENT **REFRESH AGE ANY MUST BE SET** TO ALLOW CONSIDERATION BY **OPTIMIZER FOR AUTOMATIC QUERY REWRITE, WHICH IS USED** FOR DYNAMIC SOL ONLY. TO USE AN MQT IN STATIC SQL, REFER TO THE MQT TABLE NAME DIRECTLY.

User response

As appropriate.

ANL7027I *** GOOD NEWS: THIS STATEMENT IS ACCESSING A MATERIALIZED QUERY TABLE (MQT) WHICH SHOULD RESULT IN BETTER, FASTER ACCESS TO YOUR DATA. THE OPTIMIZER CHOSE AN MQT, NOT THE BASE TABLE, USING AUTOMATIC QUERY REWRITE. MQTS ARE READ ONLY TABLES, AND ELAPSED TIME SHOULD BE REDUCED VS. BASE TABLE ACCESS.

User response

As appropriate.

ANL7028I *** GOOD NEWS: THIS SQL STATEMENT IS RETRIEVING ITS VALUES DIRECTLY FROM BUFFERS IN MEMORY, RESULTING IN QUICKER ACCESS. THIS METHOD OF INSERT WITHIN SELECT IS GOOD FOR RETURNING ROWIDS, IDENTITY COLUMNS, SEQUENCES AND THE RESULTS OF TRIGGERS FROM THE FINAL TABLE.

User response

As appropriate.

ANL7029I *** GOOD NEWS: DYNAMIC SCROLLABLE CURSORS ALLOW BACKWARD INDEX SCAN AND BACKWARD SEQUENTIAL TABLE SCAN, INCLUDING BACKWARD SCANS FOR ORDER BY SORTS.

User response

As appropriate.

ANL7030I *** GUIDELINE: AN ASENSITIVE SCROLLABLE CURSOR HAS BEEN DECLARED. IF SQL IS READ ONLY THEN IT BECOMES AN INSENSITIVE CURSOR; IF SQL IS NOT READ ONLY THEN IT BECOMES A SENSITIVE DYNAMIC CURSOR.

User response

As appropriate.

ANL7031I *** GOOD NEWS: ORDER BY SORT FOR THIS SCROLLABLE CURSOR HAS BEEN AVOIDED USING FORWARD AND/OR BACKWARD SCAN ON A SUITABLE INDEX INSTEAD.

As appropriate.

ANL7032I ***** GUIDELINE: SCROLLABLE CURSORS REQUIRING USE OF** WORK FILE CANNOT BE **DECLARED SENSITIVE DYNAMIC** (USE OF GROUP BY, FOR **EXAMPLE). CURSORS WILL NOT REFLECT CHANGES TO SUBQUERY** TABLES, AS BASIC PREDICATE IS **ONLY EVALUATED FOR OPEN CURSOR. ALSO, BUILT IN** FUNCTIONS (MIN, MAX, AVG, COUNT) GET CALCULATED ONLY **ONCE AND ARE NOT REEVALUATED; FINALLY,** PARALLELISM IS NOT SUPPORTED.

LENGTH AGAINST THIS INDEX DURING STAGE 1.

User response

As appropriate.

ANL7035I *** GOOD NEWS: THIS TABLE IS DESIGNATED AS VOLATILE, MEANING THAT DB2 WILL ALWAYS USE AN INDEX TO ACCESS ITS DATA, IF ONE IS AVAILABLE.

User response

As appropriate.

ANL7036I *** GOOD NEWS: A SPARSE INDEX WAS EMPLOYED BY DB2 TO ACCESS THIS WORK FILE DURING THE STAR JOIN PROCESS. SPARSE INDEXES ARE SUPERIOR TO TABLE SPACE SCANS.

User response

As appropriate.

ANL7033I ***** GUIDELINE: DPSI ARE** INDEXES THAT ARE ORDERED BY THE DATA WITHIN EACH PARTITION. THEY ARE GOOD FOR ACCESSING DATA ORGANIZED ALONG PARTITION BOUNDARIES. PARTITION PRUNING MAY OCCUR IF THIS IS DESIGNATED AS THE **PARTITIONING INDEX, BUT IF** COMPARING INDEX COLUMNS TO HOST VARIABLES OR PARAMETER MARKERS THEN REOPT (VARS) MUST BE USED TO PRUNE. DPSI ARE GOOD WHEN RANDOM DATA ACCESS IS LIMITED, OR WHEN PREDICATES USE COLUMNS FROM **BOTH A DPSI AND A TRADITIONAL PARTITIONING** INDEX.

User response

As appropriate.

ANL7034I *** GOOD NEWS: THIS INDEX IS NOT PADDED, MAKING IT A TRUE VARIABLE LENGTH INDEX. INDEX ONLY ACCESS IS NOW POSSIBLE AND DB2 WILL COMPARE CHAR / VARCHAR COLUMNS OF UNEQUAL

User response

As appropriate.

ANL7037I *** GUIDELINE: DYNAMIC SQL CAN CACHE THE PREPARED STATEMENT TO REDUCE REEXECUTION OVERHEAD. USE REOPT(ONCE) TO LOAD BEST FIT ACCESS PLAN; USE RUNSTATS WITH REPORTS NO UPDATE NONE TO INVALIDATE (REMOVE) PREPARED SQL STATEMENTS IN CACHE.

User response

As appropriate.

ANL7038I *** GUIDELINE: INDEX ON EXPRESSION CANNOT BE BUILT ON A SUBQUERY, AGGREGATE FUNCTION, UDF, HOST VARIABLE, PARAMETER MARKER, SPECIAL REGISTER OR CAST EXPRESSIONS. ALSO, DESC ORDER IS NOT PERMITTED.

User response

As appropriate.

Statistics collector messages

ANL8001I COLLECTING STATS FOR TABLE creator.tablename

Explanation

Informational. You have requested collection of all Table, Table space, and Index statistics pertaining to the table identified by its *creator* (8 characters) and *tablename* (18 characters). The table must be specified in fixed field A8.A18 format, with a dot or space between the names, as: COLLECT TDT690 .L1000; otherwise, the table name cannot be properly identified. Statistics are automatically collected from SYSIBM.SYSTABLES, SYSTABLESPACE and SYSINDEXES.

User response

None

ANL8002I CAF OPEN RETURN CODE IS ccc

Explanation

Informational. The Call Attach Facility Open routine has returned the value *ccc*. In normal operations, *ccc* is 0. A value of 4 or more indicates a problem attaching to the DB2 subsystem. Check the setting of the SUBSYST parameter for accuracy.

User response

As appropriate.

ANL8003I CAF CLOSE RETURN CODE IS ccc

Explanation

Informational. The Call Attach Facility Open routine has returned the value *ccc*. In normal operations, *ccc* is 0. A value of 4 or more indicates a problem disconnecting from the DB2 subsystem.

User response

As appropriate.

ANL8004E ERROR ENCOUNTERED: errcode SQLCODE: sqlcode

Explanation

Error. The program encountered an error and was unable to continue. The latest error condition (*errcode*) and DB2 SQLCODE (*sqlcode*) are returned for diagnosis.

User response

As appropriate.

ANL8005I	PROGRAM ENDS, STATISTICS
	COLLECTED FROM:
	SYSIBM.SYSTABLES n
	SYSIBM.SYSTABLESPACE n
	SYSIBM.SYSINDEXES n
	SYSIBM.SYSCOLUMNS n
	SYSIBM.SYSCOLDIST n
	SYSIBM.SYSCOLSTATS n
	SYSIBM.SYSCOLDISTSTATS n
	SYSIBM.SYSTABSTATS n
	SYSIBM.SYSINDEXSTATS n
	SYSIBM.SYSKEYTARGETS n
	SYSIBM.SYSKEYTGTDIST n

Explanation

Informational. The final count of the records extracted from each of the potential SYSIBM catalog tables at the end of the program.

n

Indicates the number of records extracted from each catalog table.

User response

None

ANL8006I DB2 SUBSYSTEM RELEASE LEVEL IS vers.

Explanation

Informational. The DB2 version for which output DDL is being written (that is, the migration target).

User response

None

ANL8007W

THE *limit object* COLLECTION LIMIT EXCEEDED. ONLY THE FIRST *limit* ENTRIES ARE PROCESSED.

Explanation

Warning. You have requested more than the maximum number of table, table space, or index entries in a single run. The program stops collecting information after the maximum number of COLLECT parameters have been processed. Those specified parameters that do not exceed *limit* continue processing.

limit

Indicates the maximum number of object entries allowed by the COLLECT parameter in a single run.

User response

Separate the job into several jobs that each have a subset of the objects that does not exceed the limit, and submit the new jobs.

```
ANL8008I PARTITION LEVEL STATISTICS
ARE REQUESTED.
```

Explanation

Informational. Partition level statistics have been requested. ANLCAT collects all entries found in SYSIBM.SYSTABSTATS, SYSINDEXSTATS. SYSCOLSTATS, and SYSCOLDISTSTATS for all partitioned tables collected.

User response

None

ANL8009W NO ENTRY FOUND FOR TABLE creator.tablename.

Explanation

Warning. The specified *creator.tablename* could not be found in the catalog table SYSIBM.SYSTABLES.

User response

Check the setting and spelling of the COLLECT parameter for fixed field (A8.A18), check the SUBSYST parameter for proper subsystem ID, and run again.

ANL8010I DB2 SUBSYSTEM NAME USED IS ssid.

Explanation

Informational. You have specified the target DB2 subsystem from which ANLCAT collects its catalog statistics.

User response

None

ANL8011W NO ENTRY FOUND FOR TABLESPACE dbname.tsname.

Explanation

Warning. The specified database name (*dbname*) and table space name (*tsname*) could not be found in the

catalog table SYSIBM.SYSTABLESPACE. These names were derived from the entry in SYSIBM.SYSTABLES.

User response

Check and correct the catalog, and run again.

ANL8012I NON-UNIFORM DISTRIBUTION STATS ARE REQUESTED.

Explanation

Informational. You have requested non-uniform column distribution statistics from SYSIBM.SYSCOLDIST to be collected for columns where they exist. This also collects concatenated index key column statistics, if they exist.

User response

None

Explanation

Informational. There were no indexes found in SYSIBM.SYSINDEXES for the table named *creator.tablename*. Processing continues.

User response

As appropriate.

ANL8014I NO COLUMN STATS FOR TABLE creator.tablename.

Explanation

Informational. There were no column entries found in SYSIBM.SYSCOLUMNS for the table named *creator.tablename*. Processing continues.

User response

As appropriate.

```
ANL8015I INDIVIDUAL COLUMN LEVEL
STATISTICS ARE REQUESTED.
```

Explanation

Informational. You have requested column level statistics from SYSIBM.SYSCOLUMNS, including COLCARDF, HIGH2KEY, and LOW2KEY.

User response

None

ANL8016I NO COLUN

NO COLUMN DIST STATS FOR creator.tablename.

Explanation

Informational. There were no non-uniform column distribution statistics found in SYSIBM.SYSCOLDIST for the table named *creator.tablename*. Processing continues.

User response

As appropriate.

ANL8017I	NO PART COLDIST STATS FOR
	creator.tablename.

Explanation

Informational. There were no partition-level nonuniform column distribution statistics found in SYSIBM.SYSCOLDISTSTATS for the partitioned table *creator.tablename*. Processing continues.

User response

As appropriate.

ANL8018I NO PART COLUMN STATS FOR creator.tablename.

Explanation

Informational. There were no partition-level column statistics (Highkey, High2key, and so on) found in SYSIBM.SYSCOLSTATS for the partitioned table *creator.tablename*. Processing continues

User response

As appropriate.

ANL8019I NO PART TABLE STATS FOR creator.tablename.

Explanation

Informational. There are no statistics in the SYSIBM.SYSTABSTATS table relating to the partitioned table identified by *creator.tablename*.

User response

As appropriate.

ANL8020I NO PART INDEX STATS FOR creator.tablename.

Explanation

Informational. There are no statistics in the SYSIBM.SYSINDEXSTATS table relating to the partitioned table identified by *creator.tablename*.

User response

As appropriate.

ANL8021E SQLCODE sqlcode UNKNOWN TO DSNTIAR ROUTINE, DB2 HAS NO EXPLAINABLE MESSAGE FOR THIS ERROR.

Explanation

Error. An SQLCODE was returned during processing that is not processed by DB2's DSNTIAR error message routine.

User response

Check the *sqlcode* in Message and Codes and take appropriate action.

ANL8022I	GLOBAL TARGET CREATOR IS
	creator.

Explanation

Informational. SQL PA acknowledges that you selected a different creator name to be applied to all output generated by the statistics migration program, rather than propagate the same creator/owner that was defined on the input system.

User response

None

ANL8023I GLOBAL TARGET DATABASE NAME IS database.

Explanation

Informational. SQL PA acknowledges that you selected a different database name to be applied to all output generated by the statistics migration program, rather than propagate the same database name as was defined on the input system.

User response

None

ANL8024I COLLE TABLE

COLLECTING STATS FOR LONG TABLE authid tabname.

Explanation

Informational. SQL PA acknowledges that you defined a long table name (in Version 8 the names of the creator and table can be up to 128 chars each), specified by separate authorization ID and tabname parameters on the COLLATH and COLLTAB parameters.

User response

None

ANL8025I

"FROM" DB2 RELEASE LEVEL IS version.

Explanation

Informational message identifies the DB2 level of the source machine.

User response

None

Db2 SQL Performance Analyzer ISPF panel messages

The following messages are issued under TSO by the SQL PA CLIST and ISPF panel processes:

ANLOO1I For statements that contain SQL errors, the explain reports and trace reports do not exist.

Explanation:

Explain and trace reports do not exist for statements that contain SQL errors.

User response:

No action is required. If you need the explain and trace reports, you must fix any statements that contain SQL errors.

ANL002E Required values are missing.

Explanation:

Some fields are missing required values. The cursor will display in the fields that do not have values.

User response:

Specify values for the required fields.

ANL003I The selected feature is unavailable when the input source is a plan, or when ANLCSPA is used, or when the SQL Display or the Tables Display is used.

Explanation

The selected feature cannot be used if any of the following occurred:

- A plan was the input source for the explain command.
- ANLCSPA was run from a browse or edit session.
- The SQL or Tables Display screens were used to issue the explain command.

User response:

No action required.

ANL005E The package does not exist in the SYSIBM.SYSPACKAGE catalog table.

Explanation:

The specified package does not exist in the catalog table.

User response:

No action required. To continue, specify a different package.

ANLOO6E The specified value is invalid.

Explanation:

An invalid value was specified.

User response:

Ensure that a valid value is specified.

ANLOO7E	The source module was not found.

Explanation:

The source module is missing.

User response:

Ensure that the ANLCSRCE source module exists.

```
ANLOO8E SQL statements for the source module could not be extracted.
```

Explanation:

User response:

Ensure that the ANLCMACR macro was specified on the command line during the ISPF edit session.

ANL009E	The specified data set is
	unavailable.

Explanation:

The specified data set is not available for use.

User response:

Ensure that the data set is available, or specify a different data set.

ANLO10E The data set that contains the specified member is not partitioned.

Explanation:

When a member was being created, a PDS with a member name was specified.

Either change the data set to a PDS or remove the member name from the data set and use as a sequential data set. If it doesn't exist, it will be created.

ANL011E	The installed base parameter file
	was not found.

Explanation:

While a new base parameter file was being created, the file that was created by Tools Customizer could not be found.

User response:

Restore the missing parameter file.

ANL012E The specified primary command is invalid

Explanation:

The specified primary command is not valid. Valid primary commands are displayed on the ISPF panel.

User response:

Specify a valid primary command. You can type the command in the command line, or you can put the cursor on the command and press Enter.

ANL015E Wildcards are not supported. Remove all specified wildcards and try again.

Explanation:

Wildcards, such as % and *, are not supported.

User response:

Remove any wildcard characters.

ANL017E The specified value contains too many characters.

Explanation:

The specified data set name was too long. Data set names can contain up to 44 characters, and each qualifier cannot be more than eight characters.

User response:

Specify a correct name for the DBRM member.

ANL018E The specified value for the degree of parallelism is invalid. Specify 1 or ANY.

Explanation:

An invalid value for the degree of parallelism was specified. Valid values are 1 and ANY.

User response:

Specify either 1 or ANY.

ANL019E An alphabetic character value is specified in a field that can contain only a numeric value. Specify a numeric value.

Explanation:

In the QUERYNO field, an alphabetic character instead of a number was specified.

User response:

Specify a numeric value.

ANL020E	The name of the table creator is
	required. Specify the name of the
	table creator.

Explanation:

A table name was specified, but the table creator was not specified for the package or plan.

User response:

Specify a table creator.

NL022E	The specified table does not exist
	in the package.

Explanation:

The package does not contain the specified table.

User response:

Specify an existing table.

ANL024E	The specified QUERYNO does exist
	in the package.

Explanation:

The package does not contain the specified QUERYNO.

User response:

Specify an existing QUERYNO.

ANLO26E	The specified value of FORCE is
	invalid.

Explanation:

An incorrect value was specified for FORCE. Valid values are Yes and No.

User response:

To explain all SQL statements in all packages that conform to the specification, specify Yes. Otherwise, specify No or leave the field blank.

ANLO27E The specified pattern was not found in the name of any member.

Explanation:

The DBRM member was not found.

User response:

Specify a valid DBRM member name.

ANL028E The data set could not be opened.

Explanation:

The data set that contains the statements to be explained could not be opened.

User response:

Ensure that you have the correct authority to use the data set.

ANL029E The specified data set does not exist or is not cataloged.

Explanation:

A data set that does not exist or is not cataloged was specified.

User response:

Ensure that the specified data set exists and is cataloged.

ANL030E	A member name was not specified
	for the PDS.

Explanation:

A PDS and a member name must be specified together.

User response:

Ensure that you specify the member with the PDS.

ANL031E	The format of the specified data
	set is invalid.

Explanation:

The data set name was specified in an invalid format.

User response

Ensure that the specified data set name adheres to the following format:

- · Contains no more than 44 characters
- Each qualifier contains no more than 8 characters
- Each qualifier is separated by a period (.)

ANL032E The specified qualifier is invalid.

Explanation:

The data set name was specified in an invalid format.

User response:

Ensure that only one qualifier is specified in the field.

ANL033E

found in the PLAN_TABLE.

The specified QUERYNO was not

Explanation:

The specified QUERYNO does not exist in the PLAN_TABLE.

User response:

Ensure that the specified information about the PLAN_TABLE is correct.

ANL034E The QMF query was not found in the OBJECT_DATA table.

Explanation:

The OBJECT_DATA table does not contain the specified QMF query.

User response:

Change the query name.

ANL036E The PLAN_TABLE was not found in the DB2 system catalog for the

specified authorization ID.

Explanation: The PLAN_TABLE could not be found.

User response:

Ensure that the specified information for the PLAN_TABLE is correct.

ANL037E The table for the specified name and creator was not found in the DB2 system catalog.

Explanation:

An incorrect table name was specified.

User response:

Ensure that the correct table name is specified.

ANLO38E	The index for the specified name
	and creator was not found in the
	DB2 system catalog.

Explanation:

An incorrect index name was specified.

User response:

Ensure that the correct table name is specified.

ANL039E	The specified member contains no
	records.

Explanation:

The member is empty.

User response:

Ensure that the member contains records, or specify a different member name.

ANL040E The specified member or data set was not found.

Explanation:

The member or data set could not be found.

User response:

Specify a valid member or data set.

ANL041E The specified index exists in the system catalog.

Explanation:

The index being created exists in the system catalog.

User response:

Specify a different index.

ANL042I The maximum number of SQL statements that can be explained was reached.

Explanation:

The maximum number of statements that can be explained is 300.

No action is required. To explain more then 300 statements, specify FORCE=Y.

ANL043E The specified DB2 run library was not found.

Explanation:

An incorrect DB2 run library was specified.

User response:

Specify a valid DB2 run library.

ANL044E The specified CEE run library was not found.

Explanation:

An incorrect CEE run library was specified.

User response:

Specify a valid CEE run library.

ANL045E The specified SQL PA load library was not found.

Explanation:

An incorrect SQL PA load library was specified.

User response:

Specify a valid SQL PA load library.

ANLO46E The specified DB2 load library was not found.

Explanation:

An incorrect DB2 load library was specified.

User response:

Specify a valid DB2 load library.

ANL047E The specified DB2 exit library was not found.

Explanation:

An incorrect DB2 exit library was specified.

User response:

Specify a valid DB2 exit library.

ANL048E The specified table space or database was not found in the DB2 system catalog.

Explanation:

An incorrect table space or database was specified.

User response:

Specify a valid table space or database.

ANL049E	The user <i>user_ID</i> is not authorized
	to set the CURRENT SQLID value
	to ID.

Explanation:

The specified user does not have the required authority to set the CURRENT SQLID to the specified value.

User response:

Ensure that the specified user has the required authority.

ANL050E ANLCHECK was not bound.

Explanation:

The SQL ID was set, but ANLCHECK was not bound.

User response:

Submit &ANLSSID.KCHE to bind the ANLCHECK package.

ANL051I If the value of SQLID is blank, it will be set to +OFF+ after you exit.

User response:

No action is required.

ANL052E	The specified line command is
	invalid.

Explanation:

The specified line command cannot be used on the panel.

User response:

Specify a valid line command.

ANL053E In DB2 10 and later releases, all plans must be bound by using the PKLIST option.

Explanation:

The plan was bound by an option that is not supported in DB2 10 and later releases.

User response:

To process plans that were bound from a package, use the option to process packages from the DB2 system catalog.

ANL054E No explainable statements were found in the selected input.

Explanation:

The output contains no explainable statements.

User response:

Ensure that the package contains statements that can be explained.

ANL055W	The ANLPARM user parameter file
	is undefined. Default values will
	be used for processing, and
	changes will not be permanently
	saved.

User response:

No action is required.

ANL056W	The ANLCNTL system parameter
	file is undefined. Default values
	will be used for processing, and
	changes will not be permanently
	saved.

No action is required.

ANL059W The specified database and table space information cannot be obtained from the SQL_ID.EEEWORK table. The default database and table space will be used.

User response:

If the default database and table space exist, this message is a warning, and no action is required. Otherwise, restore the table and ensure that the required permissions are granted.

ANLO60E A connection to the remote DB2 subsystem was not made. Ensure that SYSIBM.LOCATIONS contains the *location_name* location.

Explanation:

The DB2 subsystem might not be running, or the subsystem might be set up incorrectly.

