



DB2 Utilities Solution Pack for z/OS Overview and Customization

Version 2 Release 2



DB2 Utilities Solution Pack for z/OS Overview and Customization

Version 2 Release 2

Note:

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

First Edition (June 2015)

This edition applies to Version 2 Release 2 of IBM DB2 Utilities Solution Pack for z/OS (product number 5697-DUP) and to all subsequent releases and modifications until otherwise indicated in new editions.

© Copyright IBM Corporation 2011, 2015; Copyright Infotel, 2011, 2015; Copyright Rocket Software Inc. 2011, 2015. All Rights Reserved.

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

Contents

About this information v

Chapter 1. DB2 Utilities Solution Pack for z/OS overview 1

What is the DB2 Utilities Solution Pack for z/OS? 1
Benefits 1
| Proactive administration of your DB2 environment 4
| Scenario: Improving the performance of a DB2 utility by using the DB2 Utilities Enhancement Tool 7
| Scenario: Improving the sort processing performance of the DB2 utilities 7
| Scenario: Unloading DB2 data with DB2 High Performance Unload and DB2 Sort 8
| Scenario: Reducing processor consumption by automating routine maintenance. 9
| Scenario: Modernization of maintenance and administration tasks in Management Console 10
| Scenario: Monitoring and problem determination in Management Console 10
| Service updates and support information 11
| Product documentation and updates 11
| Accessibility features 12

Chapter 2. Preparing to customize DB2 Utilities Solution Pack 15

Environment setup 18
 DB2 Utilities Solution Pack: Set up your environment prior to customization 18
 Set up your environment prior to customization 19
 Hardware and software requirements. 22
 Set up your environment prior to customization 23
Deployment considerations 25
 Deployment considerations: DB2 Automation Tool 25
 Deployment considerations: DB2 Sort. 26
 Deployment considerations: DB2 Utilities Enhancement Tool 30
Worksheets: Data set names and parameters 37
Worksheets: DB2 Utilities Solution Pack 37
Worksheets: DB2 Automation Tool. 37
Worksheets: DB2 High Performance Unload 67
Worksheets: DB2 Sort. 132
Worksheets: DB2 Utilities Enhancement Tool 135
Worksheets: Gathering parameter values for Tools Customizer 157

Chapter 3. Pre-customization tasks 161

Pre-customization tasks: DB2 High Performance Unload 161
 Authorizing and enabling DB2 HPU. 161
 Optional: Creating a TSO command for the DB2 HPU interactive component 161
Pre-customization tasks: DB2 Sort 162
 Activating DB2 Sort 162

Chapter 4. Starting and preparing Tools Customizer for use 163

Best Practice: SMP/E and runtime libraries maintenance strategy for Tools Customizer 163
Starting Tools Customizer 167
Modifying Tools Customizer user settings 168
| Hiding and displaying panel text. 171

Chapter 5. Customizing components in the DB2 Utilities Solution Pack. . . 173

Roadmap: Customizing components for the first time 173
Roadmap: Customizing new versions of components from a previous customization 174
Roadmap: Recustomizing components 175
Specifying the metadata library for the pack to customize 177
Selecting components to customize 178
Discovering component information automatically 179
Creating and associating DB2 entries 181
Defining parameters 183
 Defining component parameters 183
 Defining LPAR parameters 185
 Defining DB2 parameters 186
Generating customization jobs. 188
Submitting customization jobs. 189
Browsing parameters 191
Copying DB2 entries 191
Removing DB2 entries 193
Deleting DB2 entries 193
Displaying customization jobs 194
Maintaining customization jobs 194
Using Tools Customizer in a multiple-LPAR environment. 194

Chapter 6. Post-customization tasks 197

Post-customization tasks: DB2 Automation Tool 197
 Creating multiple configurations of DB2 Automation Tool 197
Post-customization tasks: DB2 High Performance Unload 198
 Optional: Integrating DB2 HPU into DB2 Tools Launchpad 198
 Optional: Integrating DB2 HPU into DB2 Administration Tool 199
Post-customization tasks: DB2 Sort 200
 Accessing DB2 Sort libraries 200
 Installation Verification Program: Overview 201
 Using Installation Verification Program 201
 Generating an IVP report 202
Post-customization tasks: DB2 Utilities Enhancement Tool. 203
 Setting up the initialization options member (optional). 203
 APF-authorizing the load library 205

Copying the started task PROC	205	Tools Customizer messages.	216
Customizing DSNUTILB intercept parameters (optional).	206	Chapter 8. Reference	275
Copying the DSNUTILF module	210	Tools Customizer terminology and data sets . . .	275
Creating a security exit (optional)	210	Tools Customizer terminology.	275
Creating a pre- or post-cancel exit (optional) . .	211	Data sets that Tools Customizer uses during customization	278
Starting the started task	212	How to read syntax diagrams	280
Starting the ISPF interface	212	Notices	283
Chapter 7. Troubleshooting and messages	215	Index	287
Tools Customizer troubleshooting	215		
Gathering diagnostic information.	215		
Determining the trace data set name	215		

About this information

DB2 Utilities Solution Pack is a product that combines a number of IBM components into a consolidated solution that enables enterprises to efficiently and intelligently run utilities, while optimizing the performance of daily utilities management activities.

DB2 Utilities Solution Pack combines the following components into a single offering:

- DB2 Automation Tool for z/OS V4.2
- DB2 High Performance Unload for z/OS V4.3
- DB2 Sort for z/OS V2.1
- DB2 Utilities Enhancement Tool for z/OS V2.2

These topics provide an overview of the components and capabilities that are delivered with DB2 Utilities Solution Pack and some common usage scenarios to help you understand the capabilities of the product. These topics also provide comprehensive instructions for customizing the solution pack after a successful SMP/E installation.

These topics are designed to help database administrators, system programmers, application programmers, and system operators perform the following tasks:

- Understand the capabilities of the functions that are associated with the solution pack
- Perform the pre-customization tasks for the components
- Customize the components of the solution pack
- Troubleshoot any errors that might occur during the customization process

To use the functions that are described in this information, you must have already installed DB2 Utilities Solution Pack by completing the SMP/E installation process that is documented in the Program Directory for DB2 Utilities Solution Pack, GI10-9005.

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

- The z/OS operating system
- ISPF

Always check the DB2 Tools Product Documentation page for the most current version of this information:

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

Chapter 1. DB2 Utilities Solution Pack for z/OS overview

IBM® DB2® Utilities Solution Pack for z/OS® (also referred to as DB2 Utilities Solution Pack) combines powerful DB2 tools that, when used together, can improve utility performance and simplify utility management.

What is the DB2 Utilities Solution Pack for z/OS?

DB2 Utilities Solution Pack is a set of functionally compatible components.

DB2 Utilities Solution Pack includes the following components:

IBM DB2 Automation Tool for z/OS (also referred to as DB2 Automation Tool)

You can use this component to set up recurring utility jobs for conditional and routine DB2 maintenance tasks. You can also automate common DB2 maintenance tasks and generate JCL for more complex tasks on one or more objects.

DB2 High Performance Unload for z/OS (also referred to as DB2 High Performance Unload)

This component offers performance options for unloading and extracting large amounts of data across the enterprise. Because HPU works outside of the DB2 address space, it does not compete for the same DB2 resources.

DB2 Sort for z/OS (also referred to as DB2 Sort)

Sort processing of large volumes of data, large table spaces, or large indexes typically requires intensive processor usage and heavy I/O activity. Large amounts of memory and disk space can be required for this sort processing. DB2 Sort meets these demands by optimizing the use of memory, using zIIP technology, and optimizing the management of disk space that is used for sort work.

DB2 Utilities Enhancement Tool for z/OS (also referred to as DB2 Utilities Enhancement Tool)

This component provides additional utility features and the ability to set company-wide standards for executing DB2 syntax. The tool enhances the DB2 LOAD, REORG TABLESPACE, and CHECK DATA utilities. Additionally, DB2 Utilities Enhancement Tool can block and cancel threads on DB2 objects for other DB2 online utilities and for any programs that use a DB2 database.

Benefits

By using DB2 Utilities Solution Pack, you can simplify the management and operation of your utilities and improve their performance at the same time.

DB2 Utilities Solution Pack provides the following benefits:

Proactive administration of your DB2 environment

Today's global enterprise faces unprecedented challenges as it strives to recruit, train, deploy, and retain expert IT staff on the mainframe. Even the best information management professionals work with ever-increasing constraints of time, skill, and resources.

As database administrators retire, a global business needs innovative approaches to maintain its investment in mainframe skills. New IT professionals who replace retiring DBAs typically need up to five years of experience to effectively meet critical business requirements for a healthy, optimized, available DB2 environment. At the same time, experienced DBAs should not spend most of their time navigating ISPF panels when they might otherwise be setting strategic direction for the business. To whatever extent you can automate routine administrative tasks and minimize the need for decades of mainframe experience, the enterprise benefits.

IBM Management Console for IMS™ and DB2 for z/OS and IBM Autonomics Director for DB2 for z/OS provide the tools that you need to optimize both DB2 administration and your workforce. In combination with DB2 Utilities Solution Pack, these tools help you to meet demanding service level agreements in a complex global business environment. They provide the following advantages:

- A consolidated view of your DB2 environment an intuitive, graphical web interface
- Configuration and setup options to help you achieve the precise level of automation that your business requires for database monitoring and maintenance tasks

For more information, see “Proactive administration of your DB2 environment” on page 4.

Integrated monitoring and maintenance among pack components

DB2 Automation Tool

- User-defined thresholds enable you to monitor DB2 objects at any time interval, and then take any corrective action that you define for each symptom.
- Autonomics Director for DB2 can proactively take actions that DB2 Automation Tool does already. Instead of waiting for user intervention to resolve a condition, you can take appropriate action automatically when your tools environment detects that DB2 exceeded a specified threshold.
- Autonomics enablement features that are available in DB2 Automation Tool feed alerts and recommendations to Management Console for a one-stop view of DB2 status and health.

DB2 High Performance Unload

- LOB and XML features support unloading LOB and XML data from image copies.
- OUTPUT data supports SPANNED FORMAT.
- ORDER CLUSTER or ORDER BY options are supported for UNLOAD FORMAT INTERNAL
- When you unload from image copy, you can use OBID_REPORT with a non-production run to obtain a list of object IDs.

DB2 Sort

- DB2 Sort works with DB2 High Performance Unload and DB2 Utilities Enhancement Tool to read files directly before performing a sort.

- When you specify that DB2 Sort processes sorts for DB2 Utilities and other DB2 management tools, you can significantly reduce processor consumption and performance. You do this by running DB2 LOAD with the extended syntax that DB2 Utilities Enhancement Tool provides, such as PRESORT, CONSTANT and VALUEIF.

DB2 Utilities Enhancement Tool

- DB2 Utilities Enhancement Tool works with DB2 Automation Tool to generate the mapping table and mapping-table index when running a REORG TABLESPACE.
- DB2 Utilities Enhancement Tool sends statistics to Autonomics Director for DB2 that help with workload balancing.

Increased availability

DB2 DB2 Utilities Enhancement Tool provides greater control over utility operations. The following examples describe some of the benefits:

- Reduces application downtime by maintaining access to business-critical objects
- Reduces DBA intervention by canceling threads automatically

Optimized processor utilization

DB2 DB2 Utilities Enhancement Tool provides significant savings on costly processor consumption:

- Saves on processor utilization and elapsed time by automatically creating the mapping table during a REORG
- Reduces processor utilization and elapsed time with the PRESORT option during a LOAD
- Reduces repeated CPU consumption by ensuring utilities run successfully the first time
- Improves availability and efficiency by using the DB2 Automation Tool to generate CHECK DATA on related sets of objects. DB2 DB2 Utilities Enhancement Tool then discards data rows to a flat file.

More efficient LOAD and UNLOAD operations

The DB2 High Performance Unload component alleviates some of the problems that are associated with unloading large amounts of data.

Because of the large amount of time consumed by large sequential reads of DB2 tables, UNLOAD operations are difficult to schedule during increasingly-shrinking batch windows. Performance issues can become critical when several UNLOAD operations read the same table space concurrently.

In this situation, DB2 buffer pool management can cause multiple programs to compete for the same data. This competition among multiple programs can result in the following challenges:

- Conflicts in the use of the DB2 buffer pool
- Overwrites of buffers that are serving several UNLOAD operations
- Multiple reads of the same DB2 pages
- Channel conflicts

To address these problems, DB2 High Performance Unload provides a fast way to sequentially read and share a DB2 table space among multiple unloads. Whenever possible, DB2 High Performance Unload processes

requests to unload data from the same table space in parallel. You can create different output files during the same UNLOAD process at almost no additional cost.

For example, you can unload a list of customers who have payments due this week and another list of customers whose birthdays are on the first day of the week. You can create these lists in a single execution of DB2 High Performance Unload at a fraction of the cost that is required by traditional approaches that run UNLOAD operations twice.

DB2 High Performance Unload can decrease the time it takes to unload certain DB2 tables by invoking DB2 Sort. You can increase performance when DB2 must sort the result set or when unloading from a traditional or inline image copy.

Improved sort processing

DB2 Sort improves the performance of utility sort processing, especially in environments with large volumes of data, large table spaces, or large indexes.

Efficient and secure utility syntax management

The Utility Syntax Monitor that is available in the DB2 Utilities Enhancement Tool has robust capabilities to help you with utility syntax. The following are only some of the advantages that the monitor can provide:

- Enables you to establish and maintain company-wide DB2 utility syntax policies
- Extends utility syntax to reduce system costs and improve availability
- Provides an additional layer of security to protect business critical objects and ensure a secure recovery strategy
- Enables you to take advantage of new DB2 utility syntax without changing JCL, thereby reducing costs and avoiding the need to change multiple batch jobs

Proactive administration of your DB2 environment

Management Console and Autonomics Director for DB2 consolidate features of the tools in the DB2 Utilities Solution Pack to provide a self-monitoring, self-diagnosing, and self-healing monitoring and maintenance system.

Overview

The DB2 Utilities Solution Pack is a comprehensive monitoring and administration solution that helps you to manage, optimize, control, and automate your DB2 environment. It is the cornerstone of an *autonomics solution* that helps you to anticipate and avoid emergency situations and high-severity problems before they occur. DB2 Utilities Solution Pack also helps you to maintain availability, integrity, and security, and comply with service level agreements. The ability to optimize performance is also an essential component of the solution.

In combination with Autonomics Director for DB2 and Management Console, the DB2 Utilities Solution Pack provides the precise level of automation and scheduled maintenance that your business requires to supplement manual approaches. Thanks to a robust set of configuration options based on policies and profiles, you are in complete control. You decide which actions to automate and which to perform manually.

The pack offers easy-to-configure *policy services* that build intelligence into utilities and tools management. In addition to the *real-time statistics* (RTS) that DB2 generates, including DB2 sensor data that any tool can access, Autonomics Director for DB2 adds an additional layer of features that enable you to define policies. These policies recommend or automate recurring DB2 monitoring and maintenance tasks, including the ability to schedule the collection and evaluation of the sensor data.

Autonomics Director for DB2 collects these point-in-time statistics about database health and stores them in a central repository. It evaluates the sensor data, and then compares the values to thresholds that you define in the policies. If DB2 activities exceed a threshold that you specify, Autonomics Director for DB2 generates an exception. You then have the option to define actions, such as triggering notifications or launching processes, that Autonomics Director for DB2 takes in response to the exceptions.

Management Console

Management Console modernizes mainframe administration by providing a single, intuitive, and modern web-based interface that configures, administers, automates, and optimizes the performance of your DB2 environment. It consolidates data from multiple DB2 tools, including tools from non-IBM vendors, and presents a more complete picture of those systems and databases. You can use the comprehensive context-sensitive help system included with Management Console as a teaching aid for new DBAs and system programmers.

Today's global enterprise must manage larger and more complex information management environments. With this increased complexity comes more data and larger tables; more subsystems, objects, and bufferpools; and new utility and metadata options. As the number of experienced DBAs in the workplace declines, your business must focus on efficiency, choose the right options, and avoid unnecessary work. Management Console provides a comprehensive view of your databases and tools that builds upon the existing capabilities of DB2 Automation Tool to maximize efficiency and availability.

Administrative capabilities

- Database administration
- Catalog navigation
- Policy management
- Automated action and alert resolution
- Manual action and alert resolution

Automation and performance features

- Monitor DB2
- Read policies
- Generate or modify actions and alerts
- Generate and resolve alerts and actions

Features of an autonomics strategy

Simple setup and administration

Management Console provides central oversight, default profiles, and a new graphical user interface in addition to the 3270 interface.

Staged enablement from passive to active stages of management

You can define profiles that manage all database objects from individual tables to the entire DB2 environment.

Transparency based upon existing autonomic infrastructure

Your autonomics strategy builds on existing DB2 capabilities and infrastructure, including tables, stored procedures, and the Administrative Scheduler. Integration with batch schedulers enables you to run the maintenance jobs that you require during maintenance windows.

Preservation of existing profiles and utilities

Build on the existing profiles and logic that are found in DB2 Automation Tool and DB2 Utilities Enhancement Tool to automate REORG, COPY, and RUNSTATS operations and incorporate existing batch utilities into your autonomics strategy.

Extensibility that integrates other tools and products

The DB2 tools in the DB2 Utilities Solution Pack work with all available tools, utilities, and actions as part of a comprehensive solution. Examples of this extensibility include Optim™ Query Workload Tuner, performance tools, and activities that are not performed by utilities, such as WTOs and SMS texts.

Autonomic activities

Autonomics Director for DB2 performs both *active* and *passive* activities. The following stages describe the passive activities:

- **Collect data:** Schedule or trigger data collection; view historical data.
- **Analyze and identify symptoms:** Schedule or initiate analysis; view and dismiss symptoms.
- **Diagnose and prescribe action:** View recommendations.

The following stages are active:

- **Plan and schedule maintenance window:** Define maintenance windows; view and fine tune maintenance windows.
- **Perform and take action:** Schedule or initiate specific actions.

In any stage of the process, you decide when and how to analyze, identify, diagnose, prescribe, and take action. The following user-defined profiles put you in complete control of the actions that you undertake, either manually or automatically:

- Exception profiles (alerts, thresholds)
- Object profiles
- Utility profiles
- Alert profiles
- Maintenance profiles

The following examples describe typical actions that Autonomics Director for DB2 might recommend:

- DB2 REORG
- Image copy (IC) operations
- RUNSTATS operations

For example, you can monitor your DB2 environment continuously for thresholds and policy compliance issues. Instead of running DB2 Automation tool during batch windows, you can trigger a run in response to conditions in the environment. You can set alerts and thresholds in DB2 Automation Tool, and then when a threshold reached, initiate a REORG manually or automatically in response to the alert.

Scenario: Improving the performance of a DB2 utility by using the DB2 Utilities Enhancement Tool

You can improve the performance of a DB2 utility by using a DB2 Utilities Enhancement Tool *intercept policy*.

An intercept policy enables you to dynamically change the DB2 utility syntax at run time without manually changing JCL, which is a process that is time consuming and error prone.

Consider the following scenario: the DB2 LOAD utility is loading records from unloaded tables before the records are sorted. Because the DB2 LOAD utility loads presorted records faster, it is advantageous to have the data rows in sort order before the utility loads them. DB2 Sort can presort the records and has superior sort performance over the DB2 DFSORT utility. However, manually adding a sort step to a LOAD utility job, for the purpose of presorting a SYSREC file, can be time-consuming. Imagine the time that it could take, especially if there are dozens or perhaps hundreds of LOAD utility jobs in your environment. Can you dynamically change your run time environment for the DB2 LOAD utility so that DB2 Sort can presort the input records? Yes.

An intercept policy also lets you change the DB2 LOAD utility syntax at run time based on the rules that are defined in the policy. In other words, you can add parameters that are not currently present in the LOAD utility syntax. For example, if the PRESORT option is present in the LOAD utility syntax, the intercept policy can add the PRESORTED YES option to the LOAD control statement at run time. The PRESORT option invokes DB2 Sort to sort the rows in the input data set by table object identifier (OBID) and by clustering index key. The PRESORTED YES option tells the DB2 LOAD utility to load the input records immediately because they are presorted.

Here is the relevant <UTILITY> section in a sample intercept policy:

```
<UTILITY NAME="LOAD">
  <MONITOR>
    <SYNTAX ADD="PRESORTED YES" OPTIONIF="PRESORT"/>
    <SYNTAX REMOVE="SPANNED YES"/>
    <MESSAGE ID="DSNU1150I" RETURN_CODE="4"/>
  </MONITOR>
</UTILITY>
```

Scenario: Improving the sort processing performance of the DB2 utilities

DB2 Sort offers superior sort performance over the DB2 DFSORT utility, and can improve the sort processing performance of several DB2 utilities.

Consider the following scenario: suppose that your enterprise uses the DFSORT utility for all sort processing. A significant number of objects in your enterprise are large, and working with these objects results in an abundant amount of sorting

activity. Because sort processing is done by many of the DB2 utilities, you might reduce processor and elapsed times by using DB2 Sort.

DB2 Sort supports the following DB2 utilities:

- CHECK INDEX
- CHECK LOB
- LOAD
- REBUILD INDEX

To invoke DB2 Sort from these utilities automatically, after DB2 Utilities Solution Pack is customized, set the DB2 subsystem parameter, **DB2SORT**, to ENABLE.

You can further integrate DB2 Sort into your DB2 environment by using DB2 Automation Tool. DB2 Automation Tool supports the following DB2 utilities:

- CHECK DATA
- REORG TABLESPACE
- RUNSTATS

After the DB2 Utilities Solution Pack is customized, the CHECK DATA, REORG TABLESPACE, and RUNSTATS utilities invoke DB2 Sort by default for any sort processing.

Scenario: Unloading DB2 data with DB2 High Performance Unload and DB2 Sort

You can decrease the amount of time that is required to unload DB2 tables by using DB2 High Performance Unload in conjunction with DB2 Sort.

Consider the following scenario: a banking company in the United States is opening a European branch and needs to transfer its customer accounts to a database in Europe.

Before the data can be unloaded, the senior DBA must consider the following factors:

Size of the table

The table is large; it contains over 15 million rows.

Date format of the columns

In the United States, the MM-DD-YYYY format is used, whereas in Europe, the DD-MM-YYYY format is used. Data, therefore, must be unloaded in the European format.

Performance

Data must be unloaded as fast as possible.

Using DB2 High Performance Unload in conjunction with DB2 Sort is ideal for this scenario for the following reasons:

- DB2 High Performance Unload has a performance advantage over the DB2 Unload utility when unloading large tables.
- DB2 Sort has superior sort performance over the DFSORT utility. DB2 Sort can be invoked for complicated SELECT clauses that require access to the DB2 index (such as ORDER BY and WHERE clauses). Because DB2 Sort can be invoked, these SQL clauses do not slow down the unload process because the syntax does not need to be passed to DB2 for interpretation.

About this task

The senior DBA who implements this scenario must perform the steps outlined in this scenario.

1. Ensure that they have the required privileges for running DB2 HPU.
For a list of the privileges that are required, see the *DB2 HPU 4.2 User's Guide*.
2. Code the job with the required SYSIN DD options and save it.

Here is an example:

```
GLOBAL SORTUTIL DB2SORT ;
  UNLOAD TABLESPACE
  QUIESCE YES
  QUIESCECAT YES
  SELECT FNAME,NAME, ADDRESS, DATE_B
  FROM CUSTOMER ORDER BY NAME
  OUTDDN (SYSUT1)
  FORMAT USER (
  COL FNAME NULLID YES,
  COL 3 TYPE CHARACTER(100)
  JUST RIGHT,
  COL 004 TYPE DATE_E )
  LOADDN LOAD1
  LOADOPT (RESUME YES)
```

In this example, SYSUT1 is DD name for the output data set and LOAD1 is the DD name for the LOADDN data set.

Sample JCL for unloading a table can be found in the *DB2 HPU 4.2 User's Guide*.

3. Submit the job.

Results

DB2 High Performance Unload unloads the CUSTOMER table from the banking database and converts all of the columns that contain dates to the European date format. The DBA can now send the output and load files to colleagues in Europe so that they can load the data into their own database.

Scenario: Reducing processor consumption by automating routine maintenance

You can reduce the amount of processor resources that routine maintenance consumes by automatically generating JCL for only those objects that need maintenance.

Consider the following scenario: suppose that your enterprise has routine batch JCL jobs that are submitted on a regular basis for running utilities. Maintenance jobs, such as generating image copies, reorganizing table spaces and indexes, checking data, and so on, are critical for ensuring that business data remains accessible. Suppose further that the allocated batch maintenance window is full, and that a significant number of objects must be maintained within that window. Adding additional utilities or objects can put a strain on an already full maintenance window.

By using DB2 Automation Tool, you can apply filter criteria against the objects that you define to selectively generate utility JCL for only those objects that need maintenance. Instead of you routinely running utilities against objects whether

those objects need maintenance or not, DB2 Automation Tool determines which objects need maintenance and generates utility JCL for only objects that need it. The result: processor resources are saved.

DB2 Automation Tool can evaluate database objects by applying over 180 exception criteria to determine which objects need maintenance and which objects do not. DB2 Automation Tool can generate this JCL without any intervention from the database administrator. By running maintenance utility JCL against only those objects that need it, you consume less processor resources, and reduce the strain on your batch maintenance window.

Scenario: Modernization of maintenance and administration tasks in Management Console

A DB2 database administrator with 25 years of experience changes a reactive response to an alert to a proactive autonomic approach that anticipates similar problems in the future.

A DBA manages DB2 databases for a global retail enterprise. He also mentors peers who have less experience with DB2 databases on z/OS. One morning, he receives a mobile alert that indicates multiple problems in a DB2 subsystem. When he arrives at work, he logs in to the Alerts Dashboard in Management Console to review a set of critical exceptions.

The DBA drills down into the inventory database, where he quickly and easily identifies the problem. One of his table spaces needs to be reorganized. After the REORG process is complete, he decides to take proactive action that will detect similar problems automatically in the future. He schedules an active autonomic REORG that will perform this maintenance task automatically before the state of the database reaches a critical point again.

The DBA opens the autonomic scheduler and reviews the existing schedules. He creates a new schedule for the REORG, and then reviews the list of jobs that are in queue for the next maintenance window. He defines profiles that will monitor the required DB2 objects, generate the alerts, trigger the appropriate actions, and schedule the maintenance job in the next batch window. As he defines options for the RUNSTATS utility, best practices are displayed in the console by default.

If the same conditions recur, the REORG task will run automatically and address the problem before it reaches a critical juncture. Instead of reacting to phone calls and alerts, the DBA can avoid crisis management and attend to more urgent priorities.

Scenario: Monitoring and problem determination in Management Console

A DB2 database administrator uses Management Console to identify database changes and to determine which maintenance tasks should be automated or performed manually, resulting in more efficient staff utilization and massive performance improvements.

A DBA who works for a global banking enterprise reviews properties and key metrics for an important member of the data sharing environment. The DBA and

| system programmer recently completed a customization process that tailored
| Management Console to display dashboards and other information that are
| relevant to their site.

| In the console, she drills down for more information about the top three table
| spaces. A series of views that enable her to review properties, alerts, and *table space*
| *size over time* helps her to identify some changes that she wants to make to the
| database. The DBA selects each table space in turn, drills down, and reviews
| detailed graphs and recommendations.

| The DBA sets up an active autonomies policy to schedule the RUNSTATS utility if
| certain thresholds are exceeded. Noting that the same exception is appearing
| repeatedly for one of the table spaces, she sets a flag to prevent automatic REORGs
| for that table space. She takes an additional proactive action to avoid future
| truncation problems with long names by increasing the length of the City column
| in one of the tables from 32 to 64 characters.

| In the final stage of the review, the DBA creates an active autonomies policy to
| schedule queries for routing to the DB2 Analytics Accelerator. Queries that reach a
| certain threshold that qualifies them for acceleration now trigger a process that
| adds their tables to the accelerator automatically. The resulting improvements to
| the performance of business intelligence queries are so great that the DBA
| schedules a special meeting with upper management to report on the radically
| improved performance.

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](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:

<http://www.ibm.com/software/data/db2imstools/solutions/index.html>

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 comments@us.ibm.com. 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. Preparing to customize DB2 Utilities Solution Pack

Before you start to customize DB2 Utilities Solution Pack, 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 terminology and data sets" on page 275	
Set up your environment prior to customization		
Ensure that your environment meets hardware and software requirements before you deploy DB2 Utilities Solution Pack and its components.	<p>DB2 Utilities Solution Pack "DB2 Utilities Solution Pack: Set up your environment prior to customization" on page 18</p> <p>DB2 Automation Tool "Set up your environment prior to customization" on page 19</p> <p>DB2 High Performance Unload "Hardware and software requirements" on page 22</p> <p>DB2 Utilities Enhancement Tool "Set up your environment prior to customization" on page 23</p>	
SMP/E installation		
Verify that DB2 Utilities Solution Pack has been installed correctly. DB2 Utilities Solution Pack is installed by using standard SMP/E processing.	"Verify that DB2 Utilities Solution Pack has been installed successfully" on page 19	
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 19	
Deployment considerations		

Task	Link to detailed instructions	Status
<p>Some components of the DB2 Utilities Solution Pack have deployment issues that you should consider before you customize them.</p>	<p>DB2 Automation Tool</p> <ul style="list-style-type: none"> • “DB2 version migration and fallback” on page 25 • “DB2 version migration and fallback instructions” on page 25 • “Porting repository statistics from one DB2 subsystem to another” on page 26 <p>DB2 Sort</p> <ul style="list-style-type: none"> • “Default installation options” on page 26 • “Runtime considerations” on page 28 <p>DB2 Utilities Enhancement Tool</p> <ul style="list-style-type: none"> • “Storage requirements” on page 30 • “Deployment in a DB2 data sharing environment” on page 31 • “Started task authorization requirements” on page 33 • “Considerations for running multiple started tasks” on page 33 • “Dispatching priority” on page 35 • “Use of WTO messages for automated operations” on page 35 • “Operational issue related to the vector table” on page 36 • “If you are migrating from DB2 Thread Expert” on page 36 	
Pre-customization tasks		
<p>Ensure that you have completed all the pre-customization tasks for some components of the DB2 Utilities Solution Pack.</p>	<p>DB2 High Performance Unload</p> <ul style="list-style-type: none"> • “Authorizing and enabling DB2 HPU” on page 161 • “Optional: Creating a TSO command for the DB2 HPU interactive component” on page 161 <p>DB2 Sort</p> <p>“Activating DB2 Sort” on page 162</p>	
Gather data set names		
<p>During the customization process, you must specify names for the following types of data sets:</p> <ul style="list-style-type: none"> • Tools Customizer • DB2 Utilities Solution Pack <p>Additionally, you must specify the names of data sets for each component of the DB2 Utilities Solution Pack that you will customize.</p>	<p>DB2 Utilities Solution Pack</p> <p>“Worksheets: Gathering required data set names” on page 37</p> <p>DB2 Automation Tool</p> <p>“Worksheets: Gathering required data set names” on page 37</p> <p>DB2 Sort</p> <p>“Worksheets: Gathering required data set names and parameter values” on page 132</p> <p>DB2 Utilities Enhancement Tool</p> <p>“Worksheets: Gathering required data set names” on page 136</p>	
Gather parameter values		

Task	Link to detailed instructions	Status
<p>During the customization process, you must specify parameter values for DB2 Utilities Solution Pack, for the components that you will customize, for DB2, and for your LPAR.</p>	<p>DB2 Automation Tool “Worksheets: Gathering parameter values for DB2 Automation Tool” on page 40</p> <p>DB2 High Performance Unload “Worksheets: DB2 High Performance Unload” on page 67</p> <p>DB2 Sort “Worksheets: Gathering required data set names and parameter values” on page 132</p> <p>DB2 Utilities Enhancement Tool “Worksheets: Gathering parameter values for Tools Customizer” on page 138</p> <p>DB2 Utilities Solution Pack “Worksheets: Gathering parameter values for Tools Customizer” on page 157</p>	
Customize DB2 Utilities Solution Pack		
<p>Complete the steps in the appropriate customization roadmap based on the type of customization that you are performing.</p>		
<p>Customizing DB2 Utilities Solution Pack for the first time</p> <p>Follow this roadmap if you do not have a customized version of DB2 Utilities Solution Pack, and you need to customize it for the first time.</p>	<p>“Roadmap: Customizing components for the first time” on page 173</p>	
<p>Customizing a different version of DB2 Utilities Solution Pack</p> <p>Follow this roadmap if you have already customized a version of DB2 Utilities Solution Pack and you want to use the same parameter values to customize a different version.</p>	<p>“Roadmap: Customizing new versions of components from a previous customization” on page 174</p>	
<p>Recustomizing DB2 Utilities Solution Pack</p> <p>Follow this roadmap if you have a customized version of DB2 Utilities Solution Pack but you want to change one or more parameter values.</p>	<p>“Roadmap: Recustomizing components” on page 175</p>	
Post-customization tasks		

Task	Link to detailed instructions	Status
<p>Some components of the DB2 Utilities Solution Pack have tasks to be completed after the Tools Customizer jobs have been submitted. Ensure that the post-customization tasks that apply to your environment are complete.</p>	<p>DB2 Automation Tool “Creating multiple configurations of DB2 Automation Tool” on page 197</p> <p>DB2 High Performance Unload</p> <ul style="list-style-type: none"> • “Optional: Integrating DB2 HPU into DB2 Tools Launchpad” on page 198 • “Optional: Integrating DB2 HPU into DB2 Administration Tool” on page 199 <p>DB2 Sort</p> <ul style="list-style-type: none"> • “Accessing DB2 Sort libraries” on page 200 • “Installation Verification Program: Overview” on page 201 • “Using Installation Verification Program” on page 201 • “Generating an IVP report” on page 202 <p>DB2 Utilities Enhancement Tool</p> <ul style="list-style-type: none"> • “Setting up the initialization options member (optional)” on page 203 • “APF-authorizing the load library” on page 205 • “Copying the started task PROC” on page 205 • “Customizing DSNUTILB intercept parameters (optional)” on page 206 • “Copying the DSNUTILF module” on page 210 • “Creating a security exit (optional)” on page 210 • “Creating a pre- or post-cancel exit (optional)” on page 211 • “Starting the started task” on page 212 • “Starting the ISPF interface” on page 212 	

Environment setup

Use the information in this section to set up your environment for the DB2 Utilities Solution Pack.

DB2 Utilities Solution Pack: 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.

IBM System Modification Program Extended (SMP/E) for z/OS, V3.5 or higher (5655-G44)

Ensure that you are using one of the following supported versions of DB2 for z/OS:

- DB2 V9.1 (5635-DB2) operating in new-function mode
- DB2 Value Unit Edition V9.1 (5697-P12)
- DB2 V10 (5605-DB2)
- DB2 Value Unit Edition V10.1 (5697-P31)
- DB2 V11 (5615-DB2)
- DB2 Value Unit Edition V11.1 (5697-P43)

Verify that DB2 Utilities Solution Pack has been installed successfully

See the Program Directory for IBM DB2 Utilities Solution Pack for z/OS, GC19-4931 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.

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

DB2 Automation Tool supports the following versions of DB2 for z/OS:

- DB2 V10
- DB2 Version 11

Additional feature requirements are as follows:

- If you plan to use DB2 Automation Tool with Autonomics Director, DB2 V10 NFM or later is required.
- If you plan to use DB2 Automation Tool's interface to the DB2 administrative task scheduler:
 - The DB2 subsystems must be configured to enable DB2 administrative task scheduler capabilities.
 - RACF[®] must be configured to allow PassTickets for the DB2 administrative task scheduler task.
- If you plan to use DB2 Automation Tool's autonomic statistics functionality, the following DB2 stored procedures must also be configured, and users' authorization IDs must have execute authority on them:
 - ADMIN_UTL_EXECUTE
 - ADMIN_UTL_MONITOR
- If you plan to use DB2 Automation Tool to build utility JCL for image copies of objects on IBM ESS devices, you must have IBM Enterprise Storage Server[®] devices with Advanced Copy Services, and the DFSSMS system data mover (SDM) API (macro ANTRQST) must be at level 5 or above.