User response:

Ensure that the DB2 subsystem is running and that setup information for the remote subsystem is correct.

ANL061E Information from the QMF tables could not be retrieved.

Explanation:

QMF queries were not found because the specified owner was incorrect.

User response:

Ensure that the correct owner is specified, and retrieve the QMF queries again.

ANL062E	The QMF object type must be
	QUERY.

Explanation:

An incorrect QMF object type was specified.

User response:

Specify QUERY for the QMF object type.

ANL063E The QMF query subtype must be SQL.

Explanation:

An incorrect QMF query subtype was specified.

User response:

Specify SQL for the QMF query subtype.

ANL064E The QMF query was not saved with SHARE=YES and is restricted to the owner.

Explanation:

The user does not have permission to read the QMF query.

User response:

Specify SHARE=YES, and ensure that the user has permission to read the QMF query.

ANL065E The value specified for primary space allocation is not valid.

Explanation:

An invalid value for PRQTY was specified. The value must be a nonzero integer that is greater than or equal to -1.

User response:

Specify a nonzero integer that is greater than or equal to -1 for the PRQTY parameter.

ANLO66E The value specified for secondary space allocation is not valid.

Explanation:

An invalid value for SEQTY was specified. The value must be an integer that is greater than or equal to -1.

User response:

Specify an integer that is greater than or equal to -1 for the SEQTY parameter.

ANL067I VIEW is disabled in the ISPF Configuration Table so BROWSE is used instead.

User response:

No action is required. However, consider enabling VIEW so that the correct position can be located for you.

ANL068I VIEW is disabled in the ISPF Configuration Table so BROWSE is used instead. Query *query_number* must be found manually.

User response:

No action is required. However, consider enabling VIEW so that the specified query number can be located automatically.

```
ANL069E During an attempt to view or
browse the data_set_name report,
a return code of
return_code_number was issued.
```

Explanation:

The report could not be viewed or browsed.

User response:

Check the return code to determine the program or procedure that issued the message.

ANL070W	The user user_ID is not authorized
	to save to <i>parameter_file</i> . Changes
	will be effective for only the
	current session.

Explanation:

The user ID does not have the required authority to save changes to the parameter file.

User response:

Ensure that the user has authority to make changes in the parameter file.

ANL071E Reports are unavailable because either no previous reports exist or an error occurred when the reports were generated.

Explanation:

If Db2 SQL Performance Analyzer is being run for the first time, this message is informational. Otherwise, reports were not generated because of an error that occurred while the reports were run.

User response:

Ensure that the reports exist. If they do not exist, rerun the reports and try to view them again.

ANL072E The *SQL_ID*.DSN_VIRTUAL_INDEXES DB2 table could not be created.

Explanation:

The specified table was not created because the user does not have the authority to create tables.

User response:

Ensure that the authority to create tables is granted to the user.

ANL073E An error occurred while the data_set_name data set was being created.

Explanation:

The user parameter file could not be created because the user does not have the authority to create the file.

User response:

Ensure that the authority to create the file is granted to the user.

ANL074E	An error occurred while the
	parameter values were being
	copied to the <i>data_set_name</i> data
	set.

Explanation:

The parameter values could not be copied to the member in the data set because the user does not have the authority to write to the member.

User response:

Ensure that the authority to write to the member is granted to the user.

ANL075E The specified virtual index already exists in the DB2 system catalog.

Explanation:

The specified virtual index exists.

User response:

Specify a different name for a virtual index.

ANL076E The *index_name* virtual index created by *creator_name* was not found in the DB2 system catalog.

Explanation:

The specified virtual index could not be found.

User response:

Verify that the name of the virtual index and the name of the creator are correct.

ANL077I The specified index exists in the DB2 system catalog and was already dropped virtually.

User response:

No action is required.

ANL078E	The specified index is a virtual
	index in the DB2 system catalog.

Explanation:

The specified index is being used as a virtual index so it cannot be used as a real index.

User response:

Specify a different index.

ANL079E	The specified column name
	<i>column_name</i> is not in the
	specified table.

Explanation:

The specified column name is invalid.

User response:

Specify a different column name.

ANL080E	The value must not contain
	embedded blanks.

Explanation:

The column name contains blanks.

User response:

Specify a column name without blanks.

LE L	ser user_ID needs the privilege
f	or SELECT from
9	YSIBM.SYSINDEXES and
ę	YSIBM.SYSCOLUMNS, CREATE
I	NDEX, and CREATE TABLE
c	wner_name.DSN_VIRTUAL_INDE
)	ES, if DSN_VIRTUAL_INDEXES
t	able is not created at SQL PA
c	ustomization.
c) t	wner_name.DSN_VIRTUAL_IN ES, if DSN_VIRTUAL_INDEXEs able is not created at SQL PA ustomization.

Explanation:

If the DSN_VIRTUAL_INDEXES table was not created when Db2 SQL Performance Analyzer was customized, the user needs SELECT privilege to access or create tables.

Ensure that the user has the required privilege.

ANL082E NO cannot be specified for the INCLUDE NULL KEYS parameter when the *key_column* key column does not allow null values.

User response:

Specify YES for the INCLUDE NULL KEYS parameter.

ANL083E NO cannot be specified for the INCLUDE NULL KEYS parameter when YES is specified for the PARTITIONED parameter.

User response:

Specify YES for the INCLUDE NULL KEYS parameter.

ANL084E NO cannot be specified for the INCLUDE NULL KEYS parameter when YES is specified for the UNIQUE parameter.

User response:

Specify YES for the INCLUDE NULL KEYS parameter.

ANL086W The user-level user parameter data set is unavailable or does not exist. The job cards cannot be saved.

Explanation:

The job cards were not saved because the specified data set is not available or has not been created.

User response:

Ensure that the user-level user parameter file exists and is available for updates.

User response:

No action is required.

ANL088I The user-level user parameter data set must be created before the job cards can be saved. To create the data set with the displayed name and settings, press Enter. Otherwise, change the name, settings, or both, and press Enter.

Explanation:

The user-level user parameter data set must exist. The data set can be created based on the default settings, or the data set name and settings can be changed.

User response:

To create the data set with the displayed name and settings, press Enter. Otherwise, change the name, settings, or both, and press Enter.

ANL089I

The user-level user parameter data set was created. Press Enter to save the job cards, or press PF3 to cancel the changes.

Explanation:

The data set was created, and the job cards can be saved now.

User response:

To save the job cards, press Enter. To cancel the changes, press PF3.

ANL091E	NO GENERIC IDs ARE AVAILABLE
	IN userid.REGISTRY. EITHER
	CHANGE THE CURRENT SQL ID TO
	A VALUE OTHER THAN +OFF+ OR
	MODIFY THE REGISTRY TABLE TO
	FREE THE ENTRIES THAT ARE NO
	LONGER BEING USED BY
	CHANGING ANLINUSE FROM 'Y'
	TO 'N'.

Explanation

All generic IDs in the REGISTRY are in use, as indicated by the column ANLINUSE being set to 'Y'.

User response

Either change the current SQL ID to a value other than +OFF+ to continue processing with using generic IDs or change ANLINUSE to 'N' for those IDs in the REGISTRY that are no longer being used.

ANL092E	AN ERROR OCCURRED WHEN
	ACCESSING userid.REGISTRY
	WHILE SELECTING A GENERIC ID
	FOR USE. VERIFY THAT THE
	REGISTRY WAS SET UP AND IS
	AVAILABLE.

Explanation

An error occurred when accessing the REGISTRY while selecting a generic ID for use.

User response

Verify that the REGISTRY was set up correctly and is available.

ANL112E	The DB2 SQL authorization ID can
	only be changed when SETPLAN is
	set to ALLOW or YES. Contact the
	administrator regarding the value
	of SETPLAN.

Explanation

The DB2 SQL authorization ID can only be changed when SETPLAN is set to ALLOW. Prior to Version 5.1, YES was an accepted value for SETPLAN and will be still be processed.

User response

Verify the value of SETPLAN. It is the system parameter in the member *ssid*CNTL of *hiqual*.SANLPARM. Contact the administrator regarding the value of SETPLAN.

ANL113W The user *username* is not authorized to change *data set name*.

Explanation:

You are attempting to edit the system parameters without the authority to modify the required data set. The system parameters will be displayed in browse mode and cannot be modified.

User response

No action is required.

ANL114E The user *username* is not authorized to change the value of DATAGRP. The value of DATAGRP is unchanged.

Explanation:

You are attempting to change the system parameter, DATAGRP, without the authority to modify the required DB2 table, EEEAUTH. The value of DATAGRP will not be changed.

User response

No action is required.

ANL115E The user *username* is not authorized to change the value of system parameter DATAGRP in the EEEAUTH table. The value of DATAGRP is unchanged.

Explanation:

The value of the system parameter, DATAGRP, does not match the current value of DATAGRP in the EEEAUTH table. You do not have the authority to modify EEEAUTH. The value of DATAGRP will not be changed..

User response

No action is required.

ANL116I

The user *username* is not authorized to see user table data.

The statistics collected will be incomplete since it will not contain user table data such as HIGH2KEY and LOW2KEY.

Explanation:

You are not a member of the user group specified by the system parameter, DATAGRP, so you do not have the authority to see user table data. User table data will not be included in the statistics that are collected.

User response

No action is required.

ANL134E	userid is a generic ID and cannot
	be used as the SQL authorization
	ID. To use a generic ID, set the
	New DB2 SQL authorization ID to
	+0FF+.

Explanation:

An attempt was made to set the "New DB2 SQL authorization ID" to a generic ID. A specific generic ID cannot be used as the DB2 SQL authorization ID.

User response:

To use a generic ID, set the "New DB2 SQL authorization ID" to +0FF+.

ANL135E	If a wild card character is used in a
	Collection ID, then both Collection
	IDs must be identical.

Explanation:

The specified Collection IDs must be identical if wild card characters are used.

User response:

Change the Collection IDs to use the same value or remove all wild card characters and try again.

ANL136I	At least one table was not found in the catalog. To view the containing statements, run the E line command on the "found>" row and view the SQL report from the resulting Query
	Limits report.

Explanation:

When using the TABLES command, at least one table was not verified during processing.

User response:

To view the containing statements, explain the"<table not found>" row and then view the SQL for the statements.

ANL137I	No tables were found in the
	selected input.

Explanation:

When using the TABLES command, no tables were found.

SUBSYSTEM ID, THEN RETRY. RC: retcod

User response:

No action is required.

ANL9012 THE DATA SET OR MEMBER TO BE RESTORED DOES NOT EXIST. CHECK YOUR SPELLING FOR THIS MEMBER NAME AND THEN RETRY. SQL PA CANNOT LOCATE: prefix.ANL510.SANLSTAT (member)

Explanation

The data set member requested to supply parameters for the RESTORE function cannot be found. Under each TSO user ID, a repository of statistics is built and maintained in *prefix*.ANL510.SANLSTAT for "What If?" and Statistics Migration online.

User response

Check the spelling to verify that you specified the member name correctly. Issue the following TSO command to verify the members:

LISTD 'prefix.ANL510.SANLSTAT' ME

ANL9013 SQL PA ENCOUNTERED PROBLEMS RESTORING CATALOG STATISTICS FROM prefix.ANL510.SANLSTAT (member), RC: retcode CHECK RETURN CODE, AND CONSIDER MANUAL STATS RESTORATION

Explanation

The What If? Analysis (ANLWIF) program was unable to use the member that was requested for the RESTORE function or it had problems restoring the statistics.

User response

Run the TSO online interface from ISPF 6, and specify ANLI QDEBUG(YES). All error messages from ANLWIF are routed to SYSPRINT (screen). Check the SQL CODE for type of error reported and correct the problem. If the error is critical, restore the statistics manually under DB2I SPUFI.

ANL9014 SQL PA ENCOUNTERED PROBLEMS RETRIEVING CATALOG STATISTICS FOR DISPLAY. CHECK SPELLING OF YOUR CREATOR AND TABLE NAME AND ALSO THE

Explanation

The What If? Analysis (ANLWIF) program encountered problems while trying to access a table's statistics.

User response

Run the TSO online interface from ISPF 6, and specify ANLI QDEBUG(YES). All error messages from ANLWIF are routed to SYSPRINT (screen). Check the SQL CODE for the type of error reported and correct the problem.

```
ANL9015 SQL PA ENCOUNTERED PROBLEMS
UPDATING CATALOG STATISTICS
FOR THIS TABLE. CONTACT
SYSTEM SUPPORT. RC: retcode
SQL PA WAS UNABLE TO UPDATE
THE CATALOG FOR THIS TABLE.
```

Explanation

The What If? Analysis (ANLWIF) program encountered problems while trying to modify catalog statistics for a table.

User response

Run the TSO online interface from ISPF 6, and specify ANLI QDEBUG(YES). All error messages from ANLWIF are routed to SYSPRINT (screen). Check the SQL CODE for the type of error reported and correct the problem.

ANL9016 SQL PA ENCOUNTERED PROBLEMS RETRIEVING CATALOG STATISTICS FOR DISPLAY. CHECK SPELLING OF YOUR CREATOR AND INDEX NAME AND ALSO THE SUBSYSTEM ID, THEN RETRY. RC: retcode

Explanation

The What If? Analysis (ANLWIF) program encountered problems while trying to access an index's statistics.

User response

Run the TSO online interface from ISPF 6, and specify ANLI QDEBUG(YES). All error messages from ANLWIF are routed to SYSPRINT (screen). Check the SQL CODE for the type of error reported and correct the problem.

ANL9017 SQL PA ENCOUNTERED PROBLEMS UPDATING CATALOG STATISTICS FOR THIS INDEX. CONTACT SYSTEM SUPPORT. RC: retcode

SQL PA WAS UNABLE TO UPDATE THE CATALOG FOR THIS INDEX.

Explanation

The What If? Analysis (ANLWIF) program encountered problems while trying to modify catalog statistics for an index.

User response

Run the TSO online interface from ISPF 6, and specify ANLI QDEBUG(YES). All error messages from ANLWIF are routed to SYSPRINT (screen). Check the SQL CODE for the type of error reported and correct the problem. Possible SQL CODE is -545, a cluster ratio not between 0 and 1.

ANL9020 THE DATA SET OR MEMBER TO APPLY THE STATS DOES NOT EXIST CHECK YOUR SPELLING FOR THIS MEMBER NAME AND THEN RETRY. SQL PA CANNOT LOCATE: prefix.ANL510.SANLSTAT (member)

Explanation

The requested data set member cannot be found. A repository of statistics is built and maintained for online statistics migration under each TSO user ID. This member must contain the catalog changes to be applied.

User response

Check the spelling to verify that you specified the member name correctly. Issue the following TSO command to verify the members:

LISTD 'prefix.ANL510.SANLSTAT' ME

ANL9021 CANNOT EDIT THE JCL TO APPLY STATISTICS TO THE CATALOG. DATA SET OR MEMBER MAY NOT EXIST, OR COULD BE ALREADY IN USE OR EDIT BY ANOTHER: prefix.ANL510.SANLSTAT (JOBCARD)

Explanation

The JOBCARD data set member was requested but cannot be found or accessed. The JOBCARD member is used to dynamically build JCL for Collect and Apply statistics.

User response

Go to the Statistics Migration panel and select Default JCL to re-establish the JOBCARD member. Issue the following TSO command to verify the members:

LISTD 'prefix.ANL510.SANLSTAT' ME

ANL9022 CANNOT EDIT THE MEMBER CONTAINING COLLECTION PARAMETERS. DATA SET OR MEMBER MAY NOT EXIST, OR COULD BE ALREADY IN USE OR EDIT BY ANOTHER: prefix.ANL510.SANLSTAT (member)

Explanation

The requested data set member cannot be found. A repository of statistics is built and maintained for online statistics migration under each TSO user ID. This member contains collection statistics that were newly created for statistics migration.

User response

Check the spelling to verify that you specified the member name correctly. Issue the following TSO command to verify the members:

LISTD 'prefix.ANL510.SANLSTAT' ME

```
ANL9023
CANNOT EDIT THE JCL TO
COLLECT THE CATALOG
STATISTICS. DATA SET OR
MEMBER MAY NOT EXIST, OR
COULD BE ALREADY IN USE OR
EDIT BY ANOTHER:
prefix.ANL510.SANLSTAT
(JOBCARD)
```

Explanation

The JOBCARD data set member was requested but cannot be found or accessed. The JOBCARD member is used to dynamically build JCL for Collect and Apply statistics.

User response

Go to the Statistics Migration panel and select Default JCL to re-establish the member JOBCARD. Issue the following TSO command to verify the members:

LISTD 'prefix.ANL510.SANLSTAT' ME

Chapter 12. Messages and return codes

These topics describe the error messages and return codes that are issued by Db2 SQL Performance Analyzer.

Tools Customizer messages

These topics describe the error messages and return codes that are issued by Tools Customizer. Use the information in these messages to help you diagnose and solve Tools Customizer problems.

Tools Customizer messages

halp you diagnage and calve Teals Customizer proble

Use the in	formation in these messages to help yo	u diagnose and solv	<i>v</i> e Tools Customizer problems.
CCQB000I	The product parameter data was saved in the data store.	Processing stops.	
		User response: Select a step in the specified task or deselect the task.	
Explanation:			
were saved in the	re made to the product parameters e data store.	CCQB004I	The required information to run the Discover EXEC was saved in
System action:			the data store.
		Explanation:	ntains all the information that is
No action is requ	ired.	required to run the Discover EXEC.	
CCQB001I	The DB2 parameter data was saved in the data store.	System action: None.	
Explanation: Changes that were made to the Db2 parameters were		User response: No action is required.	
saved in the data	i store.	CCQB005E	The conflicting values for the
System action: None.			parameter_name parameter must be resolved before the information
User response:		Frankanstinna	can be saveu.
No action is required.		- Two values for one parameter conflict with each other.	
CCQB002I	The LPAR parameter data was saved in the data store.	and they must be	resolved to save the information.
Explanation: Changes that we	re made to the LPAR parameters were	System action: Processing stops.	
saved in the data store.		User response: Resolve the confl	icting values for the parameter.
System action: None.		CCQB006E	One row must be selected.
User response: No action is requ	ired.	Explanation: One row in the tal	ble must be selected.
CCQB003E	At least one step must be selected in a selected task. The selected	System action: Processing stops.	
task is <i>task_description</i> . Explanation: When a task is selected, at least one step must be		User response: Select one row.	
		CCQB007E	Only one row can be selected.
task. System action:		Explanation: Multiple rows in t	he table are selected, but only one be selected

System action:

Processing stops.

User response:

Select only one row.

CCQC000I The jobs have been customized on the selected DB2 entries.

Explanation:

The jobs were customized on the Db2 entries that were selected.

System action:

None.

User response:

Press Enter to clear the message.

CCQC001W The jobs were not generated on one or more of the selected DB2 entries. Press PF3 to check the Db2 entries that were not customized.

Explanation:

The product was not customized on one or more of the Db2 entries that were selected.

System action:

None.

User response:

Press PF3 to see the Db2 entries on which the product was not customized. The status of these Db2 entries is Errors in Customization.

CCQC002I The edit session was started automatically because values for required parameters are missing or must be verified.

Explanation:

If product, LPAR parameters, or Db2 parameters are not defined or if parameter definitions must be verified, an editing session for the undefined or unverified parameters starts automatically.

System action:

None.

User response:

Define values for all required product, LPAR parameters, or Db2 parameters.

CCQC003W The *template_name* template in the *library_name* metadata library does not contain any parameters.

Explanation:

The specified template does not have parameters.

System action:

None.

User response:

No action is required.

CCQC004S The value of the "type" attribute for the template_name template in the library_name metadata library does not match the value that was previously specified. The value is value_name, and the previously specified value is value_name.

Explanation:

The value of the "type" attribute must match the value that was previously specified.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQC005S	The <i>template_name</i> template
	exceeds the number of allowed
	templates for a customization
	sequence. The template is in the
	<i>library_name</i> metadata library.

Explanation:

The customization sequence can process only *number* templates. The specified template cannot be processed because the customization sequence already contains the maximum number of templates.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQC006E The jobs could not be generated for the *group_attach_name* Db2 group attach name.

Explanation:

The customization jobs could not be generated for the specified Db2 group attach name.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC007E The jobs could not be generated for the *subsystem_ID* Db2 subsystem.

Explanation:

The customization jobs could not be generated for the specified Db2 subsystem.

System action:

Processing stops.

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC008E The jobs could not be generated for the *member_name* Db2 member.

Explanation:

The customization jobs could not be generated for the specified Db2 member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC009S	The jobs were not generated for
	the DB2 entries.

Explanation:

One or more errors occurred while customization jobs were being generated for the selected Db2 entries.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC010S The *template_name* template could not be accessed in the *library_name* metadata library.

Explanation:

The specified template could not be accessed because the user does not have RACF access to the data set, the data set has incorrect data characteristics, or the data set is not cataloged.

System action:

Processing stops.

User response:

Ensure that you have RACF access to the data set, that the characteristics are correct according to the specifications of the product that you are customizing, and that the data set is cataloged. If the problem persists, contact IBM Software Support.

CCQC011S The template_name template could not be written to the *library_name* customization library.

Explanation:

The specified template could not be accessed because the user does not have RACF access to the data set, the data set has incorrect data characteristics, or the data set is not cataloged.

System action:

Processing stops.

User response:

Ensure that you have RACF access to the data set, that the characteristics are correct according to the specifications of the product that you are customizing, and that the data set is cataloged. If the problem persists, contact IBM Software Support.

CCQC012W The job card was generated with default values because the JOB keyword was missing.

Explanation:

Default values were used to generate the job card because the JOB keyword was not specified in the first line of the job card.

System action:

The job card was generated with default values.

User response:

No action is required. To generate the job card with your own values, add the JOB keyword in the first line of the job card.

CCQC013W The job card was generated with the default value for the programmer name because the specified programmer name exceeded 20 characters.

Explanation:

Default values were used to generate the job card because the specified programmer name contained too many characters.

System action:

The job card was generated with default values.

User response:

No action is required. To generate the job card with your own values, add a valid programmer name in the job card. A valid programmer name is 1 - 20 characters.

CCQC014W	The job card was generated with default values because the JOB keyword was not followed by a
	space.

Explanation:

Default values were used to generate the job card because a space did not follow the JOB keyword.

System action:

The job card was generated with default values.

User response:

No action is required. To generate the job card with your own values, add a space after the JOB keyword in the job card.