- If you plan to use DB2 Automation Tool to make image copies from DB2 Recovery Expert system level backups (SLBs), you must have DB2 Recovery Expert V2.2 with APAR PM06332 or higher installed on the subsystem.

Verify that DB2 Automation Tool has been installed successfully

See the Program Directory for IBM DB2 Automation Tool for z/OS, GI10-8911-01, for installation instructions.

Verify that Tools Customizer has been installed successfully

Tools Customizer is a component of IBM Tools Base for z/OS (5655-V93), which is available free of charge. Tools Customizer provides a standard approach to customizing IBM DB2 for z/OS Tools.

See the Program Directory for IBM Tools Base for z/OS, GI10-8819-05 for installation instructions.

Verify that your environment meets security requirements

DB2 Automation Tool requires no extra security measures outside of standard DB2 security.

If your site uses ACF2 to restrict TSO command use, you may need to add the TSO commands that DB2 Automation Tool uses to the ACF2 Command Limiting table. The TSO commands that DB2 Automation Tool uses are: HAA\$MAIN, HAA@BULD, and FECDBCX.

Authorize the FEC\$TSOC program

Add the program FEC\$TSOC to the AUTHPGM and AUTHTSF sections of member IKJTSO00 in SYS1.PARMLIB.

Set the MEMLIMIT parameter

DB2 Automation Tool utilizes above the bar storage (storage above the 2-gigabyte bar). To control the amount of real and auxiliary storage that an address space can use for memory objects at one time, your site may have established an installation default MEMLIMIT that sets the total number of usable virtual pages above the bar for a single address space. If the default is not sufficient, DB2 Automation Tool cannot be started.

The current default for MEMLIMIT for z/OS V1R10.0 and later is 2G. The default for z/OS versions prior to that is 0, meaning that no address space can use virtual storage above the bar. You should verify that the MEMLIMIT setting for your site is at least 2G for DB2 Automation Tool. This amount is recommended for scalability of a wide range of objects. DB2 Automation Tool only uses as much storage above the bar as it needs, and only as long as it is needed; the storage is released immediately after a job or process is complete.

To set the MEMLIMIT parameter, use one of the following methods:

- Set an installation default on the MEMLIMIT parameter in the SMFPRMxx PARMLIB member.
- Issue the SET SMF or SETSMF command.

- Add either the MEMLIMIT parameter or REGION=0 to the logon procedure TSO JCL or the job JCL.
- Specify MEMLIMIT in a IEFUSI exit routine; if you do so, this MEMLIMIT setting overrides all other MEMLIMIT settings.

The following example shows the MEMLIMIT parameter added to a TSO logon procedure:

```
//TSOPROC EXEC PGM=IKJEFT01,REGION=0M,DYNAMNBR=175,
// PARM='%LOGINIT',TIME=1440,MEMLIMIT=2G
```

See the IBM z/OS documentation for your version of z/OS for additional information about MEMLIMIT.

Add the IEFACTRT exit to the SMFPRMxx member of SYS1.PARMLIB

The IEFACTRT exit is required for successful operation of the execution reports facility. If the exit is not currently defined, add the IEFACTRT exit at either the SYS or SUBSYS level to the appropriate JES subsystem in the SMFPRMxx member of your site's SYS1.PARMLIB.

The HAAPROC member checks for IEFACTRT at the SUBSYS.JES level or SYS level upon startup. If the IEFACTRT entry is not found, DB2 Automation Tool job tracking will not work. If the IEFACTRT entry is found, the CSVDYNEX service will dynamically define the exit for DB2 Automation Tool job tracking.

In addition, ensure that type 30 SMF records are included in the SMF record types specified in SMFPRMxx.

Create an active WLM environment for integration with Autonomics Director

To use DB2 Automation Tool with Autonomics Director, the Autonomics Director stored procedures that are created during customization must be associated with an active Workload Manager (WLM) environment. You can do this by using one of the following methods:

- When customizing DB2 Automation Tool using Tools Customizer, select the Management Console Stored Procedures optional task and run the generated JCL.
- Adapt an existing WLM PROC already active for your DB2 subsystem.

Each DB2 environment (SSID or data sharing group) must have separate PROCs and separate WLM environments.

The WLM PROC must be activated using the WLM panels. This is generally done by a systems programmer. When activating the WLM environment for use with DB2 Automation Tool, the following settings are required:

Application Environment Name

Enter the WLM ENVIRONMENT parameter in the DDL that defines the stored procedure(s).

Description

(Optional) Enter a description.

Subsystem Type

Must be set to DB2.

Procedure Name

Enter the name of the JCL PROC that defines the address space in which the stored procedure runs.

Start parameters

Required parameters are:

DB2SSN=*ssid*
APPLENV=*applenv_name*

Optional parameter:

NUMTCB=*number*

applenv_name must be the same value that was entered in the **Application Environment Name** field.

NUMTCB can be left unspecified and specified in the JCL PROC. The sample PROC provided handles the NUMTCB parameter in the PROC. If specified on the WLM panel, the panel specification takes precedence over the JCL PROC parameter.

Take into account the following considerations:

- Use the same name for the application environment name and the JCL PROC.
- Adopt a naming convention for the application environment name that includes the SSID as part of the name, such as *ssidxxx* or *xxxssid*. For example, for two WLM environments that are associated with Autonomics Director (DYX) on DB2 subsystem ID DB2A, then possible application environment and procedure names might be DB2ADYX1 and DB2ADYX2, respectively.

If you choose to adapt an existing WLM PROC to run the Autonomics Director stored procedures, the PROC must include the DB2 Automation Tool load library, the FEC load library, and the Autonomics Director load library. Additionally, the PROC must have a DD named DB2PARMS that points to the DB2 Automation Tool control file.

Apply maintenance before migrating from DB2 Automation Tool V3.1 or V4.1

If you are upgrading from DB2 Automation Tool V3.1 or V4.1, and you plan to migrate your V3.1 or V4.1 data repository to use with DB2 Automation Tool V4.2, ensure that all current maintenance has been applied to DB2 Automation Tool V3.1 or V4.1 before beginning the customization process.

DB2 Automation Tool version compatibility

Restriction: DB2 Automation Tool V3.1 and DB2 Automation Tool V4.2 can be installed on the same LPAR. However, a single TSO user ID cannot run V3.1 and V4.2 at the same time. Unpredictable results may occur.

Hardware and software requirements

DB2 HPU requires the same hardware configuration that is required by DB2 for z/OS Version 10 or later.

DB2 HPU requires the following software:

- z/OS Version 1 Release 13 or later
- DB2 Version 10.1 for z/OS or later

Note:

To exploit DB2 10 for z/OS functionality, DB2 HPU V4.2 with appropriate maintenance level (PM78638 for Toleration and PM91909 for exploitation) is required. Previous versions of DB2 HPU cannot exploit DB2 10 for z/OS functionality.

If you want to perform conversions that imply non-SBCS CCSIDs or pairs of SBCS CCSIDs that are not supported by the SYSSTRINGS catalog table, you must install z/OS Support for Unicode.

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 hardware requirements

DB2 Utilities Enhancement Tool can be used on any IBM mainframe computer that is capable of running the required software.

Verify that your environment meets software requirements

Ensure that you are using z/OS V1.11 (5694-A01) or later.

Ensure that you are using one of the following versions of DB2 for z/OS:

- DB2 V8 (5625-DB2) (new function mode). Additionally, make sure that DB2 V8 maintenance has been applied.
- DB2 Value Unit Edition V8.1 (5697-N29)
- DB2 V9 (5635-DB2). Additionally, make sure that DB2 V9 maintenance has been applied.
- DB2 Value Unit Edition V9.1 (5697-P12)
- DB2 V10 (5605-DB2). Additionally, make sure that DB2 V10 APAR PM93789 has been applied.
- DB2 Value Unit Edition V10.1 (5697-P31)

Note: This is the minimum version that supports the following DB2 UET features for LOAD processing:

- SHRLEVEL REFERENCE option
- IFDISCARDS option
- DB2 V11 (5615-DB2)
- DB2 Value Unit Edition V11.1 (5697-P43)

Ensure that you are using a supported version of the following software:

- ISPF V4 (5655-042) or later
- IBM SMP/E for z/OS V03.05.00 (5655-G44) or later
- DB2 Utilities Suite for z/OS V10.01.00 (5655-V41) or later
- IBM Tools Base for z/OS V01.02.00 (5655-V93) or later

(HPU users only) To substitute the IBM DB2 High Performance Unload for z/OS (HPU) for the DB2 UNLOAD utility, HPU must be installed on the DB2 subsystem on which the UNLOAD is to be executed.

To use features that prevalidate data before loading it (IFDISCARDS and SHRLEVEL REFERENCE options):

- Ensure that your IBM Workload Manager for z/OS (WLM) environment is set up so that DB2 UET can run the SYSPROC.DSNUTILU stored procedure.
- If the DB2 UET load modules are in the WLM STEPLIB concatenation, or if the DB2 Analytics Accelerator Loader product is installed, ensure that DB2 UET APAR PI04864 is applied and that you are using the modules that are provided with it. Specifically, verify that the compile date in the DSNUTILF module in the WLM address space is the same as the date of the module in the enhancement load library.

Verify that your environment meets storage requirements

For each z/OS system on which DB2 UET will run, ensure that the common service area (CSA) or extended common service area (ECSA) meets the minimum storage requirements, that your DFSMS rules support the LOAD utility enhancements, and that your DASD storage is sufficient.

- The CSA and ECSA storage requirements are as follows. The total amount of CSA/ECSA storage that is required will depend on the number of started tasks that you run, the number of users, the level of user request activity, and the number of DB2 subsystems that are defined on the z/OS system.
 - 4 KB from subpool 228 in ECSA for the vector table
 - 1 KB from subpool 241 in CSA or ECSA for each active DB2 UET started task
 - 160 bytes from subpool 241 in CSA or ECSA for each simultaneous request that is made from a DB2 UET interface
 - 44 bytes from subpool 241 in CSA or ECSA for each DB2 subsystem that is defined on the z/OS system
 - DB2 UET supports the date, time, and timestamp values of the IBM DB2 High Performance Unload for z/OS (HPU) product. To use this feature, a minimum amount of storage will be required in CSA/ECSA. All other storage is allocated from private address spaces or data spaces.

- DFSMSdss requirements are as follows:

When performing data prevalidation for the DB2 LOAD utility and when running the LOAD utility with the parameters REPLACE SHRLEVEL REFERENCE, DB2 UET uses DFSMSdss to copy data sets from the production objects to the shadow objects. If SMS is active, you must set up SMS rules to allow the shadow data sets to reside on volumes that are allowed for DB2 objects. The underlying data sets for the shadow objects will be renamed to become the active page set for the DB2 table and index spaces; they will not be moved from the location in which SMS places them when the shadow objects are created.

The renaming or DFSMSdss COPY of DB2 VSAM objects follows the standard naming conventions for such objects:

`catname.DSNDBx.dbname.psname.y0001.znnn`

where y is a letter in the range A through Z. DB2 UET checks for existing data set names and uses the first available letter to rename the objects.

- For DASD storage requirements, see the *Program Directory for DB2 UET for z/OS*.

Verify that DB2 Utilities Enhancement Tool has been installed successfully

See the Program Directory for IBM DB2 Utilities Enhancement Tool for z/OS, GI10-8981 for installation instructions.

Verify that Tools Customizer has been installed successfully

Tools Customizer is a component of IBM Tools Base for z/OS (5655-V93), which is available free of charge. Tools Customizer 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.

Deployment considerations

Some components of the DB2 Utilities Solution Pack have deployment issues that you should consider before you customize them.

Deployment considerations: DB2 Automation Tool

Before you customize DB2 Automation Tool, determine whether a DB2 subsystem that is being used with DB2 Automation Tool was migrated to a later version, or must be reverted to a prior version.

DB2 version migration and fallback

When a DB2 subsystem that is being used with DB2 Automation Tool is migrated to a later version, or must be reverted to a prior version, follow these steps.

After your subsystem has been migrated to a later version of DB2, you can port your repository statistics to the new subsystem. For example, you can port the statistics repository from a DB2 V9 subsystem to a DB2 V10 subsystem. Information about migration is also provided in this topic.

DB2 version migration and fallback instructions

These steps must be performed when a DB2 subsystem that has been customized for use with DB2 Automation Tool is migrated to a later DB2 version, or must be reverted to a previous DB2 version.

About this task

For member names described in this procedure, *ss* is a job sequence number and *dd* is a set of alphanumeric characters assigned by Tools Customizer. Refer to “Worksheets: Gathering required data set names” on page 37 for additional information.

Procedure

1. Start Tools Customizer.
2. Run the Discover command to ensure that the Tools Customizer datastore is up to date.
3. For each DB2 subsystem that has been customized:
 - a. On the Customizer Workplace panel, edit the DB2 entry.

- b. On the DB2 Parameters panel, change the Mode and Level number fields to the new DB2 mode and level. Change other DB2 specific fields, such as library or BSDS names, as required. Save and exit the DB2 Parameters panel.
- c. On the Customizer Workplace panel, regenerate the customization jobs for the subsystem.
- d. Submit the SSID-specific update control file job (member name *ssCF2dd*).
- e. For DB2 migrations only, submit jobs to update the repository (member name *ssDD*dd*). Skip this step for fallback.
- f. For fallback only, submit the job to free previously bound plans and packages from the fallback subsystem (member name *ssFREEdd*).
- g. Submit the bind job (member name *ss#BNDdd*).

Porting repository statistics from one DB2 subsystem to another

You can use the HAARSTAT job that is provided in SHAASAMP to unload DB2 Automation Tool repository RUNSTATS statistics from one DB2 subsystem and load another DB2 subsystem with the statistics.

About this task

You can also use this job to migrate statistics between DB2 versions; for example, you can migrate the statistics repository from a DB2 V9 subsystem to a DB2 V10 subsystem.

Procedure

Edit and run the HAARSTAT sample job; the instructions are contained in the member. Information about the number of unloaded repository RUNSTATS statistics rows is displayed during the unload phase. Information is also provided about the number of RUNSTATS statistics rows that are loaded during the load phase.

Deployment considerations: DB2 Sort

Before you customize DB2 Sort, understand the default installation options and runtime considerations.

Default installation options

The DB2 Sort default installation options include options for dynamic sort work allocation, optimization mode, sensitivity to paging activity, and tuning messages.

Dynamic sort work allocation options

The following options refer to values used for the dynamic allocation of sort work data sets. These data sets are de-allocated when the sort process has completed.

Sort work DASD pool

This value specifies the 1- to 8-character name of the group of devices from which DB2 Sort dynamically allocates the sort work data sets if SMS is not installed or active for temporary DASD work data sets. If SMS is active, you can specify a value for the sort work storage class option. The default value is SYSDA.

Restriction: You cannot specify VIO as a unit device.

Note: This value can be overridden at run time.

Sort work DASD type

This value specifies the type of DASD with the smallest track capacity that might be encountered at your installation by a dynamically allocated sort work data set. If, during dynamic allocation, DB2 Sort receives a device with a track capacity that is smaller than the one specified, it might ignore the specified device type. The default value is 3390.

The following table lists common DASD types and their track capacities:

Table 1. DASD types and their track capacities

DASD type	DASD track capacity
3380	47476
3390	56664
9345	46456

Number of allocation retry attempts

This value specifies the number of times that DB2 Sort automatically retries a dynamic allocation request in situations where sufficient disk space is not immediately available to satisfy that request. Insufficient disk space can result in a Sort Capacity Exceeded condition. The minimum value allowed is 0 and the maximum value is 16. The default value is 5.

Minutes between retry attempts

This value specifies the number of minutes to wait between dynamic allocation retry attempts. The minimum value allowed is 0 and the maximum value is 15. The default value is 3.

Sort work storage class

This value specifies the storage class that DB2 Sort uses to manage temporary work data sets for the dynamic allocation of sort work data sets if SMS is active. If this value is not specified in an SMS environment, DB2 Sort dynamically allocates devices from the generic class name that is specified for the sort work DASD pool. There is no default value for storage class.

Note: An installation-written automatic class selection (ACS) routine can override the value you specify.

Maximum number of sort works

This value specifies the maximum number of sort work data sets that can be used. DB2 Sort can dynamically allocate data sets to bring the total number of sort work data sets up to the number that you specify. Specify a value from 32 through 255. DB2 Sort automatically determines the appropriate number of sort work files to use. The default value is 32.

Note: This value can be overridden at run time.

Optimization mode option

This value specifies the optimization mode for DB2 Sort.

OPTMODE

BALANCE

BALANCE provides the best mix of CPU and elapsed time performance by balancing utilization of central storage and disk space. Typically BALANCE uses less CPU time and more elapsed time than ELAP.

- CPU** CPU optimizes to minimize CPU time without any consideration for elapsed time, maximizing disk space usage.
- ELAP** ELAP optimizes to minimize elapsed time while significantly reducing CPU time, favoring central storage over disk space usage. The default value is ELAP.

Note 1: All of these options limit central storage use if auxiliary storage availability is low.

Note 2: The OPTMODE value can be overridden at run time.

Sensitivity to paging activity option

This value specifies the type of storage allocations that DB2 Sort chooses.

PAGEMON

- ON** Choose ON only if your system is highly sensitive to paging activity.
- OFF** Choose OFF if you use the defaults for DFSORT's DSPSIZE, HIPRMAX and MOSIZE installation options. The default value is OFF.

Note: Elapsed time may increase if PAGEMON is set to ON.

Tuning messages option

This value specifies whether DB2 Sort displays additional informational messages for tuning purposes. These tuning messages provide information about the product's resource utilization, such as memory usage, sort work DASD space, and sort work EXCPs.

TUNEMSG

- ON** Enables the generation of the additional tuning messages. The default value is ON.
- OFF** Suppresses the additional tuning messages.

Note: The TUNEMSG value can be overridden at run time.

Runtime considerations

To improve performance, you might want to modify the **REGION** parameter in your DB2 utility job step. You can also override some of the installation options at run time.

The REGION parameter

Because DB2 Sort manages memory allocation dynamically, setting REGION=0M is recommended for best performance.

Default installation options that you can override at run time

Although most applications can use the default installation options, some applications might require different options. You can override installation options at run time by passing parameters in the DB2SORTI DD data set.

Important: This data set must have a fixed-length format with 80-byte records.

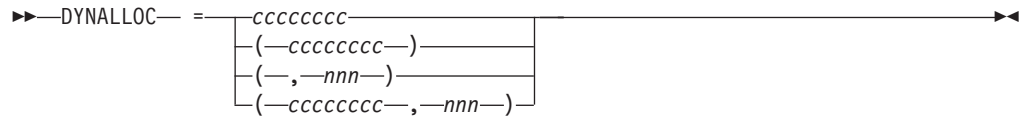
You can specify the **DYNALLOC**, **OPTMODE**, and **TUNEMSG** parameters at run time.

The parameters can start in any column but must be completed on the current record.

DYNALLOC

This parameter specifies values for dynamic sort work allocation.

Syntax



Options

- **ccccccc:**

This value specifies the 1- to 8-character name of the group of devices from which DB2 Sort dynamically allocates the sort work data sets.

This value can be omitted. If the value is omitted, specify a comma before the *nnn* value.

Restriction: You cannot specify VIO as a unit device.

- **nnn:**

This value specifies the maximum number of sort work files that can be used. DB2 Sort can dynamically allocate data sets to bring the total number of sort work data sets up to *nnn*. Specify a value from 32 through 255. DB2 Sort automatically determines the appropriate number of sort work files to use.

OPTMODE

This value specifies the optimization mode for DB2 Sort.

Syntax



Options

- **BALANCE:** BALANCE provides the best mix of CPU and elapsed time performance by balancing utilization of central storage and disk space. Typically BALANCE uses less CPU time and more elapsed time than ELAP.
- **CPU:** CPU optimizes to minimize CPU time without any consideration for elapsed time, maximizing disk space usage.
- **ELAP:** ELAP optimizes to minimize elapsed time while significantly reducing CPU time, favoring central storage over disk space usage.

Note: All of these options limit central storage use if auxiliary storage availability is low.

TUNEMSG

This parameter specifies whether DB2 Sort will display additional informational messages for tuning purposes. These tuning messages provide information about the product's resource utilization, such as memory usage, sort work DASD space, and sort work EXCPs.

Syntax

► TUNEMSG = ON
OFF ►

Options

- **ON**: Enables the generation of the additional tuning messages.
- **OFF**: Suppresses the additional tuning messages.

Example

```
//DB2SORTI DD *  
DYNALLOC=(,50),OPTMODE=ELAP
```

This example of a runtime override accomplishes the following goals:

- Specifies 50 for the maximum number of sort work data sets that can be used
- Sets the optimization mode to ELAP

Deployment considerations: DB2 Utilities Enhancement Tool

Before you customize DB2 Utilities Enhancement Tool, understand the following deployment considerations.

Storage requirements

For each z/OS system on which DB2 UET will run, ensure that the common service area (CSA) or extended common service area (ECSA) meets the minimum storage requirements. Also, ensure that your DASD storage is sufficient.

The CSA and ECSA storage requirements are as follows:

- 4 KB from subpool 228 in ECSA for the vector table
- 1 KB from subpool 241 in CSA or ECSA for each active DB2 UET started task
- 160 bytes from subpool 241 in CSA or ECSA for each simultaneous request that is made from a DB2 UET interface
- 44 bytes from subpool 241 in CSA or ECSA for each DB2 subsystem that is defined on the z/OS system
- The DB2 UET supports the date, time, and timestamp values of the IBM DB2 High Performance Unload for z/OS (HPU) product. To use this enhancement, a minimum amount of storage will be required in CSA/ECSA. All other storage is allocated from private address spaces or data spaces.

The total amount of CSA/ECSA storage that is required will depend on the number of started tasks that you run, the number of users, the level of user request activity, and the number of DB2 subsystems that are defined on the z/OS system.

For DASD storage requirements, see the *Program Directory for DB2 UET for z/OS*.

For information regarding an issue related to the 4-KB ECSA storage for the vector table, see “Operational issue related to the vector table” on page 36.

Deployment in a DB2 data sharing environment

If you plan to deploy DB2 UET in a DB2 data sharing environment, review this information to learn about deployment and configuration issues.

A DB2 data sharing group is composed of one or more DB2 subsystems that are located on the same z/OS image or on different z/OS images. The member subsystems share a common DB2 catalog and can directly access and change the same data while maintaining data integrity.

A DB2 UET started task can block and cancel threads and perform DSNUTILB intercept processing on the active subsystems within a data sharing group that have a DB2 version that DB2 UET supports. At least one of the member subsystems must be defined as the *primary subsystem* during customization and must contain the DB2 UET audit and logging tables. Because all subsystems in a data sharing group share the same DB2 catalog, they can also share the same thread-blocker and DSNUTILB-intercept tables and (for the REORG TABLESPACE mapping-table enhancement) the same mapping-table database and table space. You can define these objects once on any active member subsystem in the data sharing group. If you define these objects on a subsystem other than the primary subsystem, you will also need to define that subsystem as an *additional subsystem* during customization.

For the started task to communicate with the subsystems in a data sharing group, you must set the DB2_CONNECT_TO_ALL_SUBSYSTEMS initialization option for the started task to "Yes." If you specify "No" instead, the started task will be able to connect only to the subsystem that is specified in the DB2_SSID initialization option (that is, the primary subsystem).

Additional requirements, considerations, and restrictions depend on the DB2 UET interface that you use.

If you plan to use the DSNUTILB intercept, ensure that these requirements and recommendations are met:

- A DB2 UET started task must be running on each z/OS image where a member subsystem of the data sharing group is located. If some member subsystems are on a remote z/OS image, a started task configuration must be running on the remote z/OS image as well as on the z/OS image where the primary subsystem is located. Each started task must have its own set of DB2 objects and plans for DB2 UET. When you run the Tools Customizer for an additional started task configuration on a remote z/OS image, ensure that the field **The DB2 UET primary subsystem** on the **Product Parameters** panel contains the name of the DB2 SSID to use as the primary subsystem on the remote z/OS image.
- Each member subsystem must be specified in the DSNUTILB intercept policy for the local started task by using the <DB2SYSTEM> element.
- If the member subsystems are on different z/OS images, it is recommended that you associate the same intercept policy with the started task configuration on each of these z/OS images so that the same intercept policy will be used for all members of the data sharing group.

If you plan to use the ISPF interface or batch interface, only one started task configuration and one intercept policy are required, even if the member subsystems are on different z/OS images. All members of the data sharing group

will share a common set of DB2 objects for DB2 UET: the audit and logging tables on the primary subsystem, and the thread-blocker table, DSNUTILB-intercept worklist tables, and mapping-table database and table space on any active subsystem in the data sharing group. You will be able to cancel threads on any active subsystem that is within the same data sharing group as the primary subsystem. However, the following restrictions apply:

- You cannot perform escalated cancelations of DB2 threads on any member subsystems that are located on a z/OS image other than the z/OS image where the primary subsystem is located.
- For batch thread-cancellation operations that include thread blocking, you cannot perform the job step for blocking and canceling threads on a member subsystem that is on one z/OS image and then perform the job step for allowing new threads on another member subsystem that is on a different z/OS image. That is, the BLOCK_THREADS and ALLOW_THREADS actions must apply to the same member subsystem on one z/OS image.

The following figure shows a sample product configuration in a data sharing environment:

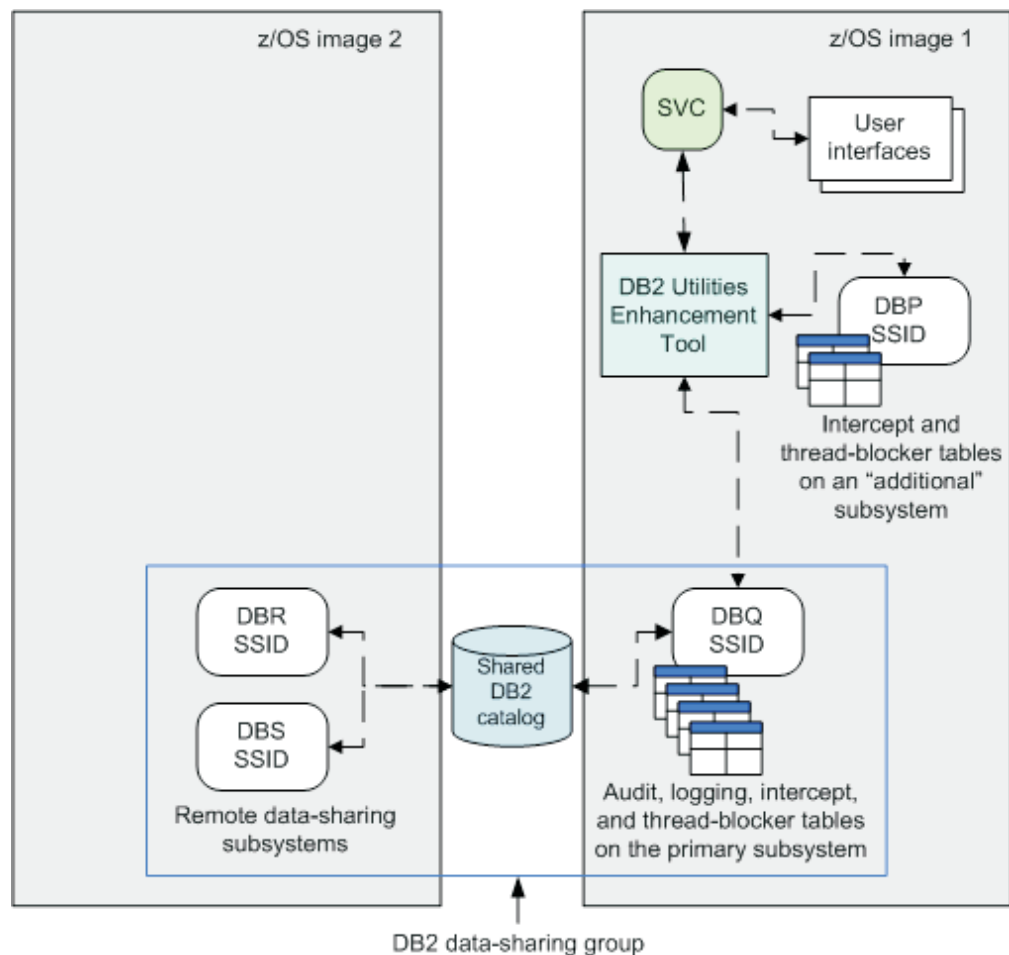


Figure 1. Sample product configuration in a DB2 data sharing environment

The DB2 subsystems DBQ, DBR, and DBS comprise a data sharing group that spans two z/OS images. DBQ is the primary DB2 subsystem for the DB2 UET started task configuration. It contains the audit and logging tables that are used for all started task operations, including those that involve the non-data sharing

subsystem DBP. DBQ also contains the thread-blocker and DSNUTILB-intercept tables; these tables are shared by all subsystems in the data sharing group. If you use the ISPF or batch interface, you could cancel threads on the DBP subsystem and on each subsystem in the data sharing group (DBQ, DBR, and DBS). However, if you use the DSNUTILB intercept, you could perform intercept processing (thread cancelation, or the utility-specific enhancements) only on the DBP and DBQ subsystems. To perform intercept processing on the DBR and DBS subsystems, another started task configuration would need to be running on z/OS image 2. Also, the intercept policy for that started task configuration would need to specify the DBR and DBS subsystems.

Started task authorization requirements

Make sure that the DB2 UET started task will run under a user ID that has the proper authority. The started task must run under a user ID that has SYSADM or SYSCTRL privileges. If you want to control security at the user level, you must create a security exit.

If you want to control access to thread-management functions or ISPF panels at the user level, you can create an optional security exit in assembler language. For example, you could create a security exit to control which users can perform thread cancelations and access the product administration panels. A sample security exit (ABPXSE00) and corresponding DSECT (ABPAPISE) are available in the *hlq.mlq.SABPSAMP* library, where *hlq* is the high-level qualifier and *mlq* is the mid-level qualifier that you specify during customization. You will need to specify the name of your security exit in the started task initialization options member by using the SECURITY_EXIT option. If you do not create a security exit, all users will be able to access all product functions.

Considerations for running multiple started tasks

A single started task is usually sufficient to handle multiple user requests from any of the DB2 UET interfaces to perform work on one or more DB2 subsystems. However, if you have a very high volume of activity, you can run multiple started tasks concurrently to handle the workload more efficiently.

If you run multiple concurrent started tasks, the SABPSAMP library must contain a separate started task initialization options member for each started task. Each initialization options member must specify a unique SVC number.

Also, each started task should have its own set of DB2 UET audit, logging, DSNUTILB-intercept, and thread-blocker tables. Although it is possible to share these tables across started tasks (because each row in each table is qualified by the *abpid* value for the started task configuration), this practice is not recommended. When multiple started tasks use the same tables, a very large volume of data might be written to the tables.

You will need to run the Tools Customizer for each started task to generate the following items: the DDL for creating the DB2 objects that the started task will use, the statements for binding the DB2 plan and packages on the DB2 subsystems that the started task will communicate with, the sample started task PROC, the started task initialization options member, and the sample DSNUTILB intercept policy. A separate DSNUTILB intercept policy is not required for each started task; the same intercept policy can be used by multiple started tasks if appropriate.

Tip: If you want to prevent a started task from performing work on a particular DB2 subsystem, you can *not* bind the plan and packages for the started task on that subsystem.

After you set up the started tasks, you can select the one that you want to use for performing product functions, as follows:

- In the ISPF interface, select a started task configuration ID on the Set ABPID panel.
- In a batch thread-cancelation job, specify a started task configuration ID by using the ABPID parameter.
- If you are using the DSNUTILB intercept, ensure that the ABPPLCY DD statement in the started task PROC specifies the DSNUTILB intercept policy that you want to use for performing intercept processing with that started task.

The following additional considerations apply if you are using the DSNUTILB intercept:

- A single started task can connect to multiple DB2 subsystems to perform intercept processing. You specify the subsystems in the DSNUTILB intercept policy for a started task by using the <DB2SYSTEM> element. However, any given DB2 subsystem can be intercepted by only one started task at a time.
- You can run multiple started tasks to perform intercept processing on different DB2 subsystems. You must specify a DSNUTILB intercept policy for each started task. If more than one started task policy identifies the same DB2 subsystem (either by specifying the same SSID or a wildcard pattern), the started task that connects to that subsystem first will have exclusive use of the subsystem until intercept processing completes. No other DB2 UET started task will be able access the subsystem while the first started task is performing intercept processing.

The following figure shows two DB2 UET started tasks performing DSNUTILB intercept processing for three subsystems:

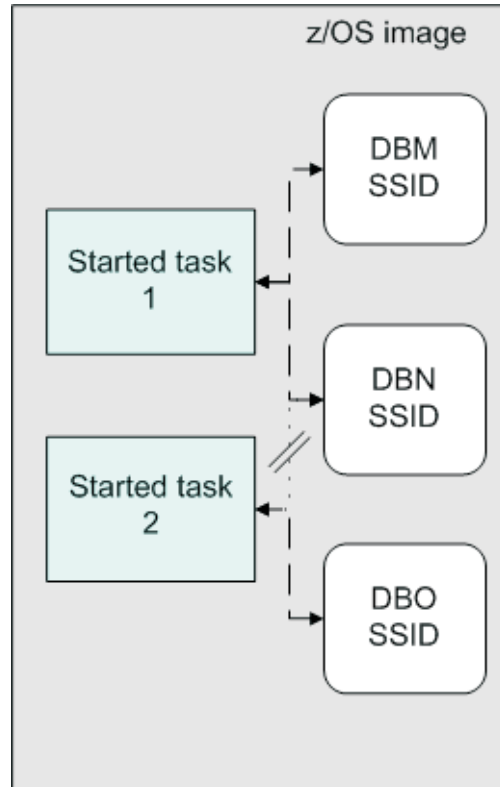


Figure 2. Using multiple started tasks for DSNUTILB intercept processing

The DSNUTILB intercept policy for started task 1 specifies the DB2 subsystems DBM and DBN. The policy for started task 2 specifies the DB2 subsystems DBN and DBO. Because started task 1 initializes first, it intercepts DBN before started task 2. Started task 2 cannot intercept DBN until after started task 1 completes its intercept processing on DBN.

Dispatching priority

Ensure that the dispatching priority for the DB2 UET task is set correctly with respect to other dispatching priorities. The dispatching priority determines the order in which a task can use the processor in a multitasking environment.

The DB2 UET dispatching priority must be lower than the priority values for the DB2 subsystems that DB2 UET will use but higher than the priority values for any utilities or batch jobs for which DB2 UET will perform processing. Set the dispatching priorities for these items in the following order (from highest to lowest priority):

1. The address spaces of the DB2 subsystems that DB2 UET will use (highest dispatching priority)
2. The DB2 UET started task
3. The DB2 utilities, batch jobs, and applications for which DB2 UET performs a function such as thread cancelation (any dispatching priority under the DB2 UET started task)

Use of WTO messages for automated operations

Consider whether you want to use the Write-to-Operator (WTO) messages that DB2 UET issues for automated operations.

DB2 UET issues some messages as WTO messages in addition to printing them to the SYSPRINT data set. If your automation tools can process WTO messages, you can use these messages to control the flow of automated operations in your environment.

The following WTO messages on the status of the DB2 UET started task are particularly useful: ABPS0001I, ABPS0002I, ABPS0003I, and ABPS0004I. These messages report the beginning and end of the started task initialization and termination phases.

Operational issue related to the vector table

Consider this operational issue regarding the common vector table.

Some IBM products for z/OS systems, including DB2 UET, share a 4-KB vector table that resides in ECSA (subpool 228, storage key 0). This table, called the RVT, serves as an anchor point for these products. The first product to start after an IPL will obtain and initialize the RVT. If that product subsequently terminates, the RVT storage is deliberately *not* released because it could be serving multiple products. A system-monitoring product might report the RVT storage as "orphaned" or "owner gone." In this situation, do *not* attempt to release the RVT storage. If you do so, the other products that are using the vector table can be severely damaged.

If you are migrating from DB2 Thread Expert

DB2 UET incorporates all of the features of the previously released product IBM DB2 Thread Expert for z/OS V1R1. It also provides additional features that enhance the DB2 online utilities. If you are migrating from DB2 Thread Expert to DB2 UET, review these migration requirements.

You can run both DB2 Thread Expert V1R1 and DB2 UET V2R1 on your z/OS system. This practice enables you to continue to use DB2 Thread Expert until you are ready to switch entirely to DB2 UET.

To run both products, ensure that the following requirements are met:

- Install DB2 UET in an SMP/E zone that different from the SMP/E zone in which DB2 Thread Expert is installed.
- Apply APAR PK48293 for DB2 Thread Expert V1R1. This APAR prevents the DB2 Thread Expert batch or ISPF interface from attempting to connect to the DB2 UET started task, thereby causing errors. You can obtain the APAR from the following Web page: <http://www-306.ibm.com/software/data/db2imstools/support.html>.
- Ensure that each product has its own DB2 plan and own set of DB2 objects (database, table spaces, and tables). DB2 Thread Expert and DB2 UET can *not* share the same DB2 plan and objects or use the same database name or creator ID for the DB2 objects. Note that if you want to use the same database name or creator ID, you can not run both products on your system; you would need to drop the existing DB2 Thread Expert database and objects during DB2 UET customization.
- Ensure that the following fields are not used in the DB2 UET DSECT (ABPAIPR) for the pre-cancel exit: UXPR_ACE_TOKEN, UXPR_FTDD_TOKEN, and UXPR_STATUS_FLAGS. These fields were included in the DB2 Thread Expert sample DSECT for this exit but they have been deprecated.
- Ensure that the following fields are not used in the DB2 UET DSECT (ABPAIPO) for the post-cancel exit: UXPO_ACE_TOKEN,

UXPO_FTDD_TOKEN, and UXPO_STATUS_FLAGS. These fields were included in the DB2 Thread Expert sample DSECT for this exit but they have been deprecated.

Worksheets: Data set names and parameters

Complete the following worksheets before you begin the customization process.

Worksheets: DB2 Utilities Solution Pack

Complete the following worksheets before you begin the customization process.

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.

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		

Data set names of DB2 Utilities Solution Pack

Identify and record the following DB2 Utilities Solution Pack data set names. During the customization process, you will enter the following values on panel CCQPPRD.

Data set name	Description	Special requirements	Your data set name
SBBYDENU	Metadata library for DB2 Utilities Solution Pack	None.	

Worksheets: DB2 Automation Tool

Complete the following worksheets before you begin the customization process.

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.

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
SCCQDENU	Metadata library for Tools Customizer		
SCCQLOAD	Executable load module library for Tools Customizer		
SCCQMENU	ISPF messages for Tools Customizer		
SCCQPENU	ISPF panels for Tools Customizer		
SCCQSAMP	Sample members for Tools Customizer		
SCCQTENU	Table library for Tools Customizer	You must have write access to this data set.	

Data set names of DB2 Automation Tool

Identify and record the following DB2 Automation Tool data set names. During the customization process, you will enter the following values on panel CCQPPRD.

Data set name	Description	Special requirements	Your data set name
SHAADB RM	DBRM library for DB2 Automation Tool		
SHAALOAD	Executable load module library for DB2 Automation Tool	You must APF authorize this data set.	
SHAAMENU	ISPF messages for DB2 Automation Tool		
SHAAPENU	ISPF panels for DB2 Automation Tool		
SHAASAMP	Sample members for DB2 Automation Tool		
SHAADENU	Metadata library for DB2 Automation Tool		
SHAASLIB	ISPF skeleton library for DB2 Automation Tool		

Data set names of FEC (common code)

If the FEC (common code) data sets were installed into their own target libraries and not into the DB2 Automation Tool libraries, identify and record the following FEC data set names. During the customization process, you will enter the following values on panel CCQPPRD.

Data set name	Description	Special requirements	Your data set name
SFECDBRM	DBRM library for FEC common code.		

Data set name	Description	Special requirements	Your data set name
SFECLOAD	Executable load module library for FEC common code.	You must APF authorize this data set.	
SFECMENU	ISPF messages for FEC common code.		
SFECPENU	ISPF panels for FEC common code.		
SFECSAMP	Sample members for FEC common code.		

Data set names of other libraries

Identify and record the following data set names. During the customization process, you will enter the following values on the Setup panel.

Data set name	Description	Special requirements	Your data set name
Discover output data set	<p>Contains the output that is generated when you run the DB2 Automation Tool Discover EXEC.</p> <p>The DB2 Automation Tool Discover EXEC retrieves the metadata and values for the parameters from a previous customization of DB2 Automation Tool.</p> <p>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.</p>	You must have write access to this data set.	
Data store data set	<p>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, and LPAR when you customize products on the local LPAR.</p> <p>The default name of the data set is DB2TOOL.CCQ110.DATASTOR. You can change the default value on the Tools Customizer Settings panel.</p>	You must have write access to this data set.	

Data set name	Description	Special requirements	Your data set name
Product customization Library	<p>Contains the customization jobs that Tools Customizer generates for DB2 Automation Tool.</p> <p>To customize DB2 Automation Tool, 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:</p> <p><i>hlq.\$LPAR-name\$.xyzvrm</i></p> <p>where:</p> <ul style="list-style-type: none"> • <i>hlq</i> is the value of the Customization library qualifier field on the Tools Customizer Settings panel (CCQPSET) • <i>LPAR-name</i> is the four-character LPAR name • <i>xyzvrm</i> is the three-letter product identifier with the version, release, and modification level <p>For example, the data set name might be DB2TOOL.PRODUCT.CUST.\$MVS1\$.XYZ420.</p>	You must have write access to this data set.	

Worksheets: Gathering parameter values for DB2 Automation Tool

During the customization process, you need to provide parameter values for DB2 Automation Tool, 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 the Tools Customizer.

After the customization jobs are generated, they are grouped by the job sequence number. In this topic, the jobs use an 8-character member naming convention that follows the format of *ssjjjdd*, where:

- *ss* is the job sequence number, which is an alphabetic character (A - Z) followed by a numeric character (0 - 9). For example, a job sequence number is A0, A1, ..., Z9.
- *jjjj* is the first four characters of the job name. The product assigns the job name.
- *dd* is a set of two alphanumeric characters (AA - 99) that Tools Customizer assigns to identify a DB2 entry.

For example, the job shown as *ssCF2Add* in this topic might be generated as A4CF2AAA.

Customization values for the Discover EXEC

Description

Use the following worksheet to identify and record the customization values for the Tools Customizer Discover EXEC. The values in this worksheet are for extracting information from a product that has already been customized. During the customization process, you will enter these values on panel CCQPDSC.

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

Parameter	Sample or default value	Your value
Discover EXEC for extracting information from an already customized product		
Discover EXEC library The fully qualified data set name that contains the product Discover EXEC.	The metadata library that you specified on the Specify the Metadata Library panel (CCQPHLQ).	
Discover EXEC name The name of the Discover EXEC.	HAADISC	
Discover output data set The name of the data set for the output from the product Discover EXEC.	The name of the discover output library that you entered on the settings panel.	
Information for Discover EXEC section		
New Automation Tool load library This parameter specifies the new DB2 Automation Tool load library that was installed as part of the SMP/E process.	DB2TOOL.HAA420.SHAALOAD	
New FEC load library This parameter specifies the new FEC load library if FEC was not installed into the same libraries as DB2 Automation Tool.	DB2TOOL.FEC130.SFECLOAD	
Previous control file This parameter specifies the DB2 control file used with the previous version of DB2 Automation Tool.	HAA.DB2.CONTROL	
Configuration ID The DB2 Automation Tool three-character configuration name.	HAA	

Parameter	Sample or default value	Your value
Previous Startup CLIST data set This parameter specifies the data set in which the CLISTs for the previous version of DB2 Automation Tool are located.	DB2TOOL.HAA410.SHAASAMP	
Previous Startup CLIST 2 This parameter specifies the name of the second of two startup CLISTs that were configured for the previous version of DB2 Automation Tool.	HAAV41C	

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 DB2 Automation Tool metadata data set.

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 Product to Customize panel (CCQPHLQ).
LPAR The LPAR field displays the LPAR on which you are customizing DB2 Automation Tool.	No	This value is supplied by Tools Customizer.
Product name This value displays the product that is being customized. In this example, IBM DB2 Automation Tool 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 <i>Vn.Rn.nn</i> .	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 Tools Customizer Settings panel (CCQPSET)

Required parameters section

Description

The parameters in this section are used by multiple steps of the customization process; refer to the parameter descriptions for details. During the customization process, you will enter these values on panel CCQPPRD.

Parameter	Required?	Discovered?	Default value	Your value
Startup CLIST library This parameter specifies the CLIST library in which to store the newly created DB2 Automation Tool CLISTs. These CLISTs are used to define the ISPF environment and to invoke the DB2 Automation Tool ISPF interface.	Yes	Yes	DB2TOOL.HAA420. CLIST	
Automation Tool load library This parameter specifies the load library for the new release of DB2 Automation Tool. This library can optionally contain the FEC load modules.	Yes	Yes	DB2TOOL.HAA420. SHAALOAD	
Solution Pack load library This parameter specifies the library name for the DB2 Utilities Solution Pack load modules. If the DB2 Utilities Solution Pack (BBY) was not purchased, this value must be left blank.	No	No	None	
Automation Tool panel library This parameter specifies the library for the DB2 Automation Tool panels.	Yes	Yes	DB2TOOL.HAA420. SHAAPENU	
User skeleton library This parameter specifies the skeleton library in which user-defined skeletons are stored. DB2 Automation Tool utilizes user-defined skeletons when the library is defined in the CLIST concatenation.	No	Yes	None	
Automation Tool skeleton library This parameter specifies the DB2 Automation Tool library that contains the product skeletons.	Yes	Yes	DB2TOOL.HAA420. SHAASLIB	
FEC common code load library This parameter specifies the library name for the FEC common code load modules if FEC was installed in a separate library from the DB2 Automation Tool load modules.	No	Yes	None	

Parameter	Required?	Discovered?	Default value	Your value
Control file This parameter specifies the control file that DB2 Automation Tool-specific information will be stored in. This file is a KSDS VSAM file, and can be shared with other DB2 tools.	Yes	No	HAA.DB2. CONTROL	
Configuration ID This parameter identifies the work environment. You can run multiple configurations of DB2 Automation Tool on the same LPAR, DB2 subsystem, or both. All of these configurations can use the same control file. See “Creating multiple configurations of DB2 Automation Tool” on page 197 for more information.	Yes	No	HAA	
Automation Tool SAMPLIB data set This parameter specifies the DB2 Automation Tool sample library.	Yes	No	DB2TOOL.HAA420. SHAASAMP	
Owner of plans and packages This parameter specifies the owner to be used in the plan and package binds. This is a global value and is used for all DB2 subsystems where a subsystem-specific value is not specified. If all of your DB2 subsystems use the same value, it can be defined once here.	Yes	No	HAAUSER	
Qualifier for repository object names This parameter specifies the qualifier for the DB2 Automation Tool repository objects. This value is used for object names in SQL, binding packages, and in the DDL that creates repository objects. This is a global value and is used for all DB2 subsystems where a subsystem-specific value is not specified. If all of your DB2 subsystems use the same value, it can be defined once here.	Yes	No	DLC	
Automation Tool DBRM library The DB2 Automation Tool library that contains the product-supplied DBRMs. The DBRMs are inputs to the bind process.	Yes	No	DB2TOOL.HAA420. SHAADBRM	
Automation Tool customization package name This parameter specifies a temporary package name that is used during customization.	Yes	No	HAA42TMK	

Parameter	Required?	Discovered?	Default value	Your value
DB2 Autonomics Director load library This parameter specifies the library name for the DB2 Autonomics Director load modules. If the DB2 Autonomics Director is not available, this field must be left blank.	No	No	None	
DB2 Autonomics Director DBRM library This parameter specifies the library name for the DB2 Autonomics Director DBRMs. If the DB2 Autonomics Director is not available, this field must be left blank.	No	No	None	

Task: Configure the product CLISTS

Description

This task builds the jobs to configure the startup CLISTS. During the customization process, you enter these values on panel CCQPPRD.

This task is *required*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Job name	Template	Description
ssV42	HAAV42	Configures and adds startup CLIST 1 to CLIST library.
ssV42C	HAAV42C	Configures and adds startup CLIST 2 to CLIST library.
ssEXECS	HAAEXECS	Adds DB2 Automation Tool required EXECs to CLIST library.

Required authority

The user ID that configures or updates CLISTS must have write access to the startup CLIST library.

Step or parameter	Required?	Discovered?	Default value	Your value
Configure Startup CLISTS Step to configure the CLISTS.	Yes	–	Selected	
Startup CLIST 1 The name of the first startup CLIST.	Yes	No	HAAV42	
Startup CLIST 2 The name of the second startup CLIST.	Yes	No	HAAV42C	
Automation Tool message library This parameter specifies the Automation Tool library that contains the product messages.	Yes	Yes	DB2TOOL.HAA420. SHAAMENU	

Step or parameter	Required?	Discovered?	Default value	Your value
FEC message library This parameter specifies the library for the FEC messages. This data set is optional if the FEC modules were installed into the same data set as the DB2 Automation Tool messages.	No	Yes	None	
FEC panel library This parameter specifies the library for the FEC panels. This data set is optional if the FEC modules were installed into the same data set as the DB2 Automation Tool panels.	No	Yes	None	
EMC load libraries This parameter specifies the EMC Symmetrix load library if you plan to use DB2 Automation Tool with DB2 Recovery Expert, and the EMC load library (not the SCF load library) is not in the z/OS LNKLIST.	No	Yes	None	
Copy the CLISTS This step copies the required product CLISTS into the specified data set.	Yes	–	Selected	
FEC SAMPLIB This parameter specifies the FEC sample library where the FECTSOC CLIST resides and is only required if FEC was not installed into the same libraries as DB2 Automation Tool	No	No	None	

Task: Create the control file

Description

This task creates a new control file. The control file is a required VSAM KSDS used by DB2 Automation Tool to store certain product and DB2 related values. It is not necessary to create a new one if one exists. DB2 Automation Tool can share a control file with other DB2 Tools products that use the same type of control file. During the customization process, you enter these values on panel CCQPPRD.

This task is *optional*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Job name	Template	Description
ssCNTFL	HAACNTFL	Creates the control file.

Required authority

The user ID that runs the job to create the control file must have the authority to create the control file data set.

Step or parameter	Required?	Discovered?	Default value	Your value
Create a new control file Step to create the control file.	No	–	Selected	
Volume serial number for control file This parameter specifies the volume serial number to use when creating the control file. If this parameter is left blank, SMS will select the appropriate volume.	No	No	None	

Task: Update the control file

Description

This task updates the DB2 Automation Tool control file. During the customization process, you enter these values on panel CCQPPRD.

This task is *required*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Job name	Template	Description
ssCF2dd	HAACF2A	Updates the control file (SSID-specific parameters).
ssCF1A	HAACF1A	Updates the control file with parameters for DB2 Recovery Expert.
ssCF1B	HAACF1B	Updates the control file with parameters for DB2 High Performance Unload.

Required authority

The user ID that runs the jobs to update the control file must have write authority to the control file data set.

Step or parameter	Required?	Discovered?	Default value	Your value
SSID-specific information Step to update the control file with SSID-specific parameters.	Yes	–	Selected	
DB2 Recovery Expert information Specify this step if you want to use DB2 Recovery Expert with DB2 Automation Tool. This step updates the control file with parameters required by DB2 Automation Tool to integrate with DB2 Recovery Expert.	No	--	Not selected	

Step or parameter	Required?	Discovered?	Default value	Your value
DB2 Recovery Expert control file This parameter is applicable only if you plan to use DB2 Recovery Expert with DB2 Automation Tool. If DB2 Recovery Expert has a separate control file from DB2 Automation Tool, specify the name. If this field is left blank, it is assumed that DB2 Recovery Expert uses the same control file as DB2 Automation Tool.	No	Yes	None	
DB2 Recovery Expert load libraries This parameter is applicable only if you plan to use DB2 Recovery Expert with DB2 Automation Tool. Specify the names of the data sets that make up the current load library concatenation for DB2 Recovery Expert.	No	Yes	DB2TOOL.ARY310. SARYLOAD	
DB2 HPU information This step updates the control file with parameters required by DB2 Automation Tool to integrate with IBM DB2 High Performance Unload.	No	--	Not selected	
DB2 HPU load libraries This parameter is applicable only if you use DB2 HPU with DB2 Automation Tool. Specify the names of the data sets that comprise the current load library concatenation for DB2 HPU.	No	Yes	DB2TOOL.HPU420. SHPULOAD	
DB2 HPU ver/rel/mod This parameter specifies the version, release, and modification values of DB2 High Performance Unload. It is required if you plan to use HPU with DB2 Automation Tool.	No	Yes	420	

Task: DB2 Utilities Solution Pack Setup

Description

This task prepares DB2 Automation Tool to run in a DB2 Utilities Solution Pack environment. If you have not purchased DB2 Utilities Solution Pack, do not select this task. During the customization process, you select this task on panel CCQPPRD.

This task is *optional*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Job name	Template	Description
ssUSLdd	HAAUSL	Sets up DB2 Automation Tool for use with DB2 Utilities Solution Pack.

Required authority

The user ID that runs this job must have SYSADM authority on the DB2 subsystems.

Step or parameter	Required?	Discovered?	Default value	Your value
Register with Autonomics Director This step generates a job that defines DB2 Automation Tool to the DB2 Utilities Solution Pack.	No	--	Not selected	

Task: Create or update DB2 objects

Description

This set of tasks creates the DB2 Automation Tool repository objects for your version of DB2. During the customization process, you enter these values on panel CCQPPRD.

The repository jobs determine whether repository objects need to be created or updated based on the version and release of DB2 Automation Tool, and the DB2 version on which DB2 Automation Tool is being installed. Ensure that the mode and levels are correct for each SSID. Run the generated job on the LPAR appropriate for the subsystem.

DDL changes to the DB2 Automation Tool repository are not common. However, missing columns or objects can result in unpredictable behavior if those items are required for functionality. If in doubt, submit the DDL and allow DB2 Automation Tool to update the repository.

This task is *required*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Job name	Template	Description
ssDELEdd	HAADELE	Drops the DB2 Automation Tool repository.
ssDDLydd	HAADDLY	Creates and updates the DB2 Automation Tool repository in preview mode.
ssDDLXdd	HAADDLX	Creates and updates the DB2 Automation Tool repository in execute mode.
ss#IDCdd	HAA#IDC1	Creates user-managed index files on SYSIBM.SYSCOPY.
ssDDL2dd	HAADDL2	Create optional STOGROUP-defined indexes on SYSIBM.SYSTABLES, SYSIBM.SYSTABSTATS, SYSIBM.SYSTABLES_HIST, SYSIBM.SYSTABSTATS_HIST, and SYSIBM.SYSCOPY.
ss#DCPdd	HAA#DCPY	Copy the DB2 Automation Tool repository.

Required authority

The user ID that runs these jobs to create DB2 objects must have SYSADM authority on the DB2 subsystems.

Step or parameter	Required?	Discovered?	Default value	Your value
<p>Drop repository This step generates a job to drop all DB2 Automation Tool repository objects. If this step is selected, you must edit the DB2 parameters for each subsystem that you want to drop the repository from and set the Drop Automation Tool repository parameter to Y.</p> <p>Attention: Submitting the DDL generated by this step will delete all existing profiles and all record of prior usage.</p>	No	–	Not selected	
<p>Preview repository changes This step generates an output file that shows a preview of the DDL that will be run. If you are creating a new repository for this installation of DB2 Automation Tool, the output lists the objects that will be created. If you are updating an existing repository, the output shows you the changes to the objects that will be made when the DDL is run.</p>	Yes	–	Selected	
<p>Execute repository changes This step executes the DDL changes shown in the preview file either to create a new repository or to update an existing repository for the new version of DB2 Automation Tool that is being installed.</p>	Yes	–	Selected	
<p>Create optional user-managed index files This step creates index files for a user managed index on SYSIBM.SYSCOPY.</p>	No	–	Not selected	
<p>Volume serial number for SYSCOPY index Specify the volume of the VSAM data set to be used for the optional SYSCOPY index. Or, leave this field blank and allow SMS to select the volume.</p>	No	No	None	

Step or parameter	Required?	Discovered?	Default value	Your value
Create optional indexes This step creates optional indexes on catalog tables SYSIBM.SYSTABLES, SYSIBM.SYSTABSTATS, SYSIBM.SYSTABLES_HIST, SYSIBM.SYSTABSTATS_HIST, and SYSIBM.SYSCOPY. These indexes speed up access to these tables when performing certain maintenance and utility functions. Adding these indexes enables DB2 to use the additional index to access the tables instead of table space scans.	No	--	Not selected	
Copy Automation Tool repository This step copies an existing DB2 Automation Tool repository for the purpose of creating multiple configurations of DB2 Automation Tool on one SSID. Multiple repositories might be useful for creating a production configuration and a test configuration on the same SSID.	No	-	Not selected	
Qualifier for previous repository object names This parameter specifies the high-level qualifier used for the previous repository when copying from one repository to another. This is a global value and can be used for all DB2 subsystems where a subsystem-specific value is not specified. If all of your DB2 subsystems use the same value, you can set it here and not set it for each subsystem.	No	No	DLC	
From ver/rel/mod This parameter specifies the version, release, and modification numbers of the source tables when copying repository tables.	No	No	410	

Task: Automation Tool Plans and Packages

Description

This task binds the plans and packages and required for using DB2 Automation Tool, optionally frees plans and packages before a rebind, and grants execute authority on the plans. During the customization process, you enter these values on panel CCQPPRD.

This task is *required*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Job name	Template	Description
ssFREEdd	HAAFREE	Frees the plans and packages.
ss#BNDdd	HAA#BND	Binds plans and packages.
ssGRANdd	HAAGRANT	Grants execute authority to use DB2 Automation Tool.

Required authority

The user ID that runs the ssFREEdd, ss#BNDdd and the ssGRANdd jobs must have SYSADM authority on the DB2 subsystems.

Step or parameter	Required?	Discovered?	Default value	Your value
FREE This step frees the Automation Tool plans and packages before they are rebound.	No	--	Not selected	
BIND This step binds the plans and packages that are required for this release of DB2 Automation Tool.	Yes	--	Selected	
Grant execute authority This step grants execute authority to the DB2 Automation Tool plans.	Yes	--	Selected	

Task: Management Console Stored Procedures

Description

This task creates and configures the stored procedures required for the Management Console. During the customization process, you enter these values on panel CCQPPRD.

This task is *optional*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Job name	Template	Description
ssFRSPdd	HAAFRSP	Frees the packages used by the Management Console stored procedures.
ssDRSPdd	HAADRSP	Drops the Management Console stored procedures.
ssDDLSDd	HAADDLS	Creates the Management Console stored procedures.
ssBNDSdd	HAABND2	Binds the DB2 Automation Tool packages used by the Management Console.
ssWLM1dd	HAAWLM1	Defines and runs the Management Console WLM address space.

Required authority

The user ID that runs the ssFRSPdd, ssDRSPdd, ssDRSPdd, ssDDLSDd, and the ssBNDSdd jobs must have SYSADM authority on the DB2 subsystems.

Step or parameter	Required?	Discovered?	Default value	Your value
Free Management Console packages This step frees the packages used by the Management Console stored procedures.	No	--	Not selected	
Drop Management Console stored procedures This step drops the Management Console stored procedures.	No	--	Not selected	
Create Management Console stored procedures This step creates the stored procedures used in Management Console to display DB2 Automation Tool information.	Yes	--	Selected	
Management Console BIND This step binds the DB2 Automation Tool packages used by Management Console.	Yes	--	Selected	
Create Management Console WLM address space This step creates JCL that can be used to run a WLM address space in which Management Console stored procedures are run.	Yes	--	Selected	

Task: Job Tracking Started Task

Description

DB2 Automation Tool uses a started task to track job execution results and to enable the restartability feature for DB2 Automation Tool-generated utility jobs. During the customization process, you enter these values on panel CCQPPRD.

This task is *optional*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Jobname	Template	Description
ssPROC	HAAPROC	Generates the job tracking started task procedure.
ssPARMS	HAAPARMS	Generates the job tracking started task parameter file.

Required authority

The user ID that runs the *ssPROC* and *ssPARMS* jobs to create the job tracking task procedure and parm file must have write access to the libraries where the started task and started task PARMLIB member will be written.

Step or parameter	Required?	Discovered?	Default value	Your value
Create the job tracking started task This step creates the job tracking started task and its associated parameters.	No	–	Not selected	

Task: Install the sample profiles

Description

This task installs sample profiles for use with DB2 Automation Tool. These profiles are optional, but can help you quickly set up job profiles to perform maintenance. During the customization process, you enter these values on panel CCQPPRD.

This task is *optional*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Jobname	Template	Description
ssLOADdd	HAALOADS	Installs the sample profiles.

Required authority

The user ID that runs the ssLOADdd job to install the sample profiles must have SYSADM authority on the DB2 subsystems and read access to the DB2 Automation Tool sample library (SHAASAMP).

Step or parameter	Required?	Discovered?	Default value	Your value
Install sample profiles This step creates a batch job that loads the DB2 Automation Tool repository tables with the sample profiles.	No	–	Selected	
Sample profile creator name This parameter specifies the creator name to be used for the sample profiles that are loaded into the DB2 Automation Tool repository tables. This value is required if installing the sample profiles.	No	No	HAAUSER	
Sample profile update option This parameter specifies the update option that is assigned to the sample profiles. The following values are valid: <ul style="list-style-type: none"> • U - any user can view, update, or export the profile • V - Only the profile creator can update the profile, but any user can view or export it • N - Only the profile creator can view, update, or export the profile 	No	No	V	

Task: Add Automation Tool to the DB2 Admin Launchpad

Description

This task customizes the EXEC that, when executed, adds DB2 Automation Tool to the DB2 Administration Tool Launchpad. During the customization process, you enter these values on panel CCQPPRD.

This task is *optional*.

Jobs generated

This task generates the following jobs based on the template that is listed:

Jobname	Template	Description
ssADBI	HAAADBI	Creates the ssADBI EXEC that, when run, adds DB2 Automation Tool to the DB2 Administration Tool Launchpad.

Required authority

The user ID that runs ssADBI job must have write access to the DB2 Automation Tool CLIST library and must have the required authorities to modify the DB2 Administration Tool Launchpad.

Step or parameter	Required?	Discovered?	Default value	Your value
Create the REXX to add Automation Tool to the Launchpad This step creates the REXX exec and stores it in the CLIST library. It then runs the REXX EXEC to add DB2 Automation Tool to the Launchpad.	No	–	Selected	
DB2 Administration Tool high-level qualifier This parameter specifies the high-level qualifier of the DB2 Administration Tool data sets.	No	No	ADBHILVL	
ADBDMTI EXEC data set This parameter specifies the data set that contains the DB2 Admin Tool ADBDMT EXEC.	No	No	ADBHILVL. SADBEXEC	
DB2 Admin version This parameter specifies the version of DB2 Administration Tool that is installed at your site.	No	No	10	

DB2 Parameters section

Description

This section contains DB2 parameters. During the customization process, you enter these values on panel CCQPDB2.

Parameter	Required?	Discovered?	Default value	Your value
Group attach name If a subsystem is the member of a data sharing group, this parameter indicates the group attach name.	No	No	None	

Parameter	Required?	Discovered?	Default value	Your value
General DB2 Information - common				
Mode This parameter indicates the mode in which the DB2 subsystem is running. The following values are valid: <ul style="list-style-type: none"> • 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. 	Yes	No	NFM	
Level number This parameter indicates the version, release, and modification level of the DB2 subsystem. The following values are valid: <ul style="list-style-type: none"> • 101 is valid only for modes CM8, CM9 or NFM. • 111 is valid only for modes CM or NFM. 	Yes	No	None	
DB2 Libraries - common				
Load library The names of the data sets that make up the current load library concatenation for DB2. The load library usually consists of a subsystem-specific DSNEXIT library, the base DSNEXIT library for the current DB2 version, and the base DSNLOAD library for the current DB2 version.	Yes	Yes	DSN.SDSNLOAD	
Run library This parameter indicates the data set name of the DB2 run library where DSN1TIAD is located.	Yes	No	DSN.RUNLIB.LOAD	
Exit library This parameter indicates the names of the data sets that make up the current DB2 exit library concatenation for DB2.	Yes	Yes	DSN.SDSNEXIT	
Bootstrap data set This parameter indicates the names of the DB2 bootstrap data sets.	Yes	Yes	DSN.SDSNBSDS	
DB2 Buffer Pools - common				

Parameter	Required?	Discovered?	Default value	Your value
Name of the 4 KB bufferpool This parameter indicates the name of the 4 KB buffer pool to be used for customization. The value must be 8 characters or fewer.	Yes	No	BP0	
Name of the 16 KB bufferpool This parameter indicates the name of the 16 KB buffer pool to be used for customization. The value must be 8 characters or fewer.	Yes	No	BP16K0	
Name of the 32 KB bufferpool This parameter indicates the name of the 32 KB buffer pool to be used for customization. The value must be 8 characters or fewer.	Yes	No	BP32K	
DB2 Utilities - common				
SYSAFF for DB2 utilities This parameter indicates the SYSAFF job parameter to use for running DB2 batch utility jobs. The value must be 8 characters or fewer.	No	No	None	
Plan name for the DSNTIAD utility This parameter indicates the plan name for the DSNTIAD utility. The value must be 8 characters or fewer.	Yes	No	DSNTIAD	
DB2 Tools Objects - common				
Storage group name This parameter indicates the name of the storage group that will be used for creating DB2 objects for customization. The value must be 128 characters or fewer.	Yes	No	SYSDEFLT	
SQL authorization ID The SQL authorization ID that will be used to create DB2 objects.	No	No	None	
Automation Tool DB2 Parameters				
SQL authorization ID for GRANT The name of the SQLID used in SET CURRENT SQLID statements for the job that issues GRANT EXECUTE for DB2 Automation Tool plans. This value overrides SQL authorization ID (but only for the job that issues GRANT EXECUTE on the plans).	No	No	None	

Parameter	Required?	Discovered?	Default value	Your value
Repository table space 4KB buffer pool This parameter indicates the name of the 4 KB buffer pool to be used for DB2 Automation Tool repository table spaces. If left blank, the value in the Name of the 4 KB buffer pool field is used.	No	No	None	
Repository index 4KB buffer pool This parameter indicates the name of the 4 KB buffer pool to be used for DB2 Automation Tool repository index spaces. If left blank, the value in the Name of the 4 KB bufferpool field is used.	No	No	None	
Storage group name for table spaces The name of the storage group that will be used for creating DB2 Automation Tool table spaces. If blank, the value in the Storage group name field is used.	No	No	None	
Storage group name for index spaces The name of the storage group that will be used for creating DB2 Automation Tool index spaces. If blank, the value in the Storage group name field is used.	No	No	None	
Owner of Automation Tool plans and packages This parameter is the BIND owner for plan and package binds. This value is subsystem-specific. If specified, it overrides the global value specified in the product parameters.	Yes	No	None	
Schema for repository object names This parameter identifies the schema used for the Automation Tool repository table names. This value is used when repository objects are created, altered, referenced in SQL, or bound into packages. This value is subsystem-specific and is optional. If specified, this value overrides the global qualifier specified in the product parameters.	No	No	None	

Parameter	Required?	Discovered?	Default value	Your value
Model DSN for GDG base If your site is managed by SMS, and you are required to include a model DSCB in your JCL, specify a model DSCB name. The GDG base must already exist. This parameter is optional if your data sets are managed by SMS. If you provide the base name, DB2 Automation Tool will use it.	No	Yes	None	
DB2 ZPARMs member The ZPARM load module member name for this specific DB2 subsystem. This module typically resides in the DB2 SDSNEXIT library.	Yes	Yes	DSNZPARM	
Plan for online navigation The name of the plan that includes most DB2 Automation Tool functions, except for the job building function and the data page display function.	Yes	Yes	HAAP4201	
Plan for building jobs The name of the plan that includes the function for building DB2 Automation Tool jobs.	Yes	Yes	HAAP4202	
Plan for Data Page Display in browse mode The name of the plan that includes the data page display browse function.	Yes	Yes	HAAP4203	
Plan for Data Page Display in edit mode The name of the plan that includes the data page display edit function.	Yes	Yes	HAAP4204	
Database for repository objects The database name for DB2 Automation Tool repository objects.	Yes	No	DLADB	
Default WLM environment The WLM environment is required only if no valid default WLM environment has been set up. If specified, it is used for a DB2 Automation Tool native SQL stored procedure. There is no default value.	No	No	None	
Table space PRIQTY This value is the primary quantity (PRIQTY) value used to create DB2 Automation Tool table spaces.	Yes	No	200	

Parameter	Required?	Discovered?	Default value	Your value
Table space SECQTY This value is the secondary quantity (SECQTY) value used to create DB2 Automation Tool table spaces.	Yes	No	200	
Index space PRIQTY This value is the primary quantity (PRIQTY) value used to create DB2 Automation Tool indexes.	Yes	No	100	
Index space SECQTY This value is the secondary quantity (SECQTY) value used to create DB2 Automation Tool indexes.	Yes	No	100	
Drop Automation Tool repository This parameter is only displayed if the Drop repository step on the Product Parameters panel (CCQPPRD) is selected. If this parameter is set to Y, a job is generated that contains DDL to drop the repository from that subsystem. If set to N, then a no-op (IEFBR14) job is generated. Attention: Submitting the DDL generated by this step will delete all existing profiles and all record of prior usage.	No	No	N	
Automation Tool User Exits				
User exits with SQL The names of the user exits that contain SQL and that must be bound into packages.	No	No	None	
User exit package list The package list name for any user exits that are used at this site.	No	No	None	
Automation Tool Shared Profile Packages				
Catalog history package list The name of the catalog history package list (PKLIST). This package stores RUNSTATS statistics in the DB2 catalog.	Yes	Yes	HAAC420C	
Shadow catalog package list The name of the shadow catalog package list (PKLIST). This package stores RUNSTATS statistics in the shadow history tables.	Yes	Yes	HAAC420S	

Parameter	Required?	Discovered?	Default value	Your value
Repository package list The package list name for most DB2 Automation Tool functions. This includes maintaining RUNSTATS statistics in DB2 Automation Tool.	Yes	Yes	HAAC420	
Automation Tool Shared Profile Devices				
Work file device type The default work file unit device to be used in generated jobs. Sample values are SYSDA and DISK.	Yes	Yes	SYSALLDA	
Sort work file device type The sort work file unit device to be used when utility JCL is generated. Sample values are SYSDA and DISK.	Yes	Yes	SYSALLDA	
Automation Tool Shared Profile Build Message DD Names				
Build informational DD The DDNAME to be used for informational messages when a job is built. All generated xxxBnnnI messages will be written to this DD.	Yes	Yes	HAAERROR	
Build warning DD The DDNAME to be used for warning messages when a job is built. All generated xxxBnnnW messages will be written to this DD.	Yes	Yes	HAAERROR	
Build error DD The DDNAME to be used for error messages when a job is built. All generated xxxBnnnE messages will be written to this DD.	Yes	Yes	HAAERROR	
Automation Tool Shared Profile Miscellaneous				
Job tracking subsystem name The name of the job tracking task subsystem that will track jobs for this DB2 subsystem. The job tracking task's subsystem name is defined in the SUBSYS parameter of the job tracking task's HAAPARMS member.	No	Yes	None	