CCQC015S	The <i>template_name</i> template in
	the library_name metadata library

contains the following filetailoring control statement: *statement_name*. This control statement is not valid in a *template_type* template.

Explanation:

The *template_type* template cannot contain the specified type of file-tailoring control statement.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC016S The)DOT file-tailoring control statement exceeded the number of allowed occurrences for the *template_name* template in the *library_name* metadata library.

Explanation:

The)DOT file-tailoring control statement can occur only a limited number of times in the specified template.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQC017S The nested)DOT file-tailoring control statements exceeded the number of allowed occurrences in the *template_name* template in the *library_name* metadata library.

Explanation:

Nested)DOT file-tailoring control statements can occur only *number* times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC018S The *template_name* template in the *library_name* metadata library is not valid because it does not contain any data.

Explanation:

The specified template is missing required data.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC019S The *template_name* template in the *library_name* metadata library is not valid because an)ENDDOT file-tailoring control statement is missing.

Explanation:

A)ENDDOT file-tailoring control statement is required in the specified template.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC021S	The <i>template_name</i> template in
	the library_name metadata library
	is not valid because the template
	must start with the
	<i>parameter_name</i> job card
	parameter.

Explanation:

The specified template must start with the specified job card parameter.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQC022SThe parameters used in a)DOT
file-tailoring control statement
exceeded the number of allowed
parameters in the template_name
template. The template is in the
library_name metadata library.
The error occurs in)DOT section
section_number.

Explanation:

A)DOT file-tailoring control statement can contain only a limited number of parameters.

System action:

Processing stops.

User response:

CCQC023S	The)DOT file-tailoring control
	statement must include the table-
	<i>name</i> table name in the
	<i>template_name</i> template. The
	template is in the <i>library_name</i>

metadata library. The error occurs in)DOT section *section_number*.

Explanation:

The)DOT file-tailoring control statement is missing a required table name.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC024S ISPF file tailoring failed for the template_name template in the library_name metadata library.

Explanation:

An error occurred during ISPF file tailoring for the specified template.

System action:

Processing stops.

User response

Review the Tools Customizer-generated trace data set and the ISPF file tailoring trace data set. To create an ISPF file tailoring trace data set, complete the following steps:

- 1. Run Tools Customizer until the error is about to occur.
- 2. Specify the ISPFTTRC command, and press Enter.
- 3. Issue the Tools Customizer command that issues the error.
- 4. Specify the ISPFTTRC command, and press Enter. The ISPF file tailoring trace data set is created. It adheres the following naming convention: *TSO_ID.*ISPFT.TRACE, where *TSO_ID* is the TSO user ID that is being used.

If the problem persists, gather the following information and contact IBM Software Support.

- A screen capture of the Tools Customizer error. Ensure that the complete error message is displayed by pressing PF1.
- The Tools Customizer trace data set. It adheres to the following naming convention: *TSO_ID*.CCQ.TRACE, where *TSO_ID* is the TSO user ID that is running Tools Customizer.
- The ISPF file tailoring trace data set.

CCQC025I Customized jobs do not exist because they have not been generated.

Explanation:

The list of customized jobs cannot be displayed because the product has not been customized for any Db2 entries.

System action:

None.

User response:

Complete the steps to customize a product. Customized jobs are generated when all required product, LPAR parameters, and Db2 parameters are defined and at least one Db2 entry on which to customize the product has been selected.

CCQC026S The value of the "customized" attribute for the *parameter_name* parameter in the *library_name* metadata library template does not match the value that was previously specified. The value is *value_name*, and the previously specified value is *value_name*.

Explanation:

The value for the "customized" attribute for a parameter must match the value that was previously specified.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC027S	The <i>job_name</i> customization job
	was not found in the <i>library_name</i>
	customization library.

Explanation:

The selected customization job does not exist in the customization library.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQC028S The *library_name* customization library was not found.

Explanation:

The customization library does not exist.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQC029I The customization jobs were generated for *Product_name*.

Explanation:

The customization jobs were generated for the specific product.

System action:

None.

User response:

No action is required.

CCQC030S The customization jobs cannot be generated because at least one DB2 entry must be associated with this product.

Explanation:

The product that you are customizing requires at least one Db2 entry to be associated with it before customization jobs can be generated.

System action:

None.

User response:

Associate a Db2 entry with the product that you are customizing, and regenerate the jobs.

CCQC031I The jobs were generated for the associated DB2 entries.

Explanation:

The customization jobs were generated for the Db2 entries that are associated with the product.

System action:

None.

User response:

No action is required.

CCQC032S

The customization jobs were not generated for *Product_name*.

Explanation:

A severe error occurred while the jobs were being generated for the specified product.

System action:

None.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQC033S The customization_library_name has no customized jobs.

Explanation:

The specified customization library cannot be browsed or edited because it is empty.

System action:

None.

User response:

Generate customization jobs for the specified library, and browse or edit the library again.

CC	QC034S	The specified operation is not
	-	allowed.
Ex	planation:	
Iss	uing comma	nds against customization jobs from
the	customizati	ion library from an ISPF browse or edi
ses	sion that wa	as started on the Finish Product

Customization panel is restricted.

System action: None.

User response:

To make changes to customization jobs, follow the steps for recustomization.

CCQC035E	Before you generate customization
	jobs, edit the product parameters
	to select one or more tasks or
	steps, and then issue the G line
	command or the GENERATEALL
	command again.

Explanation:

One or more tasks or steps must be selected before customization jobs can be generated.

System action:

None.

User response:

Edit the product parameters to select one or more tasks or steps. Then, issue the G line command or the GENERATEALL command again.

```
CCQC036E Before you exit the Product
Parameters panel, you must select
one or more tasks or steps to
generate customization jobs or
issue the CANCEL command.
```

Explanation:

One or more tasks or steps must be selected to generate customization jobs or the CANCEL command must be issued before you can exit the **Product Parameters** panel.

System action:

None.

User response:

Select one or more tasks or steps, or issue the CANCEL command.

CCQD000W The *member_name* environment index member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the specified environment index member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the warning.

CCQD001S The *member_name* environment index member is not valid. The PL/I XML parser issued the following exception error code: *code_number*.

Explanation:

While determining if the specified environment index member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the error.

CCQD002S The XML structure of the member_name environment index member is not valid. The element_name element is unknown.

Explanation:

The specified environment index member contains an unknown element.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD003S The XML structure of the member_name environment index member is not valid. Content is not allowed for the *element_name* element, but content was found.

Explanation:

Content was found in an element that cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD004S The XML structure of the member_name environment index member is not valid. Content is required for the element_name element, but content was not found.

Explanation:

The specified element does not contain required content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD005S	The XML structure of the
	<i>member_name</i> environment index
	member is not valid. The content
	length for the <i>element name</i>
	element exceeds
	maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD006S The XML structure of the member_name environment index member is not valid. The element_name element cannot occur more than maximum_number times.

Explanation:

The specified element occurs too many times in the environment index member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD007S	The XML structure of the
	<i>member_name</i> environment index
	member is not valid. The
	element_name element must occur
	at least <i>minimum_number</i> times.

Explanation:

The specified element does not occur enough times in the environment index member.

System action:

Processing stops.

User response:

CCQD008S	The XML structure of the
	member_name environment index

member is not valid. The attribute_name attribute in the element_name element cannot occur more than maximum_number times.

Explanation:

The specified attribute occurs too many times in the environment index member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD009S The XML structure of the member_name environment index member is not valid. The attribute_name attribute in the element_name element must occur at least minimum_number times.

Explanation:

The specified attribute does not occur enough times in the environment index member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD010S The XML structure of the member_name environment index member is not valid. Content is not allowed for the attribute_name attribute in the element_name element, but content was found.

Explanation:

Content was found in an attribute that cannot contain content. The name of the attribute and the name of the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD011S The XML structure of the member_name environment index member is not valid. Content is required for the attribute_name attribute in the element_name element, but content was not found.

Explanation:

An attribute does not contain required content. The name of the attribute and the name of the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD012S	The XML structure of the
	<i>member_name</i> environment index
	member is not valid. The content
	length for the <i>element_name</i>
	element exceeds
	maximum_number characters.

Explanation:

An element contains too many characters. The name of the element and the maximum number of allowed characters are indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD013S	The XML structure of the
	member_name environment index
	member is not valid. The
	<i>attribute_name</i> attribute in the
	element_name element is
	unknown.

Explanation:

The environment index member contains an unknown attribute. The name of the unknown attribute and the name of the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

```
CCQD050S The following LPAR serial number
is duplicated in the environment
index member: serial_number.
```

Explanation:

The environment index member contains duplicate LPAR serial numbers. The duplicate serial number is indicated in the message text.

System action:

Processing stops.

User response:

CCQD051S

The following DB2 serial number is duplicated in the environment index member: *serial number*.

Explanation:

The environment index member contains duplicate Db2 serial numbers. The duplicate serial number is indicated in the message text.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD052S The following Db2 group attach name is duplicated in the environment index member: group_attach_name.

Explanation:

The environment index member contains duplicate group attach names.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD053S The reference to the following DB2 subsystem for a Db2 group attach name is duplicated in the environment index member: subsystem_ID.

Explanation:

The environment index member contains duplicate references to a Db2 subsystem for a Db2 group attach name.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD054S The reference to the following DB2 subsystem for the LPAR_name LPAR is duplicated in the environment index member: subsystem_ID.

Explanation:

The environment index member contains duplicate references to a Db2 subsystem for an LPAR. The duplicate subsystem ID is indicated in the message text.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD055S The following Db2 group attach name was not found in the environment index member: group_attach_name.

Explanation:

A group attach name that is referenced by a Db2 member does not exist in the environment index member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD056S	The following LPAR was not found
	in the environment index member:
	LPAR_name.

Explanation:

The LPAR does not exist in the environment index member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

```
CCQD057S The following LPAR is duplicated
in the environment index member:
LPAR_name.
```

Explanation:

The environment index member contains duplicate LPARs. The name of the duplicate LPAR name is indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD100W The *member_name* product index member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the product index member is valid, the PL/I XML parser issued the specified exception warning code.

System action:

Processing continues.

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the specified exception warning code.

CCQD101S The member_name product index member is not valid. The PL/I XML parser issued the following exception error code: code number.

Explanation:

While determining if the product index member is valid, the PL/I XML parser issued the specified exception error code.

System action:

Processing stops.

User response:

See the Enterprise PL/I for z/OS Programming Guide for more information about the specified exception error code. Ensure that the Tools Customizer data store data set DCB is the same as the sample SCCQSAMP(CCQCDATS) data set DCB.

CCQD102S The XML structure of the member_name product index member is not valid. The element_name element is unknown.

Explanation:

The specified product index member contains an unknown element.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD103S The XML structure of the member_name product index member is not valid. Content is not allowed for the *element_name* element, but content was found.

Explanation:

Content was found for an element that cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD104S The XML structure of the member_name product index member is not valid. Content is required for the *element_name*

element, but content was not found.

Explanation:

The specified element does not contain required content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD105S The XML structure of the member_name product index member is not valid. The content length for the element_name element exceeds maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

The XML structure of the
<i>member_name</i> product index
member is not valid. The
<i>element_name</i> element cannot
occur more than
maximum_number times.

Explanation:

The specified element occurs too many times in the product index member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD107S	The XML structure of the
	<i>member_name</i> product index
	member is not valid. The
	element_name element must occur
	at least <i>minimum number</i> times.

Explanation:

The specified element does not occur enough times in the product index member.

System action:

Processing stops.

User response:

CCQD108S

The XML structure of the member_name product index member is not valid. The attribute_name attribute in the element name element cannot occur more than maximum_number times.

Explanation:

An attribute occurs too many times. The name of the attribute and the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD109S

The XML structure of the member_name product index member is not valid. The attribute_name attribute in the element_name element must occur at least minimum_number times.

Explanation:

The specified attribute does not occur enough times in the product index member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD110S The XML structure of the member_name product index member is not valid. Content is not allowed for the attribute_name attribute in the element_name element, but content was found.

Explanation:

An attribute cannot contain content. The name of the attribute and the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD111S The XML structure of the member_name product index member is not valid. Content is required for the attribute_name attribute in the element_name

element, but content was not found.

Explanation:

An attribute requires content. The name of the attribute and the name of the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD112S	The XML structure of the
	<i>member_name</i> product index
	member is not valid. The content
	length for the <i>element_name</i>
	element exceeds
	maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD1135	The XML structure of the
	<i>member_name</i> product index
	member is not valid.
	The attribute_name attribute in the
	element_name element is
	unknown.

Explanation:

The specified attribute in the product index member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD118S The content of the member_name product index member is not valid. The configuration_ID configuration ID for the configuration-name configuration name is not unique.

Explanation:

System action:

Processing stops.

User response:

CCQD120S

The content of the *member_name* product index member is not valid. The pack ID *pack_ID* that is referenced by product prefix *product_prefix* in the metadata library *library_name* could not be found.

Explanation:

The specified pack ID could not be found in the metadata library.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD121I The specified pack contains the component_name, which was previously specified as a standalone product.

Explanation:

The specified component of the pack was previously specified as a stand-alone product.

System action:

None.

User response:

No action is required.

CCQD122I The specified component metadata library was previously specified as part of the pack_name.

Explanation:

The specified metadata library for the component was previously specified as part of a pack.

System action:

None.

User response:

No action is required.

CCQD123E

23E The customization library name library_name is being used by another product or component. Specify another customization library qualifier on the Tools Customizer Settings panel.

Explanation:

A different product or component is using the specified customization library.

System action:

None.

User response:

Specify another customization library qualifier on the **Tools Customizer Settings** panel.

CCQD300W The *member_name* product environment member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the product environment member is valid, the PL/I XML parser issued the specified exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the specified exception warning code.

CCQD301S	The <i>member_name</i> product
	environment member is not valid.
	The PL/I XML parser issued the
	following exception error code:
	code_number.

Explanation:

While determining if the product environment member is valid, the PL/I XML parser issued the specified exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the specified exception error code.

```
CCQD302S The XML structure of the

member_name product

environment member is not valid.

The element_name element is

unknown.
```

Explanation:

The specified product environment member contains an unknown element.

System action:

Processing stops.

User response:

CCQD303S	The XML structure of the
	<i>member_name</i> product
	environment member is not valid.
	Content is not allowed for the
	element name element, but
	content was found.

Explanation:

Content was found for an element that cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD304S The XML structure of the member_name product environment member is not valid. Content is required for the element_name element, but content was not found.

Explanation:

The specified element does not contain required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD305S The XML structure of the member_name product environment member is not valid. The content length for the element_name element exceeds maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD306S

The XML structure of the member_name product environment member is not valid. The element_name element cannot occur more than maximum_number times.

Explanation:

The specified element occurs too many times in the product environment member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD307S

The XML structure of the *member_name* product environment member is not valid. The *element_name* element must occur at least *minimum_number* times.

Explanation:

The specified element does not occur enough times in the product environment member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD308S	The XML structure of the
	<i>member_name</i> product
	environment member is not valid.
	The attribute_name attribute in the
	<i>element name</i> element cannot
	occur more than
	maximum number times.

Explanation:

The specified attribute occurs too many times. The name of the attribute and the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

```
CCQD309S The XML structure of the

member_name product

environment member is not valid.

The attribute_name attribute in the

element_name element must occur

at least minimum_number times.
```

Explanation:

The specified attribute does not occur enough times in the product environment member.

System action:

Processing stops.

User response:

CCQD310S	The XML structure of the
	<i>member_name</i> product
	environment member is not valid.
	Content is not allowed for the
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element, but
	content was found.

Explanation:

The specified attribute cannot contain content. The name of the attribute and the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD311S The XML structure of the member_name product environment member is not valid. Content is required for the attribute_name attribute in the element_name element, but content was not found.

Explanation:

The specified attribute requires content. The name of the attribute and the name of the element that contains it are indicated in the message text.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD312S The XML structure of the member_name product environment member is not valid. The content length for the element_name element exceeds maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD313S The XML structure of the member_name product environment member is not valid. The attribute_name attribute in the element_name element is unknown.

Explanation:

The specified attribute in the product environment member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD350I The *subsystem_ID* Db2 subsystem is associated with this product.

Explanation:

The specified Db2 subsystem was added and saved in the Tools Customizer data store for the product to be customized.

System action:

Processing continues.

User response:

No action is required.

CCQD351I The *member_name* Db2 member for the *group_attach_name* Db2 group attach name is associated with this product.

Explanation:

The specified Db2 member for the group attach name was added and saved in the Tools Customizer data store for the product to be customized.

System action:

Processing continues.

User response:

No action is required.

CCQD352I The group_attach_name Db2 group attach name is associated with this product.

Explanation:

The specified Db2 group attach name was added and saved in the Tools Customizer data store for the product to be customized.

System action:

Processing continues.

User response:

No action is required.

CCQD353EThe subsystem_ID Db2 subsystemis already associated with this
product.

Explanation:

The specified Db2 subsystem cannot be added for the product to be customized because it already exists in the product environment in the data store.

System action:

None.

User response:

Ensure that the Db2 subsystem is specified correctly. If the problem persists, contact IBM Software Support.

CCQD354E

The *member_name* Db2 member for the *group_attach_name* Db2

group attach name is already associated with this product.

Explanation:

The specified Db2 member for the group attach name cannot be added for the product to be customized because it already exists in the product environment in the data store.

System action:

None.

User response:

Ensure that the Db2 group attach name is specified correctly. If the problem persists, contact IBM Software Support.

CCQD355E

The group_attach_name Db2 group attach name is already associated with this product.

Explanation:

The specified Db2 group attach name cannot be added for the product to be customized because it already exists in the product environment in the data store.

System action:

Processing stops.

User response:

Ensure that the Db2 group attach name is specified correctly. If the problem persists, contact IBM Software Support.

CCQD356S The *library_name* metadata library is already associated with the maximum number of allowed Db2 entries for this product.

Explanation:

The specified metadata library cannot be associated with more Db2 entries because it is already associated with the number of Db2 entries that are allowed.

System action:

Processing stops.

User response:

Delete an associated Db2 entry, and associate the specified library with another Db2 entry again.

CCQD357I The *subsystem_ID* Db2 subsystem is unassociated with this product.

Explanation:

The specified Db2 SSID was unassociated with the product that you are customizing.

System action:

Processing continues.

User response:

No action is required.

CCQD358I The member_name Db2 member for the group_attach_name Db2

group attach name is unassociated with this product.

Explanation:

The specified Db2 member for the Db2 group attach name was unassociated with the product that you are customizing.

System action:

Processing continues.

User response:

No action is required.

```
CCQD359I The group_attach_name Db2 group
attach name is unassociated with
this product.
```

Explanation:

The specified Db2 group attach name was unassociated with the product that you are customizing.

System action:

Processing continues.

User response:

No action is required.

CCQD360S The *library_name* metadata library is not associated with the specified Db2 subsystem *subsystem_ID*.

Explanation:

The specified Db2 subsystem and metadata library are not associated with each other.

System action:

None.

User response:

Ensure that the Db2 subsystem and the metadata library are associated. If the problem persists, contact IBM Software Support.

CCQD361S	The <i>library_name</i> metadata library
	is not associated with the
	specified DB2 data sharing group
	member <i>member_name</i> for the
	group_attach_name Db2 group
	attach name.

Explanation:

The specified Db2 data sharing group member for the group attach name and metadata library are not associated with each other.

System action:

None.

User response:

Ensure that the Db2 data sharing group member for the group attach name and the metadata library are

associated. If the problem persists, contact IBM Software Support.

CCQD362S The *library_name* metadata library is not associated with the specified *group_attach_name* Db2 group attach name.

Explanation:

The specified Db2 group attach name and metadata library are not associated with each other.

System action:

None.

User response:

Ensure that the Db2 group attach name and the metadata library are associated. If the problem persists, contact IBM Software Support.

CCQD400W

The customization parser issued the code_number warning code while it parsed the product customization member member_name. See the PL/I programming guide for more information about this XML parser continuable exception code.

Explanation:

While determining if the specified member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the warning.

CCQD401S The customization parser issued the code_number error code while it parsed the product customization member member_name. See the PL/I programming guide for more information about this XML parser terminating exception code.

Explanation:

While determining if the specified member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the error.

CCQD500W The *data_set_name* data store data set was not found.

Explanation:

Tools Customizer could not find the specified data store data set.

System action: None.

User response:

No action is required.

CCQD501W	The <i>data_set_name</i> data store
	data set was not found, so it was
	created.

Explanation:

Tools Customizer created the specified data set because it could not be found.

System action:

None.

User response:

No action is required.

CCQD502E	The <i>data_set_name</i> data store
	data set is not writable.

Explanation:

Tools Customizer cannot write to the specified data set.

System action:

None.

User response:

Ensure that the data set is writable.

CCQD503E The *data_set_name* data store data set could not be opened with the *disposition_type* disposition.

Explanation:

Tools Customizer could not open the data set with the specified disposition.

System action:

Processing stops.

User response:

Ensure that you have WRITE authority access to this data set.

```
CCQD504E
```

The *data_set_name* data store data set could not be opened with the *option_name* option.

Explanation:

Tools Customizer could not open the data set with the specified option.

System action:

Processing stops.

User response:

Ensure that you have WRITE authority access to this data set.
CCQD505E Th

The data store data set *data_set_name* already exists in a different volume.

Explanation:

Tools Customizer could not create the specified data set because the specified data set already exists in a different volume. Data store data set names must be unique.

System action:

Processing stops.

User response:

Specify a different data store data set name.

CCQD510I The DB2 SSID and Db2 group attach name were created.

Explanation:

The Db2 SSID and Db2 group attach name were created and saved in the data store.

System action:

None.

User response:

No action is required.

CCQD511E The DB2 entry already exists in the list of Db2 entries to be associated.

Explanation:

The Db2 entry cannot be added because it already exists in the list of Db2 entries to be associated.

System action:

None.

User response:

Specify a different Db2 entry.

CCQD512S An error occurred while a DB2 entry was being created.

Explanation:

A severe error occurred while a Db2 entry was being created.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD513E The specified DB2 entry already exists and is associated with the current product on the Customizer Workplace panel.

Explanation:

The Db2 entry cannot be added because it already exists, and it is already associated with the product to be customized.

System action:

None.

User response:

Press F3 to go to the **Customizer Workplace** panel to see the Db2 entry, or specify a different Db2 entry.

CCQD514E A value is required for a DB2 subsystem, a Db2 group attach name, or both before they can be created.