Parameter	Required?	Discovered?	Default value	Your value
Max primary space allocation The maximum amount of space that can be allocated for a primary allocation. You can specify up to 999999 tracks, cylinders, or megabytes. The value that you specify will be converted to cylinders when the job is built. This value affects primary space allocation for image copy DDs and temporary DDs that are used in REORGs, and it also affects utility jobs that are built with TEMPLATE syntax.	Yes	Yes	999999	
Size unit for max primary space allocation The unit for the primary space allocation. Valid values are T for tracks, C for cylinders, and M for megabytes.	Yes	Yes	M	
Secondary allocation percentage The amount of space that can be allocated for a secondary allocation. This value is expressed as a percentage of primary allocation. This value affects secondary space allocation for image copy DDs and temporary DDs that are used in REORGs, and it also affects utility jobs that are built with TEMPLATE syntax. Valid values are 1 - 999.	Yes	Yes	050	
Utility region size The default REGION size, in megabytes, to be used when generating utility JCL. The region size is set on the job step and the value is used for all job steps. If you include a REGION parameter in your job card, the job card REGION parameter overrides the REGION parameter on the EXEC statement.	Yes	Yes	1024	
DB2 fetch buffer size The size of the DB2 fetch buffer in MB. The fetch buffer is used for the DB2 V8 multi-row fetch capability and can improve SQL performance. Valid values are 1 - 2047 MB.	Yes	Yes	0004	
Parallel MVS™ catalog locates The number of parallel processing tasks to be created when doing MVS catalog LOCATE operations. Valid values are 1 - 99.	Yes	Yes	10	

Parameter	Required?	Discovered?	Default value	Your value
Terminate utility if an abend occurs This parameter indicates whether to terminate a utility if an abend occurs when the utility is run. This value can be changed after configuration by using option 0 on the DB2 Automation Tool main menu.	Yes	Yes	N	
Generate STEPLIB DDs This parameter indicates whether STEPLIB DDs are to be included in the JCL. If you specify N, STEPLIB DDs are not included. When N is specified, the DB2 product libraries must be contained in the LNKLST set.	Yes	Yes	Y	
Gen image copy DSNs in GMT This parameter indicates whether to use Greenwich Mean Time (GMT) or local time when image copy data set names are created. Valid values are Y (Greenwich Mean Time) and N (the local time). If you want date and time variables in templates to be resolved at job run time rather than job build time, specify Y.	Yes	Yes	Y	
Indexes of DEFINE NO table spaces This parameter indicates whether to include indexes when exploding a table space that was defined with DEFINE NO and does not yet exist. Even though the table space does not exist, if the index exists, this setting allows DB2 Automation Tool to process the index.	Yes	No	N	
Enable administrative task scheduler support This parameter indicates whether to enable DB2 Automation Tool support for DB2 administrative task scheduler. This support is available only if the DB2 administrative task scheduler is installed.	Yes	Yes	N	

Parameter	Required?	Discovered?	Default value	Your value
<p>Administrative task scheduler max history The number of task executions that DB2 Automation Tool retrieves from the DB2 administrative task scheduler when a task execution status list is viewed. DB2 Automation Tool uses this value only to determine how many task executions to retrieve. It does not define the Maximum history parameter to the DB2 administrative task scheduler. This parameter is used only if the DB2 administrative task scheduler product is installed and enabled.</p>	Yes	Yes	0010	
<p>Segment size for repository table spaces The segment size (number of pages per segment) for the DB2 Automation Tool repository segmented table spaces. Valid values are multiples of 4 in the range of 4 - 64.</p>	Yes	No	4	
<p>Event notifications This value controls whether a user is notified when a selected event occurs. Valid values are Y and N (yes and no). Selected events include when a batch job build begins and when a batch job build ends. This value is required.</p>	Yes	No	N	
<p>Sysout Class for SMTP output This value is defined as part of the JES set up on the LPAR for this subsystem ID. Ask your systems programmer for a valid value for your site. This value is ignored unless Event notifications events is set to Y.</p>	Yes	No	A	
<p>Z/OS Host Name This parameter is a JES node (system name). This is the JES name of the LPAR for the current subsystem ID. This name can be located in any job log, or ask your systems programmer. This value is ignored unless Event notifications is set to Y.</p>	Yes	No	JESNODE	

Parameter	Required?	Discovered?	Default value	Your value
SMTP task name This parameter is the SMTP task name used for sending and receiving event notification email notifications. This value is ignored unless Event notifications is set to Y.	Yes	No	SMPTTASK	
Automation Tool Shared Profile Sort Parameters				
Primary sort work space This parameter indicates an override amount for the calculated primary sort work space, specified in cylinders.	No	Yes	None	
Secondary sort work space This parameter indicates an override amount for the calculated secondary sort work space, specified in cylinders.	No	Yes	None	
Number of sort work DDs This parameter indicates an override amount for the calculated number of sort work DD statements.	No	Yes	None	
Automation Tool Stored Procedure Parameters				
Management Console package list This parameter specifies the name of the Management Console package list. This value is required when Management Console is used.	No	No	MCCOLL1	
Management Console PROC member This parameter specifies two values for the Management Console stored procedures. It is the PROC name and the data set member name of the JCL PROC that runs the address space for the Management Console stored procedures.	No	No	WLMPROC	
Management Console WLM environment This parameter specifies the WLM environment name for the Management Console stored procedures. The environment name is defined to WLM in WLM panels. It is defined to DB2 in the stored procedure definition (DDL). If this parameter is not specified, the WLM PROC name is used for the WLM environment name. Leaving this parameter unspecified is recommended.	No	No	None	

Parameter	Required?	Discovered?	Default value	Your value
Management Console WLM NUMTCB This parameter specifies the maximum number of concurrent stored procedures that can be active at one time in the Management Console WLM address space. This parameter is required when the Management Console stored procedures are used.	No	No	8	

LPAR Parameters section

Description

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

Parameter	Required?	Discovered?	Default value	Your value
Job Entry Subsystem (JES)				
JES version The version of the job entry system (JES) that is installed on the LPAR.	Yes	No	3	
ISPF Libraries				
Message library The data set name of the ISPF message library.	Yes	Yes	ISP.SISPMENU	
ISPF table input library The data set name of the ISPF table input library.	Yes	Yes	ISP.SISPTENU	
Language Environment® Libraries				
Load library The data set name of the Language Environment load library.	Yes	Yes	CEE.SCEELLIB	
Automation Tool LPAR Parameters				
Automation Tool PROCLIB This parameter specifies the data set name where you want to store JCL procedures (PROCS) used with DB2 Automation Tool. This data set is used for the job tracking started task and the Management Console stored procedures.	No	No	HAA.PROCLIB	
Job tracking started task PROC member This parameter specifies the member name of the job tracking started task. The member name can be the same name as the job tracking started task ID.	No	No	HAAPROC	

Parameter	Required?	Discovered?	Default value	Your value
Job tracking started task Parms data set This parameter specifies the data set where the job tracking started task parameters are located. The data set can be sequential or a PDS. If it is a PDS, the member name must be included in parentheses.	No	No	HAA.PARMLIB (HAAPARM)	
Job tracking started task ID This parameter specifies the name of the job tracking started task. Use a unique name for each job tracking task that will run on the LPAR. To prevent confusion, specify a name other than the name of a DB2 subsystem. Tip: This parameter can be the same as the job tracking started task PROC member name.	No	No	HAAT	
Job tracking started task AUTHID This parameter specifies the authorization ID of the job tracking started task and will be used to connect to DB2 when saving data to the repository. This ID must be a valid RACF user that can insert and update rows in the job tracking tables. This user ID cannot be a RACF GROUP name, but must be at the RACF USER level. You can use RACF STARTED class profiles that specify a RACF user and group name to assign to the started task. If you use RACF STARTED class profiles that specify a RACF user and group name, do not specify this parameter.	No	No	DB2USER	

Worksheets: DB2 High Performance Unload

Refer to the following information before you customize DB2 High Performance Unload.

Discover process parameters

The Discover process parameters on the Discover Customized Product Information panel (CCQPDSC) in Tools Customizer are required to run the Discover EXEC.

You can accept the default values for the Discover process parameters, or you can customize them based on your specific requirements. All Discover process parameters are required.

The following list describes the Discover process parameters.

Name of the PARMLIB library of a previous version (4.2.0 and later)

The name of PARMLIB library (INZPLIB/INFPLIB) of a previous version. Use

this field when the previous installation was done through TCz (DB2 HPU version 4.2.0 and later). Example: oldhlq.SINZAMP. There is no default value.

Previous data set of variables to be retrieved (4.1.0 and earlier)

The name of the data set that contains the variable to be discovered (INZRSAVE or INZTVAR member). Use this field when the previous installation was done without TCz (DB2 HPU version 4.1.0 and earlier). Example: oldhlq.SINZRAV(INZRSAVE). There is no default value.

Previous data set with customized file dsnames (4.1.0 and earlier)

The name of the data set that contains customized files dsnames (INZTDSN, member). Use this field when the previous installation was done without TCz (DB2 HPU version 4.1.0 and earlier). Exwample: oldhlq.SINZAMP(INZTDSN). There is no default value.

Verbose mode for discover procedure.....OFF (OFF,ON)

Specify ON to enable display of the trace information issued by the discover process. Otherwise, specify OFF.

DB2 HPU library dsnames

The DB2 HPU library dsnames section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the dsnames of the DB2 HPU libraries.

The following list describes the dsnames:

Load modules library (SINZLOAD)

This parameter is required. It specifies the dsname of the load modules library (SINZLOAD).

In previous releases of DB2 HPU, the name of this parameter was VIZ003.

APF load modules library (SINZLINK)

This parameter is required. It specifies the dsname of the APF load modules library (SINZLINK).

In previous releases of DB2 HPU, the name of this parameter was VIZ004.

Samples library (SINZSAMP)

The dsname of the samples library (INZSAMP).

PARMLIB library (INZPLIB/INFPLIB)

This parameter is required. It specifies the dsname of the PARMLIB library (INZPLIB). Do not use the SINZSAMP library. This library is handled by DB2 HPU under the INFPLIB ddname.

Requirement: This library must be the same as the one that is specified in the INZTDSN member. If you do not use this library, DB2 HPU cannot read the customization correctly.

In previous releases of DB2 HPU, the name of this parameter was VIZ007.

Product's DBRM library (SINZDBRM)

This parameter is required. It specifies the dsname of the DBRM library (SINZDBRM).

In previous releases of DB2 HPU, the name of this parameter was VIZ012.

REXX EXEC library (SINZCLST)

This parameter is required. It specifies he dsname of the REXX EXEC library (SINZCLST).

ISPF LOAD modules library (SINZLLIB)

This parameter is required to run the DB2 HPU interactive component. It specifies the dsname of the ISPF load module library (SINZLLIB) that contains the load modules.

In previous releases of DB2 HPU, the name of this parameter was VIZ013.

ISPF messages library (SINZMLIB)

This parameter is required to run the DB2 HPU interactive component. It specifies the dsname of the ISPF messages library (SINZMLIB) that contains the messages.

In previous releases of DB2 HPU, the name of this parameter was VIZ015.

ISPF panels library (SINZPLIB)

This parameter is required to run the DB2 HPU interactive component. It specifies the dsname of the ISPF panels library (SINZPLIB) that contains the ISPF panels.

In previous releases of DB2 HPU, the name of this parameter was VIZ016.

ISPF skeletons library (SINZSLIB)

This parameter is required to run the DB2 HPU interactive component. It specifies the dsname of the ISPF skeletons library (SINZSLIB) that contains the skeletons.

In previous releases of DB2 HPU, the name of this parameter was VIZ017.

ISPF tables library (SINZTLIB)

This parameter is required to run the DB2 HPU interactive component. It specifies the dsname of the ISPF tables library (SINZTLIB) that contains the tables.

In previous releases of DB2 HPU, the name of this parameter was VIZ018.

DB2 HPU DB2 parameters

The DB2 parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the DB2 parameters that are used by DB2 HPU.

You can accept the default values for the DB2 parameters, or you can configure them based on your specific requirements. Some DB2 parameters are mandatory and must be configured.

The following list describes the common DB2 parameters. The parameters are listed in the following format:

description (Vxxxxxxxx/parameter-name)

Percentage of displaced pages in inline FIC (VUX023/PGDFIN)

This parameter is optional. It specifies the estimated value of the percentage of displaced pages in inline full-image copies (FIC).

Valid values are 0 - 100. The default value is 20.

The corresponding SYSIN keyword is PGDFIN in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX023/PGDFIN.

Use real-time statistics tables for size estimation (VUX036/RTSESTIM)

This parameter is optional. Specify YES to use the real-time statistics tables, when they are available, to estimate the number of rows of the table space to

be unloaded. Otherwise, specify NO. This estimate is used to allocate the output data set when a TEMPLATE is used, to set the sort program parameters when an ORDER BY clause is specified, or both.

In previous releases of DB2 HPU, the name of this parameter was VUX036/RTSESTIM.

The default value is NO.

Maximum degree of parallelism for SELECT (VUX031/DB2PARAL)

This parameter is optional. It specifies the parallelism degree for an UNLOAD command when several SELECT statements are processed by using DB2 access (using DB2 FORCE or DB2 YES with unsupported SELECT statements). The value that you specify indicates the maximum number of SELECT statements that are processed in parallel. The value that you specify indicates the maximum number of SELECT statements that are processed in parallel.

In previous releases of DB2 HPU, the name of this parameter was VUX031/DB2PARAL.

Valid values are 1 - 65535. The default value is 5.

Application plan for DB2 HPU (VUM011/PLANOBJT)

This parameter is required. It specifies the plan name for the interactive application and the batch jobs.

Valid values are valid DB2 plan names.

In previous releases of DB2 HPU, the name of this parameter was VUM011/PLANOBJT.

Package collection for DB2 HPU (VUM030/COLLOBJT)

This parameter is required. It specifies the name of the collection that is used to bind the DB2 HPU packages.

In previous releases of DB2 HPU, the name of this parameter was VUM030/COLLOBJT.

Owner of the plan created for DB2 HPU (VUM012/PLANOWN)

This parameter is required. It specifies the name of the owner of the plan to be used to bind the DB2 HPU plan.

Specify one value for each DB2 subsystem that you defined.

In previous releases of DB2 HPU, the name of this parameter was VUM012/PLANOWN.

Grant on the plan created for DB2 HPU (VUX011)

This parameter is required. It specifies whether to use the GRANT TO PUBLIC or the GRANT TO USER statement to grant privileges to the plan that was created for DB2 HPU.

Specify one value for each DB2 subsystem that you defined with variable VZD001.

Valid values are PUBLIC and USER.

In previous releases of DB2 HPU, the name of this parameter was VUX011.

Quiesce of SYSDBASE and DBD01 for the batch utilities (VUM014/QUIESCAT)

This parameter is optional. It specifies whether a quiesce point is to be taken on the following table spaces before the job is run.

- DSNDB01.DBD01
- DSNDB06.SYSCOPY

- DSNDB06.SYSDBASE
- DSNDB06.SYSDBAUT
- DSNDB06.SYSGROUP
- DSNDB06.SYSOBJ
- DSNDB06.SYSSTATS
- DSNDB06.SYSUSER
- DSNDB06.SYSVIEWS

Specify one of the following values:

YES

A quiesce point is taken at run time unless keyword QUIESCECAT NO was specified in the SYSIN of DB2 HPU.

NO A quiesce point is not taken at run time unless keyword QUIESCECAT YES was specified in the SYSIN of DB2 HPU.

OFF

A quiesce point is never taken at run time, even if keyword QUIESCECAT YES was specified in the SYSIN of DB2 HPU.

FORCE

A quiesce point is always taken at run time, even if keyword QUIESCECAT NO was specified in the SYSIN of DB2 HPU.

QUIESCAT is forced to NO in DB2 10 for z/OS ENFM and later releases because DB2 HPU accesses the catalog in SQL only at that DB2 level.

The default value is YES.

In previous releases of DB2 HPU, the name of this parameter was VUM014/QUIESCAT.

User who quiesces the catalog table spaces (VUM020/QUIESUSR)

This parameter is optional. It specifies the user who will run the QUIESCE utility on the DB2 catalog table spaces. Specify one value for each DB2 subsystem that you defined with variable VZD001.

Specify one of the following values:

CURRENT_USER

The user who submits the job will be used to run the QUIESCE utility on the DB2 catalog table spaces.

USER *name*

A specific user name is used to the QUIESCE utility on the DB2 catalog table spaces. The name can be 1 -7 characters.

The default value is CURRENT_USER.

In previous releases of DB2 HPU, the name of this parameter was VUM020/QUIESUSR.

User who quiesces the table space to be unloaded (VUM031/QUITUSR)

This parameter is optional. If a quiesce is requested, this parameter specifies the user who will run the QUIESCE utility on the table space to be unloaded. Specify one value for each DB2 subsystem that you defined with variable VZD001.

Specify one of the following values:

CURRENT_USER

The user who submits the job is used to run the QUIESCE utility on the table space to be unloaded.

USER name

A specific user name is used to run the QUIESCE utility on the table space to be unloaded. The name can be 1 - 7 characters.

In previous releases of DB2 HPU, the name of this parameter was VUM031/QUITSUSR.

The default value is CURRENT_USER.

SQL access for reading DB2 catalog (VUM027/SQLACCES)

This parameter is optional. It specifies which of the following three methods is used to access the DB2 catalog:

DIRECT

Direct access on all catalog information

SQL

SQL access on all catalog information

MIXED

Direct access for all information except for the IPREFIX of the SYSINDEXPART and SYSTABLEPART tables (the I/J), which might change after an online reorganization

Specify one of the following values:

MINIMAL

The DB2 catalog is accessed by using DIRECT mode (DIRECT access maximum) and is completed by using the SQL mode (SQL access minimum).

AUTO

The DB2 catalog is accessed either in DIRECT mode or in SQL mode.

YES

The DB2 catalog is accessed only in SQL mode.

NO The DB2 catalog is accessed only in DIRECT mode.

When SQL access is used to read the DB2 catalog, the user ID that is specified in PARMLIB variable VUM032/ACTLGUSR must have SELECT authority on the tables of the DB2 catalog that DB2 HPU accesses.

The access method that is used depends on the SQLACCESS setting and whether QUIESCECAT comes from the PARMLIB or from the SYSIN. The following table shows the access method that is used by DB2 HPU based on the SQLACCESS and QUIESCECAT settings.

Table 2. DB2 catalog access dependencies

SQLACCES value	QUIESCECAT value	Access method used
YES	Any	SQL
NO	Any	DIRECT
AUTO	NO	SQL
AUTO	YES	DIRECT
MINIMAL	NO	MIXED

Table 2. DB2 catalog access dependencies (continued)

SQLACCES value	QUIESCECAT value	Access method used
MINIMAL	YES	DIRECT

For a table space that was not altered recently, using MIXED access with a QUIESCECAT setting of NO guarantees a consistent reading of information from the DB2 catalog, even if an online reorganization (other than an ALTER operation) was recently done. Using MINIMAL provides the best compromise between consistency and speed.

SQLACCES is forced to YES in DB2 10 for z/OS ENFM and later releases because DB2 HPU only accesses the catalog in SQL at that DB2 level.

The default value is MINIMAL.

The corresponding SYSIN keyword is SQLACCES in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUM027/SQLACCES.

User who runs the DISPLAY command (VUM028/DISPLUSR)

This parameter is optional. It specifies the user who will run the DISPLAY command.

CURRENT_USER

The user who submitted the job is used to run the DISPLAY command. CURRENT_USER is the default value.

USER *name*

A specific user name is used to run the DISPLAY command. The name can be 1 - 7 characters.

Specify one value for each DB2 subsystem that you defined with variable VZD001.

The default value is CURRENT_USER.

In previous releases of DB2 HPU, the name of this parameter was VUM028/DISPLUSR.

Use DB2 to process unsupported SELECT statements (VUU011/ULSEDB2)

This parameter is optional. It specifies whether DB2 will process SELECT statements when the statements are not supported by DB2 HPU. This parameter is used only if the DB2 option is not specified in the SYSIN. When COPYDDN_STRICT(YES) is specified in the VUU030/ULOPTNS parameter, unloading data from the table by using DB2 instead of unloading data from the image copy is not allowed. Therefore, if you are unloading from an image copy and COPYDDN_STRICT(YES) is specified, the value of the VUU001 parameter is forced to NO.

Specify one of the following values:

NO SELECT statements that are not supported by DB2 HPU will not be processed by DB2.

YES

SELECT statements that are not supported by DB2 HPU will be processed by DB2, unless DB2 NO was specified in the SYSIN of DB2 HPU.

The default value is YES.

In previous releases of DB2 HPU, the name of this parameter was VUU011/ULSEDB2.

Lock the tables in the table space (VUU012/ULLOCK)

This parameter is optional. It specifies whether to lock the tables in the table space.

Specify one of the following values:

NO Tables in the table space are not to be locked unless YES was specified in the SYSIN of DB2 HPU.

YES

Tables in the table space are locked unless NO was specified in the SYSIN of DB2 HPU.

Specifying LOCK NO does not preclude DB2 from taking locks if SQL Access is used to access the data.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU012/ULLOCK.

Quiesce the table space (VUU013/ULQSCE)

This parameter is optional. It specifies whether to quiesce the table space.

Specify one of the following values:

NO The table space is not quiesced unless YES was specified in the SYSIN of DB2 HPU.

YES

The table space is quiesced unless NO was specified in the SYSIN of DB2 HPU.

The default value is NO.

Attention: DB2 HPU operates on the physical VSAM data set level that is outside of DB2. If you run DB2 HPU on a table in which a row was just inserted, the unloaded data might not contain the row that was inserted. The unloaded data does not show the row because DB2 might not have externalized the data to DASD yet. This situation can happen when you use DB2 HPU without issuing a QUIESCE (or STOP) on the object. Be careful when using QUIESCE NO.

In previous releases of DB2 HPU, the name of this parameter was VUU013/ULQSCE.

Degree of parallel processing (VUU021/ULDEGREE)

This parameter is applicable only when DB2 extracts data. It specifies the number of parallel tasks or I/O operations that DB2 can use to extract data from a partitioned table space.

Specify one of the following values:

1 Parallelism is not used.

ANY

DB2 HPU decides whether parallelism will be used.

CURRENT_DEGREE

DB2 HPU must keep the default value for the current degree that is set in the DB2 customization. If the PARMLIB variable is set to

CURRENT_DEGREE, DB2 HPU will not issue any SET CURRENT DEGREE statements before issuing the SELECT statement.

The default value is CURRENT_DEGREE.

The corresponding SYSIN parameter is ULDEGREE in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUU021/ULDEGREE.

Default scheme for UNLOAD TABLESPACE (VUU023/UNLSCHEM)

This parameter is optional. It specifies the unload format for the data.

Specify one of the following values:

ASCII

Specifies that the unloaded data must be in ASCII format. DB2 HPU uses the ASCII CCSID of the subsystem, unless you override it by specifying the CCSID option in the SYSIN of DB2 HPU.

ASIS

Specifies that the data is unloaded in its original format. If the specification for the underlying table space cannot be determined (for example, if the data is processed by DB2), the CCSID that is returned by a standard prepare statement in SQLDA is used. You can also override ASIS by specifying the CCSID keyword.

Specifying ASIS does not mean that a conversion is not required. Conversion might still be required when columns that are not CHAR, VARCHAR, GRAPHIC, and VARGRA types are unloaded in an external format and when the schema of the unloaded table space is not system-EBCDIC.

DB2 HPU first converts the value to the external format (in system-EBCDIC), and the result is converted to the table space schema.

Similarly, the padding characters or field separators (FORMAT DELIMITED) are in system-EBCDIC by default. They are also converted to the table space schema if the table space schema is not EBCDIC.

Conversion is also required when the table space CCSID is not the same as the SYSIN CCSID and when the SELECT statement causes C'constants'.

EBCDIC

Indicates that the data is unloaded in EBCDIC format. DB2 HPU uses the EBCDIC CCSID of the subsystem unless you override it by specifying the CCSID keyword.

UNICODE

Indicates that the data is unloaded in UNICODE format. DB2 HPU uses the UNICODE CCSID of the subsystem unless you override it by specifying the CCSID option.

The default value is EBCDIC.

In previous releases of DB2 HPU, the name of this parameter was VUU023/UNLSCHEM.

Quiesce process (VUU028/ULQSCEBH)

This parameter is optional. It specifies whether the process of updating physical objects in linear data sets (LDS) must be forced or only attempted.

Specify one of the following values:

FORCE

The process of updating physical objects in LDSs is forced, which means that the object is quiesced if possible; otherwise, a STOP/START is forced. The default value is FORCE.

TRY

The process of updating physical objects in LDSs is attempted, which means that the object is quiesced if possible; otherwise, the processing terminated in error.

The default value is FORCE.

In previous releases of DB2 HPU, the name of this parameter was VUU028/ULQSCEBH.

DB2 HPU additional features (VUU030/ULOPTNS)

This parameter is optional. Use this parameter to activate functions for DB2 HPU syntax. This parameter can receive a list of optional parameters, separated by commas, that modify DB2 HPU behavior. Use a continuation comma at the end of each line or fill in the line up to the end.

LOADINDDN(YES)

The INDDN *ddname* card is generated into the LOAD command. The variable *ddname* points to the data set that contains the unloaded data.

When a TEMPLATE is used to allocate the output file, DB2 HPU also generates a TEMPLATE in the LOADDDN file to allocate the input file for the LOAD.

LOADINDDN(NO)

The INDDN card is not generated into the LOAD command. The *ddname* points to the data set that contains the unloaded data.

When a TEMPLATE is used to allocate the output file, DB2 HPU does not generate a TEMPLATE in the LOADDDN file.

DSNTIAULSTRICT(YES)

The SELECT statements that are coded with FORMAT DSNTIAUL will function as if they are coded as FORMAT DSNTIAUL STRICT.

DSNTIAULSTRICT(NO)

The SELECT statements that are coded with FORMAT DSNTIAUL are not changed into FORMAT DSNTIAUL STRICT.

Attention: Changing the behavior of the DSNTIAUL format might affect the content of the unloaded data. Use the DSNTIAUL(YES) parameter and DSNTIAUL(NO) parameter with caution. See the *IBM DB2 High Performance Unload for z/OS V4.2 User's Guide* for a description of the difference between specifying FORMAT DSNTIAUL and specifying FORMAT DSNTIAUL STRICT.

DELIMITEDGRAPHEXT(YES)

In the DELIMITED format, GRAPHIC and VARGRAPHIC columns are unloaded in ASCII or in EBCDIC as GRAPHIC EXTERNAL data, including the SO/SI characters.

DELIMITEDGRAPHEXT(NO)

In the DELIMITED format, GRAPHIC and VARGRAPHIC columns are unloaded in ASCII or in EBCDIC as GRAPHIC data, without the SO/SI characters.

The DELIMITEDGRAPHEXT parameter is used only for DB2 HPU syntax. It has no impact on Fast Unload or Unload Plus syntax and has no impact when GRAPHIC data is unloaded in UNICODE.

ONDEMAND_RESOURCE_ALLOCATION(YES)

When DB2 HPU processes a list of table spaces by using a LISTDEF and TEMPLATE in a single DB2 HPU step, TEMPLATE files and control blocks are allocated when the table space is processed and are freed afterwards. If you use LISTDEF, you can run DB2 HPU jobs with more table spaces that can be processed in a single DB2 HPU invocation. Specifying YES reduces the memory resources that DB2 HPU uses. If any part of the unload process fails, processing continues with other unload tasks, and the process that failed will be displayed in the list of failed processes that is indicated after message INZU361I.

ONDEMAND_RESOURCE_ALLOCATION(NO)

When DB2 HPU processes a list of table spaces by using a LISTDEF and TEMPLATE in a single DB2 HPU step, all resources are allocated at the beginning of the step. If any part of the unload process fails, processing stops, and DB2 HPU terminates immediately after the error.

PADDING_STRICT(YES)

The previous syntax for the PADDING option that allowed you to code a two-byte hexadecimal constant, such as PADDING x'hhhh', is not accepted. The default value for the DBCS padding character is always the DBCS space that was converted in the output CCSID.

COPYDDN_STRICT(YES) or COPYDDN_STRICT(NO)

In previous releases, when both COPYDDN and DB2 FORCE were used, COPYDDN was ignored, and the data was unloaded from the table. When COPYDDN and DB2 YES were used, results could change along with DB2 HPU maintenance. When an unsupported SELECT statement became supported because maintenance was applied, DB2 HPU unloaded the data from the image copy instead of from the table.

In both cases, the data source that was selected by DB2 HPU might not be the expected one. In later releases, COPYDDN and DB2 YES or DB2 FORCE will be made incompatible to avoid this ambiguous behavior.

You can enable the behavior of DB2 HPU V4.1 and later releases by specifying COPYDDN_STRICT(YES). Otherwise, specify COPYDDN_STRICT(NO), which is the default value.

In later releases, COPYDDN_STRICT(YES) or COPYDDN_STRICT(NO) will be ignored, and only the enhanced behavior, corresponding to the behavior when COPYDDN_STRICT(YES) is specified, will be available.

BY_SQL_ONLY(WITH_UR)

SELECT statements with the WITH_UR clause are always processed in SQL mode. SELECT statements without the WITH_UR clause are always processed in native mode.

GBLPARAL_MAXPART_SET_NULL(YES) or GBLPARAL_MAXPART_SET_NULL(NO)

Processing subsets of partitions (when you use MAXPART or ULMAXPAR when the specified value is less than the number of partitions to be unloaded) and unloading without parallelism because GBLPARAL (NO) is specified are incompatible.

When VUU060/ULMAXPAR *n* and VUU036/GBLPARAL NO are specified together and are applicable to the unload in progress, GBLPARAL NO is

ignored, and the data is unloaded without parallelism at the partition level unless `GBLPARAL_MAXPART_SET_NULL(YES)` is specified.

Specify `GBLPARAL_MAXPART_SET_NULL(YES)` to disable the processing of the subset of partitions and unload the data without partition parallelism instead. If you specify either the `PARALLELISM` or the `MAXPART` keyword in the `UNLOAD` statement, specifying `GBLPARAL_MAXPART_SET_NULL(YES)` does not affect parallelism.

`AUTO_UNCNT_MAX(unit_count_value)`

Sets the maximum value to be used as the unit count when the unit count is dynamically determined by DB2 HPU. The value is determined by DB2 HPU when the value is not specified for the `UNCNT` option in the `TEMPLATE` statement.

Valid values are 0 - 59.

`CHECK_CCSID_STRICT(YES|NO)`

Specifies with CCSID classes, such as `SBCS` or `MIXED` or `DBCS`, are allowed.

YES Using an irrelevant CCSID class, such as `SBCS` or `MIXED` or `DBCS`, in the `CCSID(ccsid_sbc, ccid_mixed, ccid_dbc)` option is not allowed.

NO Using an irrelevant CCSID class, such as `SBCS` or `MIXED` or `DBCS`, in the `CCSID(ccsid_sbc, ccid_mixed, ccid_dbc)` option is allowed.

The default value is `YES`.

The default values are `LOADINDDN(NO)`, `DSNTIAULSTRICT(NO)`, `DELIMITEDGRAPHEXT(NO)`, `ONDEMAND_RESOURCE_ALLOCATION(NO)`, `COPYDDN_STRICT(NO)`, and `BY_SQL_ONLY()`.

`INSTREAM_XML_AS_CLOB(YES|NO)`

Specifies with CCSID classes, such as `SBCS` or `MIXED` or `DBCS`, are allowed.

`INSTREAM_XML_AS_CLOB(YES)`

XML data unloaded in-stream are unloaded as CLOB data.

`INSTREAM_XML_AS_CLOB(NO)`

XML data unloaded in-stream are unloaded as XML unless `FORMAT DSNTIAUL STRICT` is requested in which case the XML data is unloaded as CLOB.

The default value is `YES`.

`INSTREAM_XML_AS_CLOB(YES|NO)`

Specifies with CCSID classes, such as `SBCS` or `MIXED` or `DBCS`, are allowed.

`INSTREAM_XML_AS_CLOB(YES)`

XML data unloaded in-stream are unloaded as CLOB data.

`INSTREAM_XML_AS_CLOB(NO)`

XML data unloaded in-stream are unloaded as XML unless `FORMAT DSNTIAUL STRICT` is requested in which case the XML data is unloaded as CLOB.

The default value is `YES`.

`TRUE_UNICODE(YES|NO)`

TRUE_UNICODE(YES)

When conversion into unicode is requested with no target CCSID specification, the CSSID UTF8 is used as a target CCSID.

TRUE_UNICODE(NO)

When conversion into unicode is requested with no target CCSID specification, the default SBCS CSSID of the subsystem is used as a target CCSID.

The default value is NO.

CHECK_CCSID_STRICT(YES|NO)**CHECK_CCSID_STRICT(YES)**

The use of irrelevant CCSID class (i.e. SBCS or MIXED or DBCS) in the CCSID (ccsid_sbc, ccsid_mixed, ccsid_dbc) option is prohibited.

CHECK_CCSID STRICT(NO)

The use of irrelevant CCSID class (i.e. SBCS or MIXED or DBCS) in the CCSID (ccsid_sbc, ccsid_mixed, ccsid_dbc) option is allowed.

The default behavior of DB2 HPU is CHECK_CCSID_STRICT (YES).

In previous releases of DB2 HPU, the name of this parameter was VUU030/ULOPTNS.

U_NULLCHAR

UNLOAD PLUS NULLCHAR parameter default. This parameter is used for UNLOAD PLUS syntax and allows you to change the null indicator character in the output record.

Allowed values: xx where xx is any valid hexadecimal constant, except X'00'.

The default value is 6F.

U_NULLTYPE

UNLOAD PLUS NULLCHAR parameter default. This parameter is used for UNLOAD PLUS syntax and allows you to specify the location and length of the null indicator field in the output record.

Allowed values: T1/L1/T2/L2

The default value is T1.

DB2 SETTINGS_SOURCE (AUTO|DSNHDECP)**AUTO**

The way DB2 HPU determines the DB2 settings depends on the DB2 version. Prior to DB2 V9, the DSNHDECP member from the DSNEXIT library is loaded; in DB2 V9 and later versions, the GETVARIABLE SQL function is used

DSNHDECP

the DSNHDECP member from the DSNEXIT library is used regardless of the DB2 release.

The default value is DB2_SETTINGS_SOURCE (AUTO).

TEMPLATE_SSID_SUBSTITUTION (MEMBER_NAME)

The SSID template variable is substituted with the current DB2 member name to which DB2 HPU has connected.

TEMPLATE_SSID_SUBSTITUTION (GROUP_NAME)

The SSID template variable is substituted with the DB2 member name in a non datasharing environment and with the group attachment name for datasharing environment.

Here is an example that shows how to specify a very long value for ULOPTNS

```
TCUSTMZR           Multiple Value Parameter           Row 1 to 4 of 4

To add another value, issue the ADD command, and specify another value
for the parameter in the blank field in the Value column. Press enter to
save your values, or press End do save and exit.

Commands: ADD

Parameter description:
  DB2 HPU additional features (VUU030/UNLOPTNS)

Line commands: D - Delete

Cmd Value:
  (LOADINDDN (YES),DSNTIAULSTRICT(YES),DELIMITEDGRAPHEX
  T(YES),ONDEMANDE_RESOURCE_ALLOCATION(YES),PADDING_STRIC
  T(YES),COPYDDN_STRICT(YES),BY_SQL_ONLY(WITH_UR),GBLPAR
  RAL_MAXPART_SET_NULL(YES) )
-----End of values-----

Command ==>>>                               Scroll ==>>>PAGE
```

Check image copy before unloading (VUU033/ULCHKCPY)

This parameter is optional. It specifies whether the dsname that was specified by the COPYDDN parameter in the SYSIN should be checked.

Specify one of the following values:

YES

The dsname that was provided by the COPYDDN parameter in the SYSIN is checked. This PARMLIB parameter is considered if CHECK or INLINE is not specified in the SYSIN. The return code is 4, with a warning message in the SYSOUT if the check against the SYSIBM.SYSCOPY failed, and the FIC corresponding to the dsname is considered as a non-inline FIC.

NO The dsname is not checked.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU033/ULCHKCPY.

Option to modify the like behavior in SQL statement (VUU034/ULLIKE)

This parameter is optional. It specifies how DB2 HPU handles UNICODE MIXED strings in a LIKE predicate.

Specify one of the following values:

STRICT

Indicates that SQL rules are used to handle UNICODE mixed strings. To handle UNICODE mixed strings by using SQL rules, DB2 HPU must convert all operands (columns and masks of the LIKE predicate) in UNICODE DBCS, which increases CPU consumption due to conversions for each unloaded row.

FAST

Indicates that UNICODE mixed strings are considered as SBCS strings in LIKE predicates. The FAST option avoids MIXED to DBCS conversions and improves UNLOAD performances.

The default value is FAST.

In previous releases of DB2 HPU, the name of this parameter was VUU034/ULLIKE.

DB2 HPU response when a LOADDN cannot be generated (VUU038/UNLLDER)

This parameter is optional. It specifies whether DB2 HPU will stop when it is unable to generate a requested LOADDN. This situation can occur when the output format that was specified for DATE, TIME, or TIMESTAMP is not supported by the DB2 LOAD utility.

Specify one of the following values:

STOP

DB2 HPU stops with return code 8.

IGNORE

DB2 HPU issues a warning message and continues processing.

The default value is STOP.

In previous releases of DB2 HPU, the name of this parameter was VUU039/UNLLDER.

Unload HIDDEN Column by select * (VUU042/ULHIDDEN)

This parameter is optional. It specifies whether hidden columns will be selected when a SELECT * statement from a table name or from a LISTDEF is used. This parameter applies only to SELECT statements that are processed natively. Use the HIDDEN parameter of the EXEC card or the HIDDEN option of the OPTIONS block to override this parameter.

Specify one of the following values:

NO Specifies that hidden columns will not be unloaded when a SELECT * statement is used.

YES

Specifies that hidden columns will be unloaded when a SELECT * statement is used. For SELECT statements that are processed by DB2, the HIDDEN option is ignored.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU042/ULHIDDEN.

Unload controlled access table in native mode (VUU062/ULACCTRL)

This parameter is optional. Use it to unload data from tables whose access is controlled at the row level or the column level. This parameter applies only when the data of the table is accessed in native mode, which means a physical unload or a logical unload with DB2 NO specified and a supported SQL statement. To unload a row-controlled table, specify DB2 FORCE or specify DB2 NO, and set the VUU062 parameter to YES.

YES Tables that are controlled at the row or column level can be unloaded in native mode.

NO Tables that are controlled at the row or column level cannot be unloaded.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU062/ULACCTRL.

Technical parameter to unload XML columns (VUU063/ULOPTLX)

This parameter is optional. Use this parameter to tune the low-level technical algorithm that DB2 HPU uses to unload XML data. If the default optimized values do not apply to your specific case, you can use the VUU063 parameter to enhance performance.

Important: Do not specify a value for this parameter unless IBM Software Support recommends a specific value.

This parameter does not have a default value.

An equivalent SYSIN keyword for this parameter does not exist.

In previous releases of DB2 HPU, the name of this parameter was VUU063/ULOPTLX.

Parameter to identify ROW-TRANSACTION-TIMESTAMP (VUU065/ULRTTST)

This parameter is optional. It specifies the technique to be used when generating the load SYSIN to identify the columns that are defined as ROW-TRANSACTION-TIMESTAMP.

This parameter applies only to natively processed SELECT statements.

Unsupported SELECT statements and SELECT statements that are processed with the DB2 FORCE option are ignored. In this case, the IGNOREFIELDS and PERIODOVERRIDE keywords are not generated.

IGNORE

The load generates the column data. The generated load control statement contains the combination of IGNOREFIELDS keyword and a dummy field name for the identify column.

IGNOREOVERRIDE

The unloaded data is loaded into the identify column. The PERIODOVERRIDE keyword is added to the load control statement, and the exact name of the identify column is kept. This option is available for DB2 Version 9 and later releases. For earlier version of DB2, the ignore version is used.

IGNORE the default value.

Parameter to identify ROW-TRANSACTION-START-ID (VUU066/ULRTSID)

This parameter is optional. It specifies the technique to be used when generating the load SYSIN to identify the columns that are defined as ROW-TRANSACTION-START-ID.

This parameter applies only to natively processed SELECT statements.

Unsupported SELECT statements and SELECT statements that are processed with the DB2 FORCE option are ignored. In this case, the IGNOREFIELDS and PERIODOVERRIDE keywords are not generated.

IGNORE

The load generates the column data. The generated load control statement contains the combination of IGNOREFIELDS keyword and a dummy field name for the identify column.

TRANSIDOVERRIDE

The unloaded data is loaded into the identify column. The TRANSIDOVERRIDE keyword is added to the load control statement, and the exact name of the identify column is kept. This option is available for DB2 Version 9 and later releases. For earlier version of DB2, the ignore version is used.

IGNORE the default value.

Option to specify conversion truncation policy (VUU067-/ULCNVTRC)

This parameter is optional. It allows you to specify what policy to apply when data truncation is needed because of a CCSID conversion. The value of ULCNVTRC is overridden by the CONVERSION_TRUNCATION_ALLOWED sysin option (see Technical Parameters option block). Allowed values: YES/NO

- YES** Truncation of the output field after a CCSID conversion is allowed.
- NO** Truncation of the output field after a CCSID conversion is forbidden. Records requesting truncation are discarded and the return code is set to 4.

The default value is YES.

DB2 HPU output data parameters

The DB2 HPU output data parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the parameters for configuring output data.

You can accept the default values for the DB2 HPU output data parameters, or you can configure them based on your specific requirements.

The following list describes the DB2 HPU parameters. The parameters are listed in the following format:

description (Vxxxxxxxx/parameter-name)

Override DCB DB2 HPU parameters (VUM029/DFSIGDCB)

This parameter is optional. It specifies whether the DCB JCL parameter can be overridden in DB2 HPU syntax.

Specify one of the following values:

- YES** For jobs that use DB2 HPU syntax, DB2 HPU ignores the DCB parameter in the JCL and sets the DCB attributes to the appropriate values.
- NO** For jobs that use DB2 HPU syntax, DB2 HPU uses the DCB parameter in the JCL.

The default value is NO.

The corresponding SYSIN keyword is DFSIGDCB in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUM029/DFSIGDCB.

Override DCB UNLOAD PLUS syntax (VUM029/DFSIGDCB)

This parameter is optional. It specifies whether the DCB JCL parameter can be overridden in UNLOAD PLUS syntax.

Specify one of the following values:

- YES** For jobs that use UNLOAD PLUS syntax, DB2 HPU ignores the DCB parameter in the JCL and sets the DCB attributes to the appropriate values.
- NO** For jobs that use UNLOAD PLUS syntax, DB2 HPU uses the DCB parameter in the JCL.

The default value is NO.

The corresponding SYSIN keyword is DFSIGDCB in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUM029/DFSIGDCB.

Override DCB FAST UNLOAD syntax (VUM029/DFSIGDCB)

This parameter is optional. It specifies whether the DCB JCL parameter can be overridden in Fast Unload syntax.

Specify one of the following values:

- YES** For jobs that use Fast Unload syntax, DB2 HPU ignores the DCB parameter in the JCL and sets the DCB attributes to the appropriate values.
- NO** For jobs that use Fast Unload syntax, DB2 HPU uses the DCB parameter in the JCL.

The default value is NO.

The corresponding SYSIN keyword is DFSIGDCB in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUM029/DFSIGDCB.

NULL indicator, format USER (VUU014/ULNULL)

This parameter is optional. Use this parameter to modify the null or not-null indicator.

OFF

The null indicator is not present in the output data set.

hhhh

The first two digits (one hexadecimal character) represent the null indicator for a null column. The last two digits (one hexadecimal character) represent the null indicator for a not-null column.

The default value is FF00.

The corresponding SYSIN parameter is OPTIONS NULL.

In previous releases of DB2 HPU, the name of this parameter was VUU014/ULNULL.

NULL indicator, format DSNTIAUL (VUU014/ULNULL)

This parameter is optional. Use this parameter to modify the null or not-null indicator.

OFF

The null indicator is not present in the output data set.

hhhh

The first two digits (one hexadecimal character) represent the null indicator for a null column. The last two digits (one hexadecimal character) represent the null indicator for a not-null column.

The default value is FF00.

The corresponding SYSIN parameter is OPTIONS NULL.

In previous releases of DB2 HPU, the name of this parameter was VUU014/ULNULL.

NULL indicator, format VARIABLE (VUU014/ULNULL)

This parameter is optional. Use this parameter to modify the null or not-null indicator.

OFF

The null indicator is not present in the output data set.

hhhh

The first two digits (one hexadecimal character) represent the null indicator for a null column. The last two digits (one hexadecimal character) represent the null indicator for a not-null column.

The default value is FF00.

The corresponding SYSIN parameter is OPTIONS NULL.

In previous releases of DB2 HPU, the name of this parameter was VUU014/ULNULL.

NULL indicator, format EXTERNAL (VUU014/ULNULL)

This parameter is optional. Use this parameter to modify the null or not-null indicator.

OFF

The null indicator is not present in the output data set.

hhhh

The first two digits (one hexadecimal character) represent the null indicator for a null column. The last two digits (one hexadecimal character) represent the null indicator for a not-null column.

The default value is FF00.

The corresponding SYSIN parameter is OPTIONS NULL.

In previous releases of DB2 HPU, the name of this parameter was VUU014/ULNULL.

DATE, format USER (VUU015/ULDATE)

This parameter is optional. It specifies the default conversion type for a date column.

The default value is DATE_C.

The corresponding SYSIN parameter is OPTIONS DATE.

In previous releases of DB2 HPU, the name of this parameter was VUU015/ULDATE.

DATE, format DSNTIAUL (VUU015/ULDATE)

This parameter is optional. It specifies the default conversion type for a date column.

The default value is DATE_DB2.

The corresponding SYSIN parameter is OPTIONS DATE.

In previous releases of DB2 HPU, the name of this parameter was VUU015/ULDATE.

DATE, format DELIMITED (VUU015/ULDATE)

This parameter is optional. It specifies the default conversion type for a date column.

The default value is DATE_C.

The corresponding SYSIN parameter is OPTIONS DATE.

In previous releases of DB2 HPU, the name of this parameter was VUU015/ULDATE.

DATE, format VARIABLE (VUU015/ULDATE)

This parameter is optional. It specifies the default conversion type for a date column.

The default value is DATE_C.

The corresponding SYSIN parameter is OPTIONS DATE.

In previous releases of DB2 HPU, the name of this parameter was VUU015/ULDATE.

DATE, format EXTERNAL (VUU015/ULDATE)

This parameter is optional. It specifies the default conversion type for a date column.

The default value is DATE_DB2.

The corresponding SYSIN parameter is OPTIONS DATE.

In previous releases of DB2 HPU, the name of this parameter was VUU015/ULDATE.

TIME, format USER (VUU016/ULTIME)

This parameter is optional. It specifies the default conversion type for a time column.

The default value is TIME_A.

The corresponding SYSIN parameter is OPTIONS TIME.

In previous releases of DB2 HPU, the name of this parameter was VUU016/ULTIME.

TIME, format DSNTIAUL (VUU016/ULTIME)

This parameter is optional. It specifies the default conversion type for a time column.

The default value is TIME_DB2.

The corresponding SYSIN parameter is OPTIONS TIME.

In previous releases of DB2 HPU, the name of this parameter was VUU016/ULTIME.

TIME, format DELIMITED (VUU016/ULTIME)

This parameter is optional. It specifies the default conversion type for a time column.

The default value is TIME_A.

The corresponding SYSIN parameter is OPTIONS TIME.

In previous releases of DB2 HPU, the name of this parameter was VUU016/ULTIME.

TIME, format VARIABLE (VUU016/ULTIME)

This parameter is optional. It specifies the default conversion type for a time column.

The default value is TIME_A.

The corresponding SYSIN parameter is OPTIONS TIME.

In previous releases of DB2 HPU, the name of this parameter was VUU016/ULTIME.

TIME, format EXTERNAL (VUU016/ULTIME)

This parameter is optional. It specifies the default conversion type for a time column.

The default value is TIME_DB2.

The corresponding SYSIN parameter is OPTIONS TIME.

In previous releases of DB2 HPU, the name of this parameter was VUU016/ULTIME.

TIMESTAMP, format USER (VUU017/ULTMSTP)

This parameter is optional. It specifies the default conversion type for a timestamp column.

The default value is TMSTP_B

The corresponding SYSIN parameter is OPTIONS TIMESTAMP.

In previous releases of DB2 HPU, the name of this parameter was VUU017/ULTMSTP.

TIMESTAMP, format DSNTIAUL (VUU017/ULTMSTP)

This parameter is optional. It specifies the default conversion type for a timestamp column.

The default value is TMSTP_B

The corresponding SYSIN parameter is OPTIONS TIMESTAMP.

In previous releases of DB2 HPU, the name of this parameter was VUU017/ULTMSTP.

TIMESTAMP, format DELIMITED (VUU017/ULTMSTP)

This parameter is optional. It specifies the default conversion type for a timestamp column.

The default value is TMSTP_B

The corresponding SYSIN parameter is OPTIONS TIMESTAMP.

In previous releases of DB2 HPU, the name of this parameter was VUU017/ULTMSTP.

TIMESTAMP, format VARIABLE (VUU017/ULTMSTP)

This parameter is optional. It specifies the default conversion type for a timestamp column.

The default value is TMSTP_B

The corresponding SYSIN parameter is OPTIONS TIMESTAMP.

In previous releases of DB2 HPU, the name of this parameter was VUU017/ULTMSTP.

TIMESTAMP, format EXTERNAL (VUU017/ULTMSTP)

This parameter is optional. It specifies the default conversion type for a timestamp column.

The default value is TMSTP_B

The corresponding SYSIN parameter is OPTIONS TIMESTAMP.

In previous releases of DB2 HPU, the name of this parameter was VUU017/ULTMSTP.

Decimal picture, format USER (VUU018/ULPIC)

This parameter is optional. It specifies the numeric data display format.

Specify a value in the following format: *signpositionseparator*.

sign

Specify one of the following values to print the sign:

- + The plus sign (+) is used for positive values.
- The minus sign (-) is used for negative values.
- P The padding character is used for positive values, and the minus sign (-) is used for negative values.

The default value for specifying the rules for printing is the minus sign.

position

Specify one of the following values to position the sign:

LEAD

The sign is placed in front of the numeric value. The LEAD value is ignored for floating point numbers.

TRAIL

The sign is placed after the numeric value. The TRAIL value is ignored for floating point numbers.

The default value for where to position the sign is LEAD.

separator

Specify one of the following values for the decimal separator:

- . Use a period (.) as the decimal separator.
- , Use a comma (,) as the decimal separator.

The default value -LEAD., which means that the sign is printed before the numeric value, the sign is shown only for negative values, and the period is used as the decimal separator.

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU018/ULPIC.

Decimal picture, format DSNTIAUL (VUU018/ULPIC)

This parameter is optional. It specifies the numeric data display format.

Specify a value in the following format: *signpositionseparator*.

sign

Specify one of the following values to print the sign:

- + The plus sign (+) is used for positive values.
- The minus sign (-) is used for negative values.
- P The padding character is used for positive values, and the minus sign (-) is used for negative values.

The default value for specifying the rules for printing is the minus sign.

position

Specify one of the following values to position the sign:

LEAD

The sign is placed in front of the numeric value. The LEAD value is ignored for floating point numbers.

TRAIL

The sign is placed after the numeric value. The TRAIL value is ignored for floating point numbers.

The default value for where to position the sign is LEAD.

separator

Specify one of the following values for the decimal separator:

- . Use a period (.) as the decimal separator.
- , Use a comma (,) as the decimal separator.

The default value -LEAD., which means that the sign is printed before the numeric value, the sign is shown only for negative values, and the period is used as the decimal separator.

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU018/ULPIC.

Decimal picture, format DELIMITED (VUU018/ULPIC)

This parameter is optional. It specifies the numeric data display format.

Specify a value in the following format: *signpositionseparator*.

sign

Specify one of the following values to print the sign:

- + The plus sign (+) is used for positive values.
- The minus sign (-) is used for negative values.
- P The padding character is used for positive values, and the minus sign (-) is used for negative values.

The default value for specifying the rules for printing is the minus sign.

position

Specify one of the following values to position the sign:

LEAD

The sign is placed in front of the numeric value. The LEAD value is ignored for floating point numbers.

TRAIL

The sign is placed after the numeric value. The TRAIL value is ignored for floating point numbers.

The default value for where to position the sign is LEAD.

separator

Specify one of the following values for the decimal separator:

- . Use a period (.) as the decimal separator.
- , Use a comma (,) as the decimal separator.

The default value -LEAD., which means that the sign is printed before the numeric value, the sign is shown only for negative values, and the period is used as the decimal separator.

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU018/ULPIC.

Decimal picture, format VARIABLE (VUU018/ULPIC)

This parameter is optional. It specifies the numeric data display format.

Specify a value in the following format: *signpositionseparator*.

sign

Specify one of the following values to print the sign:

- + The plus sign (+) is used for positive values.
- The minus sign (-) is used for negative values.
- P The padding character is used for positive values, and the minus sign (-) is used for negative values.

The default value for specifying the rules for printing is the minus sign.

position

Specify one of the following values to position the sign:

LEAD

The sign is placed in front of the numeric value. The LEAD value is ignored for floating point numbers.

TRAIL

The sign is placed after the numeric value. The TRAIL value is ignored for floating point numbers.

The default value for where to position the sign is LEAD.

separator

Specify one of the following values for the decimal separator:

- . Use a period (.) as the decimal separator.
- , Use a comma (,) as the decimal separator.

The default value -LEAD., which means that the sign is printed before the numeric value, the sign is shown only for negative values, and the period is used as the decimal separator.

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU018/ULPIC.

Decimal picture, format EXTERNAL (VUU018/ULPIC)

This parameter is optional. It specifies the numeric data display format.

Specify a value in the following format: *signpositionseparator*.

sign

Specify one of the following values to print the sign:

- + The plus sign (+) is used for positive values.
- The minus sign (-) is used for negative values.
- P The padding character is used for positive values, and the minus sign (-) is used for negative values.

The default value for specifying the rules for printing is the minus sign.

position

Specify one of the following values to position the sign:

LEAD

The sign is placed in front of the numeric value. The LEAD value is ignored for floating point numbers.

TRAIL

The sign is placed after the numeric value. The TRAIL value is ignored for floating point numbers.

The default value for where to position the sign is LEAD.

separator

Specify one of the following values for the decimal separator:

- . Use a period (.) as the decimal separator.
- , Use a comma (,) as the decimal separator.

The default value -LEAD., which means that the sign is printed before the numeric value, the sign is shown only for negative values, and the period is used as the decimal separator.

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU018/ULPIC.

Display format for numeric values, format USER (VUU032/ULMASK)

This parameter is optional. It specifies the display format for numeric values (zero and separator).

The value of this parameter consists of four separate columns:

- The Left padding column indicates whether the number is padded on the left with nonsignificant zeroes to fill up the output field. The sign character is added to the left of these zeroes.

- The Left zero column indicates whether a zero is placed to the left of the decimal separator when the value is 0.
- The Decimal separator column indicates whether the decimal separator is displayed. The Decimal separator column applies only to decimal data. The decimal separator is never displayed for SMALLINT or INTEGER values.
- The Right padding column indicates whether the number is padded on the right of the decimal separator with zeroes, up to the number of digits of the decimal scale.

Important:

1. The ULMASK variable has no impact on the formatting of FLOAT values.

Table 3. Nonsignificant zero values for DECIMAL and SMALLINT/INTEGER

ULMASK value	Left padding	Left zero	Decimal separator	Right padding
.*	No	Only when the value is 0	Only if the decimal value is not 0	No
0.*	No	Always	Only if the decimal value is not 0	No
*.0	No	Only when the decimal scale is 0	Only if the decimal scale is not 0	Yes
0.0	No	Always	Only if the decimal scale is not 0	Yes
00.0	Yes	Always	Only if the decimal scale is not 0	Yes
*.Z	No	Only when the decimal scale is 0	Always	Yes
0.Z	No	Always	Always	Yes
00.Z	Yes	Always	Always	Yes

For examples of the effect of using ULMASK, see the *IBM DB2 High Performance Unload for z/OS V4.2 User's Guide*.

The following example shows how to specify a ULMASK value for the USER format:

```
00.0
```

The default value when a LIKE value is used to force conversion to a CHAR, an INTO clause, or a REFORMAT clause is *.*

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU032/ULMASK.

Display format for numeric values, format DSNTIAUL (VUU032/ULMASK)

This parameter is optional. It specifies the display format for numeric values (zero and separator).

The value of this parameter consists of four separate columns:

- The Left padding column indicates whether the number is padded on the left with nonsignificant zeroes to fill up the output field. The sign character is added to the left of these zeroes.
- The Left zero column indicates whether a zero is placed to the left of the decimal separator when the value is 0.
- The Decimal separator column indicates whether the decimal separator is displayed. The Decimal separator column applies only to decimal data. The decimal separator is never displayed for SMALLINT or INTEGER values.
- The Right padding column indicates whether the number is padded on the right of the decimal separator with zeroes, up to the number of digits of the decimal scale.

Important:

1. The ULMASK variable has no impact on the formatting of FLOAT values.

Table 4. Nonsignificant zero values for DECIMAL and SMALLINT/INTEGER

ULMASK value	Left padding	Left zero	Decimal separator	Right padding
.*	No	Only when the value is 0	Only if the decimal value is not 0	No
0.*	No	Always	Only if the decimal value is not 0	No
*.0	No	Only when the decimal scale is 0	Only if the decimal scale is not 0	Yes
0.0	No	Always	Only if the decimal scale is not 0	Yes
00.0	Yes	Always	Only if the decimal scale is not 0	Yes
*.Z	No	Only when the decimal scale is 0	Always	Yes
0.Z	No	Always	Always	Yes
00.Z	Yes	Always	Always	Yes

For examples of the effect of using ULMASK, see the *IBM DB2 High Performance Unload for z/OS V4.2 User's Guide*.

The following example shows how to specify a ULMASK value for the USER format:

```
00.0
```

The default value when a LIKE value is used to force conversion to a CHAR, an INTO clause, or a REFORMAT clause is *.*

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU032/ULMASK.

Display format for numeric values, format DELIMITED (VUU032/ULMASK)

This parameter is optional. It specifies the display format for numeric values (zero and separator).

The value of this parameter consists of four separate columns:

- The Left padding column indicates whether the number is padded on the left with nonsignificant zeroes to fill up the output field. The sign character is added to the left of these zeroes.
- The Left zero column indicates whether a zero is placed to the left of the decimal separator when the value is 0.
- The Decimal separator column indicates whether the decimal separator is displayed. The Decimal separator column applies only to decimal data. The decimal separator is never displayed for SMALLINT or INTEGER values.
- The Right padding column indicates whether the number is padded on the right of the decimal separator with zeroes, up to the number of digits of the decimal scale.

Important:

1. The ULMASK variable has no impact on the formatting of FLOAT values.

Table 5. Nonsignificant zero values for DECIMAL and SMALLINT/INTEGER

ULMASK value	Left padding	Left zero	Decimal separator	Right padding
,	No	Only when the value is 0	Only if the decimal value is not 0	No
0.*	No	Always	Only if the decimal value is not 0	No
*.0	No	Only when the decimal scale is 0	Only if the decimal scale is not 0	Yes
0.0	No	Always	Only if the decimal scale is not 0	Yes
00.0	Yes	Always	Only if the decimal scale is not 0	Yes
*.Z	No	Only when the decimal scale is 0	Always	Yes
0.Z	No	Always	Always	Yes
00.Z	Yes	Always	Always	Yes

For examples of the effect of using ULMASK, see the *IBM DB2 High Performance Unload for z/OS V4.2 User's Guide*.

The following example shows how to specify a ULMASK value for the USER format:

```
00.0
```

The default value when a LIKE value is used to force conversion to a CHAR, an INTO clause, or a REFORMAT clause is *.*

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU032/ULMASK.

Display format for numeric values, format VARIABLE

This parameter is optional. It specifies the display format for numeric values (zero and separator).

The value of this parameter consists of four separate columns:

- The Left padding column indicates whether the number is padded on the left with nonsignificant zeroes to fill up the output field. The sign character is added to the left of these zeroes.
- The Left zero column indicates whether a zero is placed to the left of the decimal separator when the value is 0.
- The Decimal separator column indicates whether the decimal separator is displayed. The Decimal separator column applies only to decimal data. The decimal separator is never displayed for SMALLINT or INTEGER values.
- The Right padding column indicates whether the number is padded on the right of the decimal separator with zeroes, up to the number of digits of the decimal scale.

Important:

1. The ULMASK variable has no impact on the formatting of FLOAT values.

Table 6. Nonsignificant zero values for DECIMAL and SMALLINT/INTEGER

ULMASK value	Left padding	Left zero	Decimal separator	Right padding
,	No	Only when the value is 0	Only if the decimal value is not 0	No
0.*	No	Always	Only if the decimal value is not 0	No
*.0	No	Only when the decimal scale is 0	Only if the decimal scale is not 0	Yes
0.0	No	Always	Only if the decimal scale is not 0	Yes
00.0	Yes	Always	Only if the decimal scale is not 0	Yes
*.Z	No	Only when the decimal scale is 0	Always	Yes
0.Z	No	Always	Always	Yes
00.Z	Yes	Always	Always	Yes

For examples of the effect of using ULMASK, see the *IBM DB2 High Performance Unload for z/OS V4.2 User's Guide*.

The following example shows how to specify a ULMASK value for the USER format:

00.0

The default value when a LIKE value is used to force conversion to a CHAR, an INTO clause, or a REFORMAT clause is *.*

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU032/ULMASK.

Display format for numeric values, format EXTERNAL (VUU032/ULMASK)

This parameter is optional. It specifies the display format for numeric values (zero and separator).

The value of this parameter consists of four separate columns:

- The Left padding column indicates whether the number is padded on the left with nonsignificant zeroes to fill up the output field. The sign character is added to the left of these zeroes.
- The Left zero column indicates whether a zero is placed to the left of the decimal separator when the value is 0.
- The Decimal separator column indicates whether the decimal separator is displayed. The Decimal separator column applies only to decimal data. The decimal separator is never displayed for SMALLINT or INTEGER values.
- The Right padding column indicates whether the number is padded on the right of the decimal separator with zeroes, up to the number of digits of the decimal scale.

Important:

1. The ULMASK variable has no impact on the formatting of FLOAT values.

Table 7. Nonsignificant zero values for DECIMAL and SMALLINT/INTEGER

ULMASK value	Left padding	Left zero	Decimal separator	Right padding
.	No	Only when the value is 0	Only if the decimal value is not 0	No
0.*	No	Always	Only if the decimal value is not 0	No
*.0	No	Only when the decimal scale is 0	Only if the decimal scale is not 0	Yes
0.0	No	Always	Only if the decimal scale is not 0	Yes
00.0	Yes	Always	Only if the decimal scale is not 0	Yes
*.Z	No	Only when the decimal scale is 0	Always	Yes
0.Z	No	Always	Always	Yes
00.Z	Yes	Always	Always	Yes

For examples of the effect of using ULMASK, see the *IBM DB2 High Performance Unload for z/OS V4.2 User's Guide*.

The following example shows how to specify a ULMASK value for the USER format:

```
00.0
```

The default value when a LIKE value is used to force conversion to a CHAR, an INTO clause, or a REFORMAT clause is *.*

The corresponding SYSIN parameter is OPTIONS PIC.

In previous releases of DB2 HPU, the name of this parameter was VUU032/ULMASK.

Options for global LOAD statement (VUU019/ULOPTLDT)

This parameter is optional. It defines the parameters of the LOAD statement that were generated at the table space level.

Enclose all sub-parameters in parentheses, as shown in the following example:

```
SORTDEVT(SYSDA) SORTNUM(32)
```

To generate a parenthesis in the LOADDDN, include two parentheses in this parameter, as shown in the following statement:

```
(ENFORCE((NO)),LOG(NO),COPYDDN((SYSCOPY)))
```

This statement generates this option in the LOAD statement:

```
ENFORCE(NO) LOG NO COPYDDN(SYSCOPY)
```

To code the load options on several lines, follow these rules:

- Use a continuation comma at the end of each line.
- Begin the first string with a left parenthesis, and code the corresponding right parenthesis only on the last line used.

The following parameters are valid. If you code a parameter that is not in this list, it is not controlled, and it is copied in the generated LOAD SYSIN.

- COPYDDN
- DISCARDN
- DISCARDS
- ENFORCE(CONSTRAINTS/NO)
- INDDN
- KEEPDICTIONARY
- LOG(YES/NO)
- RECOVERYDDN
- REPLACE
- RESUME(YES|NO)
- SORTDEVT
- SORTKEYS
- SORTNUM
- PARALLEL or PARALLEL (num-subtask) where num-subtask specifies the maximum number of sub-tasks that are to be processed in parallel by the LOAD utility. The value must be an integer between 0 and 32767, inclusive.

Important: If you specify the SORTKEYS keyword, the value that you specify is substituted with a value that is calculated according to the number of unloaded records, except when you run DB2 HPU in EXECUTE NO mode. When you run DB2 HPU in EXECUTE NO mode, the &SORTKEYS variable is replaced with 0.

If you code a parameter that is not in the previous list, it will not be controlled, and will only be copied in the generated LOAD SYSIN.

When the output file is a spanned file, NUMRECS is always generated in the SYSIN for the LOAD utility, therefore the SORTKEYS parameter is not taken into account.

The default value is (LOG (NO), ENFORCE(NO)).

In previous releases of DB2 HPU, the name of this parameter was VUU019/ULOPTLDT.

Options of partition LOAD statement (VUU020/ULOPTLDP)

This parameter is optional. Defines the parameters of the LOAD statement that was generated at the partition level.

To code the load options on several lines, follow these rules:

- Use a continuation comma at the end of each line.
- Begin the first string with a left parenthesis, and code the corresponding right parenthesis only on the last line used.

The following parameters are accepted:

- RESUME(YES|NO)
- REPLACE
- KEEPDICTIONARY

The default value is (RESUME(YES)).

In previous releases of DB2 HPU, the name of this parameter was VUU020/ULOPTLDP.

Options of the table LOAD statement (VUU070/ULOPTLDP)

This parameter is optional. Defines the parameters of the LOAD statement to be generated at the table level.

To code the load options on several lines, follow these rules:

- Use a continuation comma at the end of each line.
- Begin the first string with a left parenthesis, and code the corresponding right parenthesis only on the last line used.

The following parameters are accepted:

- NUMREC(&NUMREC): the number of unloaded records is specified as the operand of the NUMRECS options written in the generated LOAD statement
- NUMRECS(integer): the specified integer value is used as the operand of the NUMRECS options written in the generated LOAD statement.

If you code a parameter that is not in the above, it will not be controlled and will only be copied in the generated LOAD SYSIN as specified.

The default value is (NUMRECS(&NUMRECS)).

In previous releases of DB2 HPU, the name of this parameter was VUU020/ULOPTLDP.

Position for NULL indicator, format USER (VUU022/NULLPOS)

This parameter is optional. It specifies the position of the NULL indicator within the DB2 HPU output data sets.

Specify one of the following values:

AFTER

The NULL indicator will be set after the column data.

BEFORE

The NULL indicator will be set before the column data.

The default value is BEFORE.

The corresponding SYSIN parameter is OPTIONS NULLPOS.

In previous releases of DB2 HPU, the name of this parameter was VUU022/NULLPOS.

Position for NULL indicator, format DSNTIAUL (VUU022/NULLPOS)

This parameter is optional. It specifies the position of the NULL indicator within the DB2 HPU output data sets.

Specify one of the following values:

AFTER

The NULL indicator will be set after the column data.

BEFORE

The NULL indicator will be set before the column data.

The default value is BEFORE.

The corresponding SYSIN parameter is OPTIONS NULLPOS.

In previous releases of DB2 HPU, the name of this parameter was VUU022/NULLPOS.

Position for NULL indicator, format VARIABLE (VUU022/NULLPOS)

This parameter is optional. It specifies the position of the NULL indicator within the DB2 HPU output data sets.

Specify one of the following values:

AFTER

The NULL indicator will be set after the column data.

BEFORE

The NULL indicator will be set before the column data.

The default value is BEFORE.

The corresponding SYSIN parameter is OPTIONS NULLPOS.

In previous releases of DB2 HPU, the name of this parameter was VUU022/NULLPOS.

Position for NULL indicator, format EXTERNAL (VUU022/NULLPOS)

This parameter is optional. It specifies the position of the NULL indicator within the DB2 HPU output data sets.

Specify one of the following values:

AFTER

The NULL indicator will be set after the column data.

BEFORE

The NULL indicator will be set before the column data.

The default value is BEFORE.

The corresponding SYSIN parameter is OPTIONS NULLPOS.

In previous releases of DB2 HPU, the name of this parameter was VUU022/NULLPOS.

Use the same DDN for UNLDDN, OUTDDN, and LOADDN (VUU029/UNLDDDN)

This parameter is optional. Use this parameter to prevent the use of the same DDN for UNLDDN, OUTDDN, and LOADDN.

Specify one of the following values:

YES

If you use the same DDN in a single SELECT statement with the keywords UNLDDN, OUTDDN, and LOADDN, DB2 HPU issues error INZU124E.

NO An error is not issued when you use the same DDN in a single SELECT statement with the keywords UNLDDN, OUTDDN, and LOADDN.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU029/UNLDDDN.

DATE/TIME delimiter option (VUU031/DTDELIM)

This parameter is optional. It applies to FORMAT DELIMITED to delimit the column types DATE, TIME, and TIMESTAMP in the output data set.

Specify one of the following values:

YES

Column types DATE, TIME, and TIMESTAMP are delimited by the column delimiter that is specified by the option DELIM *literal*.

NO If the NULL DELIM option is used in FORMAT DELIMITED, this parameter will not apply to the DATE, TIME, and TIMESTAMP columns.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU031/DTDELIM.

NULL DATE/TIME delimiter option (VUU043/DTNULDLM)

This parameter is optional. Specifies that the NULL DELIM option of FORMAT DELIMITED should also be used for DATE, TIME, and TIMESTAMP columns. This parameter is used only when PARMLIB variable VUU031/DTDELIM is set to YES.

YES

If the NULL DELIM option is used in FORMAT DELIMITED, it will also apply to the DATE, TIME, and TIMESTAMP columns.

NO If the NULL DELIM option is used in FORMAT DELIMITED, it will not apply to the DATE, TIME, and TIMESTAMP columns.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU043/DTNULDLM.

Technique to generate SYSIN for identity column (VUU058/ULIDENT)

This parameter is optional. It specifies the technique to be used when generating the load SYSIN for identity columns that are defined as GENERATED ALWAYS.

This parameter applies only to natively processed SELECT statements.

Unsupported SELECT statements and SELECT statements that are processed with the DB2 FORCE option are ignored. In this case, the IGNOREFIELDS and IDENTITYOVERRIDE keywords are not generated.

Specify one of the following values:

IGNORE

The load generates the column data. The generated load control statement contains the combination of IGNOREFIELDS keyword and a dummy field name for the identity column.

OVERRIDE

The unloaded data is loaded into the identity column. The IDENTITYOVERRIDE keyword is added to the load control statement, and the exact name of the identity column is kept. This option is available for DB2 Version 9 and later releases. For earlier versions of DB2, the IGNORE option is used.

The default value is IGNORE.

In previous releases of DB2 HPU, the name of this parameter was VUU058/ULIDENT.

Positive sign for zoned-decimal numeric values (VUU037/ULSIGZ)

This parameter is optional. It specifies the positive sign for decimal zoned values in hexadecimal format. Valid values are A, B, C, D, E, and F. Each value represents the positive sign. Using this parameter helps you create a customized format that meets the requirements of the programs that process the unloaded data.

The default value is C.

In previous releases of DB2 HPU, the name of this parameter was VUU037/ULSIGZ.

Positive sign for date-time packed values (VUU038/ULSIGDTP)

This parameter is optional. It specifies the positive sign for date- and time-packed values. Valid values are A, B, C, D, E, and F. Each value represents the positive sign. Using this parameter helps you create a customized format that meets the requirements of the programs that process the unloaded data.

The default value is F.

In previous releases of DB2 HPU, the name of this parameter was VUU038/ULSIGDTP.