Explanation:

Required information is missing. A Db2 subsystem, a Db2 group attach name, or both must be specified.

System action:

None.

User response:

Specify a Db2 subsystem, a Db2 group attach name, or both.

CCQD515E The specified DB2 entry already exists in the list of Db2 entries and is already associated with the current product.

Explanation:

The Db2 entry has already been created and associated with the product that you want to customize.

System action:

None.

User response:

Specify a different Db2 entry.

CCQD516E The specified DB2 entry already exists in the list of Db2 entries on the Associate DB2 Entry with Product panel but is not associated with the current product.

Explanation:

The Db2 entry exists, but it must be associated with the product to be customized.

System action:

None.

User response:

On the **Customizer Workplace** panel, issue the ASSOCIATE command to associate the Db2 entry with the product.

CCQD517S An error occurred while a DB2 entry was being copied.

Explanation:

A severe error occurred while a Db2 entry was being copied

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD518E A value is required for a DB2 subsystem, a Db2 group attach name, or both before they can be copied.

Explanation:

Required information is missing. A Db2 subsystem, a Db2 group attach name, or both must be specified.

System action:

None.

User response:

Specify a Db2 subsystem, a Db2 group attach name, or both.

CCQD519I The DB2 entry was copied.

Explanation:

The Db2 entry was copied and saved in the Tools Customizer data store.

System action:

None.

User response:

No action is required.

CCQD520S The DB2 entry was copied to the list of Db2 entries but was not associated because the product is already associated with the allowed number of Db2 entries.

Explanation:

The Db2 entry was not completely copied because a product can be associated with only 1200 Db2 entries.

System action:

Processing stops.

User response:

Remove a Db2 entry from the list, and copy the specified Db2 entry again.

CCQD521E Line_command is not a valid line command.

Explanation:

The specified line command is not valid. Valid line commands are on the panel.

System action:

Processing stops.

User response:

Specify a valid line command.

CCQD522E

The *subsystem_ID* Db2 subsystem ID occurs more than once in the list. Each row must be unique.

Explanation:

The specified Db2 subsystem ID can be used only once.

System action:

Processing stops.

User response:

Specify a different Db2 subsystem ID.

CCQD523E	The group_attach_name Db2 group
	attach name occurs more than
	once in the list. Each row must be
	unique.

Explanation:

The specified Db2 group attach name can be used only once.

System action:

Processing stops.

User response:

Specify a different Db2 group attach name.

CCQD524E	The <i>member_name</i> Db2 member
	for the Db2 group attach name
	occurs more than once in the list.
	Each row must be unique.

Explanation:

The specified Db2 member for the Db2 group attach name can be used only once.

System action:

Processing stops.

User response:

Specify a different Db2 member for the Db2 group attach name.

CCQD525I The DB2 entries were created.

User response:

No action is required.

CCQD526E The *subsystem_ID* Db2 subsystem ID occurs more than once in the list. Each Db2 subsystem ID must be unique.

Explanation:

The specified Db2 subsystem ID can be used only once.

System action:

Processing stops.

User response:

Specify a different Db2 subsystem ID.

CCQD527I DB2 group attach names cannot be created during the copy process.

Explanation:

The ability to create Db2 group attach names is not available during the copy process.

System action:

None.

User response:

Create Db2 group attach names by issuing the CREATE command on the **Customizer Workplace** panel.

CCQD528E	The <i>metadata_library</i> metadata
	library is already associated with
	number Db2 entries. The
	maximum number of associated
	Db2 entries for this metadata
	library is 256.
	-

Explanation:

A metadata library can be associated with a maximum of 256 Db2 entries. The specified metadata library is already associated with 256.

System action:

Processing stops.

User response:

Remove an existing association between the specified metadata library and a Db2 entry, and associate the specified the metadata library with another entry.

CCQD529I	At least one row is required.
CCQD560E	The <i>subsystem_ID</i> Db2 subsystem already exists and is associated with the current product on the Customizer Workplace panel.

Explanation:

The specified Db2 subsystem exists and is associated with the product that you are customizing.

System action:

None.

User response:

Specify another Db2 subsystem.

CCQD561E	The <i>member_name</i> Db2 member
	for the group_attach_name Db2
	group attach name already exists
	and is associated with the current
	product on the Customizer
	Workplace panel.

Explanation:

The specified Db2 data sharing group for the Db2 group attach namer exists and is associated with the product that you are customizing.

System action:

None.

User response:

Specify another Db2 subsystem.

CCQD562E The group_attach_name Db2 group attach name already exists and is associated with the current

product on the Customizer Workplace panel.

Explanation:

The specified Db2 group attach name exists and is associated with the product that you are customizing. The subsystem is in the table on the **Customizer Workplace** panel.

System action: None.

User response:

Specify another Db2 group attach name.

CCQD563E A value is required for a DB2 subsystem, a Db2 group attach name, or both before they can be created.

Explanation:

A Db2 subsystem, a Db2 group attach name, or both are not specified so one or both of them cannot be created.

System action:

None.

User response:

Specify a value for the Db2 subsystem, the Db2 group attach name, or both.

CCQD565E	The subsystem_ID Db2 subsystem
	already exists in the list of Db2
	entries and is already associated
	with the current product.

Explanation:

The specified subsystem is already associated.

System action:

None.

User response:

Specify a different Db2 subsystem.

CCQD566E	The <i>member_name</i> Db2 member
-	for the group attach name Db2
	group attach name already exists
	in the list of Db2 entries and is
	already associated with the
	current product.

Explanation:

The specified Db2member is already associated.

System action:

None.

User response:

Specify a different Db2 member.

CCQD567E The group_attach_name Db2 group attach name already exists in the list of Db2 entries and is already

associated with the current product.

Explanation:

The specified Db2 group attach name is already associated.

System action:

None.

User response:

Specify another Db2 group attach name.

CCQD568I product_name is not associated with a DB2 entry.

Explanation:

The product that you are trying to customize is not associated with a Db2 entry. Before a product can be customized, it must be associated with at least one Db2 entry.

System action:

None.

User response:

Associate one or more Db2 entries with the product.

CCQD569I The *product_name* product configuration is not associated with a DB2 entry.

Explanation:

The configuration for the specified product is not associated with a Db2 entry.

System action:

None.

User response:

Associate one or more Db2 entries with the configuration.

CCQD577W

The *mode_name* Db2 mode of the *subsystem_ID* Db2 subsystem is not supported by the product.

Explanation:

The product does not support the specified Db2 mode.

System action:

None.

User response:

Specify a supported Db2 mode.

CCQD578W The mode_name Db2 mode of the member_name Db2 member for the Db2 group is not supported by the product.

Explanation:

The product does not support the specified Db2 mode.

System action:

None.

User response:

Specify a supported Db2 mode.

CCQD579W The *mode_name* Db2 mode of the group_name Db2 group attach name is not supported by the product.

Explanation:

The product does not support the specified Db2 mode.

System action:

None.

User response:

Specify a supported Db2 mode.

The subsystem_ID Db2 subsystem
was copied to the list of Db2
entries but was not associated
because the product is already
associated with the allowed
number of Db2 entries.

Explanation:

The copied Db2 subsystem was not associated with the product because the product is associated with the maximum number of Db2 entries.

System action:

None.

User response:

Remove an associated Db2 entry and associate the product with the copied Db2 subsystem.

CCQD581S	The <i>member_name</i> Db2 member
	for the group_attach_name Db2
	group attach name was copied to
	the list of Db2 entries but was not
	associated because the product is
	already associated with the
	allowed number of Db2 entries.

Explanation:

The copied Db2 member for the Db2 group attach name was not associated with the product because the product is associated with the maximum number of Db2 entries.

System action:

None.

User response:

Remove an associated Db2 entry and associate the product with the copied Db2 member.

CCQD582S	The group_attach_name Db2 group
	attach name was copied to the list
	of Db2 entries but was not
	associated because the product is
	already associated with the
	allowed number of Db2 entries.

Explanation:

The copied Db2 group attach name was not associated with the product because the product is associated with the maximum number of Db2 entries.

System action:

None.

User response:

Remove an associated Db2 entry and associate the product with the copied Db2 group attach name.

CCQD583I The *from_DB2* Db2 subsystem was copied to the *to_DB2* subsystem.

System action:

None.

User response:

No action is required.

CCQD584I The *member_name* DB2 member for the *group_attach_name* DB2 group attach name is copied to the *subsystem_ID* DB2 subsystem.

Explanation:

The specified Db2 member was copied.

System action:

None.

User response:

No action is required.

CCQD585I The group_attach_name Db2 group attach name cannot be copied because a Db2 member is required.

Explanation:

The specified Db2 group attach name was not copied because a Db2 member was missing.

System action:

None.

User response:

No action is required.

CCQD586S The current LPAR is LPAR_name, but the data store contains information about the LPAR_name LPAR. You must use the LPAR_name LPAR to customize the product.

Explanation:

The LPAR that is stored in the data store data set must be used to customize the product.

System action:

Processing stops.

User response:

Use the LPAR that is stored in the data store data set.

	<i>subsystem_name</i> Db2 subsystem is not supported by the product.
Explanation: The product doe	es not support the specified Db2 level.
System action: Processing conti	inues.
User response: Specify a suppor	rted level of Db2.
CCQD588W	The level_number Db2 level of the member_name Db2 member of the group_name Db2 group is not supported by the product.
Explanation: The product doe	es not support the specified Db2 level.
System action: Processing conti	inues.
User response: Specify a suppor	rted level of Db2.
CCQD589W	The <i>level_number</i> Db2 level of the <i>group_name</i> Db2 group attach name is not supported by the product.
Explanation: The product doe	es not support the specified Db2 level.
System action: Processing conti	inues.
User response: Specify a suppor	rted level of Db2.
CCQD593I	The <i>subsystem_ID</i> Db2 subsystem was deleted.
User response: No action is requ	uired.
CCQD594I	The <i>member_name</i> Db2 for the group_attach_name Db2 group attach name was deleted.
User response: No action is requ	uired.
CCQD595I	The group_attach_name Db2 group attach name was deleted.
User response: No action is requ	uired.
CCQD596E	The <i>subsystem_ID</i> Db2 subsystem was not deleted.
Explanation: An internal error	occurred while the specified Db2

The *level_number* Db2 level of the

CCQD587W

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD597E The *member_name* Db2 member for the *group_attach_name* Db2 group attach name was not deleted.

Explanation:

An internal error occurred while the specified Db2 member was being deleted.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD598E The group_attach_name Db2 group attach name was not deleted.

Explanation:

An internal error occurred while the specified Db2 group attach name was being deleted.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD600W The *member_name* product customization member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the XML structure of the product customization member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQD601S The member_name product customization member is not valid. The PL/I XML parser issued the following exception error code: code_number.

Explanation:

While determining if the XML structure of the product customization member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception error code.

CCQD602S The XML structure of the member_name product customization member is not valid. The element_name element is unknown.

Explanation:

The data store member contains an unknown element.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

The XML structure of the
<i>member_name</i> product
customization member is not
valid. Content is not allowed for
the <i>element name</i> element, but
content was found.

Explanation:

The specified element cannot contain content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD604S	The XML structure of the
	<i>member_name</i> product
	customization member is not
	valid. Content is required for the
	element_name element, but
	content was not found.

Explanation:

The specified element is missing required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD605S	The XML structure of the
	<i>member_name</i> product
	customization member is not
	valid. The content length for the
	element_name element exceeds
	<i>maximum_number</i> characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD606S The XML structure of the member_name product customization member is not valid. The element_name element cannot occur more than maximum_number times.

Explanation:

The specified element occurs too many times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD607S The XML structure of the member_name product customization member is not valid. The element_name element must occur at least minimum_number times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

The XML structure of the
<i>member_name</i> product
customization member is not
valid. The <i>attribute_name</i> attribute
in the <i>element_name</i> element
cannot occur more than
<i>maximum_number</i> times.

Explanation:

The specified attribute occurs too many times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD609S The XML structure of the member_name product customization member is not valid. The attribute_name attribute

in the *element_name* element must occur at least *minimum_number* times.

Explanation:

The specified attribute does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD610S The XML structure of the member_name product customization member is not valid. Content is not allowed for the attribute_name attribute in the element_name element, but content was found.

Explanation:

The specified attribute cannot contain content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD611S	The XML structure of the
	<i>member_name</i> product
	customization member is not
	valid. Content is required for the
	attribute_name attribute in the
	<i>element_name</i> element, but
	content was not found.

Explanation:

The specified attribute does not contain required content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQD612S The XML structure of the member_name product customization member is not valid. The content length for the element_name element exceeds maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD613S The XML structure of the member_name product customization member is not valid. The attribute_name attribute in the element_name element is unknown.

Explanation:

The specified attribute in the data store member is unknown.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD614S The content of the *member_name* product customization member is not valid. The value of the *element_name* element is not valid. The value is *value_name*.

Explanation:

The specified value is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQD700W The *member_name* Db2 data member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the XML structure of the Db2 data member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQD701S	The <i>member_name</i> Db2 data
	member is not valid. The PL/I XML
	parser issued the following
	exception error code:
	code_number.

Explanation:

While determining if the XML structure of the Db2 data member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception error code.

CCQD750W The value_number value in the Db2 parameter parameter_name was skipped because only maximum_number values are allowed.

Explanation:

The specified value was skipped because it exceeds the number of allowed values in the Db2 parameter.

System action:

Processing continues.

User response:

No action is required. To stop this message from being issued, remove the extra values from the Db2 parameter.

The <i>member_nαme</i> LPAR data
member is not valid. The PL/I XML
parser issued the following
exception warning code:
code_number.

Explanation:

While determining if the XML structure of the LPAR data member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQD801S	The <i>member_nαme</i> LPAR data
	member is not valid. The PL/I XML
	parser issued the following
	exception error code:
	code_number.

Explanation:

While determining if the XML structure of the LPAR data member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception error code.

CCQD850W The value_number value in the LPAR parameter parameter_name was skipped because only

maximum_number values are allowed.

Explanation:

The specified value was skipped because it exceeds the number of allowed values in the LPAR parameter.

System action:

Processing continues.

User response:

No action is required. To stop this message from being issued, remove the extra values from the LPAR parameter.

CCQD851I	The <i>subsystem_ID</i> Db2 subsystem
	is copied to the member_name Db2
	member for the
	group_attach_name Db2 group
	attach name.

User response:

No action is required.

CCQD852I The member_name Db2 member for the group_attach_name Db2 group attach name is copied to the member_name Db2 member for the group_attach_name Db2 group attach name.

User response:

No action is required.

CCQD854I	The <i>member_name</i> Db2 member
	for the group_attach_name Db2
	group 'attach name is copied to
	multiple Db2 entries.

User response:

No action is required.

CCQD900W The *member_name* product data member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the XML structure of the product data member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQD901S The *member_name* product data member is not valid. The PL/I XML parser issued the following exception error code: *code_number*.

Explanation:

While determining if the XML structure of the product data member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQD950W The value_number value in the product parameter parameter_name was skipped because only maximum_number values are allowed.

Explanation:

The specified value was skipped because it exceeds the number of allowed values in the product parameter.

System action:

Processing continues.

User response:

No action is required. To stop this message from being issued, remove the extra values from the product parameter.

CCQD960I	The subsystem_ID Db2 subsystem
	was changed to the member_name
	Db2 member for the
	group_attach_name Db2 group
	attach name.

User response:

No action is required.

CCQD961I	The <i>member_name</i> Db2 member
	for the group_attach_name Db2
	group attach name was changed to
	the <i>subsystem_ID</i> Db2 subsystem.

User response:

No action is required.

CCQD962I The *member_name* Db2 member for the *group_attach_name* Db2 group attach name was changed to the *member_name* Db2 member for the *group_attach_name* Db2 group attach name.

User response:

No action is required.

CCQD963E	The DB2 group attach name
	cannot be blank when the Db2
	subsystem ID is blank.

Explanation:

A Db2 group attach name, Db2 subsystem ID, or both must be specified.

System action:

Processing stops.

User response:

Specify a Db2 group attach name, Db2 subsystem ID, or both.

CCQE000S The specified message field name or message *message_ID* was not found.

Explanation:

An error occurred while displaying a message field name or the specified message.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQE001E An incorrect trace level was specified. Valid trace levels are 0 -4.

Explanation:

A wrong trace level was specified. Valid trace levels are 0 - 4.

System action:

Processing stops.

User response:

Specify a valid trace level 0 - 4.

CCQH001W The specified option *option_name* is not valid.

Explanation:

The option that was specified is not a valid option on the panel.

System action:

Tools Customizer stops.

User response:

Specify a valid option on the panel.

CCQH006W Before you customize a product, verify your user settings.

Explanation:

The user settings must be verified before a product can be customized.

System action:

Tools Customizer stops.

User response:

Verify the user settings.

CCQH007E Check the user settings. One or more current values are not valid.

Explanation:

One or more of the values in the user settings is not valid.

System action:

Tools Customizer stops.

User response:

Ensure that the specified values for the user settings are valid.

CCQH008W	Before you use Tools Customizer,
	you must select option 0 to verify
	your user settings.

Explanation:

The user settings must be changed before a product can be customized.

System action:

Tools Customizer stops.

User response:

Change the user settings.

CCQH009E You must select option 0 to change your user settings.

Explanation:

User settings must be changed before a product can be customized.

System action:

Tools Customizer stops.

User response:

Change the user settings.

CCQ1000W	The XML structure of the
	<i>member_name</i> Db2 parameter
	metadata member is not valid. The
	PL/I XML parser issued the
	following exception warning code:
	code_number.

Explanation:

While determining if the Db2 parameter metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI001S	The XML structure of the
	<i>member_nαme</i> Db2 parameter
	metadata member is not valid. The
	PL/I XML parser issued the
	following exception error code:
	code number.

Explanation:

While determining if the Db2 parameter metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI002S The XML structure of the member_name Db2 parameter metadata member is not valid. The element name element is unknown.

Explanation:

The specified element in the Db2 parameter metadata member is unknown.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI003S The XML structure of the member_name Db2 parameter metadata member is not valid. Content is not allowed for the element_name element, but content was found.

Explanation:

The specified element cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI004S

The XML structure of the member_name Db2 parameter metadata member is not valid. Content is required for the element_name element, but content was not found.

Explanation:

The specified element requires content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI005S The XML structure of the member_name Db2 parameter metadata member is not valid. The

content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI006S The XML structure of the member_name Db2 parameter metadata member is not valid. The content length for the element_name element must be at least minimum_number characters.

Explanation:

The specified element does not contain enough characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI007S The XML structure of the member_name Db2 parameter metadata member is not valid. The element_name element must occur at least minimum_number times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI008S The XML structure of the member_name Db2 parameter metadata member is not valid. The attribute_name attribute in the element_name element cannot occur more than maximum_number times.

Explanation:

The specified attribute occurs too many times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI009S The XML structure of the *member name* Db2 parameter metadata member is not valid. The attribute_name attribute in the element_name element must occur at least *minimum_number* times.

Explanation:

The specified attribute did not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI010S The XML structure of the *member name* Db2 parameter metadata member is not valid. Content is not allowed for the attribute_name attribute in the element_name element, but content was found.

Explanation:

The specified attribute cannot have content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI011S

The XML structure of the *member_name* Db2 parameter metadata member is not valid. Content is required for the attribute_name attribute in the element_name element, but content was not found.

Explanation:

The specified attribute is missing required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI012S The XML structure of the member_name Db2 parameter metadata member is not valid. The content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI0135	The XML structure of the
	<i>member_nαme</i> Db2 parameter
	metadata member is not valid. The
	<i>attribute_name</i> attribute in the
	element_name element is
	unknown.

Explanation:

The specified attribute in the Db2 parameter metadata member is unknown.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI014S

The content of the *member_name* **Db2** parameter metadata member is not valid because the value of the element_name element is incorrect. The value is value_name.

Explanation:

The specified value of the element is not a valid value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI015S	The content of the DB2 parameter
	metadata member is not valid
	because the value of the
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element is
	incorrect. The value of the
	attribute is <i>value name</i> .

Explanation:

The specified value of the attribute is not a valid value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI016S	The content of the DB2 parameter
	metadata member is not valid

because the data type of the element_name element is incorrect. The value is value name.

Explanation:

The specified data type is not a valid data type.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI017S The content of the DB2 parameter metadata member is not valid because the data type of the *attribute_name* attribute in the *element_name* element is incorrect. The value of the attribute is value_name.

Explanation:

The specified data type is not a valid data type.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI050S The *member_name* Db2 parameter metadata member was not found in the *data_set_name* data set.

Explanation:

Tools Customizer could not find the specified Db2 parameter metadata member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI051S The parameter_name LPAR parameter in the template_name template does not have associated metadata in the member_name LPAR parameter metadata member.

Explanation:

The specified template does not contain metadata for an LPAR parameter. The name of the LPAR parameter metadata member, the name of the LPAR parameter, and the name of the template are indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI052S The parameter_name product parameter in the template_name template does not have associated metadata in the member_name product parameter metadata member.

Explanation:

The specified template does not contain metadata for a product parameter. The name of the product parameter metadata member, the name of the product parameter, and the name of the template are indicated in the message text.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI053E	The following metadata data set
	was not found: <i>data_set_name</i> .

Explanation:

Tools Customizer could not find the specified metadata data set.

System action:

Processing stops.

User response:

Ensure that the metadata data set is specified correctly. If the problem persists, contact IBM Software Support.

CCQI054E The following metadata data set could not be opened: data_set_name.

Explanation:

Tools Customizer could not open the specified LPAR metadata data set.

System action:

Processing stops.

User response:

Ensure the metadata data set was specified correctly.

CCQI055S	The CCQ\$\$DB2 Db2 parameter
	metadata member was not found
	in the <i>data_set_name</i> Tools
	Customizer metadata data set.

Explanation:

Tools Customizer could not find the Db2 parameter metadata member in the specified Tools Customizer metadata data set.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI056S The CCQ\$\$LPR LPAR parameter metadata member was not found in the *data_set_name* data set.

Explanation:

Tools Customizer could not find the specified LPAR parameter metadata member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI057S The *member_name* product parameter metadata member was not found in the *data_set_name* data set.

Explanation:

The product parameter metadata member was not found in the specified data set.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ10581

Product_name does not have any Db2 parameters.

Explanation:

Db2 parameters are not required to customize the specified product.

System action:

Processing continues.

User response:

No action is required.

CCQ10591

Product_name does not have any LPAR parameters.