Truncate variable records exceeding 32756 (VUU040/ULTR32K)

This parameter is optional. Specifies whether DB2 HPU will truncate the variable records that exceed the physical limit of 32756.

Specify one of the following values:

YES

The output file record format is variable, and the LRECL is greater than 32756 (LRECL > 32756). LRECL is truncated to 32756, and message INZU297I is issued.

NO The output file record format is fixed, and the LRECL is greater than 32756 (LRECL > 32756). Processing ends, and message INZU298E is issued.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU040/ULTR32K.

Default output format for a logical unload (VUU045/ULFORMAT)

This parameter is optional. It specifies the value of the output format when the FORMAT parameter is not specified in the SYSIN for a SELECT statement.

This parameter applies only to DB2 HPU syntax.

Attention: Changing this value in the PARMLIB will affect existing production jobs that do not specify the FORMAT parameter. Use caution when changing the value of this parameter in the PARMLIB.

Specify one of the following values:

DSNTIAUL

Specifies that the default output format for SELECT statements is FORMAT VARIABLE. This format is the default value.

STRICT

Use this value to unload data in the same format that the DSNTIAUL program produces. DSNTIAUL STRICT affects the formatting of constant character strings that are specified in SELECT statements.

Specifying STRICT in VUU045/ULFORMAT overrides the value of DSNTIAUL STRICT in the PARMLIB variable ULOPTNS. If STRICT is not specified in VUU045/ULFORMAT, the content of the PARMLIB variable ULOPTNS is used.

Important: FORMAT DSNTIAUL STRICT applies only to DB2 HPU syntax.

DELIMITED

Specifies that the default output format for SELECT statements is FORMAT DELIMITED.

SEP=*val* | BLANK

Specifies the separator character, *val*, to separate fields in the output data set. *val* can be specified in character ('c') or hexadecimal (X'hh') format. Values that are specified in hexadecimal are not converted in CCSID conversion. Values that are specified in character are converted, if necessary, into the output CCSID.

Use BLANK to specify a space character. BLANK is the default value for SEP.

DELIM=*val* | BLANK | NONE

Specifies the delimiter character, *val*, to be used to enclose CHAR, VARCHAR, GRAPHIC, and VARGRAPHIC fields in the output data set. *val* can be specified in character ('c') or hexadecimal (X'hh') format. Values that are specified in hexadecimal are not converted in CCSID conversion. Values that are specified in character are converted, if necessary, into the output CCSID.

BLANK

Specifies a space character.

NONE

Specifies that you do not want to use a delimiter.

The default value for the delimiter is NONE.

NULLDELIM

Specifies that null values are not enclosed by the delimiter character that is specified by the *val* of DELIM.

The DELIM and NULL DELIM keyword can also apply to DATE, TIME, TIMESTAMP format depending on the settings of VUU043/DTNULDLM.>

VARIABLE

Specifies that the default output format for SELECT statements is FORMAT VARIABLE.

ALL

If the column that was selected last is variable, the output data set is VB, and this last column is written on its effective length. The effective length is the actual length of the data that is contained in a column that has a variable type. Both length bytes precede the column.

END

All the variable columns are written by using their actual length.

The default value is END if only VARIABLE is specified.

EXTERNAL

Specifies that the default output format for SELECT statements is FORMAT EXTERNAL.

USER

Specifies that the default output format for SELECT statements is FORMAT USER.

In previous releases of DB2 HPU, the name of this parameter was VUU045/ULFORMAT.

Size of the autotag in the output file (VUU046/ULAUTAG)

This parameter is optional. Use this parameter to specify a number for the value of the autotag that is generated in the output file. It applies only to DB2 HPU syntax, and it applies to all output formats.

Valid values are 0 - 8.

The default value is 0. When the default value is specified, an autotag is not generated.

The corresponding SYSIN parameter is OPTIONS AUTOTAG.

Attention: Changing this value in the PARMLIB will affect existing production jobs. Use caution when you change this value.

In previous releases of DB2 HPU, the name of this parameter was VUU046/ULAUTAG.

Full compatibility of LOAD statement with DB2 LOAD (VUU054/CTRLLIBM)

This parameter is optional. It specifies whether the LOAD statement that is generated by DB2 HPU must be fully compatible with the DB2 LOAD utility.

Valid values are YES and NO.

When this parameter is set to YES, additional controls are made when the DELIMITED format is requested, and the LOAD statement is generated only if it is compatible with the DB2 LOAD utility.

In previous releases of DB2 HPU, the name of this parameter was VUU054/CTRLLIBM.

Delimiter for external date, format USER (VU055/DATEDEL)

This parameter is optional. It specifies the default delimiter that is used in external date representations.

Specify one of the following values:

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character is used as the default delimiter for external date representation. By default, this variable applies to FORMAT USER. It can be used to change the default value of the delimiter for date external representation for any format by using the syntax for defining formatting options for each output format.

The default value is ' '.

The corresponding SYSIN parameter is OPTIONS DATEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU055/DATEDEL.

Delimiter for external date, format DSNTIAUL (VU055/DATEDEL)

This parameter is optional. It specifies the default delimiter that is used in external date representations.

Specify one of the following values:

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character is used as the default delimiter for external date representation. By default, this variable applies to FORMAT USER. It can be used to change the default value of the delimiter for date external representation for any format by using the syntax for defining formatting options for each output format.

The default value is ' '.

The corresponding SYSIN parameter is OPTIONS DATEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU055/DATEDEL.

Delimiter for external date, format DELIMITED (VU055/DATEDEL)

This parameter is optional. It specifies the default delimiter that is used in external date representations.

Specify one of the following values:

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character is used as the default delimiter for external date representation. By default, this variable applies to FORMAT USER. It

can be used to change the default value of the delimiter for date external representation for any format by using the syntax for defining formatting options for each output format.

The default value is '-'.

The corresponding SYSIN parameter is OPTIONS DATEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU055/DATEDEL.

Delimiter for external date, format VARIABLE (VU055/DATEDEL)

This parameter is optional. It specifies the default delimiter that is used in external date representations.

Specify one of the following values:

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character is used as the default delimiter for external date representation. By default, this variable applies to FORMAT USER. It can be used to change the default value of the delimiter for date external representation for any format by using the syntax for defining formatting options for each output format.

The default value is '-'.

The corresponding SYSIN parameter is OPTIONS DATEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU055/DATEDEL.

Delimiter for external date, format EXTERNAL (VU055/DATEDEL)

This parameter is optional. It specifies the default delimiter that is used in external date representations.

Specify one of the following values:

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character is used as the default delimiter for external date representation. By default, this variable applies to FORMAT USER. It can be used to change the default value of the delimiter for date external representation for any format by using the syntax for defining formatting options for each output format.

The default value is '-'.

The corresponding SYSIN parameter is OPTIONS DATEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU055/DATEDEL.

Delimiter in external time, format USER (VUU056/TIMEDEL)

This parameter is optional. It specifies the default delimiter that will be used in external time representations.

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character will be used as the default delimiter for external time representation.

The default value is '.'.

The corresponding SYSIN parameter is OPTIONS TIMEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU056/TIMEDEL.

Delimiter in external time, format DSNTIAUL (VUU056/TIMEDEL)

This parameter is optional. It specifies the default delimiter that will be used in external time representations.

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character will be used as the default delimiter for external time representation.

The default value is '.'.

The corresponding SYSIN parameter is OPTIONS TIMEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU056/TIMEDEL.

Delimiter in external time, format DELIMITED (VUU056/TIMEDEL)

This parameter is optional. It specifies the default delimiter that will be used in external time representations.

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character will be used as the default delimiter for external time representation.

The default value is '.'.

The corresponding SYSIN parameter is OPTIONS TIMEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU056/TIMEDEL.

Delimiter in external time, format VARIABLE (VUU056/TIMEDEL)

This parameter is optional. It specifies the default delimiter that will be used in external time representations.

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character will be used as the default delimiter for external time representation.

The default value is '.'.

The corresponding SYSIN parameter is OPTIONS TIMEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU056/TIMEDEL.

Delimiter in external time, format EXTERNAL (VUU056/TIMEDEL)

This parameter is optional. It specifies the default delimiter that will be used in external time representations.

'c'

The specified value must be in character format, coded between single quotation marks, and one byte long.

BLANK

Specifies that a space character will be used as the default delimiter for external time representation.

The default value is '.'.

The corresponding SYSIN parameter is OPTIONS TIMEDELIM.

In previous releases of DB2 HPU, the name of this parameter was VUU056/TIMEDEL.

Options apply to all formats (VUU057/OPALLFMT)

This parameter is optional. It specifies whether the formatting options that are specified in SYSIN in the OPTIONS block at the GLOBAL level or the UNLOAD level apply only to FORMAT USER or to all output formats. VUU057/OPALLFMT applies to the following SYSIN parameters of the OPTIONS block:

- NULL
- DATE
- TIME
- TIMESTAMP
- PIC
- DATEDELIM
- TIMEDELIM
- NULLID
- NULLPOS
- NULLPAD
- PADDING
- TRIM

Specify one of the following values:

NO The SYSIN parameters in the previous list apply only to FORMAT USER when they are specified at the GLOBAL level or at the UNLOAD level.

YES

The SYSIN parameters in the previous list apply to all output formats when they are specified at the GLOBAL level or at the UNLOAD level.

Attention: Specifying YES for this PARMLIB variable might affect the output data that is generated by existing production jobs. Change this variable to YES only after you have verified that existing DB2 HPU production jobs will not be affected by the change.

When the SYSIN parameters in the previous list are specified at the SELECT level (after the OUTDDN keyword), they always apply to all output formats. In this case, the value of OPALLFMT is ignored.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU057/OPALLFMT.

Trim the trailing blanks, format USER (VUU059/ULTRIM)

This parameter is optional. Use it to specify whether DB2 HPU will remove the trailing blanks for character data that is unloaded into a variable length field. It applies to CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, CLOB, and DBCLOB output fields. However, it does not apply to numeric external data.

Specify one of the following values:

NO Trailing blanks are not removed from variable length strings.

YES

Trailing blanks are removed when data is unloaded into VARCHAR, VARGRAPHIC output fields, or CLOB, and DBCLOB output field except if SPANNED YES is requested. The length of the output field is adjusted to match the effective number of characters that are written.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU059/ULTRIM.

Trim the trailing blanks, format DSNTIAUL (VUU059/ULTRIM)

This parameter is optional. Use it to specify whether DB2 HPU will remove the trailing blanks for character data that is unloaded into a variable length field. It applies to CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, CLOB, and DBCLOB output fields. However, it does not apply to numeric external data.

Specify one of the following values:

NO Trailing blanks are not removed from variable length strings.

YES

Trailing blanks are removed when data is unloaded into VARCHAR, VARGRAPHIC output fields, or CLOB, and DBCLOB output field except if SPANNED YES is requested. The length of the output field is adjusted to match the effective number of characters that are written.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU059/ULTRIM.

Trim the trailing blanks, format DELIMITED (VUU059/ULTRIM)

This parameter is optional. Use it to specify whether DB2 HPU will remove the trailing blanks for character data that is unloaded into a variable length field. It applies to CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, CLOB, and DBCLOB output fields. However, it does not apply to numeric external data.

Specify one of the following values:

NO Trailing blanks are not removed from variable length strings.

YES

Trailing blanks are removed when data is unloaded into VARCHAR, VARGRAPHIC output fields, or CLOB, and DBCLOB output field except if

SPANNED YES is requested. The length of the output field is adjusted to match the effective number of characters that are written.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU059/ULTRIM.

Trim the trailing blanks, format VARIABLE (VUU059/ULTRIM)

This parameter is optional. Use it to specify whether DB2 HPU will remove the trailing blanks for character data that is unloaded into a variable length field. It applies to CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, CLOB, and DBCLOB output fields. However, it does not apply to numeric external data.

Specify one of the following values:

NO Trailing blanks are not removed from variable length strings.

YES

Trailing blanks are removed when data is unloaded into VARCHAR, VARGRAPHIC output fields, or CLOB, and DBCLOB output field except if SPANNED YES is requested. The length of the output field is adjusted to match the effective number of characters that are written.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU059/ULTRIM.

Trim the trailing blanks, format EXTERNAL (VUU059/ULTRIM)

This parameter is optional. Use it to specify whether DB2 HPU will remove the trailing blanks for character data that is unloaded into a variable length field. It applies to CHAR, VARCHAR, GRAPHIC, VARGRAPHIC, CLOB, and DBCLOB output fields. However, it does not apply to numeric external data.

Specify one of the following values:

NO Trailing blanks are not removed from variable length strings.

YES

Trailing blanks are removed when data is unloaded into VARCHAR, VARGRAPHIC output fields, or CLOB, and DBCLOB output field except if SPANNED YES is requested. The length of the output field is adjusted to match the effective number of characters that are written.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU059/ULTRIM.

String constant is unloaded as CHAR/VARCHAR data types (VUU064/STRNGCST)

This parameter is optional. It specifies whether string constants in the SELECT statements are handled as CHAR or VARCHAR data when the SELECT statement is processed in native mode.

CHAR

The string constants are handled as CHAR data unless the DSNTIAUL STRICT format is requested either by specifying DSNTIAULSTRICT(NO) as a subparameter of the ULOPTNS parameter or by using the FORMAT DSNTIAUL STRICT option in the UNLOAD command. When you specify CHAR, the output for

expressions that have string constants might depend on whether the unload is performed in native or in SQL mode.

VARCHAR

The string constants are handled as VARCHAR data. When you specify VARCHAR, the output for expressions that have string constants does not depend on whether the unload is processed in native or in SQL mode. Consider specifying VARCHAR for unloads that use the DB2 YES option.

Important: If some of your processes use strings that are handled as CHAR data when DB2 HPU processes the unload in native mode, use the default value. Otherwise, specify VARCHAR to ensure that the output of the unload for expressions that have string constants do not depend on the processing mode of the unload.

If you want to specify VARCHAR so that string constants are processed as VARCHAR data in native mode or SQL mode but you have some jobs that expect string constants to be processed as CHAR data, consider changing the SELECT statements for these unloads by replacing any string constant *string_constant* with CHAR(*string_constant*).

The default value is CHAR.

In previous releases of DB2 HPU, the name of this parameter was VUU064/STRNGCST.

Manage character loss, DB2 HPU syntax (VUU071/LOSSCHAR)

This parameter is optional. It specifies the behavior of the product when significant data from a character column is lost during unload formatting because of an INTO or REFORMAT clause and a DB2 HPU syntax is specified.

DB2 HPU considers a data loss has occurred for a character column if the data truncation removes any character that does not belong to the padded part of the character string, i.e. a sequence of padding characters that ends at the right end of the character string.

The data loss is checked for the following data types:

- CHAR
- VARCHAR
- GRAPHIC
- VARGRAPHIC
- BINARY
- VARBINARY

Specify one of the following values:

IGNORE

No control is done. The return code of the job is 0 unless another issue occurs.

WARNING

A warning message is issued for any column affected by a data loss. The return code is 4 unless another issue occurs.

REJECT

An error message is issued for any column affected by a data loss. The return code is 8 unless another issue occurs.

The default value is IGNORE.

The corresponding SYSIN parameter is CHARACTER_LOSS from the OPTIONS block.

Manage character loss, UNLOAD PLUS syntax (VUU071/LOSSCHAR)

This parameter is optional. It specifies the behavior of the product when significant data from a character column is lost during unload formatting because of an INTO or REFORMAT clause and an UNLOAD PLUS syntax is specified.

DB2 HPU considers a data loss has occurred for a character column if the data truncation removes any character that does not belong to the padded part of the character string, i.e. a sequence padding characters that ends at the right of the character string..

The data loss is checked for the following data types:

- CHAR
- VARCHAR
- GRAPHIC
- VARGRAPHIC
- BINARY
- VARBINARY

Specify one of the following values:

IGNORE

No control is done. The return code of the job is 0 unless another issue occurs.

WARNING

A warning message is issued for any column affected by a data loss. The return code is 4 unless another issue occurs.

REJECT

An error message is issued for any column affected by a data loss. The return code is 8 unless another issue occurs.

The default value is IGNORE.

The corresponding SYSIN parameter is CHARACTER_LOSS from the OPTIONS block.

Manage character loss, FAST UNLOAD syntax (VUU071/LOSSCHAR)

This parameter is optional. It specifies the behavior of the product when significant data from a character column is lost during unload formatting because of an INTO or REFORMAT clause and a FAST UNLOAD syntax is specified.

DB2 HPU considers a data loss has occurred for a character column if the data truncation removes any character that does not belong to the padded part of the character string, i.e. a sequence padding characters that ends at the right end of the character string.

The data loss is checked for the following data types:

- CHAR
- VARCHAR
- GRAPHIC
- VARGRAPHIC
- BINARY
- VARBINARY

Specify one of the following values:

IGNORE

No control is done. The return code of the job is 0 unless another issue occurs.

WARNING

A warning message is issued for any column affected by a data loss. The return code is 4 unless another issue occurs.

REJECT

An error message is issued for any column affected by a data loss. The return code is 8 unless another issue occurs.

The default value is IGNORE.

The corresponding SYSIN parameter is CHARACTER_LOSS from the OPTIONS block.

Manage numeric loss, DB2 HPU syntax (VUU072/LOSSNUM)

This parameter is optional. It specifies the behavior of the product when significant data from a numeric column is lost during unload formatting (via INTO or REFORMAT clause) and a DB2 HPU syntax is specified.

DB2 HPU considers a data loss has occurred for a numeric column the numeric value has changed.

The data loss is checked for the following data types:

- SMALLINT
- INTEGER
- BIGINT
- DECIMAL

Specify one of the following values:

IGNORE

No control is done. The return code of the job is 0 unless another issue occurs.

WARNING

A warning message is issued for any column affected by a data loss. The return code is 4 unless another issue occurs.

REJECT

An error message is issued for any column affected by a data loss. The return code is 8 unless another issue occurs.

The default value is IGNORE.

The corresponding SYSIN parameter is NUMERIC_LOSS from the OPTIONS block.

Manage numeric loss, UNLOAD PLUS syntax (VUU072/LOSSNUM)

This parameter is optional. It specifies the behavior of the product when significant data from a numeric column is lost during unload formatting (via INTO or REFORMAT clause) and an UNLOAD PLUS syntax is specified.

DB2 HPU considers a data loss has occurred for a numeric column the numeric value has changed.

The data loss is checked for the following data types:

- SMALLINT
- INTEGER

- BIGINT
- DECIMAL

Specify one of the following values:

IGNORE

No control is done. The return code of the job is 0 unless another issue occurs.

WARNING

A warning message is issued for any column affected by a data loss. The return code is 4 unless another issue occurs.

REJECT

An error message is issued for any column affected by a data loss. The return code is 8 unless another issue occurs.

The default value is IGNORE.

The corresponding SYSIN parameter is NUMERIC_LOSS from the OPTIONS block.

Manage numeric loss, FAST UNLOAD syntax (VUU072/LOSSNUM)

This parameter is optional. It specifies the behavior of the product when significant data from a numeric column is lost during unload formatting (via INTO or REFORMAT clause) and a FAST UNLOAD syntax is specified.

DB2 HPU considers a data loss has occurred for a numeric column the numeric value has changed.

The data loss is checked for the following data types:

- SMALLINT
- INTEGER
- BIGINT
- DECIMAL

Specify one of the following values:

IGNORE

No control is done. The return code of the job is 0 unless another issue occurs.

WARNING

A warning message is issued for any column affected by a data loss. The return code is 4 unless another issue occurs.

REJECT

An error message is issued for any column affected by a data loss. The return code is 8 unless another issue occurs.

The default value is IGNORE.

The corresponding SYSIN parameter is NUMERIC_LOSS from the OPTIONS block.

Example: Manage numeric loss, with DB2 HPU, UNLOAD PLUS, FAST UNLOAD syntaxes

Assume C1 has the INTEGER data-type and is unloaded into a SMALLINT data-type with the help of a REFORMAT clause:

- a row for which C1's value is 100 and is unloaded without error as the value 100 is kept (not altered) in the SMALLINT data-type output.

- a row for which C1's value is 1 000 000 (the internal representation is x'000F4240') is processed as follows according to the parameter value:
 - it is unloaded with C1 field containing the value 16960 (the internal representation is x'4240') without any message related to data loss if IGNORE is specified,
 - it is unloaded with C1 field containing the value 16960 (with internal representation is x'4240') with a warning message if WARNING is respected,
 - it is rejected with an error message if REJECT is specified.

Error message limit, DB2 HPU parameters (VUU073/MAXERR)

This parameter is optional. It specifies the maximum number of messages that are issued if , while processing an unload requested via the DB2 HPU syntax, DB2 HPU encounters an error (conversion, formatting...) while unloading the rows of a table. When the limit is reached, the process stops and the return code is set to 8. Use this parameter to limit the number of messages that are written into the spool.

Valid values are: 0-2147483647 (0 means 'no limit').

The default value is 0.

The corresponding SYSIN parameter is MAXERR.

Error message limit, UNLOAD PLUS syntax (VUU073/MAXERR)

This parameter is optional. It specifies the maximum number of messages that are issued if while processing an unload requested via the UNLOAD PLUS syntax, DB2 HPU encounters an error (conversion, formatting...) while unloading the rows of a table. When the limit is reached, the process stops and the return code is set to 8. Use this parameter to limit the number of messages that are written into the spool.

Valid values are: 0-2147483647 (0 means 'no limit').

The default value is 0.

The corresponding SYSIN parameter is DISCARDS.

Error message limit, FASTUNLOAD syntax (VUU073/MAXERR)

This parameter is optional. It specifies the maximum number of messages that are issued if while processing an unload requested via the FASTUNLOAD syntax, DB2 HPU encounters an error (conversion, formatting...) while unloading the rows of a table. When the limit is reached, the process stops and the return code is set to 8. Use this parameter to limit the number of messages that are written into the spool.

Valid values are: 0-2147483647 (0 means 'no limit').

The default value is 0.

The corresponding SYSIN parameter is DISCARDS.

DB2 HPU DB2 Administration Tool and DB2 Launchpad parameters

The DB2 Administration Tool and DB2 Launchpad parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the parameters that are required to integrate DB2 HPU with DB2 Admin.

The following list describes the DB2 HPU DB2 Administration Tool and DB2 Launchpad parameters. The parameters are listed in the following format:

description (Vxxxxxxx/parameter-name)

High-level qualifier for the DB2 Admin data sets

This parameter is optional. It specifies the high-level qualifier of the DB2 Administration Tool libraries. This information is used by the sample programs INZADBI and INZDB2IX (in the SINZCLIST library) to update the DB2 Administration Tool and the Data Management Tools Launchpad tables.

In previous releases of DB2 HPU, the name of this parameter was VUU025.

Library which contains DB2 Admin commands tables

This parameter is optional. It specifies the name of the library that contains the DB2 Administration Tool command tables. This information is used by the sample programs INZADBI and INZDB2IX (in the SINZCLIST library) to update the DB2 Administration Tool and the DB2 Tools Launchpad tables.

In previous releases of DB2 HPU, the name of this parameter was VUU026.

Library which contains the ADBMTI EXEC

This parameter is optional. It specifies the name of the library that contains the ADBMTI EXEC. This information is used by the sample programs INZADBI and INZDB2IX (in the SINZSAMP library) to update the DB2 Administration Tool and the DB2 Tools Launchpad tables.

In previous releases of DB2 HPU, the name of this parameter was VUU027.

DB2 HPU conversion parameters

The DB2 HPU conversion parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the conversion parameters that are used by DB2 HPU.

The following list describes the DB2 HPU conversion parameters. The parameters are listed in the following format:

description (Vxxxxxxx/parameter-name)

Unicode Conversion Services load library (VZM006/SCUNMOD)

This parameter is optional. It specifies the name of the IBM Conversion Service Load Library. If you want to perform conversions that imply non-SBCS CCSIDs or pairs of SBCS CCSIDs that are not supported by the SYSSTRINGS catalog table, you must first install IBM OS/390® Support for Unicode. For more information about this program, see *Program Directory for z/OS Support for Unicode* and *z/OS Support for Unicode Using Conversion Services*.

In previous releases of DB2 HPU, the name of this parameter was VZM006/SCUNMOD.

Unicode Conversion Services technique search order (VZM007/SCUNTSO)

This parameter is optional. It specifies a list of technique search orders to be searched for the z/OS Unicode Services. Separate each value with a comma. When a conversion between two CCSIDs is required, HPU looks for a conversion that specifies one of the listed technique search orders until it finds one valid conversion.

Example: SCUNTSO = ER,,RE means that DB2 HPU will for the ER technique search order, followed by the *blank* technique search order, and then the RE technique search order.

The default value is blank.

If DB2 Version 8 or later is installed, usually the technique search order is equal to ER.

See *z/OS Support for Unicode Using Conversion Services* for more information.

In previous releases of DB2 HPU, the name of this parameter was VZM007/SCUNTSO.

Unicode Conversion Services substitution character mode (VZM008/SCUNSUB)

This parameter is optional. It specifies how the UNLOAD utility manages conversions through z/OS Unicode Services.

YES Indicates that if a character in a string cannot be converted between two CCSIDs, it is replaced by the substitution character of the converter.

NO Indicates that if a character in a string cannot be converted, the conversion of all the strings fails.

The default value is YES.

In previous releases of DB2 HPU, the name of this parameter was VZM008/SCUNSUB.

DB2 HPU file management parameters

The DB2 HPU file management parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the file management parameters that are used by DB2 HPU.

The following list describes the DB2 HPU file management parameters. The parameters are listed in the following format:

description (Vxxxxxxx/parameter-name)

Volume name of migrated object (VZM005/INFVSMIG)

This parameter is optional. It specifies the volume name that is located in the ICF catalog for migrated files. The default value is MIGRAT, which corresponds to the value that is used by the DFSMSHsm component of the IBM Data Facility Storage Management Subsystem (DFSMS).

In previous releases of DB2 HPU, the name of this parameter was VZM005/INFVSMIG.

Assignable devices number per tape unit/storage class (VZM010/TAPEUNIT)

This parameter is optional. Use this parameter to specify multiple unit names, storage class names, or both, and the number of associated devices that can be used by DB2 HPU when you request that output files be allocated on tape. DB2 HPU ensures that the maximum number of tape devices is not exceeded by automatically reducing the parallelism degree, if necessary. When the number of tape devices that are associated with a unit name or a storage class name is not specified, the value is set to 1.

This parameter does not have a default value.

The corresponding SYSIN keyword is TAPEUNIT in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VZM010/TAPEUNIT.

Maximum number of disk units to allocate a work file (VUX010/LIMUNIT)

This parameter is optional. It specifies the maximum number of disk units to be used when DB2 HPU allocates a temporary work file. It is also taken into account to determine the number of work data sets specified as the DYNALLOC first operand of the SORT program when a DB2 HPU needs to perform a sort task.

The default value is 9.

The corresponding SYSIN keyword is LIMUNIT in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX010/LIMUNIT.

Note: When allocations are performed by DB2 HPU, the LIMUNIT value is used as the volume count (VOLCNT) allocation parameter unless it exceeds the maximum value accepted by the system for the device type involved. For example, if LIMUNIT is set to 255 and the work data sets are to be allocated on DASD devices for which the limit is 59, DB2 HPU allocates the work data sets with VOLCNT=59.

Maximum size for primary allocation of a work data set (VUX019/WRKMXPR)

This parameter is optional. It specifies the maximum size, in kilobytes, for the primary allocation of a work data set on DASD. When very large work data sets are used, the primary allocation might be distributed among several volumes according to the limit that was specified in the VUX010/LIMUNIT parameter.

Requirement: Regardless of the limit that was specified on the VUX010/LIMUNIT parameter, the value that you provide for the VUX019/WRKMXPR parameter must be lower than the capacity of the units that is used for these work data sets (VUM013). You must also consider that these units might be distributed among several volumes.

Valid values are 1 - 16777215.

The default value is 500000.

The corresponding SYSIN keyword is WRKMXPR in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX019/WRKMXPR.

Note: When DB2 HPU invokes the sort program, the number of work datasets to be used by the sort program is determined according to the following rules:

- The size of a work data set is assumed to be VUX019/WRKMXPR.
- The number of work files per allowed volume (i.e. VUX010/LIMUNIT) must not exceed 2.

These rules are implemented via the following formula: Number of work data sets = $\min(\text{upper_round}(2 * \text{FILSZ} * \text{RECORD_LENGTH} / \text{WRKMXPR} / 1024), 2 * \text{LIMUNIT})$ where FILSZ stands for the estimated number of records to be sorted and RECORD_LENGTH for the mean value of the record length.

Although the above formula is always used to determine the allowed number of work data sets, the actual number of data sets might be lower if this calculated number exceeds the number of files the sort program can handle. For example, DFSORT V1R12.0 cannot use more than 255. Therefore, although LIMUNIT is set to a value greater than or equal to 128, which would theoretically allow the sort program, to use more than 256 work data sets, no more than 255 files will be actually used.

DSCB model for allocation of GDS (VUX033/TMPLDSCB)

This parameter is optional. It specifies the model data set control block (DSCB)

to use when allocating generation data sets (GDS) and when a TEMPLATE is used to allocate new generations of a generation data group (GDG).

In previous releases of DB2 HPU, the name of this parameter was VUX033/TMPLDSCB.

Volumes for allocation of temporary data sets (VUM018/WRKVOL)

This parameter is optional. It specifies the name of the volume where temporary data sets will reside.

In previous releases of DB2 HPU, the name of this parameter was VUM018/WRKVOL.

Tape unit where the work data sets must be allocated (VUA007/WRKTUNIT)

This parameter is optional. It specifies the name of the tape unit that is used to allocate temporary files.

If you use temporary files on tape, specify a tape unit or DASD device on this parameter. If you specify a DASD device, ensure that the pool of volumes that are associated with that unit has enough free space to store large data sets. If you do not specify a value, the utility allocates temporary files on the unit that is specified on the WRKUNIT parameter.

The corresponding SYSIN keyword is WRKTUNIT of the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUA007/WRKTUNIT.

Maximum size for work data set on DASD (VUX016/WRKUNTSW)

This parameter is optional. It specifies a threshold size (in kilobytes) for work data sets. All work data sets that exceed this threshold size will be allocated on the unit that is specified on the VUA007/WRKTUNIT parameter.

The corresponding SYSIN keyword is WRKUNTSW in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX016/WRKUNTSW.

Maximum number of unit for tape temporary data set (VUX017/MAXTUNIT)

This parameter is optional. It specifies the maximum number of tape units that are provided for work data sets that are used by a DB2 HPU job.

Valid values are 1 - 255.

The default value is 2.

The corresponding SYSIN keyword is MAXTUNIT in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX017/MAXTUNIT.

Number of volumes for temporary data set on tape (VUX032/WRKTVcnt)

This parameter is optional. It specifies the number of volumes to use for temporary data sets that are allocated on tape. Specifying 0 indicates that no VOLCOUNT parameter will be used for allocating tape files. In this case, up to five volumes are used.

Valid values are 0 - 255.

The default value is 0.

The corresponding SYSIN keyword is WRKTVcnt in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX032/WRKTVcnt.

BUFNO for sequential QSAM (VUM022/QSBUFNO)

This parameter is optional. It specifies the number of data buffers for sequential QSAM (the BUFNO parameter of the DCB for QSAM).

Valid values are 1 - 255.

The default value is 60.

In previous releases of DB2 HPU, the name of this parameter was VUM022/QSBUFNO.

BUFND for sequential VSAM (VUM023/VSBUFND)

This parameter is optional. It specifies the number of data buffers for sequential VSAM (the BUFND parameter of the ACB for VSAM).

Note: Specifying a large value might increase the amount of memory that DB2 HPU requires.

Valid values are 0 - 65535.

The default value is 360.

In previous releases of DB2 HPU, the name of this parameter was VUM023/VSBUFND.

DB2 HPU reporting parameters

The DB2 HPU reporting parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the reporting parameters that are used by DB2 HPU.

The following list describes the reporting parameters. The parameters are listed in the following format:

description (Vxxxxxxx/parameter-name)

List of system codes for which no dump is produced (VZM009)

This parameter is optional. It defines a list of system codes for which no dump is produced if an abend occurs during execution of DB2 HPU. You can specify up to eight lines. Each line can contain up to eight values. Separate each value by a comma.

After changing the content of this variable, you must regenerate the installation job related to the INZZSCOD template to activate the change.

The list can contain specific system codes that are coded on three hexadecimal digits or generic system codes that begin with X and include all the system codes for the number that follows it. For example, 0c1 is a specific system code, and X78 is a generic system code that includes all system codes that end with 78 (B78, D78, E78, and so on).

The default value is X22, X06, X37, 913, X78, X0A.

In previous releases of DB2 HPU, the name of this parameter was VZM009.

Generate a tape usage report in SYSPRINT (VZM011/TAPERPT)

This parameter is optional. Use it to specify whether to generate a report about the tape usage. The report displays the data set names (DSN) and their positions on the tape (FILESEQ) for each volume that is used.

Specify one of the following values:

YES The tape usage report is generated.

NO The tape usage report is not generated.

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VZM011/TAPERPT.

Maximum number of messages for row structure errors (VUX018/LDSERRLM)

This parameter is optional. It specifies the maximum number of messages that are issued if DB2 HPU encounters a row structure error while reading the rows of a table space. Use this parameter to limit the number of messages that are written into the spool.

Valid values are 0 - 2147483647.

Important: Specifying a large number of records might increase the amount of storage that DB2 HPU requires.

This parameter does not have a default value.

In previous releases of DB2 HPU, the name of this parameter was VUX018/LDSERRLM.

Return code for unloaded rows (VUU024/UNLZLRC)

This parameter is optional. It specifies the return code that applies when no rows are unloaded by at least one of the SELECT statements of the UNLOAD command.

Valid values are 0 - 4 095.

The default value 4.

Alteration applies only when original return code is zero unless keyword MAXRC is specified.

In previous releases of DB2 HPU, the name of this parameter was VUU024/UNLZLRC.

Return code if an object is missing (VUU024/UNLZLRC)

This parameter is optional. It specifies the return code that applies when an object included in a LIS TDEF does not exist, or when a LISTDEF expands to contain no objects. When this parameter is not specified, no specific return code applies for this situation.

Valid values are 1 - 4095.

There is no default value.

Alteration applies only when original return code is zero unless keyword MAXRC is specified.

In previous releases of DB2 HPU, the name of this parameter was VUU024/UNLZLRC.

Return code if zero line is unloaded (VUU024/UNLZLRC)

This parameter is optional. It specifies the return code when a LISTDEF clause returns no objects. When this parameter is not specified, no specific return code applies for this situation.

Valid values are 1 - 4095.

There is no default value.

Alteration applies only when original return code is zero unless keyword MAXRC is specified.

In previous releases of DB2 HPU, the name of this parameter was VUU024/UNLZLRC.

MAXRC alteration return code (VUU024/UNLZLRC)

This parameter is optional. When MAXRC is not specified, the original return code can be altered only when it is 0. When MAXRC is specified, it can also be altered when its value is 1 - 8.

Frequency to display information messages (VUU050/ULFRQMSG)

This parameter is optional. It specifies that DB2 HPU issues informational messages that display the current number of unloaded rows for each output file.

n At the end of processing, DB2 HPU issues message INZX089, which indicates the total number of rows for each output file. A non-null integer value means that this message is issued for each *n* rows.

The default value is 0.

In previous releases of DB2 HPU, the name of this parameter was VUU050/ULFRQMSG.

Additional parameter for information messages (VUU050/ULFRQMSG)

This parameter is optional. It specifies interval time since the previous message was issued or since the first record was unloaded.

Specify one of the following values:

DELTA

Messages INZX089 and INZX090 are issued to indicate the elapsed time since the previously issued message.

TOTAL

Messages INZX089 and INZX090 are issued to indicate the total elapsed time since the beginning of the unload process.

TOTAL is the default when a non-null value is specified and neither DELTA nor TOTAL is specified.

TOTAL and DELTA are mutually exclusive.

The default value is TOTAL.

In previous releases of DB2 HPU, the name of this parameter was VUU050/ULFRQMSG.