Explanation:

LPAR parameters are not required to customize the specified product.

System action:

Processing continues.

User response:

No action is required.

CCQI060S

The parameter_name Db2 parameter in the task_description task condition does not have associated metadata in the

member_name Db2 parameter metadata member.

Explanation:

Associated metadata is missing for the specified Db2 parameter in a task.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI061S The parameter_name LPAR parameter in the task_description task condition does not have associated metadata in the member_name LPAR parameter metadata member.

Explanation:

Associated metadata is missing for the specified LPAR parameter in a task.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI062S	The <i>parameter_name</i> product
	parameter in the <i>task_description</i>
	task condition does not have
	associated metadata in the
	<i>member_name</i> product parameter
	metadata member.

Explanation:

Associated metadata is missing for the specified product parameter in a task.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI063S The parameter_name Db2 parameter in the task_description task and the step_description step does not have associated metadata in the member_name Db2 parameter metadata member.

Explanation:

Associated metadata is missing for the specified Db2 parameter in a task and step.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI064S The parameter_name LPAR parameter in the task_description task and the step_description step does not have associated metadata in the member_name LPAR parameter metadata member.

Explanation:

Associated metadata is missing for the specified LPAR parameter in a task and step.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI065S The parameter_name product parameter in the task_description task and the step_description step does not have associated metadata in the member_name parameter metadata member.

Explanation:

Associated metadata is missing for the specified parameter in a task and step.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI066S The parameter_name Db2 parameter in the task_description task, step_description step, and template_name template condition does not have associated metadata in the member_name Db2 parameter metadata member.

Explanation:

Associated metadata is missing for the specified Db2 parameter in a task, step, and template.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI067S The parameter_name LPAR parameter in the task_description task, step_description step, and template_name template condition does not have associated

metadata in the *member_name* LPAR parameter metadata member.

Explanation:

Associated metadata is missing for the specified LPAR parameter in a task, step, and template.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI068S	The <i>parameter_name</i> product
	parameter in the task_description
	task, <i>step_description</i> step, and
	template_name template condition
	does not have associated
	metadata in the <i>member_name</i>
	product parameter metadata
	member.

Explanation:

Associated metadata is missing for the specified product parameter in a task, step, and template.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1069S	Product metadata does not
	support multiple configurations,
	but the <i>template_name</i> product
	template contains the
	parameter_name parameter.
	Enable multiple configurations
	support for this product, and try
	again.

Explanation:

The specified template contains a parameter for multiple configurations, but the product is not enabled to support multiple configurations.

System action:

Processing stops.

User response:

Enable multiple configurations support, and try again.

CCQI070E	The parameter_name Db2
	parameter metadata member is
	not valid. The default length for
	the parameter-element_name
	parameter element exceeds the
	length of the parameter. The
	default length is <i>default_length</i> ,
	and the specified length is
	specified_length. The default

length will be truncated accordingly.

Explanation:

The specified length cannot be shorter than the default length.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI071E The parameter_name LPAR parameter metadata member is not valid. The default length for the parameter-element_name parameter element exceeds the length of the parameter. The default length is default_length, and the specified length is specified_length. The default length will be truncated accordingly.

Explanation:

The specified length cannot be shorter than the default length.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI072E

The parameter_name product parameter metadata member is not valid. The default length for the parameter-element_name parameter element exceeds the length of the parameter. The default length is *default_length*, and the specified length is *specified_length*. The default length will be truncated accordingly.

Explanation:

The specified length cannot be shorter than the default length.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI073S The XML structure of the member_name Db2 parameter metadata member is not valid. The following value of the

attribute_name attribute in the *element_name* element already exists: *value_name*.

Explanation:

The specified value already exists for an attribute.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI074S The XML structure of the member_name LPAR parameter metadata member is not valid. The following value of the attribute_name attribute in the element_name element already exists: value_name.

Explanation:

The specified value already exists for an attribute.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI075S	The XML structure of the
	<i>member_name</i> product parameter
	metadata member is not valid. The
	following value of the
	attribute_name attribute in the
	element_name element already
	exists: value_name.

Explanation:

The specified value already exists for an attribute.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI076S The XML structure of the member_name Db2 parameter metadata member is not valid. The parameter_name parameter refers to the section-name section. This section was not found in the Db2 parameter metadata member.

Explanation:

The specified value already exists for an attribute.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI077S The XML structure of the member_name LPAR parameter metadata member is not valid. The parameter_name parameter refers to the section-name section. This section was not found in the LPAR parameter metadata member.

Explanation:

The specified parameter refers to a section that is not in the LPAR parameter metadata member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI078S The XML structure of the member_name product parameter metadata member is not valid. The parameter_name parameter refers to the section-name section. This section was not found in the product parameter metadata member.

Explanation:

The specified parameter refers to a section that is not in the product parameter metadata member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ1080S	The content of the <i>member_name</i>
	Db2 parameter metadata member
	is not valid because the value of
	the attribute_name attribute in the
	element_name element is
	incorrect. The value of the
	attribute is <i>value_name</i> .

Explanation:

The specified value for an attribute in the Db2 parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI081S The content of the *member_name* LPAR parameter metadata member is not valid because the value of the *attribute_name* attribute in the *element_name* element is incorrect. The value of the attribute is *value_name*.

Explanation:

The specified value for an attribute in the LPAR parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI082S	The content of the <i>member_name</i>
	product parameter metadata
	member is not valid because the
	value of the <i>attribute_name</i>
	attribute in the <i>element name</i>
	element is incorrect. The value of
	the attribute is <i>value name</i> .

Explanation:

The specified value for an attribute in the product parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1090S	The product-defined Db2
	parameter <i>parameter_name</i> in the
	<i>member_name</i> parameter
	metadata member references the
	section_ID section ID, but this ID
	does not exist in either the
	parameter metadata member or
	the Db2 parameter metadata
	member.

Explanation:

A section that does not exist in the parameter metadata member or the Db2 parameter metadata member is referenced by the specified Db2 parameter.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ10915	The product-defined LPAR
	parameter in the <i>member_name</i>
	parameter metadata member
	references the section ID section
	ID, but this ID does not exist in
	either the parameter metadata

member or the LPAR parameter metadata member.

Explanation:

A section that does not exist in the parameter metadata member or the LPAR parameter metadata member is being referenced by the specified LPAR parameter.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI092S The overridden DB2 parameter parameter_name in the member_name parameter metadata member does not exist in the Db2 parameter metadata member.

Explanation:

The specified parameter does not exist.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI093S The overridden LPAR parameter parameter_name in the member_name parameter metadata member does not exist in the LPAR parameter metadata member.

Explanation:

The specified parameter does not exist.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI094S

The CCQ\$\$PRD product customization parameter metadata member was not found in the *data_set_name* data set.

Explanation:

The specified data set must contain the CCQ\$\$PRD product customization parameter metadata member

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI100W

The XML structure of the *member_name* LPAR parameter metadata member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the LPAR parameter metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI101S	The XML structure of the
	<i>member_name</i> LPAR parameter
	metadata member is not valid. The
	PL/I XML parser issued the
	following exception error code:
	code_number.

Explanation:

While determining if the LPAR parameter metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI102S The XML structure of the member_name LPAR parameter metadata member is not valid. The element_name element is unknown.

Explanation:

The specified element in the LPAR parameter metadata member is unknown.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI103S The XML structure of the member_name LPAR parameter metadata member is not valid. Content is not allowed for the element_name element, but content was found.

Explanation:

The specified element cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI104S The XML structure of the member_name LPAR parameter metadata member is not valid. Content is required for the element_name element, but content was not found.

Explanation:

The specified element requires content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI105S

The XML structure of the member_name LPAR parameter metadata member is not valid. The content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI106S The XML structure of the member_name LPAR parameter metadata member is not valid. The content length for the element_name element must be at least minimum_number characters.

Explanation:

The specified element does not contain enough characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI107S The XML structure of the member_name LPAR parameter metadata member is not valid. The

element_name element must occur at least *minimum_number* times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI108S	The XML structure of the
	<i>member_name</i> LPAR parameter
	metadata member is not valid. The
	attribute name attribute in the
	<i>element name</i> element cannot
	occur more than
	maximum_number times.

Explanation:

The specified attribute occurs too many times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI109S The XML structure of the member_name LPAR parameter metadata member is not valid. The attribute_name attribute in the element_name element must occur at least minimum_number times.

Explanation:

The specified attribute did not occur enough times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI110S	The XML structure of the
	<i>member_name</i> LPAR parameter
	metadata member is not valid.
	Content is not allowed for the
	attribute name attribute in the
	<i>element name</i> element, but
	content was found.

Explanation:

The specified attribute cannot have content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI111S The XML structure of the member_name LPAR parameter metadata member is not valid. Content is required for the attribute_name attribute in the element_name element, but content was not found.

Explanation:

The specified attribute is missing required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI1125	The XML structure of the
	<i>member_name</i> LPAR parameter
	metadata member is not valid. The
	content length for the
	<i>element_name</i> element cannot
	exceed maximum_number
	characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI113S The XML structure of the member_name LPAR parameter metadata member is not valid. The attribute_name attribute in the element_name element is unknown.

Explanation:

The specified attribute in the LPAR parameter metadata member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI114S The content of the member_name LPAR parameter metadata member is not valid because the value of the element_name element is incorrect. The value is value_name.

Explanation:

The specified value for an element in the LPAR parameter metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI1155	The content of the <i>member_name</i>
	LPAR parameter metadata
	member is not valid because the
	value of the <i>attribute_name</i>
	attribute in the <i>element_name</i>
	element is incorrect. The value of
	the attribute is <i>value_name</i> .

Explanation:

The specified value for an attribute in the LPAR parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI116S The content of the member_name LPAR parameter metadata member is not valid because the data type of the *element_name* element is incorrect. The value is value name.

Explanation:

The specified data type value for an element in the LPAR parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI1175	The content of the <i>member_name</i>
	LPAR parameter metadata
	member is not valid because the
	data type of the <i>attribute_name</i>
	attribute in the <i>element_name</i>
	element is incorrect. The value is
	value_name.

Explanation:

The specified data type value for an attribute in the LPAR parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI120S

The XML structure of the *member_name* Db2 parameter metadata member is not valid. The *element_name* element in the *parameter_name* parameter contains duplicate values for the *element_name* element. The duplicate value is *value_name*.

Explanation:

An element contains the specified duplicate value.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI121S

The XML structure of the *member_name* LPAR parameter metadata member is not valid. The *element_name* element in the *parameter_name* parameter contains duplicate values for the *element_name* element. The duplicate value is *value_name*.

Explanation:

An element contains the specified duplicate value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI122S The XML structure of the member_name parameter metadata member is not valid. The element_name element in the parameter_name parameter contains duplicate values for the element_name element. The duplicate value is value_name.

Explanation:

An element contains the specified duplicate value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI123S The XML structure of the member_name discover metadata member is not valid. The element_name element in the parameter_name parameter contains duplicate values for the

element_name element. The duplicate value is *value_name*.

Explanation:

An element contains the specified duplicate value.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI124S	The XML structure of the member name product
	customization parameter
	metadata member is not valid. The element_name element in the
	<i></i>
	contains duplicate values for the element_name element. The duplicate value is value, name
	auplicate value 15 value_nume.

Explanation:

An element contains the specified duplicate value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI200W	The XML structure of the
	<i>member_name</i> information
	metadata member is not valid. The
	PL/I XML parser issued the
	following exception warning code:
	code_number.

Explanation:

While determining if the information metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI2015	The XML structure of the
	<i>member_name</i> information
	metadata member is not valid. The
	PL/I XML parser issued the
	following exception error code:
	code number.

Explanation:

While determining if the information metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI202S The XML structure of the member_name information metadata member is not valid. The element name element is unknown.

Explanation:

The specified element in the information metadata member is unknown.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI2035

The XML structure of the member_name information metadata member is not valid. Content is not allowed for the element_name element, but content was found.

Explanation:

The specified element cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI204S

member_name information metadata member is not valid. Content is required for the *element_name* element, but content was not found.

The XML structure of the

Explanation:

The specified element requires content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI205S The XML structure of the member_name information metadata member is not valid. The content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI206S	The XML structure of the
	<i>member_name</i> information
	metadata member is not valid. The
	content length for the
	<i>element_name</i> element must be at
	least minimum number
	characters.

Explanation:

The specified element does not contain enough characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI207S The XML structure of the

member_name information metadata member is not valid. The *element_name* element must occur at least *minimum_number* times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI208S The XML structure of the member_name information metadata member is not valid. The attribute_name attribute in the element_name element cannot occur more than maximum_number times.

Explanation:

The specified attribute occurs too many times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI209S	The XML structure of the
	member name information

metadata member is not valid. The attribute_name attribute in the element_name element must occur at least minimum_number times.

Explanation:

The specified attribute did not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI210S

The XML structure of the member_name information metadata member is not valid. Content is not allowed for the attribute_name attribute in the element_name element, but content was found.

Explanation:

The specified attribute cannot have content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI211S The XML structure of the member_name information metadata member is not valid. Content is required for the attribute_name attribute in the element_name element, but content was not found.

Explanation:

The specified attribute is missing required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI212S The XML structure of the member_name information metadata member is not valid. The content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action: Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI213S The XML structure of the member_name information metadata member is not valid. The attribute_name attribute in the element_name element is unknown.

Explanation:

The specified attribute in the information metadata member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI2145	The content of the <i>member_name</i>
	information metadata member is
	not valid because the value of the
	<i>element_name</i> element is
	incorrect. The value is
	value_name.

Explanation:

The specified value for an element in the information metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI215S	The content of the <i>member_name</i>
	information metadata member is
	not valid because the value of the
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element is
	incorrect. The value is
	value name.

Explanation:

The specified value for an attribute in the information metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI216S The content of the *member_name* information metadata member is not valid because the data type of the *element_name* element is

incorrect. The value is *value_name*.

Explanation:

The specified data type value for an element in the information metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI217S The content of the member_name information metadata member is not valid because the data type of the attribute_name attribute in the element_name element is incorrect. The value is value_name.

Explanation:

The specified data type value for an attribute in the information metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI218S

The content of the *member_name* information metadata member is not valid. The length of the *value_name* value that of the *attribute_name* attribute is longer than the *value_name* value of the *attribute_name* attribute.

Explanation:

The first specified value cannot be longer than the second specified value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI219S The content of the member_name information metadata member is not valid. The value_name value of the attribute_name attribute contains the value_name value.

Explanation:

The first specified value cannot be longer than the second specified value.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI220S The XML structure of the member_name information metadata member is not valid. Content for the attribute_name attribute in theelement_name element exceed maximum_number characters.

Explanation:

The specified attribute contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI223S The XML structure of the member_name information metadata member is not valid. The value that is specified for the Db2 Level already exists. The value is value_name.

Explanation:

The specified value already exists.

System action:

Processing stops.

User response:

Specify a different Db2 level. If the problem persists, contact IBM Software Support.

CCQI224S	The XML structure of the
	<i>member_name</i> information
	metadata member is not valid. The
	value that is specified for the Db2
	Mode already exists. The value is
	value name.

Explanation:

The specified value already exists.

System action:

Processing stops.

User response:

Specify a different Db2 mode. If the problem persists, contact IBM Software Support.

CCQI250S	The information metadata
	member was not found in the
	<i>data_set_name</i> data set.

Explanation:

Tools Customizer could not find the information metadata member in the specified data set.

System action:

Processing stops.

User response

If this message was issued on the **Specify the Metadata Library** (CCQPHLQ) panel, specify the product metadata library. The name of this library is *hlq*.SANLDYXDENU.

Do not specify the Tools Customizer metadata library, which is *hlq*.SCCQDENU.

If the problem persists, identify the name of the Tools Customizer trace data set and contact IBM Software Support.

CCQI251E The *member_name* member was not accessible in the *data_set_name* data set.

Explanation:

The specified member could not be accessed in the data set.

System action:

Processing stops.

User response:

Specify the correct metadata library.

CCQI252S The information metadata member was not found in the *library_name* component metadata library that is part of the *library_name* pack metadata library. The name of the pack is *pack_name*.

Explanation:

The specified component metadata library does not contain the information metadata member.

System action:

Processing stops.

User response:

CCOI253E

Specify the correct metadata library.

The *library_name* Tools Customizer metadata library is not current. Update the metadata library on the Tools Customizer Settings panel.

Explanation:

The specified metadata library is not current.

System action:

Processing stops.

User response:

Specify a current metadata library on the **Tools Customizer Settings** panel.

CCQ1300W

The XML structure of the *member_name* sequence metadata member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation:

While determining if the sequence metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI301S The XML structure of the member_name sequence metadata member is not valid. The PL/I XML parser issued the following exception error code: code_number.

Explanation:

While determining if the sequence metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the Enterprise PL/I for z/OS Programming Guide for more information about the exception error code, and contact IBM Software Support.

CCQI302S The XML structure of the member_name sequence metadata member is not valid. The element_name element is unknown.

Explanation:

The specified element in the sequence metadata member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI303S The XML structure of the

member_name sequence metadata member is not valid. Content is not allowed for the *element_name* element, but content was found.

Explanation:

The specified element cannot contain content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI3045	The XML structure of the
	member_name sequence metadata
	member is not valid. Content is
	required for the <i>element_name</i>
	element, but content was not
	found.

Explanation:

The specified element is missing required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI305S The XML structure of the member_name sequence metadata member is not valid. Content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI306S The XML structure of the member_name sequence metadata member is not valid. The element_name element cannot occur more than maximum_number times.

Explanation:

The specified element occurs too many times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI307S The XML structure of the member_name sequence metadata member is not valid. The element_name element must occur at least minimum_number times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI308S	The XML structure of the
	<i>member_name</i> sequence metadata
	member is not valid. The
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element cannot
	occur more than
	maximum_number times.

Explanation:

The specified attribute occurs too many times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI3095	The XML structure of the
	member_name sequence metadata
	member is not valid. The
	<i>attribute_name</i> attribute in the
	element_name element must occur
	at least <i>minimum_number</i> times.

Explanation:

The specified attribute does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI310S	The XML structure of the
	<i>member_name</i> sequence metadata
	member is not valid. Content is not
	allowed for the <i>attribute_name</i>
	attribute in the <i>element name</i>
	element, but content was found.

Explanation:

The specified attribute cannot contain content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI3115	The XML structure of the
	<i>member_name</i> sequence metadata
	member is not valid. Content is
	required for the attribute_name

attribute in the *element_name* element, but content was not found.

Explanation:

The specified attribute is missing required content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI312S The XML structure of the member_name sequence metadata member is not valid. The content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI313S The XML structure of the member_name sequence metadata member is not valid. The attribute_name attribute in the element_name element is unknown.

Explanation:

The specified attribute in the sequence metadata member is unknown.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI314S The content of the member_name sequence metadata member is not valid because the value of the *element_name* element is incorrect. The value is *value_name*.

Explanation:

The specified value for an element in the sequence metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI315S The content of the member_name sequence metadata member is not valid because the value of the *attribute_name* attribute in the *element_name* element is incorrect. The value is *value_name*.

Explanation:

The specified value for an attribute in the sequence metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI316S	The content of the <i>member_name</i>
	sequence metadata member is not
	valid because the data type of the
	element_name element is
	incorrect. The value is
	value_name.

Explanation:

The specified data type value for an element in the sequence metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI317S	The content of the <i>member_name</i>
	sequence metadata member is not
	valid because the data type of the
	attribute name attribute in the
	<i>element name</i> element is
	incorrect. The value is
	value name.

Explanation:

The specified data type value for an attribute in the sequence metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI350S The XML structure of the member_name sequence metadata member is not valid because the value of the attribute_name attribute in the element_name

element is incorrect. The value is *value_name*.

Explanation:

A specified value for an attribute in the sequence metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI351S The *member_name* sequence metadata member was not found in the *data_set_name* metadata data set.

Explanation:

Tools Customizer could not find the specified sequence metadata member in the metadata data set.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI352S The *template_name* product template was not found in the *data_set_name* metadata data set.

Explanation:

Tools Customizer could not find the specified product template in the data set.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI353S The sequence metadata member was not found in the *data_set_name* component data set that is part of the *data_set_name* pack.

Explanation:

Tools Customizer could not find the sequence metadata member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI360S The XML structure of the member_name sequence metadata member is not valid. The value of the attribute_name attribute in the

Explanation:

The specified attribute contains a value that already exists.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI361S The XML structure of the member_name sequence metadata member is not valid. The condition element on the *level_type* level already contains a relational operator.

Explanation:

A relational operator already exists for the condition element on the specified level.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

```
CCQI362S The XML structure of the

member_name sequence metadata

member is not valid. The condition

element on the level_type level

must contain only one content

string or content number element.
```

Explanation:

Only one content string element or content number element can be contained in the condition element on the specified level.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

```
CCQI363S The XML structure of the

member_name sequence metadata

member is not valid. The condition

element in the element_name

element with the attribute_name

attribute must contain either the

content string element or the

content number element.
```

Explanation:

Either the content string element or the content number element must be in the condition element.

System action:

Processing stops.

User response:

Contact IBM Software Support.

CCQI400W The XML structure of the member_name parameter metadata member is not valid. The PL/I XML parser issued the following exception warning code: code_number.

Explanation:

While determining the parameter metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI401S The XML structure of the member_name parameter metadata member is not valid. The PL/I XML parser issued the following exception error code: code_number.

Explanation:

While determining if the parameter metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQI402S The XML structure of the member_name parameter metadata member is not valid. The element name element is unknown.

Explanation:

The specified element in the parameter metadata member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI403S The XML structure of the member_name parameter metadata member is not valid. Content is not allowed for the

element_name element, but content was found.

Explanation:

The specified element cannot contain content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI404S	The XML structure of the
	<i>member_name</i> parameter
	metadata member is not valid.
	Content is required for the
	element name element, but
	content was not found.

Explanation:

The specified element requires content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI405S The XML structure of the

member_name parameter metadata member is not valid. The content length for the *element_name* element cannot exceed *maximum_number* characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI406S	The XML structure of the
	<i>member_nαme</i> parameter
	metadata member is not valid. The
	content length for the
	<i>element_name</i> element must be at
	least minimum number
	characters.

Explanation:

The specified element does not contain enough characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI407S The XML structure of the member_name parameter metadata member is not valid. The element_name element must occur at least minimum_number times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI408S The XML structure of the member_name parameter metadata member is not valid. The attribute_name attribute in the element_name element cannot occur more than maximum_number times.

Explanation:

The specified attribute occurs too many times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI409S The XML structure of the member_name parameter metadata member is not valid. The attribute_name attribute in the element_name element must occur at least minimum_number times.

Explanation:

The specified attribute does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI410S	The XML structure of the
	<i>member_name</i> parameter
	metadata member is not valid.
	Content is not allowed for the
	attribute_name attribute in the
	<i>element_name</i> element, but
	content was found.

Explanation:

The specified attribute cannot have content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI411S	The XML structure of the
	<i>member_name</i> parameter
	metadata member is not valid.
	Content is required for the
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element, but
	content was not found.

Explanation:

The specified attribute is missing required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI412S	The XML structure of the
	<i>member_name</i> parameter
	metadata member is not valid. The
	content length for the
	element_name element cannot
	exceed maximum_number
	characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI413S	The XML structure of the
	member name parameter
	member_nume parameter
	metadata member is not valid. The
	<i>attribute_name</i> attribute in the
	element_name element is
	unknown.

Explanation:

The specified attribute in the parameter metadata member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI414S The content of the *member_name* parameter metadata member is not valid because the value of the *element_name* element is

incorrect. The value is *value_name*.

Explanation:

The specified value for an element in the parameter metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI415S The content of the *member_name* parameter metadata member is not valid because the value of the *attribute_name* attribute in the *element_name* element is incorrect. The value is *value_name*.

Explanation:

The specified value for an attribute in the parameter metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI416S The content of the *member_name* parameter metadata member is not valid because the data type of the *element_name* element is incorrect. The value is *value_name*.

Explanation:

The specified data type value for an element in the parameter metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI417S The content of the member_name parameter metadata member is not valid because the data type of the attribute_name attribute in the element_name element is incorrect. The value is value_name.

Explanation:

The specified data type value for an attribute in the parameter metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI420S The XML structure of the member_name parameter metadata member is not valid. The element_name element is unknown for the overridden Db2 parameter.

Explanation:

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI421S The XML structure of the member_name parameter metadata member is not valid. The element_name element is unknown for the overridden LPAR parameter.

Explanation:

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI422S	The XML structure of the
	<i>member_name</i> parameter
	metadata member is not valid. The
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element is
	unknown for the overridden Db2
	parameter.

Explanation:

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI423S The XML structure of the member_name parameter metadata member is not valid. The attribute_name attribute in the element_name element is unknown for the overridden LPAR parameter.

Explanation:

System action: Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI450S The member_name product parameter metadata member was not found in the data_set_name data set.

Explanation:

Tools Customizer could not find the specified product parameter metadata member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI510W The *data_set_name* data store data set does not exist.

Explanation:

The specified data store data set does not exist.

System action:

Processing continues.

User response:

Ensure that the data store data set exists.

CCQI511S The *data_set_name* data store data set cannot be opened by using the *disposition_type* disposition.

Explanation:

The specified data store data set could not be opened with the specified disposition.

System action:

Processing continues.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI512S The *data_set_name* data store data set cannot be opened by using the *option-type* option.

Explanation:

The specified data store data set was unable to be opened with the specified option.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI600W The XML structure of the member_name product customization parameter

metadata member is not valid. The PL/I XML parser issued the following exception warning code: code number.

Explanation:

While determining if the product customization parameter metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the warning.

CCQI601S

The XML structure of the member_name product customization parameter metadata member is not valid. The PL/I XML parser issued the following exception error code: code_number.

Explanation:

While determining if the product customization parameter metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the warning.

CCQ1602S	The XML structure of the
	<i>member_name</i> product
	customization parameter
	metadata member is not valid. The
	element_name element is
	unknown.

Explanation:

The specified product customization parameter metadata member contains an unknown element.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI603S	The XML structure of the
	<i>member_name</i> product
	customization parameter
	metadata member is not valid.
	Content is not allowed for the
	<i>element_name</i> element, but
	content was found.

Explanation:

Content was found in an element that cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI604S The XML structure of the member_name product customization parameter metadata member is not valid. Content is required for the element_name element, but content was not found.

Explanation:

The specified element does not contain required content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1605S	The XML structure of the
	<i>member_name</i> product
	customization parameter
	metadata member is not valid. The
	content length for the
	<i>element_name</i> element 'cannot
	exceed maximum_number
	characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ1606S

member_name product customization parameter metadata member is not valid. The *element_name* element cannot occur more than *maximum_number* times.

The XML structure of the

Explanation:

The specified element occurs too many times in the product customization parameter metadata member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI6075 The XML structure of the member_name product customization parameter metadata member is not valid. The element_name element must occur at least minimum_number times.

Explanation:

The specified element does not occur enough times in the product customization parameter metadata member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1608S	The XML structure of the
	<i>member_name</i> product
	customization parameter
	metadata member is not valid. The
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element cannot
	occur more than
	maximum_number times.

Explanation:

The specified attribute occurs too many times in the product customization parameter metadata member.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ1609S	The XML structure of the
	<i>member_name</i> product
	customization parameter
	metadata member is not valid. The
	<i>attribute_name</i> attribute in the
	element_name element must occur
	at least <i>minimum number</i> times.

Explanation:

The specified attribute does not occur enough times in the product customization parameter metadata member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI610S The XML structure of the member_name product customization parameter metadata member is not valid. Content is not allowed for the *attribute_name* attribute in the *element_name* element, but content was found.

Explanation:

Content was found in an element that cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI611S The XML structure of the member name product

member_name product customization parameter metadata member is not valid. Content is required for the *attribute_name* attribute 'in the *element_name* element, but content was not found.

Explanation:

The specified attribute does not contain required content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI612S

The XML structure of the member_name product customization parameter metadata member is not valid. The content length for the attribute_name attribute in the element_name element cannot exceed maximum_number characters.

Explanation:

The specified attribute contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI613S The XML structure of the member_name product customization parameter metadata member is not valid. The attribute_name attribute in the

element_name element is unknown.

Explanation:

The specified product customization parameter metadata member contains an unknown attribute.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI614S	The XML structure of the
	<i>member_name</i> product
	customization parameter
	metadata member is not valid. The
	value of the <i>element_name</i>
	element is not valid. The value
	value_name.

Explanation:

The specified value of the element is not a valid value.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

The XML structure of the
<i>member_name</i> product
customization parameter
metadata member is not valid. The
value of the <i>attribute_name</i>
attribute for the <i>element_name</i>
element is not valid. The value is
value_name.

Explanation:

The specified value of the attribute is not a valid value.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1616S	The XML structure of the
	<i>member_name</i> product
	customization parameter
	metadata member is not valid. The
	data type of the <i>element name</i>
	element is 'not valid. The value of
	the element is <i>value name</i> .

Explanation:

The specified data type is not a valid data type.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI617S The XML structure of the member_name product customization parameter metadata member is not valid. The data type of the attribute_name attribute for the element_name element is not valid. The value of the attribute is value_name..

Explanation:

The specified data type is not a valid data type.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

The XML structure of the

CCQI650S

member_name product customization parameter metadata member is not valid. The following value of the *attribute_name* attribute in the *element_name* element already exists: *value_name*.

Explanation:

The specified value for an attribute already exists.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI651S The XML structure of the member_name product customization parameter metadata member is not valid. The parameter_name parameter refers to the following section, which was not found in the member_name product customization parameter metadata member: section-name.

Explanation:

The specified section is not in the product customization parameter metadata member.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1652S	The <i>member_name</i> product
	customization metadata member
	not valid. The default length for
	the <i>element_nαme</i> parameter
	element exceeds the length of the
	parameter. The default length is

parameter. The default length is default_length, and the specified length is specified_length. The default length will be truncated accordingly.

Explanation:

The specified length cannot be shorter than the default length.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI653S	The content of the member name
	product customization parameter
	metadata member is not valid. The
	value of the attribute_name
	attribute in the <i>element_name</i>
	element is not valid. The value of
	the attribute is <i>value_name</i> .

Explanation:

The specified value of the attribute is not a valid value.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1700W	The XML structure of the
	<i>member_name</i> solution pack
	metadata member is not valid. The
	PL/I XML parser issued the
	following exception warning code:
	code_number.

Explanation:

While determining if the specified solution pack metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the warning.

CCQI701S	The XML structure of the
	<i>member_name</i> solution pack
	metadata member is not valid. The
	PL/I XML parser issued the

following exception error code: *code_number*.

Explanation:

While determining if the specified solution pack metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the error.

CCQI702S The XML structure of the member_name solution pack metadata member is not valid. The element_name element is unknown.

Explanation:

The specified solution pack metadata member contains an unknown element.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI703S The XML structure of the member_name solution pack metadata member is not valid. Content is not allowed for the element_name element, but content was found

Explanation:

Content was found in an element that cannot contain content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI704S The XML structure of the member_name solution pack metadata member is not valid. Content is required for the element_name element, but content was not found.

Explanation:

The specified element does not contain required content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI705S The XML structure of the member_name solution pack metadata member is not valid. The content length for the element_name element cannot exceed maximum_number characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI706S The XML structure of the member_name solution pack metadata member is not valid. The element_name element cannot occur more than maximum_number times.

Explanation:

The specified element occurs too many times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI707S The XML structure of the member_name solution pack metadata member is not valid. The element_name element must occur at least minimum_number times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ1708S	The XML structure of the
	<i>member_name</i> solution pack
	metadata member is not valid. The
	attribute name attribute in the
	<i>element_nαme</i> element cannot
	occur more than
	maximum_number times.

Explanation:

The specified attribute occurs too many times.
System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI709S The XML structure of the member_name solution pack metadata member is not valid. The attribute_name attribute in the element_name element must occur at least minimum_number times.

Explanation:

The specified attribute does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI710S The XML structure of the member_name solution pack metadata member is not valid. Content is not allowed for the attribute_name attribute in the element_name element, but content was found.

Explanation:

The specified attribute cannot have content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI711S

The XML structure of the member_name solution pack metadata member is not valid. Content is required for the attribute_name attribute in the element_name element, but content was not found.

Explanation:

The specified attribute is missing content.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI712S The XML structure of the member_name solution pack metadata member is not valid. The content length for the attribute_name attribute in the

element_name element cannot exceed *maximum_number* characters.

Explanation:

The specified attribute contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI713S The XML structure of the member_name solution pack metadata member is not valid. The attribute_name attribute in the element_name element is unknown.

Explanation:

The specified attribute in the solution pack metadata member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI714S	The XML structure of the
	<i>member_name</i> solution pack
	metadata member is not valid
	because the value of the
	<i>element_name</i> element is
	incorrect. The value is
	value_name.

Explanation:

The specified value of the element is not a valid value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI715S The XML structure of the member_name solution pack metadata member is not valid because the value of the attribute_name attribute in the element_name element is incorrect. The value of the attribute is value_name.

Explanation:

The specified value of the attribute is not a valid value.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI716S The XML structure of the member_name solution pack metadata member is not valid because the data type of the element_name element is incorrect. The value is value_name.

Explanation:

The specified data type is not a valid data type.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI717SThe XML structure of the
member_name solution pack
metadata member is not valid
because the data type of the
attribute_name attribute in the
element_name element is
incorrect. The value of the
attribute is value_name.

Explanation:

The specified data type is not a valid data type.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI720S The XML structure of the member_name solution pack metadata member is not valid. The msg element is required for the component_name component that is not customizable.

Explanation:

The msg element is required for the specified component, which cannot be customized by using Tools Customizer.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQI750S The solution pack metadata member was not found in the *library_name* metadata library.

Explanation:

Tools Customizer could not find the solution pack metadata member in the specified library.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ1751S	The version in the <i>library_name</i>
	solution pack metadata library is
	different than the version in the
	<i>library_name</i> component metadata
	library. The name of the pack is
	pack_name, and the name of the
	component is component_name.

Explanation:

The version in the solution pack metadata library does not match the version in the component metadata library.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI752S The release in the *library_name* solution pack metadata library is different than the release in the *library_name* component metadata library. The name of the pack is *pack_name*, and the name of the component is *component_name*.

Explanation:

The release in the solution pack metadata library does not match the release in the component metadata library.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQI753S The modification level in the library_name solution pack metadata library is different than the modification level in the library_name component metadata library. The name of the pack is pack_name, and the name of the component is component_name.

Explanation:

The modification level in the solution pack metadata library does not match the modification level in the component metadata library.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQM002E The command_name line command is not valid: .

Explanation:

The specified line command is not valid.

System action:

Processing continues.

User response:

Specify a valid line command on the panel.

CCQ0000W The XML structure of the member_name discover parameter metadata member is not valid. The PL/I XML parser issued the following exception warning code: code_number.

Explanation:

While determining if the discover parameter metadata member is valid, the PL/I XML parser issued an exception warning code.

System action:

Processing continues.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code.

CCQ0001S The XML structure of the member_name discover parameter metadata member is not valid. The PL/I XML parser issued the following exception error code: code_number.

Explanation:

While determining if the Discover metadata member is valid, the PL/I XML parser issued an exception error code.

System action:

Processing stops.

User response:

See the *Enterprise PL/I for z/OS Programming Guide* for more information about the exception warning code. Contact IBM Software Support.

CCQ0002S	The XML structure of the
	<i>member_name</i> discover parameter
	metadata member is not valid. The
	<i>element_name</i> element is
	unknown.

Explanation:

The specified element in the discover parameter metadata member is unknown.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0003S	The XML structure of the
	member_name discover parameter
	metadata member is not valid.
	Content is not allowed for the
	<i>element_name</i> element, but
	content was found.

Explanation:

The specified element cannot contain content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0004S	The XML structure of the
	member_name discover parameter
	metadata member is not valid.
	Content is required for the element
	name element, but content was
	not found.

Explanation:

The specified element is missing required content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ00055	The XML structure of the
	<i>member_name</i> discover parameter metadata member is not valid. The
	content length for the
	<i>element_name</i> element cannot
	exceed maximum_number
	characters.

Explanation:

The specified element contains too many characters.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0006S The XML structure of the member_name discover parameter metadata member is not valid. The

element_name element cannot	
occur more than	
maximum number times.	

Explanation:

The specified element occurs too many times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0007S The XML structure of the member_name discover parameter metadata member is not valid. The element_name element must occur at least minimum_number times.

Explanation:

The specified element does not occur enough times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0008S The XML structure of the member_name discover parameter metadata member is not valid. The attribute_name attribute in the element_name element cannot occur more than maximum_number times.

Explanation:

The specified attribute occurs too many times.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0009S The XML structure of the member_name discover parameter metadata member is not valid. The attribute_name attribute in the element_name element must occur at least minimum_number times.

Explanation:

The specified attribute does not occur enough times.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

<u>CCQ0010S</u>	The XML structure of the
	<i>member_name</i> discover parameter
	metadata member is not valid.
	Content is not allowed for the
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element, but
	content was found.

Explanation:

The specified attribute cannot contain content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

The XML structure of the
member_name discover parameter
metadata member is not valid.
Content is required for the
attribute_name attribute in the
<i>element_name</i> element, but
content was not found.

Explanation:

The specified attribute requires content.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ00125	The XML structure of the
	<i>member_name</i> discover parameter
	metadata member is not valid. The
	content length for the
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element in the
	cannot exceed maximum_number
	characters.

Explanation:

The specified attribute contains too many characters.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ00135	The XML structure of the
	<i>member_name</i> discover parameter
	metadata member is not valid. The
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element is
	unknown.

Explanation:

The specified attribute is unknown.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0014S The content of the *member_name* discover parameter metadata member is not valid because the value of the *element_name* element is incorrect. The value is *value_name*.

Explanation:

A The specified value for an element in the discover parameter metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0015S

The content of the *member_name* discover parameter metadata member is not valid because the value of the *attribute_name* attribute in the *element_name* element is incorrect. The value is *value_name*.

Explanation:

The specified value for an attribute in the discover parameter metadata member is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0016S The content of the member_name discover parameter metadata member is not valid because the data type of the *element_name* element is incorrect. The value is *value_name*.

Explanation:

The specified data type value for an element in the discover parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0017S The content of the *member_name* product parameter metadata member is not valid because the

data type of the *attribute_name* attribute in the *element_name* element is incorrect. The value is *value_name*.

Explanation:

The specified data type value for an attribute in the product parameter metadata member is not valid.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0050S The *data_set_name* Discover REXX EXEC data set could not be initialized or was not found.

Explanation:

Tools Customizer could not find or could not initialize the specified Discover REXX EXEC data set.

System action:

Processing stops.

User response:

Ensure that the Discover REXX EXEC is specified correctly.

CCQ0051W	The data_sharing_group_ID data
	sharing group ID cannot contain
	more than four characters.

Explanation:

The specified data sharing group ID contains too many characters.

System action:

Processing continues.

User response:

Ensure that the specified data sharing group ID does not exceed four characters.

CCQ0052S The REXX_EXEC_name Discover REXX EXEC was not found in the data_set_name Discover data set.

Explanation:

Tools Customizer could not find the Discover REXX EXEC in the specified data set.

System action:

Processing stops.

User response:

Ensure that the Discover data set was specified correctly.

CCQ0053W	The LPAR_name LPAR name
	cannot contain more than eight
	characters.

Explanation:

The specified LPAR name contains too many characters.

System action:

Processing continues.

User response:

Ensure that the specified LPAR name does not exceed eight characters.

CCQ0054W	The <i>subsystem_ID</i> Db2 SSID
	cannot contain more than four
	characters. The record was not
	processed.

Explanation:

The specified Db2 SSID contains too many characters.

System action:

Processing continues.

User response:

Ensure that the specified Db2 SSID does not exceed four characters.

CCQ0055W The *parameter_name* Db2 group attach name parameter is in the *record_name* Discover record, but a Db2 group attach name was not specified. The record was not processed.

Explanation:

The Discover record contains a data sharing group parameter, but a Db2 group attach name was not specified.

System action:

Processing continues.

User response:

Ensure that information is specified correctly on the **Discover Customized Product Information** panel.

CCQ0056W	The parameter_name Db2
	parameter in the record_name
	Discover record did not have a Db2
	group attach name or a Db2 SSID.
	The record was not processed.

Explanation:

The Discover record did not have a Db2 group attach name or a Db2 subsystem ID in the Db2 parameter.

System action:

Processing continues.

User response:

Ensure that information is specified correctly on the **Discover Customized Product Information** panel.

CCQ0057W The Discover EXEC could not find the *parameter_name* parameter in the metadata for the product to be

customized. The record was not processed.

Explanation:

The specified parameter could not be found in the metadata for the product to be customized.

System action:

Processing continues.

User response:

Ensure that information is specified correctly on the **Discover Customized Product Information** panel.

CCQ0058W The parameter_name product parameter name in the record_type Discover record does not start with CCQ_LPR_, CCQ_DB2_, or CCQ_PRD_. The record was not processed.

Explanation:

The parameter in the record does not start with CCQ_DB2_, CCQ_LPAR_, or CCQ_PRD_.

System action:

Processing continues.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0059W	The <i>parameter_name</i> product
	parameter cannot contain more
	than 72 characters. The record
	was not processed.

Explanation:

The specified product parameter contains too many characters.

System action:

Processing continues.

User response:

Ensure that the specified product parameter does not exceed 72 characters.

CCQ0060W	The record_name Discover record
	from the REXX EXEC output must
	start with the following record
	type: <i>record_type</i> . The record was
	not processed.

Explanation:

A Discover record from the REXX EXEC output must start with the specified Db2 record type.

System action:

Processing continues.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0061I

If you do not have a previously customized version of the product, do not run the Discover EXEC. Press END to go to the Customizer Workplace panel.

Explanation:

This message is issued when you customize a product for a the first time. It prompts you to use the Discover EXEC to discover data from a previous customization of the specified product.

System action:

Processing continues.

User response

Tip: Using the Discover EXEC saves time and reduces errors that can error when parameters are specified manually. If you want to use the Discover EXEC, specify the required information on the **Discover Customized Product Information** panel. Otherwise, press End to continue without discovering data from a previous customization of the product.

CCQ0062W The Discover EXEC could not find the following *parameter_name* parameter in the Db2 metadata. The record was not processed.

Explanation:

The specified parameter is missing in the Db2 metadata.

System action:

Processing continues.

User response:

If this parameter is required, contact IBM Software Support.

CCQ0064W

The *Discover-record* Discover record did not have a parameter name. The record was not processed.

Explanation:

A parameter name was missing in the Discover record.

System action:

Processing continues.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0065W The value for the parameter_name parameter is ignored because it has more than maximum_number characters, which is the maximum length that is defined in the metadata. The value is parameter_value.

Explanation:

The specified value exceeded the maximum allowed length, which was defined in the metadata. Tools Customizer truncated the extra characters.

System action:

Processing continues.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQ0066W The *record_name* Discover record from the Discover REXX EXEC output does not have a parameter value. The record was not processed.

Explanation:

The Discover record was missing a parameter value from the Discover EXEC output.

System action:

Processing continues.

User response:

Ensure that information was specified correctly on the **Discover Customized Product Information** panel.

CCQ0067W The *parameter_name* parameter is defined in the metadata to support one value, but more than one value was found. The last value was used.

Explanation:

The definition of the parameter in the metadata supports one value, but more than one value was specified. Only the last value was used.

System action:

Processing continues.

User response:

Ensure that information was specified correctly on the **Discover Customized Product Information** panel.

CCQ0068W	The value of the <i>parameter_name</i>
	parameter is ignored because the
	parameter is defined as
	internal=true. The value is
	value_name.

Explanation:

The specified value of the parameter is ignored because it is defined as internal=true.

System action:

Processing continues.

User response:

Ensure that information was specified correctly on the **Discover Customized Product Information** panel.

CCQ0069W

The Discover EXEC did not find the *parameter_name* parameter in the LPAR metadata. The record was not processed.

Explanation:

The specified parameter is missing from the LPAR metadata.

System action:

Processing continues.

User response:

Ensure that information was specified correctly on the **Discover Customized Product Information** panel.

CCQ0070W The *record_type* Discover record contains an incorrect delimiter between the Environment section and the Data section. The record was not processed.

Explanation:

Tools Customizer found an incorrect delimiter between the Environment section and the Data section.

System action:

None.

User response:

No action is required.

CCQ0071W The *member_name* member could not be found in the *data_set_name* Discover data set.

Explanation:

Tools Customizer could not find the specified Discover data set.

System action:

None.

User response:

No action is required.

CCQ0072S The *member_name* discover metadata member was not found in the *data_set_name* metadata data set.