The location where DB2 HPU writes information messages (VUU051/PROCMSG)

This parameter is optional. It specifies where to write informational messages that correspond to the unload process of the table space, image copy, or partitions and indexes.

Specify one of the following values:

WTO

Messages are issued as write to operator messages in the system log.

ddname

Messages are issued in the corresponding ddname. If the corresponding ddname is not already allocated in the JCL, it is dynamically allocated as SYSOUT=*. This ddname can be equal to SYSPRINT, in which case these messages might be mixed with other messages that DB2 HPU issues.

All messages that are issued in the output file are prefixed with the system time in *hh:mm:ss* format.

The default value is WTO.

The corresponding SYSIN parameter is OPTIONS PROCMSG.

In previous releases of DB2 HPU, the name of this parameter was VUU051/PROCMSG.

Location of messages issued for each SELECT statement (VUU052/SEMMSG)

This parameter is optional. It specifies where to write informational messages INZX089 and INZX090 that are issued for each SELECT statement.

Specify one of the following values:

WTO

Messages that correspond to the SELECT statements are issued as write to operator messages in the system log.

NUMBERED

Messages that correspond to the SELECT statements are issued in a ddname for each SELECT statement. If this ddname is not already allocated in the JCL, it is dynamically allocated as SYSOUT=*. The format of the generated ddname is UxxSyyyy, where *xx* is the two-digit UNLOAD number and *yyyy* is the SELECT number for this unload.

ddname

Messages that correspond to the SELECT statements are issued in the corresponding ddname. If this ddname is not already allocated in the JCL, it will be dynamically allocated as SYSOUT=*. This ddname can be equal to SYSPRINT, in which case these messages might be mixed with other messages that DB2 HPU issues.

The default value is WTO.

The corresponding SYSIN parameter is OPTIONS SELMSG.

In previous releases of DB2 HPU, the name of this parameter was VUU052/SELMSG.

Generate a list of all OBIDs in the FIC (VUU053/OBIDRPT)

This parameter is optional. It specifies whether to generate a list of all object IDs (OBIDs) that are found in the full image copy (FIC). It applies only when you are unloading from an image copy.

YES

Generates a list of all OBIDs that are found in the FIC.

NO Does not generate a list of all OBIDs that are found in the FIC.

The default value is NO.

The corresponding SYSIN parameter is OBID_REPORT in the COPYDDN option.

In previous releases of DB2 HPU, the name of this parameter was VUU053/OBIDRPT.

Return code when the utility has switched to SQL mode (VUU069/SQLSWIRC)

This parameter is optional. It specifies the return code that applies when the processing of a select statement has automatically switched to SQL mode when

the statement is not supported in native mode. It is not applicable when the SQL mode is explicitly requested (e.g. DB2 HPU FORCE is specified).

Valid values are 0 - 4 095.

The default value is 4.

DB2 HPU tuning parameters

The DB2 HPU tuning parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the tuning parameters that are used by DB2 HPU.

The following list describes the tuning parameters. The parameters are listed in the following format:

description (Vxxxxxxx/parameter-name)

Default size of the input file or object for DB2 HPU (VUX003/SIZE)

This parameter is optional. It specifies the default number of records when allocating the resources (work data sets and FILSZ parameter) that are required by the SORT processing. The default value is used only when DB2 HPU is unable to estimate the number of rows to be unloaded, which happens only in very specific cases.

This parameter is ignored when the rows are selected by DB2. Therefore, set VUX003/SIZE to a typical or maximum number of rows for the table spaces that are usually processed by DB2 HPU. VUX003/SIZE is expressed as a number of records.

Valid values are 1000 - 2147483647

Important: Specifying a large number of records might increase the amount of storage that DB2 HPU requires.

The default value is 1000000 records.

In previous releases of DB2 HPU, the name of this parameter was VUX003/SIZE.

Minimum memory size for each sort process (VUX004/LOWMEM)

This parameter is optional. It specifies the memory size, in bytes, below the 16 MB line that is used by the sort process. DB2 HPU considers LOWMEM only during parallel processing. When DB2 HPU prepares to start one or more SORT tasks (with the exception of the first SORT), it checks whether the amount of remaining memory under the line is greater or equal to LOWMEM.

Take the following into account in order to determine the MAXSORT and LOWDEM parameters:

1. Determine the maximum storage that is allowed for a SORT below the 16 MB limit; this value is referred as SORTMAXMEMBELOW hereafter. For example, SORTMAXMEMBELOW is the value of VSCORE parameter of the DFSORT installation and the value of VSCORE parameter of the SYNCSORT installation. Refer to the documentation of your SORT program to determine the SORTMAXMEMBELOW value if you use another one.
2. Determine the minimum amount of memory required below the 16 MB line by your SORT program: This value is referred as SORTMINMEMBELOW hereafter. To do so, either refer to your SORT program documentation or run a sample of SORT program sorting no rows or a very little number of rows and check the amount of memory this program has used below the

16MB line (see the SYS value given by the IEF371I message).
SORTMINMEMBELOW is about 2070 000 bytes for DFSORT VR1R10.

3. Determine the remaining memory below the 16 MB limit when the INZUTILB module is loaded in memory: this value is referred as HPUMEMBELOW hereafter. To do so, run a sample DB2 HPU job requesting a basic unload (the IVP job is suitable) with the QUIESCE YES option and no SORT request. Check the amount of memory this program has used below the 16 MB line (see the SUS value given by the IEF374I message).

Set the LOWMEM and MAXSORT according to the following rules:

- $\text{SORTMINMEMBELOW} < \text{LOWMEM} < \text{SORTMAXMEMBELOW}$
- $1 < \text{MAXSORT} < (16 \text{ MB} - \text{HPUMEMBELOW}) / \text{LOWMEM}$

To prevent failures, do not set LOWMEM lower than SORTMINMEMBELOW.

To maximize the amount of memory that is used, set LOWMEM to a value less than or equal to the value of SORTMAXMEMBELOW. Setting LOWMEM to a higher value would lead of a waste of memory equal to $\text{LOWMEM} - \text{SORTMAXMEMBELOW}$. Setting LOWMEM equal to LOWMAXMEMBELOW ensures that any SORT task launched by DB2 HPU can use an amount of memory up to that value of SORTMAXMEMBELOW. However, as SORTMAXMEMBELOW might not always be needed,, consider setting LOWMEM to a lower value so that DB2 HPU will attempt to launch more SORT tasks. Do not decrease LOWMEM to a too low value as it can lead to failures because some SORT tasks might run out of memory. The default setting (in parmlib) might not be relevant for some complex unload jobs where alot of tasks are involved and parallelism is enabled. In such a case, it is recommended to adjust the value of LOWMEM and MAXSORT via the Technical Parameters option block in the GLOBAL block in SYSIN.

To improve performance, increase MAXSORT and decrease LOWMEM.

To minimize the consumption of memory below the 16 MB line, decrease MAXSORT and increase LOWMEM.

Valid values are 1 - 2147483647.

The default value is 270000.

The corresponding SYSIN keyword is LOWMEM in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX004/LOWMEM.

Wait unit in seconds/100 (VUX007/WAITUNIT)

This parameter is required. It specifies the wait time (in hundredths of seconds) between two unsuccessful tests of the STOP command.

Valid values are 1 - 2147483647.

The default value is 100 (1 second).

In previous releases of DB2 HPU, the name of this parameter was VUX007/WAITUNIT.

Number of wait periods before issuing a WTOR (VUX008/WAITQTY)

This parameter is required. It specifies the number of times an unsuccessful STOP command will be tested before a WTOR is sent to the console. If the operator answers CANCEL (C), DB2 HPU will stop with return code 8. If the operator answers WAIT (W), the wait process starts again.

Valid values are 1 - 2147483647.

The default value is 20.

In previous releases of DB2 HPU, the name of this parameter was VUX008/WAITQTY.

Wait time for the WTOR reply (VUX009/WAITQTYM)

This parameter is required. It specifies the maximum wait time (in seconds) before an answer is sent to the WTOR message. Utility execution will stop beyond this limit (return code 8).

Valid values are 1 - 2147483647.

The default value is 60.

In previous releases of DB2 HPU, the name of this parameter was VUX009/WAITQTYM.

Size of work areas (VUX022/VBUFSIZE)

This parameter is optional. It sets the size, in bytes, of the buffers that are used for communication between DB2 HPU tasks such as reading LDS, formatting data, and sorting data. The value that you set depends on the speed of the system. A higher value has little impact on performance, but a lower value forces DB2 HPU to change tasks more frequently, which increases WAIT TIME and CPU TIME.

Valid values are 1 - 214748 3647.

The recommended value is 1000000. Do not change VBUFSIZE unless performance problems occur.

The default value is 1000000.

The corresponding SYSIN keyword is VBUFSIZE in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX022/VBUFSIZE.

Maximum degree of parallelism for LDS reading (VUX025/PARALLEL)

This parameter is optional. It specifies the parallelism degree for an unload job when you are unloading a partitioned table space. This parameter indicates the maximum number of partitions that are processed in parallel. However, if the unload job includes a sort process, the VUX025 parameter is limited to the use of the VUX005/MAXSORT parameter. When a sort process is included, if the value of the VUX005/MAXSORT parameter is lower than the value that is specified in the VUX025/PARALLEL parameter, the VUX005/MAXSORT parameter is used. You can specify a value of 1 or greater.

Valid values are 1 - 65535.

The default value is 20.

In previous releases of DB2 HPU, the name of this parameter was VUX025/PARALLEL.

Maximum degree of parallelism between unload tasks (VUX030/UTLPARAL)

This parameter is optional. It specifies the parallelism degree when you are starting several unload tasks for the same UNLOAD command. An unload task can be when you unload a table space (each table space that is accessed counts for one task, whether it is partitioned or not), or a SELECT statement

that is processed by DB2. If more than one SELECT statement is processed by using DB2, DB2 HPU will count one task for all of these SELECT statements. You can specify a value of 1 or greater.

Valid values are 1 - 65535.

The default value is 5.

In previous releases of DB2 HPU, the name of this parameter was VUX030/UTLPARAL.

Number of rows retrieved by an SQL FETCH function (VUU035/ULROWSET)

This parameter is optional. It specifies the number of rows that will be retrieved by a single SQL FETCH. Specifying the number of rows that will be retrieved improves the performance for SELECT statements that are processed by DB2 (using DB2 FORCE or DB2 YES with unsupported SELECT statements). When you specify 1, the multi-fetch function is not used. A standard single-row FETCH is used instead. This function is available only in DB2 Version 8 and later releases.

Valid values are 1 - 32767.

The default value is 100.

In previous releases of DB2 HPU, the name of this parameter was VUU035/ULROWSET.

Enforce partition parallelism for unloading a table into a single file (VUU036/GBLPARAL)

This parameter is optional. It specifies whether partition parallelism will be enforced when a table space is unloaded into a single output file.

Specify one of the following values:

YES

Parallelism is enforced when table spaces are unloaded. The parallelism degree is set in variables VUX025/PARALLEL and VUX005/MAXSORT when a sort is requested.

NO Parallelism is determined by the number of output files that are coded in the JCL and the explicit selection of partitions in the SYSIN. You can override this PARMLIB variable by using the PARALLELISM SYSIN keyword (DB2 HPU syntax only).

The default value is NO.

In previous releases of DB2 HPU, the name of this parameter was VUU036/GBLPARAL.

Support parallelism for unloading several partitions (VUU044/SUBTKSOF)

This parameter is optional. It specifies whether parallelism is supported when data is unloaded from several partitions into a single output data set.

Specify one of the following values:

NO Does not support parallelism when data is unloaded from several partitions into a single output data set. NO is the recommended value.

YES

Supports parallelism when data is unloaded from several partitions into a single output data set.

The default value is NO.

Attention: Do not change this value unless IBM asks you to change it.

In previous releases of DB2 HPU, the name of this parameter was VUU044/SUBTKSOF.

Maximum number of partitions processed in one unload (VUU060/ULMAXPAR)

This parameter is optional. It specifies the maximum number of partitions that DB2 HPU can process in a single unload operation. ULMAXPAR affects logical unload operations (SELECT) that are processed natively and physical unload operations only. You can use this parameter to reduce the memory consumption of DB2 HPU.

If the total number of partitions that have to be processed for a table space is larger than the value that is specified in ULMAXPAR, DB2 HPU automatically splits the single unload operation into several unload operations to respect the value of ULMAXPAR. If the value of ULMAXPAR is smaller than the value of PARALLELISM at the partition level, the effective partition parallelism is limited by ULMAXPAR.

Valid values are 0 - 4096.

The default value is 0, which means that splitting is not done. When you specify a non-zero value, ORDER BY and ORDER CLUSTER clauses can be used only when each partition is unloaded into a separate file. When you specify a non-zero value and a split is done, the OUTMAXROWS or UNLMAXROWS setting applies to each partition.

The corresponding SYSIN keyword is MAXPART, which can be specified in the GLOBAL OPTIONS block or the UNLOAD block.

In previous releases of DB2 HPU, the name of this parameter was VUU060/ULMAXPAR.

Size of the buffer used to retrieve LOB data (VUU041/ULOCSIZE)

This parameter is optional. It specifies the size of the buffer, in bytes, to be used to retrieve LOB data by using a LOB locator through DB2. This parameter is used in DB2 Versions 7 and 8 when retrieving LOB data by using LOBFILE REFERENCE (CLOBF, BLOBF, or DBCLOBF) in DB2 FORCE or DB2 YES with an unsupported SELECT statement. In DB2 Version 9, DB2 HPU uses LOB FILE REFERENCE variables and does not require intermediate buffers.

Valid values are 1 - 16000000.

The default value is 1000000.

In previous releases of DB2 HPU, the name of this parameter was VUU041/ULOCSIZE.

Wait/retry function when resources are unavailable (VUU047/RETRYMOD)

This parameter is optional. If a resource is not available, it specifies whether DB2 HPU uses wait/retry logic. This parameter controls the following availability criteria:

- The status of the DB2 object (table space, partition, or index) that prevents processing
- If LOCK YES is specified in the SYSIN and a LOCK cannot be taken on the corresponding object
- If the dynamic allocation of the LDS files of the table space or index to be unloaded fails

Specify one of the following values:

NONE

DB2 HPU does not use wait/retry logic.

ALL

DB2 HPU uses wait/retry logic for the three kinds of situations in the previous list.

STATUS

DB2 HPU uses wait/retry logic when the status of the DB2 object, table space, partition, or index prevents processing. The following statuses prevent processing:

- CHKP
- GRECP
- UTUT
- PSRBD
- WEPR
- RBDP
- RBDP*
- REORP
- DBETE
- RESTP
- PSRCP
- LPL
- RELDP
- REFP
- RECP
- RECP*

LOCK

DB2 HPU uses wait/retry logic when a LOCK is requested by the LOCK YES option and the LOCK cannot be taken.

LDS

DB2 HPU uses wait/retry logic when LDS allocation fails.

You can specify multiple values except when you specify ALL or NONE. If you specify multiple values, separate them by using commas without spaces. Specify the wait time (in hundredths of a second) between two unsuccessful tests of the resource availability with PARMLIB parameter VUU048/RETRYW, and specify the number of retries with PARMLIB parameter VUU049/RETRYNB.

The default value is NONE.

In previous releases of DB2 HPU, the name of this parameter was VUU047/RETRYMOD.

Wait time between tries to access unavailable resources (VUU048/RETRYW)

This parameter is optional. It specifies the wait time (in hundredths of seconds) between two unsuccessful tests of the resource availability that is defined by the VUU047/RETRYMOD parameter.

Valid values are 1 - 32767.

The default value is 100 (1 second).

In previous releases of DB2 HPU, the name of this parameter was VUU048/RETRYW.

Number of retries when resources are unavailable (VUU049/RETRYNB)

This parameter is optional. It specifies the number of times to test the availability of a resource before stopping with return code 8.

Valid values are 1 - 32767.

The default value is 20.

In previous releases of DB2 HPU, the name of this parameter was VUU049/RETRYNB.

Concurrent access resolution for prepared statement (VUU061/ACCPREP)

This parameter is optional. Use it to specify the CONCURRENT ACCESS RESOLUTION option for the PREPARE statement that DB2 HPU will use to process SELECT statements by using DB2.

Specify one of the following values:

WAIT

The WAIT FOR OUTCOME clause is specified to wait for the commit or rollback of data that is being updated or deleted.

USE

The USE CURRENTLY COMMITTED clause is specified to use the currently committed version of the data that is being updated or deleted. The USE option is supported only in DB2 10 and later releases.

The VUU061 parameter does not have a default value.

In previous releases of DB2 HPU, the name of this parameter was VUU061/ACCPREP.

DB2 HPU sort parameters

The DB2 HPU sort parameters section on the Product Parameters panel (CCQPPRD) in Tools Customizer contains the sort parameters that are used by DB2 HPU.

The following list describes the common sort parameters. The parameters are listed in the following format:

description (Vxxxxxxx/parameter-name)

Maximum number of active sorts processes (VUX005/MAXSORT)

This parameter is optional. It specifies the maximum number of active sorts that can run in the same step when parallel processing is involved. IBM DFSORT for z/OS supports MAXSORT values 1 - 20, and all of its modules are reusable.

Most of the non-IBM sort products contain modules that require MAXSORT=1 because not all of their modules are reusable.

To improve performance, increase MAXSORT and decrease LOWMEM.

To minimize the consumption of memory below the line, decrease MAXSORT and increase LOWMEM.

Valid values are 1 - 32767.

The default value is 20.

The corresponding SYSIN keyword is MAXSORT in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX005/MAXSORT.

Allocation distribution for the sort input file (VUX006/WRKSPACE)

This parameter is optional. It specifies the percentage of space that is allocated for the sort input file. WRKSPACE can be one of the following values:

Specify one of the following values:

PARTIAL

Corresponds to a primary and secondary allocation that is equal to 50 percent of the estimated size of the file to be sorted. Two volumes are allowed for this allocation.

FULL

Corresponds to a primary allocation that is equal to 100 percent of the estimated size of the file to be sorted, and to a secondary allocation of 33 percent.

The default value is PARTIAL.

The corresponding SYSIN keyword is WRKSPACE in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX006/WRKSPACE.

Sort program that DB2 HPU uses to sort data (VUX037/SORTUTIL)

This parameter is optional. It allows you to specify which sort program to use when DB2 HPU needs to perform a sort operation outside of DB2.

Specify one of the following values:

SORT

The on-site sort program is used. The load module name is DFSORT.

DB2SORT

The IBM DB2 Sort for z/OS utility (DB2 Sort) is used. DB2 Sort can improve the performance of utility sort processing, especially in environments with large volumes of data, large table spaces, large indexes, or batch window constraints. These environments require a more sophisticated sorting approach than the approach that is used by tools that are used for general sorting purposes.

The load module name is DB2SORT. The relevant entry point load module of DB2 Sort must be installed in the system libraries. Specifying DB2SORT does not affect the DB2 settings. Therefore, if you want the sorts performed by DB2 (for unsupported SELECT statements or when DB2 FORCE is specified), you must change the appropriate DB2 settings.

Tip: Specify DB2SORT when you need to perform several sort operations in parallel.

For more information about DB2 Sort, see the *DB2 Sort for z/OS User's Guide*.

The default value is SORT.

The corresponding SYSIN keyword is SORTUTIL in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUX037/SORTUTIL.

Number of records in sort work areas (VUM024/SRTNBVRE)

This parameter is optional. It specifies the number of records in the sort work areas.

Important: Specifying a large number of records might increase the amount of storage that DB2 HPU requires.

Valid values are 0 - 2147483647.

The default value is 800.

In previous releases of DB2 HPU, the name of this parameter was VUM024/SRTVNBRE.

Minimum size in bytes for sort work areas (VUM025/SRTVSMIN)

This parameter is optional. It specifies the minimum size, in bytes, of the sort work areas.

Important: Specifying a large value might increase the amount of storage or memory that DB2 HPU requires.

Valid values are 0 - 2147483647.

The default values are the values that are set for VUX022.

The corresponding SYSIN keyword is SRTVSMIN in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUM025/SRTVSMIN.

Maximum size for sort work areas (VUM026/SRTVSMAX)

This parameter is optional. It specifies the maximum size, in bytes, of the sort work areas.

Note: Specifying a large value might increase the amount of memory that DB2 HPU requires.

Valid values are 0 - 2147483647.

The default value is 2000000.

The corresponding SYSIN keyword is SRTVSMAX in the Technical Parameters options block, which can be coded in the GLOBAL OPTIONS block.

In previous releases of DB2 HPU, the name of this parameter was VUM026/SRTVSMAX.

Customization templates

Tools Customizer uses the following templates to generate the customization jobs for DB2 HPU.

The following table lists and describes the templates:

Table 8. Template names and descriptions

Template	Description
INZTVAR	Generates the INZUTIL member containing the non DB2-dependent settings

Table 8. Template names and descriptions (continued)

Template	Description
INZPARM	Defines the dsname of the DB2 HPU PARMLIB to the load modules
INZHPUCL	Generates the members required to run the DB2 HPU ISPF interface
INZLAUNC	Generates the clist to be run to add DB2 HPU to DB2 Tools Launchpad
INZADTOO	Generates a member with instructions for adding DB2 HPU into DB2 Admin
INZBIND	Binds the plans and packages that are used by DB2 HPU and grants access
INZEXEUE	Generates a sample job for running DB2 HPU against a given subsystem
INZZSCOD	Defines the list of abend codes trapped for dump
INZCHECK	Generates members with the DB2-dependent settings and runs DB2 HPU IVP
INZFREE	Frees the DB2 HPU plan for a DB2 subsystem or a data sharing group

Worksheets: DB2 Sort

Complete the following worksheets before you customize DB2 Sort.

Worksheets: Gathering required data set names and parameter values

During the customization process, you will need to provide parameter values for DB2 Sort, and in some cases, for DB2.

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.

Discover customized product information

If you have already customized a version of DB2 Sort and you want to use the same parameter values to customize a different version, you can use the Discover EXEC in Tools Customizer. The Discover panel will prompt you for the name of the library from the previous version of DB2 Sort.

Parameter	Description	Your value
(Prior) SCNKLINK	The DB2 Sort SCNKLINK library from the previous version, such as CNK130.SCNKLINK	

Parameters: Data set names and installation options

Data set names for Tools Customizer

If you have not previously used Tools Customizer, you will need to provide values for Tools Customizer data sets on the Tools Customizer Settings panel. In this worksheet, record the names of the Tools Customizer data sets.

Data set names for Tools Customizer

Data set name / hlq	Description	Special requirements	Your data set name / hlq
DB2TOOL.PRODUCT.CUST	Customization library qualifier	You must have write access to this data set qualifier.	
SCCQDENU	Metadata library for Tools Customizer	None.	
DISCOVER	Discover output data set	You must have write access to this data set.	
DATASTOR	Data store data set	You must have write access to this data set.	

Data set names for DB2 Sort

In this worksheet, record the names of the data sets for the DB2 Sort version that you are now customizing.

Data set names for DB2 Sort

Data set name	Description	Special requirements	Your data set name
SCNKLINK	DB2 Sort SCNKLINK library	This data set must be APF-authorized.	
SCNKLPA	DB2 Sort SCNKLPA library	If this data set is not in the LPA, it must be APF-authorized.	
SCNKUSER	DB2 Sort SCNKUSER library	You must have read access to this data set.	

DB2 Sort installation options

In this worksheet, record the values for the DB2 Sort installation options.

DB2 Sort installation options

Installation option	Delivered default value	Discovered?	Your value
Sort work DASD pool	SYSDA	Yes	
Sort work DASD type	3390	Yes	
Number of allocation retry attempts	5	Yes	
Minutes between retry attempts	3	Yes	
Sort work storage class	None	Yes	
Maximum number of sort works	32	Yes	
TUNEMSG	ON	Yes	
OPTMODE	ELAP	Yes	
PAGEMON	OFF	Yes	

IVP parameters

If you want to run the Installation Verification Program (IVP), you will need to supply values on the DB2 Parameters panels for the items in the following table. This task is optional.

IVP parameters

Parameter	Required?	Default value	Your value
Mode This parameter indicates the mode in which the DB2 subsystem is running.	Yes	NFM	
Level number This parameter indicates the version, release, and modification level of the DB2 subsystem.	Yes	blank	
Load library This parameter indicates the data set name of the DB2 load library.	Yes	DSN.SDSNLOAD	
Run library This parameter indicates the data set name of the DB2 run library.	Yes	DSN.RUNLIB.LOAD	
SYSAFF for DB2 utilities	No	blank	
Plan name for the DSNTEP2 utility	Yes	DSNTEP2	
Schema name	Yes	SYSTOOLS	
SQL authorization ID This ID is used in IVP JCL while creating or altering DB2 objects used by DB2 Sort.	Yes	blank	
Database name for the IVP	Yes	CNKDBNAM	
Storage group name for the IVP	Yes	CNKSTORG	
VCAT name for DB2 storage group for the IVP	Yes	CNKVCAT	
Unique identifier for DB2 utilities for the IVP	Yes	CNKUID	

IVP parameters

Parameter	Required?	Default value	Your value
Spare 4K buffer pool for IVP table space	Yes	BP48	
Current VPSIZE for 4K BP used by IVP table space	Yes	2000	
Spare 4K buffer pool for IVP index space	Yes	BP49	
Current VPSIZE for 4K BP used by IVP index space	Yes	2000	
Table space name for IVP table 1	Yes	CNKTBSP1	
Table space name for IVP table 2	Yes	CNKTBSP2	
Table name for IVP table 1	Yes	CNKTBLN1	
Table name for IVP table 2	Yes	CNKTBLN2	
Index name 1 for IVP table 1	Yes	CNKIDX11	
Index name 2 for IVP table 1	Yes	CNKIDX12	
Index name 3 for IVP table 1	Yes	CNKIDX13	
Index name 4 for IVP table 1	Yes	CNKIDX14	
Index name 5 for IVP table 1	Yes	CNKIDX15	
Index name 1 for IVP table 2	Yes	CNKIDX21	
Index name 2 for IVP table 2	Yes	CNKIDX22	
Index name 3 for IVP table 2	Yes	CNKIDX23	

Worksheets: DB2 Utilities Enhancement Tool

Refer to the following information before you customize DB2 Utilities Enhancement Tool.

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.

Data set names for Tools Customizer

Identify and record the following Tools Customizer data set names:

Data set name	Special requirements	Your data set name
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	You must have write access to this data set.	

Data set names for DB2 Utilities Enhancement Tool

Identify and record the following DB2 Utilities Enhancement Tool data set names. During the customization process, you will enter the following values on panel CCQPPRD.

Data set name and description	Special requirements	Your data set name
SABPDBRM DBRM library for DB2 Utilities Enhancement Tool	None	
SABPLOAD Executable load module library for DB2 Utilities Enhancement Tool	You must APF authorize this data set.	
SABPMENU ISPF messages for DB2 Utilities Enhancement Tool	None	
SABPPENU ISPF panels for DB2 Utilities Enhancement Tool	None	

Data set name and description	Special requirements	Your data set name
SABPSAMP Sample members for DB2 Utilities Enhancement Tool	None	
SABPDENU Metadata library for DB2 Utilities Enhancement Tool product parameters	None	

Data set names of other libraries used by Tools Customizer

Identify and record the following data set names. During the customization process, you will enter the following values on the Tools Customizer Settings panel (CCQPSET).

Data set name and description	Special requirements	Your data set name
<p>Product customization library Contains the customization jobs that Tools Customizer generates for DB2 Utilities Enhancement Tool.</p> <p>To customize DB2 Utilities Enhancement Tool, 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:</p> <p><i>hlq.\$LPAR-name\$.xyzvrm</i></p> <p>where:</p> <ul style="list-style-type: none"> • <i>hlq</i> is the value of the Customization library qualifier field on the Tools Customizer Settings panel (CCQPSET) • <i>LPAR-name</i> is the four-character LPAR name • <i>xyzvrm</i> is the three-letter product identifier with the version, release, and modification level <p>For example, the data set name might be DB2TOOL.PRODUCT.CUST.\$MVS1\$.XYZ410.</p>	You must have write access to this data set.	

Data set name and description	Special requirements	Your data set name
<p>BBY load library data set Specifies the fully qualified library name for the DB2 Utilities Solution Pack load module (BBY\$NMIC).</p>	<p>If the DB2 Utilities Solution Pack (BBY) was not purchased, this value should be left blank.</p>	
<p>Discover output data set Contains the output that is generated when you run the DB2 Utilities Enhancement Tool Discover EXEC.</p> <p>The DB2 Utilities Enhancement Tool Discover EXEC retrieves the metadata and values for the parameters from a previous customization of DB2 Utilities Enhancement Tool.</p> <p>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.</p>	<p>You must have write access to this data set.</p>	
<p>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, and LPAR when you customize products on the local LPAR.</p> <p>The default name of the data set is DB2TOOL.CCQ110.DATASTOR. You can change the default value on the Tools Customizer Settings panel.</p>	<p>You must have write access to this data set.</p>	

Worksheets: Gathering parameter values for Tools Customizer

During the customization process, you will need to provide parameter values for DB2 Utilities Enhancement Tool, 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 the Tools Customizer.

Metadata library for DB2 Utilities Enhancement Tool

Description

Use the following worksheet to identify and record the value of the metadata library for DB2 Utilities Enhancement Tool. During the customization process, you will enter this value on the Specify the Metadata Library panel (CCQPHLQ).

Parameter	Discovered?	Your value
Metadata library The default name of the metadata library after the product has been SMP/E installed is <i>hlq.SABPDENU</i> , where <i>hlq</i> is the high-level qualifier for DB2 Utilities Enhancement Tool.	No	

Customization values for the Discover EXEC

Description

Use the following worksheet to identify and record the required customization values for the Tools Customizer Discover EXEC. The values in this worksheet are for extracting information from a product that has already been customized. During the customization process, you will enter these values on panel CCQPDSC.

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

Parameter	Sample or default value	Your value
Discover EXEC for Extracting Information from an Already Customized product		
Discover EXEC library The fully qualified data set name that contains the product Discover EXEC.	<i>hlq.mlq.SABPDENU</i>	
Discover EXEC name The name of the Discover EXEC.	ABPDISCO	
Discover output data set The fully qualified name of the data set for the output from the product Discover EXEC.	The name that you specified in option 0 User Settings from the Tools Customizer main menu.	
Information for Discover EXEC section		
DB2 UET V2.2 Configuration ID The configuration ID for which the current DISCOVER process is to be performed.	ABP1	

Parameter	Sample or default value	Your value
DB2 UET V2.1 SABPSAMP data set The SABPSAMP library from the previous installation of DB2 UET that was created as part of the SMP/E installation process.	ABP.V210.SABPSAMP	
DB2 UET V2.1 options module name The customized options module name in the sample library of the previous installation of DB2 UET that is specified in the previous field.	No default	
Copy existing policy member? Specify Y to copy the existing policy member from DB2 UET v2.1 for use with DB2 UET V2.2. Specify N to create a default policy in the DB2 UET v2.2 SABPSAMP data set.	Y	
DB2 UET V2.1 policy name If you would like to copy the policy member already in use, provide the name of the policy that you would like to re-use from the previous installation of DB2 UET utility.	No default	
DB2 UET V2.2 data set HLQ The high-level qualifier for the DB2 UET V2.2 data sets, which you specified during installation, or when you copied the target libraries.	ABP.MLQ	

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 DB2 Utilities Enhancement Tool metadata data set.

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.	Yes	This value is specified on the Specify the Product to Customize panel (CCQPHLQ)

Parameter	Discovered?	Source of this value
LPAR The LPAR field displays the LPAR on which you are customizing DB2 Utilities Enhancement Tool.	Yes	This value is supplied by Tools Customizer.
Product name This value displays the product that is being customized. In this example, DB2 Utilities Enhancement Tool 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.	Yes	The default value DB2 UET is provided by the product metadata file.
Version The Version field displays the version, release and maintenance of the product that you are customizing in the format <i>Vn.Rn.mn</i> .	Yes	This value is provided by the product metadata file. The default value for this release is 2.2.0.
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 panel CCQPPRD.

Note: Tools Customizer displays some parameters only after you have selected tasks or specified values on the Product Parameters panel. Therefore, you must first define a primary SSID on the DB2 Parameters panel, then select values on the Product Parameters panel. Return to the DB2 Parameters panel to review options that were added as a result of your specifications on the Product Parameters panel.

Parameter	Required?	Discovered?	Default value	Your value
DB2 UET data sets HLQ The DB2 UET high-level qualifier for the product data sets, which you specified at installation time or when you copied the target libraries.	Yes	Yes	No default	
DB2 UET plan qualifier The qualifier for the DB2 UET plan. This variable is also used as the collection ID for the DB2 UET packages and as the creator ID for the DB2 UET tables and indexes. Restriction: DB2 UET does not support double-byte character set (DBCS) characters in this value.	Yes	Yes	ABP22PLN	

Parameter	Required?	Discovered?	Default value	Your value
<p>DB2 UET started task user ID The IBM RACF® ACF2 or TSS user ID under which the started task will run. This ID will also be used as the OWNER identifier when binding the plan and packages and as the CURRENT SQLID (SQL authorization ID) when creating the product's DB2 objects and when issuing GRANT statements during object creation and following bind processing. Important: Ensure that this user ID has SYSADM or SYSCtrl authority on each DB2 subsystem where the DB2 UET plan will be bound.</p>	Yes	No	No default	
<p>BBY load library DSN Specifies the fully qualified library name for the DB2 Utilities Solution Pack load module (BBY\$NMIC). If the DB2 Utilities Solution Pack (BBY) with Autonomics Director was not purchased, this value should be left blank.</p>	No	No	No default.	
<p>DB2 UET plan name The name of the DB2 plan that DB2 UET uses. The plan name must be unique on the DB2 subsystem where the plan is bound or within the data sharing group to which that subsystem belongs.</p>	Yes	No	No default	
<p>DB2 UET configuration name An identifier for the DB2 UET started task configuration (also referred to as the ABPID). This value must be four alphanumeric characters long. You will need to specify an ABPID to access the ISPF interface and to create a batch job that blocks threads. Also, this ID serves as the basis of the sample started task name that Tools Customizer generates and inserts into the started task PROC. (The started task name is used in various product commands.) Tools Customizer adds ABP to the beginning of the ABPID to create the sample started task name. For example, if the ABPID is ABP1, the sample started task name would be ABPABP1. You can tailor the sample started task name in the started task PROC, if necessary.</p>	Yes	Yes	No default	

Parameter	Required?	Discovered?	Default value	Your value
<p>Work database name The name of the database in which you want to store work objects that are created for the extended functionality of the CHECK DATA utility.</p>	Yes	No	No default	
<p>Thread cancel member name Specifies the name of the member that contains the optional cancel-control parameters for any thread cancelations that you perform by using the DSNUTILB intercept. These options are:</p> <ul style="list-style-type: none"> • CANCEL_TYPE • ESCALATE • CHECK_THDTERM_RETRY_COUNT • CHECK_THDTERM_RETRY_INTERVAL <p>When Tools Customizer ran, DB2 UET generated the sample member <i>abpidBCAN</i> (where <i>abpid</i> is the started task configuration ID that you specified) in the <i>hlq.mlg.SABPSAMP</i> library for your use. If you changed the name of this member, you must specify the new name in this option. The member must reside in the same partitioned data set (a PDS or PDSE) as the started task initialization options member (that is, the data set that is allocated by the ABPOPTS DD statement in the started task PROC).</p>	Yes	No	ABP@BCAN	

Parameter	Required?	Discovered?	Default value	Your value
<p>Global parameters member name Specifies the name of the member that contains the optional global parameters for any thread cancelations that you perform by using the DSNUTILB intercept. These options are:</p> <ul style="list-style-type: none"> • ESCAPE • EXEC_TYPE • ON_FAILURE, REPORT_TYPE • THREAD_QUIESCE_TIME <p>When Tools Customizer ran, DB2 UET generated the sample member <i>abpidBGLB</i> (where <i>abpid</i> is the started task configuration ID that you specified) in the <i>hlq.mlg.SABPSAMP</i> library for your use. If you changed the name of this member, you must specify the new name in this option. This member must reside in the same PDS or PDSE as the started task initialization options member (that is, the data set that is allocated by the ABPOPTS DD statement in the started task PROC).</p>	Yes	No	ABP@BGLB	
<p>The DB2 UET primary subsystem ID The subsystem identifier for the primary DB2 subsystem where the audit and logging tables reside.</p> <p>If you are customizing parameters for a Secondary Subsystem, then leave this value as the Primary Subsystem name.</p>	Yes	Yes	No default	
<p>SYSOUT class The SYSOUT class of the DB2 UET started task.</p>	Yes	Yes	*	

Task: Create customized DB2 UET jobs

Description

This required task must be selected before you can select any of the subtasks that are described in the following sections. To create JCL that customizes an installation of DB2 UET, you must have first defined product-specific parameters and have associated a DB2 SSID with this customization. The generated job names themselves might vary, however, the template names do not.