Explanation:

Tools Customizer could not find the specified metadata member in the data set.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0073E The *member_name* discover metadata member is not valid because the default length for the *element_name* parameter element exceeds the length of the parameter. The default length is *default_length*, and the specified length is *specified_length*. The default length will be truncated accordingly.

Explanation:

The default length for the specified parameter element is longer than the parameter.

System action:

Processing continues.

User response:

No action is required.

CCQ0074S	The content of the <i>member_name</i>
	discover metadata member is not
	valid. The value of the
	<i>attribute_name</i> attribute in the
	<i>element_name</i> element is not
	valid. The value of the attribute is
	value_name.

Explanation:

The specified value is not valid.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

```
CCQ0075W The configuration_ID configuration
ID in the record_name Discover
record is incorrect. The record was
not processed.
```

Explanation:

The specified configuration ID is not correct.

System action:

Processing continues.

User response:

No action is required.

CCQ0076W The configuration_ID configuration ID cannot contain more than maximum_number characters. The record was not processed.

Explanation:

The specified configuration ID contains too many characters.

System action:

Processing continues.

User response:

No action is required.

CCQ0077S

The discover metadata member was not found in the *data_set_name* component data set that is part of the *data_set_name* pack.

Explanation:

The discover metadata member was not found in the specified component data set.

System action:

Processing stops.

User response:

See "Gathering diagnostic information" on page 217. Contact IBM Software Support.

CCQ0080I *Product_name* does not support the Discover process.

Explanation:

The specified product does not support the Discover process.

System action:

None.

User response:

No action is required.

CCQP000E The value of the *mode_name* Db2 mode is not valid for the *level_name* Db2 level.

Explanation:

The specified Db2 mode is not valid for the Db2 level.

System action:

Processing stops.

User response:

Specify a valid Db2 mode for the Db2 level.

CCQP001E The value of the *mode_name* Db2 mode is missing.

Explanation:

The specified Db2 mode is not defined.

System action: Processing stops.

User response:

Specify a value for the Db2 mode.

CCQP002E The value of the *mode_name* Db2 level is missing.

Explanation:

The specified Db2 level is not defined.

System action:

Processing stops.

User response: Specify a value for the Db2 level.

CCQP003E The value of the *level_name* Db2 level is not valid.

Explanation:

The specified Db2 level does not have a valid name.

System action:

Processing stops.

User response:

Specify a valid value for the Db2 level.

CCQP004S The *parameter_name* parameter does not exist in the CCQ\$\$DB2 Db2 parameter metadata member.

Explanation:

The CCQ\$\$DB2 Db2 parameter metadata member does not contain the specified parameter.

System action:

Processing stops.

User response:

See <u>"Gathering diagnostic information" on page 217</u>. Contact IBM Software Support.

CCQP005E The value of the *subsystem_ID* Db2 SSID is missing.

Explanation:

The specified Db2 SSID is not defined.

System action:

Processing stops.

User response:

Specify a valid value for the Db2 SSID.

CCQP006E The value of the

group_attach_name Db2 group attach name is missing.

Explanation:

The specified Db2 group attach name is not defined.

System action:

Processing stops.

User response:

Specify a valid Db2 group attach name.

CCQQ000E Specify a valid metadata library. Each qualifier of the library must start with an alphabetic character and must be 1-8 alphanumeric characters. The library name must be 1-44 characters.

Explanation:

The metadata library was not specified in the correct format. The high-level qualifier must contain alphanumeric characters, and the first character cannot be numeric. The name cannot contain wildcard characters, such as asterisks (*) and percent signs (%).

System action:

Tools Customizer prompts for the correct library name.

User response

Specify a library in the correct format. If the message was issued on the **Specify the Metadata Library** (CCQPHLQ) panel, specify the product metadata library. The name of this library is *hlq*.SANLDYXDENU.

Do not specify the Tools Customizer metadata library, which is *hlq*.SCCQDENU.

CCQQ001E The *data_set_name* data set name that was specified for the metadata library was not found.

Explanation:

The data set does not exist, or the data set name was written in the incorrect format. The high-level qualifier must contain alphanumeric characters, and the first character cannot be numeric. The name cannot contain wildcard characters, such as asterisks (*) and percent signs (%).

System action:

Tools Customizer prompts for the correct data set name.

User response:

Specify a data set name in the correct format.

CCQQ002E The data set name that was specified for the *library_name* metadata library cannot be opened.

Explanation:

Tools Customizer could not open the data set.

System action:

Tools Customizer prompts for an available data set.

User response:

Ensure that the specified data set is available for Tools Customizer to open it.

CCQQ003E The data_set_name data set name that was specified for the metadata sample library is not valid. The data set must be in the following format: HLQ.SxxxSAMP.

Explanation:

The specified data set name was not specified in the correct format.

System action:

None.

User response:

Specify the data set name in the following format: HLQ.SxxxSAMP, where xxx is the three-character prefix for the product.

CCQQ004E

The *data_set_name* data set is being used by another user. Try again when the data set is not being used.

Explanation:

Another user is using the specified data set.

System action:

None.

User response:

Ensure that the specified data set is not being used.

CCQQ009E The *data_set_name* data set name that was specified for the metadata library is not valid because the data set is empty.

Explanation:

The specified data set is empty.

System action:

Tools Customizer prompts for an available data set.

User response:

Ensure that the specified data set is available for Tools Customizer to open it.

CCQQ011E	The <i>library_name</i> metadata library
	for the component that is part of
	the <i>library_name</i> pack was not
	found in the catalog. The name of
	the pack is <i>pack_name</i> , and the
	name of the component is
	component_name.

Explanation:

The specified metadata library is not in the catalog.

System action:

None.

User response:

Specify another metadata library.

CCQQ012E	The <i>library_name</i> metadata library
	for the component that is part of
	the <i>library_name</i> pack cannot be
	opened.

Explanation:

The specified metadata library cannot be opened.

System action:

None.

User response:

Ensure that the name of the library is specified correctly.

CCQS000I	Tools Customizer is being invoked
	for the first time or the previous
	ISPF session ended before Tools
	Customizer was exited. In both

cases, the fields on this panel are populated with default values. Review these default values or specify new values to be used to customize products or packs.

Explanation:

When you customize a stand-alone product or a solution pack for the first time, or when an ISPF session unexpectedly ends before the ISPF profile is saved, you must specify or review your Tools Customizer user settings.

System action:

Processing stops.

User response:

Review and accept the default settings, or specify new settings.

CCQS001E The following command is not valid: command_name.

Explanation:

The specified command is not a valid command on the panel.

System action:

Processing stops.

User response:

Specify a valid command.

CCQS002W The *data_set_name* Discover data set could not be found.

Explanation:

Tools Customizer could not find the specified data set.

System action:

The data set will be allocated, and processing continues.

User response:

Ensure that the data set name is specified correctly because the data set will be allocated with this name after the values are saved.

CCQS003W	The <i>data_set_name</i> Discover data
	set was not found so it was
	created.

Explanation:

Tools Customizer could not find the specified data set.

System action:

Processing continues.

User response:

Ensure that the data set name is specified correctly.

CCQS004I The settings were saved.

Explanation:

The settings that you changed were saved.

System action:

Processing continues.

User response:

No action is required.

CCQS006W The length of a qualifier for the data_set_name customization library data set exceeds 26 characters.

Explanation:

The qualifier for the customization library data set is too long. The qualifier cannot exceed 26 characters.

System action:

Processing continues.

User response:

Specify a qualifier that is 26 characters or less.

CCOS007E	The discover data set
-	data_set_name could not be
	opened with the option-type
	option.

Explanation:

The specified option could not open the Discover data set.

System action:

None.

User response:

Specify a data set to which you have WRITE access.

CCQS008E	The Discover data set
	<i>data_set_name</i> exists on a
	different volume.

Explanation:

The specified Discover data set must exist on the same volume as where it was created.

System action:

Processing continues.

User response:

Specify a different Discover data set name.

CCQS010E	The customization library qualifier
	is not valid.

Explanation:

The customization library qualifier that was specified is not valid.

System action:

None.

User response:

Specify a valid qualifier for the customization library.

CCQS011E The group attach option is not valid.

Explanation:

The group attach option that was specified is not valid.

System action:

None.

User response:

Specify a valid option for the group attach option.

CCQS012E The Tools Customizer metadata library is not valid.

Explanation:

The metadata library that was specified is not a valid data set.

System action:

None.

User response:

Specify a valid data set for the metadata library.

CCQS013E The Discover data set is not valid.

Explanation:

The Discover data set that was specified is not a valid data set.

System action:

None.

User response:

Specify a valid Discover data set.

CCQS014E The data store data set is not valid.

Explanation:

The data set that was specified is not a valid data set.

System action:

None.

User response:

Specify a valid data store data set.

CCQS015E	Tools Customizer is already
	running.

Explanation:

A session of Tools Customizer is already running in your environment. Only one Tools Customizer session is allowed.

System action:

None.

User response:

The trace data set is being used. Free the trace data set, and start Tools Customizer again.

CCQS018E Information on the first line of the job card exceeds 57 characters.

Explanation:

The first line of the job card can contain only 57 characters. This character limit includes a continuation character.

System action:

Tools Customizer clears the first line of the job card.

User response:

Specify information that does not exceed 57 characters on the first line of the job card.

CCQS019E The required trace data set, data_set_name, is currently not accessible.

Explanation:

The trace data set must be accessible.

System action:

Processing stops.

User response:

Ensure that the trace data set is accessible.

An error occurred while the
customization library data set was
being created. ALTER authority on
the high-level qualifier for the
customization library data set is
required.

Explanation:

To create the customization library data set, ALTER authority on the specified high-level qualifier must be granted.

System action:

None.

User response:

Ensure that ALTER authority for the specified customization library data set is granted.

CCQS021EThe value value_name in the field
that contains the cursor position is
not valid.

Explanation:

The specified value is not valid.

System action:

None.

User response:

Specify a valid value.

CCOS022E An error occurred while the customization library data set was being opened. UPDATE authority on the high-level qualifier for the customization library data set is required.

Explanation:

To open the customization library data set, UPDATE authority on the specified high-level qualifier must be granted.

System action: None.

User response:

Ensure that UPDATE authority for the specified customization library data set is granted.

CCQS023E An error occurred while the customization library data set was being opened. UPDATE authority on the high-level qualifier for the customization library data set is required.

Explanation:

To open the customization library data set, UPDATE authority on the specified high-level qualifier must be granted.

System action:

None.

User response:

Ensure that UPDATE authority for the specified customization library data set is granted, or specify a different high-level qualifier for the customization library data set on the **Tools Customizer Settings** panel.

CCQS024E An error occurred while the customization library data set was being created. ALTER authority on the high-level qualifier for the customization library data set is required.

Explanation:

To create the customization library data set, ALTER authority on the specified high-level qualifier must be granted.

System action:

None.

User response:

Ensure that ALTER authority for the specified customization library data set is granted, or specify a different high-level qualifier for the customization library data set on the **Tools Customizer Settings** panel.

CCQS025I The display options were saved.

System action:

None.

User response:

No action is required.

CCQS030E The following command is not a valid CREATE statement: command_statement.

Explanation:

The specified CREATE command statement is invalid because it contains blanks or alphabetic characters.

System action:

Processing stops.

User response:

Specify a valid CREATE command statement. The correct syntax is CREATE *nn*, where *nn* is 1 - 99.

CCQS031E The following command is not a valid CREATE statement: *command_statement*. The number that can be specified with the CREATE command is 1 - 99.

Explanation:

The specified CREATE command statement is invalid because it contains either 0 or a number greater than 99.

System action:

Processing stops.

User response:

Specify a valid CREATE command statement. The correct syntax is CREATE *nn*, where *nn* is 1 - 99.

The product configuration ID
copied_configuration_ID was
successfully copied from
configuration_ID.

Explanation:

The specified configuration ID was copied.

System action:

None.

User response:

No action is required.

CCQT001E	The command_name line
	command was specified more than
	once, which is not allowed.

Explanation:

The specified line command cannot be specified more than one time.

System action:

Processing stops.

User response:

Specify the line command only once.

CCQT002E The configuration_ID configuration ID already exists. Specify a different configuration ID.

Explanation:

The specified configuration ID exists.

System action:

Processing stops.

User response:

Ensure that the specified configuration ID is unique.

CCQT003I	The product configuration ID
	configuration_ID was created.

Explanation:

The specified configuration ID was created.

System action:

None.

User response:

No action is required.

CCQT004I The product configuration ID configuration_ID was removed.

The specified configuration ID was removed.

System action:

Explanation:

None.

User response:

No action is required.

CCQT005E The product configuration ID configuration_ID is not valid. The product configuration ID cannot contain a colon (:).

Explanation:

The specified configuration ID contains a colon (:), but a colon is not valid.

System action:

Processing stops.

User response:

Specify a configuration ID that does not contain a colon.

CCQT006E The *configuration_ID* configuration ID exists. Specify a different configuration ID.

Explanation:

The specified configuration ID exists.

System action:

Processing stops.

User response:

Specify another configuration ID.

CCQT007E The configuration_ID configuration ID exists but was removed from the list of configurations. To use this configuration ID, you must restore it.

Explanation:

The specified configuration ID exists but was removed from the list of available configuration.

System action:

Processing stops.

User response:

Specify another configuration ID. To restore the specified configuration ID, issue the CREATE command, and specify the same configuration ID again.

CCQT008E The configuration_ID configuration ID exceeds maximum_number characters.
Explanation: The specified configuration ID contains too many characters.
System action: Processing stops.
User response: Specify another configuration ID that does not exceed the maximum number of characters that was set by Db2 SQL Performance AnalyzerDb2 Autonomics Director.
CCQT010I Create request for configuration_ID configuration was cancelled by user.
Explanation: The request to create the specified configuration was canceled.
System action: Processing stops.
User response: No action is required.
CCQT011I The <i>configuration_ID</i> configuration was not copied.
Explanation: The specified configuration was not copied.
System action: Processing stops.
User response: No action is required.

CCQT012I The *configuration_ID* configuration was not removed.

Explanation:

The specified configuration was not removed.

System action:

Processing stops.

User response:

No action is required.

CCQT013I None of the configurations were copied or removed. All of the previously selected configurations are deselected.

Explanation:

The selected configurations were not copied or removed, and they are deselected.

System action:

Processing stops.

User response:

No action is required.

CCQT014E Specify Y or N and press Enter to continue, or press End to cancel.

Explanation:

A function requires input.

System action:

Processing stops.

User response:

To continue, specify Y or N and press Enter. Otherwise, press End to cancel.

CCQT015E The command_name command is not allowed during the process of "Select" configuration line command.

Explanation:

The specified command is not allowed while the line command for selecting configurations is processing.

System action:

Processing stops.

User response:

Remove the specified line command.

CCQT016I The *configuration_ID* configuration was not created

Explanation:

The specified configuration was not created.

System action:

Processing stops.

User response:

No action is required.

The *configuration_ID* configuration was not copied.

Specify Y or N, and press Enter.

Explanation:

CCOT017I

The specified configuration was not copied.

System action:

Processing stops.

User response:

No action is required.

CCQT018E

Explanation:

A function requires input.

System action:

Processing stops.

User response:

To continue, specify Y or N, and press Enter.

CCQT019I	The select configuration ID
•	configuration process ended.

Explanation:

The select process for the specified configuration is finished.

System action:

Processing stops.

User response:

No action is required.

CCQT020E The *configuration_ID* configuration was not created because the data store was not accessible.

Explanation:

The specified configuration was not created because the data store could not be accessed.

System action:

Processing stops.

User response:

Ensure that the data store is accessible and create the configuration again.

CCQT021E The configuration_ID configuration was not copied because the data store was not accessible.

Explanation:

The specified configuration was not copied because the data store could not be accessed.

System action:

Processing stops.

User response:

Ensure that the data store is accessible and copy the configuration again.

CCQT025I The *configuration_ID* configuration was not updated.

Explanation:

The specified configuration was not updated because the edit process was canceled.

System action:

Processing stops.

User response:

No action is required.

CCQT027I The product configuration ID has been updated from *edit_from_id* to *edit_to_id*.

System action:

Processing continues.

User response:

No action is required.

CCQX001S Product_name has already been customized by using values from data_set_name data store data set. Switch to the specified data

store data set to continue customizing this product.

Explanation:

The specified product was customized by using values from the specified data store data set.

System action:

Processing stops.

User response:

Use the specified data store data set to continue customizing the product.

CCQX002S component_name has already been customized by using values from data_set_name data store data set. Switch to the specified data store data set to continue customizing this component.

Explanation:

The specified component was customized by using values from the specified data store data set.

System action:

Processing stops.

User response:

Use the specified data store data set to continue customizing the component.

CCQX011I Product_name was not found.

Explanation:

The specified product was not found.

System action: Processing stops.

User response: Specify another product.

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 US

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 US 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 discussed herein is presented as derived under specific operating conditions. Actual results may vary.

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.

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.

Index

A

access path 197, 203 access path hint 144 accessed table information 197 accessibility overview 14 acctype subparameter 142 ACTION 88, 89 active pages 203 address space startup procedure 79 stored procedure 79 ADVISOR user parameter 230, 233 ALLindex 140 ANLCNTL customizing parameters 80 ANLCNTL configuration parameters AUTHIDO 226 BUFF08K 226 BUFF16K 226 BUFF32K 226 **BUFFERS 226** COMMENT 226 COSTING 226 COSTQUN 226 CPUCOST 227 **CPUTIME 227** CURRSYM 227 DATAGRP 227 DSGROUP 227 DYNAMIC 227 ELAPSED 227 FASTTBL 227 **IOCALLS 228 IOSCOST 228** MONEYIS 228 SETPLAN 228 SUBSYST 228 SUBVERS 229 TIMCOST 229 **USRPARM 229** ANLCNTL host parameter file 91, 127 ANLCNTL parameter configuring 130 ANLCOST.LOG file 96 ANLCSPA 78 ANLCSPA edit macro in TSO 103 ANLDEBUG 92 ANLI command in TSO 93 ANLIN file 91, 95 ANLINnn file 91 ANLOUT file 91 ANLPARM file 129 ANLPARM user parameter file 91, 127 ANLPARM user parameters ADVISOR 230, 233

ANLPARM user parameters (continued) **BUFFHIT 230 BYPASSRC 238** COMMENT 230 CONNECT 230 CSVDCSP 231 CSVDLIM 231 CSVHDRW 231 CSVRPT 231 CSVSEL 231 DBRMKEY 231 DEGREES 232 DELIMIT 232 DSPPARM 233 DSPVARS 233 DSPWCOC 233 HPOOLRD 233 **INLISTS 233 ISCANPG 233 IXUPDAT 233** JOINTAB 233 **KEEPLAN 234** MATCOLS 234 NEWROTA 234 NEWSEEK 234 NEWSTOR 234 NEWXFER 234 NLSCODE 234 NONIXPG 235 NOSTATS 235 NUMBERS 235 **OBJECTS 235** PKGDRDA 235 PRECISE 235 PREDPCT 236 PROCESS 236 **QLIMSQL 236** OUALIFY 236 **REPORTS 237** RETCODE 237 **RETPLAN 238** SHOWALT 238 STORAGE 238 **TSCANPG 238 USEPLAN 239** VERSION 239 VIADRDA 239 ANLPRCR stored procedure CALL parameters 80 using 119 ANLPRER stored procedure CALL parameters 80 using 119 ANLREP file 91 ANLSQLPA QMF intercept 9 SQL parser routine 8

ANLSRSET SQL <u>213</u> ANLSTER program <u>120</u> ANLSTPR program <u>120</u> ANLWIF analyzing indexes <u>106</u> restoring indexes <u>112</u> restoring tables <u>112</u> ANLWORK file <u>91</u> apply jobs setting up <u>112</u> applying catalog statistics <u>114</u> AUTHIDO configuration parameter <u>226</u> authorization ID <u>149, 246</u> authorization table <u>242</u>

В

batch 12 batch JCL 87, 91 batch SQL input, ANLIN file 91 BIND information 197 browsing reports in TSO 114 BUFF08K configuration parameter 226 BUFF16K configuration parameter 226 BUFF32K configuration parameter 226 BUFFERS configuration parameter 226 BUFFHIT user parameter 230 BYPASSRC user parameter 238

С

call attachment facility 9 canceling queries in QMF 118 catalog access 3 catalog statistics applying 114 collecting 113 job set up 112 Change CSV report options panel 97 Change user parameters panel 96 changing display options 45 charge back cost estimates 155 cluster matching index scan info in Enhanced Explain report 160 cluster nonmatching index scan info in Enhanced Explain report 162, 164 clustered 203 clustering 203 codes return 249 collect jobs setting up 112 collecting catalog statistics 113 COLLID 88, 89 commands examples of invalid 135 examples of valid 135 LASTREPORTS 103 LEVEL 135 LVL 135 package 133 PKG 133

commands (continued) Plan 133 PLN 133 **QMF 133** QNO 133 QUERYNO 133 SET 142 SOL 133 COMMENT configuration parameter 226 COMMENT user parameter 230 comments 135 compare packages 99 COMPARE 88 COMPARE for batch processing 88 comparing packages in batch 88 CONdensed 141 condensed report reference Explain 208 configuration file target host for stored procedure 79 configuration parameters ANLCNTL 226 **BUFF08K 226** BUFF16K 226 BUFF32K 226 **BUFFERS 226** COMMENT 226 COSTING 226 COSTOUN 226 CPUCOST 227 **CPUTIME 227** CURRSYM 227 DATAGRP 227 DSGROUP 227 DYNAMIC 227 ELAPSED 227 FASTTBL 227 **IOCALLS 228** configuring Plan Table tables EEEAUTH 242 EEEDBRM 243 EEEHINT 243 EEEPATH 245 **EEEPLAN 244** EEEWORK 242 CONNECT user parameter 230 cookie policy 403 Copy DB2 Entries panel 67 correlated and noncorrelated query info in Enhanced Explain report 170 cost estimates, influenced by parameters 127 Cost Summary report in batch 91 in TSO 96 overview 153 warning messages 155 COSTING configuration parameter 226 COSTQUN configuration parameter 226 CPU time estimates in Detail Trace report 183 CPUCOST configuration parameter 227 CPUTIME configuration parameter 227

CSV report changing options 97 CSV report spreadsheet 211 CSVDCSP user parameter 231 CSVDLIM user parameter 231 CSVHDRW user parameter 231 CSVRPT user parameter 231 CSVSEL user parameter 231 CURRSYM configuration parameter 227 CURSQLID 88, 89 CURVERS 88 customization associated list adding Db2 entries 53 overview 221 associating Db2 entries 53 browsing parameters 67 changing display options 45 changing parameters 46 component 221 copying Db2 entries 67 Create a DB2 Entry panel 53 creating Db2 entries 53 customization jobs deleting 70 displaying 69 generating 64 maintaining 70 regenerating 64 renaming 70 sort sequence 64 submitting 64, 69 customization library deleting jobs 70 maintaining 70 overview 224 recustomizing 70 renaming jobs 70 customization library qualifier specifying 42 Customized status 221 Customizer Workplace panel 64 customizing a new version of a product 46 customizing a product for the first time 46 customizing settings 42 data sets customization library 224 data store 224 Discover EXEC library 224 metadata library 224 data store overview 224 data store data set specifying 42 Db2 data sharing members adding 53 associating 53 copying 67 creating 53 Db2 entries adding 53 associating 53 copying 67 creating 53

customization (continued) Db2 entries (continued) defining 64 deleting 68, 69 generating jobs for 64 removing 68 selecting 64 specifying 64 unassociating 68, 69 Db2 group attach field specifying 42 Db2 group attach names adding 53 associating 53 copying 67 creating 53 Db2 parameters defining 61 editing 61 DB2 Parameters panel 61 Db2 subsystems adding 53 associating 53 copying 67 creating 53 defining Db2 parameters 61 defining LPAR parameters 59 defining parameters 55, 64 defining product parameters 56 deleting Db2 entries 69 deleting jobs 49 **Discover Customized Product Information panel 52 Discover EXEC** customizing a new version of a product 46, 48 overview 224 retrieving product information automatically 52 Discovered status 221 discovering previous versions 48 discovering product information 52 display options 45 displaying jobs 69 displaying panel text 45 editing LPAR parameters 59 editing parameters 46 editing product parameters 56 Errors in Customization status 221 finding trace data set 217 Finish Product Customization panel 64 first-time 47 first-time customization 46 generating jobs 64 hiding panel text 45 high-level qualifier 221 Incomplete status 221 job sort order 64 jobs deleting 70 displaying 69 maintaining 70 renaming 70 sort order 64 submitting 64, 69 LPAR parameters defining 59

customization (continued) LPAR parameters (continued) editing 59 LPAR Parameters panel 59 LPARs 70 maintaining jobs 70 master list adding Db2 entries 53 Associate DB2 Entry for Product panel 53 overview 221 maximizing information panels 45 metadata libraries specifying 50 metadata library overview 224 specifying 42 modifying parameters 46 modifying settings 42 multiple instances 42 multiple-LPAR environment 70 Not Required status 221 options 45 panel display options 45 panels Associate DB2 Entry for Product 53 Create a DB2 Entry 53 Customizer Workplace 64 DB2 Parameters 61 **Discover Customized Product Information 52** Finish Product Customization 64 LPAR Parameters 59 Product Parameters 56 Specify the Metadata Library 50 parameters browsing 67 defining 55, 64 viewing 67 preparing to use Tools Customizer 42 product 221 product parameters changing 49 defining 56 editing 49, 56 modifying 49 Product Parameters panel 56 Ready to Customize status 221 recustomization 46, 49 recustomizing 49 recustomizing a product 46 removing Db2 entries 68 roadmaps customizing for the first time 47 first-time customization 47 reustomizing 49 Specify the Metadata Library panel 50 specifying data sets 42 specifying metadata libraries 50 starting Tools Customizer 41 status types Customized 221 Discovered 221 Errors in Customization 221 Incomplete 221 Not Required 221

customization (continued) status types (continued) Ready to Customize 221 submitting jobs 64 terminology 221 trace data set 217 troubleshooting finding trace data set 217 user job card settings specifying 42 viewing parameters 67 customization library overview 224 customization library qualifier specifying 42 customizing Db2 SQL Performance Analyzer 127 customizing settings 42

D

data definition 203 data store overview 224 data store data set specifying 42 DATAGRP 124 DATAGRP configuration parameter 227 Db2 12 function level support 1 DB2 Administration Tool interface 77 DB2 call attachment facility 9 DB2 catalog 8 DB2 catalog access 3 DB2 catalog tables 133 DB2 Explain 8 Db2 group attach field specifying 42 DB2 release 203 Db2 SQL Performance Analyzer generic plan tables introduction 2 performance features 2 stored procedure calling 80 customizing parameters 80 Db2 SQL Performance Analyzer panels main control panel 95 welcome panel 93 Db2 SQL Performance Analyzer stored procedure using 119 DBRM information 197, 211 DBRMKEY parameter 91 DBRMKEY user parameter 231 **DBRMLIB 89** defaults data set 150 definition of data 203 DEGREES user parameter 232 delete info in Enhanced Explain report 174 **DELIMIT** user parameter 232 DETail 140 Detail trace report 187 Detail Trace report 91, 177 diagnostic information gathering 213, 217 **Discover EXEC**

Discover EXEC (continued) overview 224 display options 45 displaying panel text 45 documentation accessing 13 sending feedback 13 documentation changes 1 DRDA 3 DSGROUP configuration parameter 227 DSN3@ATH exit secondary authorization IDs 9 DSN3@SGN exit secondary authorization IDs 9 DSNSPAS startup procedure 79 DSPPARM user parameter 233 DSPVARS user parameter 233 DSPWCOC 5 DSPWCOC user parameter 233 **DYNAMIC configuration parameter 227** dynamic explain 10

E

ELAPSED configuration parameter 227 eloc subparameter 143 Enhanced Explain report description 156 in batch 91 in TSO 95 Explain condensed report reference 208 tabular report reference 205 Explain report 197, 203

F

FASTTBL <u>4</u>, <u>131</u> FASTTBL configuration parameter <u>227</u> FETCH FIRST n ROWS ONLY clause <u>180</u> first subparameter <u>143</u> first-time customization <u>47</u> force subparameter <u>144</u> format subparameter <u>144</u> FROMCATLG 89

G

gen subparameter <u>144</u> generating batch JCL for SQL PA <u>91</u> generation specification <u>200</u> generic names <u>200</u> generic plan tables creating <u>71</u> introduction <u>2</u> resetting the REGISTRY 213

Η

hiding panel text <u>45</u> hint specification <u>144</u> hint table 243 host configuration file <u>79</u> host variable <u>203</u> Host Variables report <u>195</u> hostvar subparameter <u>145</u> HPOOLRD user parameter <u>233</u>

I

I/O Estimate 154 I/O estimates in Detail Trace report 182 import to spreadsheet 211 index analysis 106 index information 197 index on PLAN_TABLE 141 index subparameter 135, 145 indexes creating 108 dropping 108 removing virtual changes 112 indicator variable 203 influence access paths 106 **INLISTS user parameter 233** Input command requirements 134 input specifications 135 input to Db2 SQL Performance Analyzer 95 input to SQL PA 91 insert info in Enhanced Explain report 174 installation limits report 191 Installation Limits report 91 invalid commands 135 invoking SQL PA in TSO 93 **IOCALLS configuration parameter 228** IOSCOST configuration parameter 228 ISCANPG user parameter 233 IXUPDAT user parameter 233

J

JCL for batch processing <u>87</u>, <u>91</u> JCL, ANLDEBUG <u>92</u> JOINTAB user parameter 233

Κ

KEEPLAN user parameter 234 key columns information 197 KEYdist 141

L

last subparameter <u>143</u> LASTREPORTS <u>103</u> legal notices cookie policy <u>403</u> notices <u>403</u> programming interface information <u>403</u> trademarks <u>403</u> level commands ALLindex <u>135</u>, <u>140</u> CONdensed <u>141</u> detail <u>135</u> DETail <u>140</u> level commands *(continued)* KEYdist <u>141</u> NOCatalog <u>135</u>, <u>141</u> NOKeys <u>135</u>, <u>141</u> SQLsummary <u>141</u> SUMmary <u>135</u>, <u>141</u> TABular <u>135</u>, <u>141</u> TINo <u>135</u>, <u>141</u> TIYes <u>135</u>, <u>141</u> XOFf <u>141</u> XOFf <u>141</u> XON <u>141</u> level subparameter <u>146</u> location <u>202</u> LVL, level <u>135</u>

Μ

main control panel 95 matching columns 203 matching index scan 142 MATCOLS user parameter 234 maximizing information on panels 45 merge scan join info in Enhanced Explain report 168 messages ANLSQLPA 249, 294 general information 249 good news and guidelines 316 methods for accessing 219 notes and recommendations 307 metadata library overview 224 specifying 42 migrating SQL Performance Analyzer 83 migrating statistics 112 migration parameters applying 114 specifying 113 modifying parameters 96 modifying settings 42 modifying statistics 112 MONEYIS configuration parameter 228 multiple DBRM support 91 multiple index scan info in Enhanced Explain report 166

Ν

nested loop join info in Enhanced Explain report <u>168</u> NEWROTA user parameter <u>234</u> NEWSEEK user parameter <u>234</u> NEWSTOR user parameter <u>234</u> NEWXFER user parameter <u>234</u> NLS parameter <u>148</u> NLSCODE user parameter <u>234</u> NOCatalog <u>141</u> NOKeys <u>141</u> non-matching index scan <u>142</u> NONIXPG user parameter <u>235</u> NOSTATS user parameter <u>235</u> notices <u>403</u> NUMBERS user parameter <u>235</u>

0

OBJECTS user parameter 235 OPTIMIZE FOR n ROWS clause 180 options 45 output report panel 96 owner subparameter 135, 147

Ρ

package report 200 package reports remote 202 packages compare 99 pages with rows 203 panel display options 45 panels Copy DB2 Entries 67 Parameters Data Set 96 **User Parameters 96** parameter configuration using ISPF 129 parameters ADVISOR 230, 233 AIXUPDAT 233 AUTHIDO 226 BUFF08K 226 BUFF16K 226 BUFF32K 226 **BUFFERS 226 BUFFHIT 230 BYPASSRC 238** COMMENT 226, 230 configuration 226 CONNECT 230 COSTING 226 COSTQUN 226 CPUCOST 227 **CPUTIME 227** CSVDCSP 231 CSVDLIM 231 CSVHDRW 231 CSVRPT 231 CSVSEL 231 CURRSYM 227 DATAGRP 227 DBRMKEY 231 DEGREES 232 DELIMIT 232 DSGROUP 227 DSPPARM 233 DSPVARS 233 DSPWCOC 233 DYNAMIC 227 ELAPSED 227 FASTTBL 227 **HPOOLRD 233 INLISTS 233 IOCALLS 228 IOSCOST 228 ISCANPG 233** JOINTAB 233 **KEEPLAN 234**

parameters (continued) MATCOLS 234 modifying 96 MONEYIS 228 NEWROTA 234 NEWSEEK 234 NEWSTOR 234 NEWXFER 234 NLSCODE 234 NONIXPG 235 NOSTATS 235 NUMBERS 235 **OBJECTS 235** PKGDRDA 235 PRECISE 235 PREDPCT 236 PROCESS 236 QLIMSQL 236 **QUALIFY 236 REPORTS 237 RETCODE 237 RETPLAN 238** SETPLAN 228 SHOWALT 238 STORAGE 238 SUBSYST 228 SUBVERS 229 TIMCOST 229 **TSCANPG 238** USEPLAN 239 USRPARM 229 VERSION 239 VIADRDA 239 Parameters Data Set panel 96 PARMS field 142, 148 Path Checker input to Db2 SOL Performance Analyzer 187 path length estimates in Detail Trace report 183 path table 245 performance features 2 PKG 133 PKGDRDA user parameter 235 **PKGNAME 88, 89 PLAN 133** plan history report 203 plan table 244 Plan Table data source 123 Plan Table report 10 PLAN_TABLE 133, 246 PLAN_TABLE a 134 PLAN_TABLE authorization ID 149 PLN 133 PLOC subparameter 147 PRECISE parameter 130 PRECISE user parameter 235 predicate analysis report 193 predicate analysis report, enabling 130 PREDPCT user parameter 236 preparing to use Tools Customizer 42 PREVSQLID 88 PREVVERS 88 problems diagnostic information about 213, 217 PROCESS user parameter 236

processing remotely <u>123</u> processing step info in Detail Trace report <u>180</u> programming interface information <u>403</u> PROMPTED query <u>133</u>

Q

QBE query 133 **QLIMIT** file example 191 **QLIMIT** file record format 191 **OLIMIT** report in batch 91, 191 in TSO 191 QLIMSQL user parameter 236 OMF batch processing 119 canceling queries 118 governor exit 9, 117 intercept exit 117 interface creating 73 described 3 PRECISE parameter in QMF 130 literals 133 parameters 133 system tables 133 **QMF** governor exit 117 QNO 133 QTRACE file 91 QUALIFY user parameter 236 Query Block Report 115 query CPU time 5 Query Limits report What If? testing 116 Query Monitor interface 73 QUERY number 154 **QUNITS 155**

R

random matching index scan info in Enhanced Explain report 163 reader comment form 13 relative cost, QUNITS 155 remote operation 3 remote package reports 202 remote processing 123 report CSV report 97 report append option 96 report overlay option 96 reports remote package 202 Reports Query Block 115 reports menu 114 **REPORTS user parameter 237** resource consumption estimates 154 resource estimator 8 Resource Limit Facility (RLF) 4 restoring indexes 112 restoring tables 112 **RETCODE user parameter 237**

RETPLAN user parameter <u>238</u> return codes <u>249</u> roadmaps customizing for the first time <u>47</u> first-time customization <u>47</u> RUN statement <u>148</u>

S

SANLDATA and SANLPARM, difference 131 screen readers and magnifiers 14 scrollable fields 94 secondary authorization IDs 9 secondary SQL ID 246 SECONDS OF ELAPSED TIME 154 service information 13 SET command 142 SETPLAN configuration parameter 228 setting up apply jobs 112 setting up collect jobs 112 SHOWALT user parameter 238 SMF accounting 5 sort info in Enhanced Explain report 173 source type 203 specification of Comment statement 135 Level statement 135 Package statement 135 Plan statement 135 **QMF** statement 135 QNO statement 135 SET statement 135 SQL statement 135 specifying batch JCL for SQL PA 87 host parameter file in batch 91 input from Path Checker 91 report file disposition TSO 96 report files in batch 91 report files in TSO 95 SQL input in batch 91 in TSO 95 temporary files in batch 91 user parameter file in batch 91 work file for multiple DBRM support in batch 91 specifying data sets 42 SOL Advisor 4 SQL input in batch 91 in OMF 117 in TSO 95 SOL PA codes 249 generic plan tables resetting the REGISTRY 213 messages 249 predicate analysis report 130 return codes 249 SQL PA panels output report panel 96 SQL PA parameters influence on cost estimates 127

SOL PA parameters (continued) listed in Enhanced Explain report 156 SOL parser routine 8 SQL Performance Analyzer stored procedure operational requirements 18 SQL, in batch JCL 87 SOLsummary 141 startup procedure **DSNSPAS** 79 statistics modification 106 STEPLIB user parameter file 91 STMTMATCH 88, 89 STORAGE user parameter 238 stored procedure address space 79 call parameters 80 calling 80 customizing parameters 80 operational requirements 18 target host config file 79 using ANLPRCR 119 using ANLPRER 119 using SQL PA 119 stored procedure customization 79 subparameter acctype 142 eloc 143 first 143 force 144 format 144 gen 144 hostvar 145 index 135. 145 last 143 level 146 owner 135, 147 **PLOC 147** table 147 **TOPA** 147 SUBSYST configuration parameter 228 SUBVERS configuration parameter 229 SUMmary 141 summary of changes 1 summary report 121, 133, 199 support required information 213, 217 support information 13 SYSPRINT file 91 system parameter DATAGRP 124 system parameters system 96 user 96 viewing or changing 96 system tables DB2 133 **OMF 133** system thresholds 96

T

table data <u>124</u> table space scan 142, 146

listed in Detail Trace report 179

table space scan info in Enhanced Explain report 158 table subparameter 147 tables FFFAUTH 242 EEEDBRM 243 EEEHINT 243 EEEPATH 245 **EEEPLAN 244** EEEWORK 242 PLAN TABLE contents 246 Tables and index names 133 tables per table space 203 TABular 141 tabular report reference Explain 205 target host configuration parameters ANLCNTL parameter file 91 target host parameter file 91 target host system, selecting 130 technotes 13 temporary files in batch ANLOUT file 91 test programs ANLSTPR 120 testing access paths for packages in batch 89 Testing access paths for batch processing 89 thresholds system 96 TIMCOST configuration parameter 229 TINo 141 TIYes 141 **Tools Customizer** associated list adding Db2 entries 53 overview 221 associating Db2 entries 53 browsing parameters 67 changing display options 45 component 221 Copy DB2 Entries panel 67 copying Db2 entries 67 Create a DB2 Entry panel 53 creating Db2 entries 53 customization jobs deleting 70 displaying 69 generating 64 maintaining 70 renaming 70 sort sequence 64 submitting 64, 69 customization library deleting jobs 70 maintaining 70 recustomizing 70 renaming jobs 70 customization library qualifier specifying 42 Customized status 221 Customizer Workplace panel 64 customizing a new version of a product 46, 48 customizing a product for the first time 46, 47 data sets

Tools Customizer (continued) data sets (continued) customization library 224 data store 224 **Discover EXEC library 224** metadata library 224 data store data set specifying 42 Db2 data sharing members adding 53 associating 53 copying 67 creating 53 Db2 entries adding 53 associating 53 copying 67 creating 53 defining 64 deleting 68 generating jobs for 64 removing 68 selecting 64 specifying 64 unassociating 68 DB2 entries 221 Db2 group attach field specifying 42 Db2 group attach names adding 53 associating 53 copying 67 creating 53 Db2 parameters defining 61 editing 61 DB2 Parameters panel 61 Db2 subsystems adding 53 associating 53 copying 67 creating 53 DBb2 entries deleting 69 unassociating 69 defining Db2 parameters 61 defining LPAR parameters 59 defining parameters 55, 64 defining product parameters 56 deleting Db2 entries 69 deleting jobs 49 Discover Customized Product Information panel 52 **Discover EXEC** customizing a new version of a product 46, 48 retrieving product information automatically 52 **Discovered status 221** discovering product information 52 displaying jobs 69 displaying panel text 45 editing LPAR parameters 59 editing product parameters 56 Errors in Customization status 221 features 10 finding trace data set 217

Tools Customizer (continued) Finish Product Customization panel 64 first-time customization 46, 47 generating jobs 64 hiding panel text 45 high-level qualifier 221 Incomplete status 221 job sort order 64 iobs deleting 70 displaying 69 maintaining 70 renaming 70 submitting 69 LPAR Parameters panel 59 maintaining jobs 70 master list adding Db2 entries 53 Associate DB2 Entry for Product panel 53 overview 221 maximizing information on panels 45 metadata libraries specifying 50 metadata library specifying 42 multiple instances 42 multiple-LPAR environment 70 Not Required status 221 overview 10 panels Associate DB2 Entry for Product 53 Copy DB2 Entries 67 Create a DB2 Entry 53 Customizer Workplace 64 DB2 Parameters 61 Discover Customized Product Information 52 Finish Product Customization 64 LPAR Parameters 59 **Product Parameters 56** Specify the Metadata Library 50 parameters browsing 67 viewing 67 preparing to use 42 product 221 product parameters changing 49 editing 49 modifying 49 Product Parameters panel 56 Ready to Customize status 221 recustomization 46 recustomizing a product 46, 49 removing Db2 entries 68 roadmaps customizing a new version of a product 48 recustomizing a product 49 using the Discover EXEC 48 Specify the Metadata Library panel 50 specifying metadata libraries 50 starting 41 status types Customized 221 Discovered 221

Tools Customizer (continued) status types (continued) Errors in Customization 221 Incomplete 221 Not Required 221 Ready to Customize 221 submitting jobs 64 terminology 221 trace data set 217 troubleshooting 217 user job card settings specifying 42 using the Discover EXEC 48 viewing parameters 67 **TOPA** subparameter 147 trace data set finding 217 trademarks 403 troubleshooting 213 TSCANPG user parameter 238 TSO 12 TSO parameter panel 96 TSO SQL input file 95, 96

U

unqualified objects in packages 132 Unqualified objects In DBRMs and flat files 132 update info in Enhanced Explain report 174 **USEPLAN** user parameter 239 user authorization 149 user authorization table 149 user defaults data set 150 user job card settings specifying 42 USER parameter 148 user parameter file in batch 91 user parameters ADVISOR 230, 233 ANLPARM 129 ANLPARM parameter file 91 **BUFFHIT 230 BYPASSRC 238** changing 96 COMMENT 230 CONNECT 230 CSVDCSP 231 CSVDLIM 231 CSVHDRW 231 CSVRPT 231 CSVSEL 231 DBRMKEY 231 DEGREES 232 DELIMIT 232 **DSPPARM 233** DSPVARS 233 DSPWCOC 233 HPOOLRD 233 in TSO 96 **INLISTS 233 ISCANPG 233**

user parameters (continued) **IXUPDAT 233** JOINTAB 233 **KEEPLAN 234** MATCOLS 234 NEWROTA 234 NEWSEEK 234 NEWSTOR 234 NEWXFER 234 NLSCODE 234 NONIXPG 235 NOSTATS 235 NUMBERS 235 **OBJECTS 235** PKGDRDA 235 PRECISE 235 PREDPCT 236 PROCESS 236 QLIMSQL 236 QUALIFY 236 **REPORTS 237 RETCODE 237 RETPLAN 238** SHOWALT 238 STEPLIB parameter file 91 STORAGE 238 **TSCANPG 238 USEPLAN 239** VERSION 239 VIADRDA 239 user table data 124 USRPARM configuration parameter 229

V

valid commands <u>135</u> version generation <u>200</u> version identification <u>200</u> version report <u>200</u> VERSION user parameter <u>239</u> VIADRDA parameter <u>3</u> VIADRDA remote connect <u>123</u> VIADRDA user parameter <u>239</u> virtual indexes creating <u>108</u> dropping <u>108</u>

W

What if? analyzing indexes <u>106</u> modifying statistics <u>106</u> restoring indexes <u>112</u> restoring tables <u>112</u> What If? testing <u>116</u> What if? analysis <u>106</u> What If? Query Limits report <u>116</u> what's new <u>1</u> wild card <u>200</u> working table 242

Х

XOFf <u>141</u> XON <u>141</u>



Product Number: 5697-W51

SC27-6771-02