Task: Create DB2 UET objects

Description

This task creates the required DB2 objects for use with DB2 UET for the subsystem on which DB2 UET was customized.

Template name

ABPOBJCR. You must run this template job once for each SSID on which you will use DB2 UET. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Task: Create additional indexes**Description**

This optional (but highly recommended) task creates three additional indexes on DB2 catalog tables to improve the performance of DB2 catalog queries.

Template name

ABPIDXCR. If you decide to use the performance indexes, you must run this template job at least once for each SSID on which you will use DB2 UET. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Tip: Generate the job to see the columns that are needed for the additional performance indexes, but before running the job, check for a suitable existing index that can be used.

If any of the indexes that the DB2 UET packages depend upon are dropped, then those packages are invalidated, causing a REBIND the next time that the package is invoked.

Task: Create LOG/AUDIT Migration job**Description**

This optional task extracts data from your existing LOG and AUDIT tables and loads the data into the new tables. This task is recommended if you want to use fully populated tables on which to base the statistics and plan.

Template name

ABPMIGRT. This template, when submitted, will use different methods to extract data from your existing Log and Audit tables and load the data into the new tables. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Step or parameter	Required?	Discovered?	Default value	Your value
V2.1 LOG and AUDIT tables CREATOR name Enter the CREATOR name value for your existing tables.	Yes	No	ABP21	

Task: Create JOURNAL Migration job**Description**

This optional task extracts data from your existing JOURNAL tables and loads the data into the new tables. This task is recommended if you want to use fully populated tables on which to base the statistics and plan.

Template name

ABPMIGR2. This template, when submitted, will use different methods to extract data from your existing Journal tables and load the data into the new tables. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Step or parameter	Required?	Discovered?	Default value	Your value
V2.1 JOURNAL tables CREATOR name Enter the CREATOR name value for your existing tables.	Yes	No	ABP21	

Task: Create RUNSTATS job

Description

This optional task runs the RUNSTATS utility to collect catalog statistics for the optional indexes on DB2 catalog tables. This task is recommended if you chose to create the three optional performance indexes on the DB2 catalog.

Template name

ABPRSTCR. You must run this template job at least once for each SSID on which DB2 UET performance indexes will be used. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Task: Create BIND job

Description

This task binds the required DB2 UET packages and plans for the subsystem on which DB2 UET was customized.

Template name

ABPBNDCCR. You must run this template job once for each SSID on which DB2 UET will be used. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Task: Create cleanup job for a worklist table

Description

This optional task creates SAMPLIB maintenance job *ssid#ERR* for the DB2 subsystem designated as the primary DB2 subsystem. *ssid* is the primary DB2 subsystem on which DB2 UET is being customized.

Job *ssid#ERR* is created for DB2 subsystems monitored by DB2 UET. The job contains SQL statements for manually deleting rows from the DSNUTILB-intercept worklist error tables that are older than the number of days specified. The job is stored in the *hlq.mlq.SABPSAMP* data set, where *hlq* and *mlq* are the product data set high-level qualifier and mid-level qualifier that were specified on the Product Parameters panel.

Template name

ABPWKLCR. You must run this template job at least once for each primary DB2 subsystem. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Task: Create the options module

Description

This task creates the options module *abpidOPTS* for use with the DB2 UET started task. *abpid* is the value that you specified in the **DB2 UET configuration name** field on the Product Parameters panel.

Template name

ABPOPTCR. This template is created once per DB2 UET configuration, and

you must run it at least once for each DB2 UET configuration that you customize. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Step or parameter	Discovered?	Default value	Your value
<p>STC audit active Controls whether DB2 UET records audit information for thread cancelation activities in a DB2 table. Specify YES to record this information, or specify NO to not record this information. The default value is YES. The product administrator will be able to temporarily override this setting by specifying another value in the Audit status field on the Control System panel.</p>	Yes	YES	
<p>Max STC audit age Indicates the maximum number of days to retain rows for audit information in the audit table (ABPAUDIT). This number of days is counted from the time when the rows are inserted into the table. When a row reaches this age limit, it is eligible for deletion. It will be automatically deleted from the table the next time a new row is inserted into the table. Valid values are from 0 through 32,767. The default value is 0, which prevents the automatic deletion of old rows from the audit table. If you accept the default value, you will need to manually delete old rows from the audit table periodically to prevent the table from becoming too large. Use the sample SQL that is provided in the SABPSAMP member <i>ssid#AUD</i>, where <i>ssid</i> is the 4-character subsystem ID where the auditing tables reside.</p>	Yes	0	
<p>Escalated cancel active Controls whether DB2 UET users are allowed to perform escalated cancelations of threads. An escalated cancelation uses the z/OS cancel command to terminate the job, TSO user, or started task that is associated with the thread. Specify YES to allow escalated cancelations, or specify NO to not allow escalated cancelations (to use only the DB2 -CANCEL THREAD command). The default value is NO.</p>	Yes	NO	

Step or parameter	Discovered?	Default value	Your value
<p>Connect to all DB2 subsystems? Controls whether DB2 UET attempts to connect to all active DB2 subsystems on the z/OS system on which it is configured, or only to the DB2 subsystem that is specified in the DB2_SSID initialization option (the subsystem that contains audit and logging information). If you specify YES (the default value), or if you omit this option from the initialization options member, DB2 UET will attempt to connect to all active DB2 subsystems by default. If you specify NO, DB2 UET will attempt to connect only to the primary subsystem that is specified in the DB2_SSID option; only that subsystem will be listed on the Set DB2 System panel in the ISPF interface.</p>	Yes	YES	
<p>Connection idle timeout Specifies the maximum amount of time (in seconds) that the DB2 connection for a DB2 UET task can have no activity. When this time limit is reached, the connection to DB2 closes. Valid values are from 0 through 32,767. The default value is 300. If you specify 0, this timeout option is disabled and will not cause an inactive connection to close. This timeout option does not apply to the subtask for the DB2 UET connection to the DB2 subsystem that is specified by the DB2_SSID option.</p>	Yes	300	
<p>DB2 tasks count Specifies the maximum number of z/OS tasks that DB2 UET can start for connection to a single DB2 subsystem. Valid values are from 1 through 2,147,483,647. The default value is 2.</p>	Yes	2	
<p>DB2 task idle timeout Specifies the maximum amount of time (in seconds) that a subtask for a DB2 UET connection to DB2 can remain inactive after the connection closes (that is, after the DB2_CONNECTION_IDLE_TIMEOUT limit has been met). When this time limit is reached, the subtask ends. Valid values are from 0 through 32,767. The default value is 900. If you specify 0, this timeout option is disabled and will not cause an inactive subtask to end. This timeout option does not apply to the subtask for the DB2 UET connection to the DB2 subsystem that is specified by the DB2_SSID option.</p>	Yes	900	

Step or parameter	Discovered?	Default value	Your value
<p>STC logging active Controls whether DB2 UET logs messages about product performance and operations in its DB2 log table. Specify YES to log messages, or specify NO to not log messages. The default value is YES. The product administrator will be able to temporarily override this setting by specifying another value in the Logging status field on the Control System panel.</p>	Yes	YES	
<p>Maximum STC log age Indicates the maximum number of days to retain rows for logged messages in the logging table (ABPLOG). This number of days is counted from the time when the rows are inserted into the table. When a row reaches this age limit, it is eligible for deletion. It will be automatically deleted from the table the next time a new row is inserted into the table. Valid values are from 0 through 32,767. The default value is 0, which prevents the automatic deletion of old rows from the logging table. If you accept the default value, you will need to manually delete old rows from the logging table periodically to prevent the table from becoming too large. To do so, use the sample SQL that is provided in the SABPSAMP member <i>ssid#LOG</i>, where <i>ssid</i> is the 4-character subsystem ID where the logging tables reside.</p>	Yes	0	
<p>Work object bufferpool The name of the buffer pool for any mapping-table indexes that DB2 UET automatically generates for the DB2 REORG TABLESPACE utility when the SHRLEVEL CHANGE option is specified. Tools Customizer will require you to specify a value for this variable.</p>	Yes	BP0	
<p>Work object storage group The name of the DB2 storage group (STOGROUP) that you want to use for the mapping-table indexes that DB2 UET can automatically generate for the DB2 REORG TABLESPACE utility when the SHRLEVEL CHANGE option is specified. Tools Customizer will require you to specify a value for this variable.</p>	Yes	SYSDEFLT	

Step or parameter	Discovered?	Default value	Your value
<p>Override mapping table Whether you want to replace any MAPPINGTABLE specifications that you manually defined in the input data sets for the DB2 REORG TABLESPACE utility with the MAPPINGTABLE specifications that DB2 UET automatically generates for the utility. Valid values are: Y (Yes) and N (No).</p>	Yes	NO	
<p>Post-cancel user exit Specifies the name of the user exit that you optionally created for performing some processing that you determined is necessary after thread cancelations. For example, you might create a post-cancel exit to notify users when thread-cancelation processing completes. If you are not using a post-cancel exit, specify NONE (the default value).</p>	Yes	NONE	
<p>Pre-cancel user exit Specifies the name of the user exit that you optionally created for performing some processing that you determined is necessary prior to thread cancelations. For example, you might create a pre-cancel exit to determine the DB2 objects that an application or utility will need to access. If you are not using a pre-cancel exit, specify NONE (the default value).</p>	Yes	NONE	
<p>Security-cancel exit Specifies the name of the user exit that you optionally created for verifying user authority to perform thread-management functions such as blocking and canceling threads and to access specific ISPF panels. If you are not using a security exit, specify NONE (the default value).</p>	Yes	NONE	
<p>Shadow database prefix Specifies a prefix for the names of shadow objects that are used with the IFDISCARDS and SHRELEVEL REFERENCE options in the LOAD utility. The prefix can be one to four characters, and must follow standard DB2 naming conventions for databases. For each invocation of the Load Prevalidate or LOAD REPLACE SHRLEVEL REFERENCE feature, DB2 UET generates a unique set of shadow object names by using the specified prefix and appending a numeric suffix.</p>	Yes	ABPS	

Step or parameter	Discovered?	Default value	Your value
<p>Shadow schema Specifies the schema (creator) to be for the shadow objects that DB2 UET creates. When executing the Load Prevalidate or the LOAD REPLACE SHRLEVEL REFERENCE feature, DB2 UET creates a unique set of shadow objects using the value of Shadow database prefix as the prefix for the object names. DB2 UET uses the value that you specify for Shadow schema as the schema name for the shadow objects that it creates.</p>	Yes	ABPSTC	
<p>SVC number The DB2 UET supervisor call (SVC) number. This number must be an integer from 200 through 255. Check with your systems programmer to ensure that you choose an SVC number that is available. The SVC will be dynamically installed when the DB2 UET started task starts and will be dynamically removed when the started task stops.</p>	Yes	255	
<p>DB2 UET trace active Controls whether DB2 UET collects trace information. Specify YES to enable tracing, or specify NO to disable tracing. The default value is YES. A trace is a record of DB2 UET internal processing that is primarily used by Support for diagnosing a problem. The product administrator can temporarily override this setting by specifying the another value in the Trace status field on the Control System panel.</p>	Yes	YES	
<p>Size of trace table Specifies the size (in MB) of the table in which DB2 UET stores trace information. Valid values are 1 - 2147483647. The default value is 1. A value of 0 will result in no trace table allocation. A trace is a record of internal processing that is primarily used by Support for diagnosing a problem.</p>	Yes	1	
<p>Workfile data class The name of a valid SMS data class for the temporary DASD data sets that are allocated by DB2 UET, or the value NONE.</p>	Yes	NONE	
<p>Workfile management class The name of a valid SMS management class for the temporary DASD data sets that are allocated by DB2 UET, or the value NONE.</p>	Yes	NONE	

Step or parameter	Discovered?	Default value	Your value
Workfile storage class The name of a valid SMS storage class for the temporary DASD data sets that are allocated by DB2 UET, or the value NONE.	Yes	NONE	
Workfile unit The unit name for the location where the temporary DASD data sets that are allocated by DB2 UET are stored. Specify a valid unit name of a storage device or the value NONE. You can specify VIO if VIO (virtual input/output) is supported on your system and you want the temporary data sets to reside entirely in paging storage to improve performance. Alternatively, you can specify SYSALLDA to use any available DASD device.	Yes	SYSALLDA	
Maximum worklist table age Specifies the maximum number of days to retain rows in the DSNUTILB intercept worklist-error tables. A DSNUTILB intercept worklist contains the enhanced SYSIN information for a DB2 utility and can be used for restart purposes if a utility terminates. Worklist data is moved to worklist-error tables for diagnostic use by Customer Support in certain situations. After rows in the worklist-error tables reach the specified age limit, they are eligible for deletion. The next time a new row is inserted into a worklist-error table, the rows that meet the age limit are deleted. You can specify a value from 0 through 32,767 for this option. The default value is 0, which prevents the deletion of old rows from the worklist-error tables based on this option. If you accept the default value, you might need to manually delete old rows from these tables periodically to prevent the tables from becoming too large. Use the sample SQL that is provided in the SABPSAMP member <i>ssid#ERR</i> , where <i>ssid</i> is the 4-character subsystem on which the tables reside.	Yes	0	
WTO ROUTCDE Specifies the routing code for write-to-operator (WTO) messages regarding DB2 UET operations. Routing codes identify the z/OS console to which to send WTO messages and are defined when DB2 is installed. Valid values are from 1 through 28. The default value is 11. The product administrator will be able to temporarily override this value by entering another routing code on the Control System panel.	Yes	11	

Step or parameter	Discovered?	Default value	Your value
Work database storage group The name of the storage group that you want to use for the mapping-table objects that are created for the extended functionality of the REORG TABLESPACE utility.	No	SYSDEFLT	
Work bufferpool name The name of the buffer pool that you want to use for the mapping-table objects that are created for the extended functionality of the REORG TABLESPACE utility.	No	BP0	
DB2 UET system table spaces STOGROUP Specifies the name of the storage group that is used internally by DB2 UET for table spaces that contain audit and logging tables and the thread-blocker and DSNUTILB intercept tables. Valid values must follow standard DB2 naming conventions for storage groups.	No	SYSDEFLT	
DB2 UET system table spaces buffer pool Specifies the buffer pool that is used for table spaces that contain the DB2 UET audit and logging tables and the thread-blocker and DSNUTILB intercept tables. Valid values must follow standard DB2 naming conventions for buffer pools.	No	BP0	
DB2 UET system index spaces STOGROUP Specifies the name of the storage group that is used internally by DB2 UET for index spaces that contain indexes on audit and logging tables and the thread-blocker and DSNUTILB intercept tables. Valid values must follow standard DB2 naming conventions for storage groups.	No	SYSDEFLT	
DB2 UET system index spaces buffer pool Specifies the buffer pool that is used for index spaces that contain indexes on the DB2 UET audit and logging tables and the thread-blocker and DSNUTILB intercept tables. Valid values must follow standard DB2 naming conventions for buffer pools.	No	BP0	

Task: Create BCAN, BGLB, PROC, PLCY members

Description

This optional task creates the following SAMPLIB members that are used by the DSNUTILB Intercept to block and cancel threads. In each member name, *abpid* is the value that you specified in the **DB2 UET configuration name** field on the Product Parameters panel. Each member is created once per DB2 UET configuration that is being customized. Each member is stored in the *hlq.mlq.SABPSAMP* data set, where *hlq* and *mlq* are the

product data set high-level qualifier and mid-level qualifier that were specified on the Product Parameters panel.

- *abpidBCAN* contains the optional thread-cancel parameters that the DSNUTILB intercept uses for each cancel request for a DB2 utility.
- *abpidBGLB* contains the optional global parameters that pertain to intercept processing activities that the DSNUTILB Intercept performs for the started task configuration.
- *abpidPROC* contains the started task PROC for DB2 UET.
- *abpidPLCY* contains a sample policy for use with the started task for the DB2 UET.

Template name

ABPMODCR. This template is created once per DB2 UET configuration, and you must run it at least once for each DB2 UET configuration that you customize. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Task: Create maintenance members

Description

This optional task creates the following SAMPLIB members. In each member name, *abpid* is the value that you specified in the **DB2 UET configuration name** field on the Product Parameters panel. Each member is created once per DB2 UET configuration that is being customized.

- *abpidCOMX*, which contains a utility that resets DB2 UET COMX control blocks. Use this utility only when IBM Software Support directs you to do so.
- *abpidMAIN*, which contains the job that runs the DB2 UET batch interface program.
- *abpidMNT*, which contains the job that runs the ABPMAINT utility. This utility maintains the intercept worklist tables on the primary SSID. You can use the utility to
 - terminate a DB2 utility for which DSNUTILB interception has occurred and to remove the worklist data that is associated with the utility ID
 - restart a DB2 utility from the appropriate point after it ended because of exceptional circumstances that prevented a normal DB2 restart

Task **Create maintenance members** also creates the following SAMPLIB members for DB2 subsystems that are designated as the primary subsystem:

- *ssid#LOG*, which contains SQL statements for manually deleting rows from the log tables that are older than the number of days specified.
- *ssid#AUD*, which contains SQL statements for manually deleting rows from the audit table that are older than the number of days specified.

All members are stored in the *hlq.mlq.SABPSAMP* data set, where *hlq* and *mlq* are the product data set high-level qualifier and mid-level qualifier that were specified on the Product Parameters panel.

Template name

ABPSMPCR. This template is created once per DB2 UET configuration, and must be run at least once for each DB2 UET configuration that is being customized. The generated job is stored in the Product Customization Library that is displayed on the Finish Product Customization panel.

Task: Create product CLISTS

Description

This optional task creates CLIST members ABPRUN and ABPF.

Template name

ABPRUNCR. This template is created once per DB2 UET configuration and is required to be run at least once. This generated job is stored in the **Product Customization Library** that is displayed on the Finish Product Customization panel.

DB2 Parameters section

Description

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

You can create a DB2 entry as the primary subsystem or secondary subsystem and associate it with DB2 UET. When customizing DB2 UET, you must first define a primary subsystem before you can define product parameters.

You can customize DB2 UET only on DB2 entries that are associated with DB2 UET. The list of DB2 entries is on the Customizer Workplace panel. You can customize any associated DB2 entries for DB2 UET.

Note: Tools Customizer displays some parameters only after you have selected tasks or specified values on the Product Parameters panel. Therefore, you must first define a primary SSID on the DB2 Parameters panel, then select values on the Product Parameters panel. Return to the DB2 Parameters panel to review options that were added as a result of your specifications on the Product Parameters panel.

Parameter	Discovered?	Default value	Your value
DB2 subsystem ID The name of the DB2 subsystem you want to associate with DB2 UET. The value must be 4 characters or less. An example of a DB2 subsystem name is DB01.	No	No default	
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.	No	No default	
Is this DB2 subsystem the primary subsystem? Specify YES if this DB2 SSID is to be used as the primary subsystem. Auditing and logging tables will be created for DB2 UET on this subsystem. Specify NO if this DB2 SSID is being configured as a secondary subsystem. Auditing and logging tables will not be created for DB2 UET on this subsystem.	No	NO	

Parameter	Discovered?	Default value	Your value
Use DBCS with this SSID? Specify if this DB2 SSID is defined for use with double-byte character sets.	No	NO	
DB2 UET database name The name of the database (up to eight alphanumeric characters) that will be used for the DB2 UET auditing and logging tables.	No	No default	
Drop DB2 UET database first? Specify whether you would like to create a DROP DATABASE statement within the SQL to first drop any previously created database. This option is helpful if you have already customized DB2 UET V2.2 and want to drop the database that was previously created.	No	YES	
Drop DB2 UET work database first? Specify if you would like to create a DROP DATABASE statement within the SQL to first drop any previously-created databases are used for work objects that DB2 UET creates. This option is helpful if you are re-customizing DB2 UET and want to drop the work database that was created before.	No	YES	
Mode The mode in which the DB2 subsystem is running. Valid values are CM, CM8, CM9, and NFM. CM is compatibility mode on V8. CM8 is conversion mode on V9 and V10. CM9 is conversion mode on V10. NFM is new-function mode on any DB2 version.)	No	No default	
Level number The version, release, and modification level of the DB2 subsystem. Valid values are 810, 910, and 101.	No	No default	
Load library The fully qualified data set name of the DB2 load library.	No	No default	
Run library The fully qualified data set name of the DB2 run library.	No	No default	
Plan name for the DSNTEP2 utility The name of the plan (up to eight alphanumeric characters) that is used for the DB2 DSNTEP2 program.	No	No default	

Worksheets: Gathering parameter values for Tools Customizer

During the customization process, you will need to 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.

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 “Data sets that Tools Customizer uses during customization” on page 278

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 output data set that is dynamically generated during the customization process.	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 fully qualified name of the Tools Customizer SCCQDENU data set.	DB2TOOL.CCQ110.SCCQDENU	
Discover output data set The fully qualified name of the data set in which the output from the DB2 Utilities Solution Pack Discover EXEC is stored. This data set is dynamically generated during the customization process.	DB2TOOL.CCQ110.DISCOVER	

Tools Customizer Library Settings

Parameter	Sample or default value	Your value
Data store data set The fully qualified name of the output data set where Tools Customizer stores information about product or component, LPAR, and DB2 parameter values. This data set is dynamically generated during the customization process.	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 Utilities Solution Pack

Description

Use the following worksheet to identify and record the value of the metadata library for DB2 Utilities Solution Pack. During the customization process, you will enter this value on the Specify the Metadata Library panel (CCQPHLQ).

DB2 Utilities Solution Pack metadata library

Parameter	Sample or default value	Your value
Metadata library The fully qualified name of the DB2 Utilities Solution Pack SADBDENU data set.	hlq.SBBYDENU	

Product to customize section

Description

The parameters that are listed in the Product to Customize section on the Product Parameters panel (CCQPPRD) are read-only. They contain information that was provided on other panels, by Tools Customizer, or by the DB2 Utilities Solution Pack metadata data set.

Read-only Product to Customize parameters

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).

Read-only Product to Customize parameters

Parameter	Discovered?	Source of this value
LPAR The LPAR field displays the LPAR on which you are customizing DB2 Utilities Solution Pack.	No	This value is provided by Tools Customizer.
Product name This value displays the product that is being customized. In this example, DB2 Utilities Solution Pack 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 <i>Vn.Rn.nn</i> .	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).

Chapter 3. Pre-customization tasks

Some components of the DB2 Utilities Solution Pack has pre-customization tasks that must be completed before you can customize the component. These tasks are identical to the ones that apply to the stand-alone version of a component (that is, as a separately orderable product).

Pre-customization tasks: DB2 High Performance Unload

Before you customize DB2 High Performance Unload, ensure that your environment meets the hardware and software requirements for this component.

Authorizing and enabling DB2 HPU

Before you can use DB2 HPU for the first time, you must APF authorize the SINZLINK library and enable the DB2 HPU batch utility.

About this task

Enabling the DB2 HPU batch utility lets you unload DB2 data. You must create a batch utility job before you can run the DB2 HPU batch utility.

Procedure

1. Specify the SETPROG command in the following format:
`SETPROG APF,ADD,DSNAME=SINZLINK-data-set-name,VOLUME=volume-name`

The following example shows the SETPROG command in the correct format:

```
SETPROG APF,ADD,DSNAME=DMT.PTF420.SINZLINK,VOLUME=V01001
```

You can include this command in a MVS procedure, in a job, or in both.

2. Use one of the following methods to enable the DB2 HPU batch utility.
 - Put the SINZLINK library in LINKLIST.
 - Specify the SINZLINK library in the STEPLIB statement of all jobs. (Be careful not to lose the APF authorizations with the concatenations.)

Optional: Creating a TSO command for the DB2 HPU interactive component

You can make the DB2 HPU interactive component available as a TSO command.

About this task

With the DB2 HPU interactive component, you can generate the batch utility job to be used to invoke the DB2 HPU batch utility program.

You can run the batch utility program and the DB2 HPU interactive component simultaneously.

Procedure

Use one of the following methods to create a TSO command:

- Concatenate the SINZCLST library to the SYSPROC concatenation.

- Copy the INZHPU procedure into a library in the SYSPROC concatenation.

What to do next

Invoke the DB2 HPU interactive component by issuing the following command:
TSO INZHPU.

Pre-customization tasks: DB2 Sort

You must activate DB2 Sort after it is installed on your system. After activating DB2 Sort, you can use Tools Customizer to change the default installation options.

Note: For the best resource utilization, set the **UTSORTAL** parameter to YES. This parameter enables you to use real-time statistics to provide the most accurate values for resource allocation.

For information about setting either the **UTSORTAL** or **SORTNUM** parameter, see one of the following documents:

- DB2 10 for z/OS Installation and Migration Guide (GC19-2974)
- DB2 Version 9.1 for z/OS Installation Guide (GC18-9846)

Activating DB2 Sort

Before customizing or using DB2 Sort, you must activate it or verify that it has been activated.

About this task

You may have to activate DB2 Sort for use in DB2 for z/OS if it has been migrated from a release earlier than DB2 10 for z/OS.

Important: If you are using a newly installed DB2 10 for z/OS or DB2 11 for z/OS, DB2 Sort is automatically activated, so no action is required.

To activate DB2 Sort, use the following procedure.

Procedure

1. Apply the following program fix APAR PI19784: DB2 SORT FOR Z/OS PRECONDITIONING.
2. Set the DB2 subsystem parameter **DB2SORT** to ENABLE.

For information about setting the **DB2SORT** parameter, see one of the following documents:

- *DB2 11 for z/OS Installation and Migration Guide (GC19-4056)*
- *DB2 10 for z/OS Installation and Migration Guide (GC19-2974)*

What to do next

1. Stop and restart the DB2 subsystem.
2. Customize, test or begin using DB2 Sort.

Chapter 4. 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 your components.

Best Practice: SMP/E and runtime libraries maintenance strategy for Tools Customizer

Tools Customizer creates relationships between the values for the Product Parameters, LPAR Parameters, and DB2 Subsystem Parameters for each Tools Customizer enabled product. Determining the correct maintenance strategy for your Tools Customizer runtime libraries, after SMP/E processing, can reduce problems working with Tools Customizer and the enabled products through their life cycles.

Tools Customizer has very specific requirements for data set names:

- Only one DATASTOR data set exists per LPAR
- The product metadata library data set names do not change during the life of that release of the Tools Customizer enabled product.

The DATASTOR data set is the repository for all the information that Tools Customizer requires to generate customization JCL for enabled products.

When you update and save the Tools Customizer Settings panel (CCQPSET), as described in “Modifying Tools Customizer user settings” on page 168, the name of the DATASTOR data set is saved in the ISPF profile. This allows Tools Customizer to know the active DATASTOR data set when the TSO user id logs in and starts the Tools Customizer EXEC.

Maintenance scenarios

IBM expects maintenance to be applied to libraries which are then used by Tools Customizer. In practice, different customer shops distribute SMP/E APPLY maintenance in different ways.

The following scenarios explains some considerations and alternatives for determining your maintenance strategy. The one overriding objective is to preserve and maintain the same data set names for the Tools Customizer instance.

Apply SMP/E maintenance to the same data sets (using the SMP/E APPLY command):

If you apply SMP/E maintenance (using the SMP/E APPLY command) using the same data set names with each maintenance cycle, you can either use these target libraries as your Tools Customizer runtime libraries or you can copy the SMP/E target data sets to the runtime libraries that are used by Tools Customizer to customize enabled products.

Tools Customizer assumes that if the product metadata library (*DENU) has the same name, this metadata library is for the same release of the enabled product. For example, assume that you customize DB2 Log Analysis Tool v3.3 and name the metadata library SYS2.DB2T00L.SALADENU, with no indication of the version or release. You then upgrade to DB2 Log

Analysis Tool v3.4 and employ the same naming convention, SYS2.DB2T00L.SALADENU. Tools Customizer will assume that you are continuing to work with DB2 Log Analysis Tool v3.3 and will report v3.3 on panels and continue to use the same v3.3 Customization Library data set.

A more sustainable approach to naming the data sets is to include a product version, release identifier, or other distinguishing qualifier in the name of the metadata library, so that Tools Customizer can determine the new product release when you upgrade. For example, using metadata library names that include a product version, release identifier, or other distinguishing qualifier, similar to the following, can make product maintenance and upgrades easier:

- DB2T00L.R330.SALADENU for DB2 Log Analysis Tool v3.3
- DB2T00L.R340.SALADENU for DB2 Log Analysis Tool v3.4

Apply SMP/E maintenance to new data sets (using the SMP/E APPLY command):

If you apply SMP/E maintenance (using the SMP/E APPLY command) to new data sets rather than to the same data sets, the next time you open the product metadata library, Tools Customizer will return a data set error that indicates that the library name is being used by another product or component.

For example, assume that you name the DB2 High Performance Unload for z/OS target metadata libraries to reflect the date of an upgrade or to reflect a specific RSU, as follows:

- To reflect a specific upgrade date (August 2014):
 - DB2T00L.PTF420.SINZDBRM.D201408
 - DB2T00L.PTF420.SINZDENU.D201408
 - DB2T00L.PTF420.SINZLOAD.D201408
- To reflect a specific RSU (RSU 1406):
 - DB2T00L.PTF420.SINZDBRM.RSU1406
 - DB2T00L.PTF420.SINZDENU.RSU1406
 - DB2T00L.PTF420.SINZLOAD.RSU1406

Using either of these naming conventions, the next time you start the Tools Customizer EXEC, it will return a data set error.

To handle this type of SMP/E maintenance processing, you can do either of the following:

- Define aliases (using ALIAS control statements) to reference the appropriate libraries for Tools Customizer processing.
- Copy the SMP/E libraries to a set of runtime libraries that are specifically for Tools Customizer processing.

Define aliases (using ALIAS control statements) to reference the appropriate libraries for Tools Customizer processing:

Defining aliases (using ALIAS control statements) for the SMP/E created new product data set names is likely the best strategy when planning for Tools Customizer.

If you are setting up Tools Customizer for the first time, consider specifying the product library data set names with an indicator that these data sets will be used for Tools Customizer processing. For example, use data set names similar to the following names:

- TCZ.PTF420.SINZDBRM
- TCZ.PTF420.SINZDENU
- TCZ.PTF420.SINZLOAD

After applying maintenance using SMP/E, which creates new product library data sets, you should define aliases (using ALIAS control statements) for the new data set names to the data set names that Tools Customizer originally processed. For example, the following maintenance data sets have aliases defined to the original data sets:

- DB2TOOL.PTF420.SINZDBRM.RSU1406 --> TCZ.PTF420.SINZDBRM
- DB2TOOL.PTF420.SINZDENU.RSU1406 --> TCZ.PTF420.SINZDENU
- DB2TOOL.PTF420.SINZLOAD.RSU1406 --> TCZ.PTF420.SINZLOAD

You will need to define an alias (using ALIAS control statements) to each of the following IBM-distributed Tools Customizer data sets:

- SCCQDENU
- SCCQEXEC
- SCCQLOAD
- SCCQMENU
- SCCQPENU
- SCCQSAMP
- SCCQTENU

If you have already set up Tools Customizer and customized the product, you will have to define aliases (using ALIAS control statements) for the newly created data set names to the data sets that were specified when the product was originally customized using Tools Customizer.

After defining the aliases, you should be able to run Tools Customizer successfully.

Note: Only define aliases for IBM-distributed SMP/E libraries.

Restriction: Do not define an alias for any Tools Customizer created data sets, like the following three data sets on the Tools Customizer Settings panel (CCQPSET):

- Customization library qualifier
- Discover output data set
- Data store data set

Do not define an alias for any data sets that you create on behalf of a product, for example the DB2 High Performance Unload for z/OS parameter library (INFPLIB).

Copy the SMP/E libraries to a set of runtime libraries that are specifically for Tools Customizer processing:

If you are setting up Tools Customizer for the first time, consider specifying the product library data set names with an indicator that these data sets will be used for Tools Customizer processing. For example, use data set names similar to the following names:

- TCZ.R420.SINZDBRM
- TCZ.R420.SINZDENU

- TCZ.R420.SINZLOAD

After applying maintenance using SMP/E, which creates new product library data sets, you should copy the new data sets to the data sets that Tools Customizer originally processed. For example, copy the following maintenance data sets to the original data sets:

- DB2T00L.PTF420.SINZDBRM.RSU1406 --> TCZ.R420.SINZDBRM
- DB2T00L.PTF420.SINZDENU.RSU1406 --> TCZ.R420.SINZDENU
- DB2T00L.PTF420.SINZLOAD.RSU1406 --> TCZ.R420.SINZLOAD

You will need to copy each of the following IBM-distributed Tools Customizer data sets:

- SCCQDENU
- SCCQEXEC
- SCCQLOAD
- SCCQMENU
- SCCQPENU
- SCCQSAMP
- SCCQTENU

If you have already set up Tools Customizer and customized the product, you will have to copy the newly created data sets to the data sets that were specified when the product was originally customized using Tools Customizer).

After copying the SMP/E data sets to the Tools Customizer instance libraries, you should be able to run Tools Customizer successfully.

Note: Only copy IBM-distributed SMP/E libraries.

Restriction: Do not copy any Tools Customizer created data sets, like the following three data sets on the Tools Customizer Settings panel (CCQPSET):

- Customization library qualifier
- Discover output data set
- Data store data set

Do not copy any data sets that you create on behalf of a product, for example the DB2 High Performance Unload for z/OS parameter library (INFPLIB).

Related tasks:

Modifying Tools Customizer user settings

Before you can customize a product or a component with Tools Customizer, you must review the settings that Tools Customizer uses.

Related information:

The SMP/E APPLY command

The APPLY command specifies which of the received SYSMODs are to be selected for installation in the target libraries.

Alias processing: SMP/E for z/OS Commands

When an element with aliases is processed, both the element and its aliases are updated. SMP/E does not check the aliases against elements maintained in the target zone.

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'

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 Utilities Solution Pack.

Modifying Tools Customizer user settings

Before you can customize components in the DB2 Utilities Solution Pack 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 components in the DB2 Utilities Solution Pack 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:

```
CCQPSET          Tools Customizer Settings          14:03:51
Command ==>
Enter the settings for customizing a product or press End to save and exit.

Commands: SAVE - Save user settings

Product Customization Settings
Customization library qualifier . . DB2TOOL.PRODUCT.CUST
Use DB2 group attach . . . . . YES (YES/NO)

Tools Customizer Library Settings
Metadata library . . . . . DB2TOOL.CCQ110.SCCQDENU
Discover output data set . DB2TOOL.CCQ110.DISCOVER
Data store data set . . . DB2TOOL.CCQ110.DATASTOR

User Job Card Settings for Customization Jobs
====> //          JOB
====>
====>
====>
====>
```

Figure 3. The Tools Customizer Settings panel (CCQPSET)

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 components in the DB2 Utilities Solution Pack are stored. Write access to this qualifier is required.

For each component 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 component 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 (').

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 component 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:

*Table 9. The effect of the value of the **Use DB2 group attach** field in a data sharing environment*

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

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:

- V91C, which is a stand-alone DB2 subsystem
- V91A, 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 Lvl Mode User ID Date      Status      Message
V91C  --          910 NFM  SYSADM  2010/11/09 Ready to Customize
V91A  DSG1         910 NFM  SYSADM  2010/11/09 Ready to Customize
--    DSGA         910 NFM  SYSADM  2010/11/09 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 10. Value that is used in the CONNECT statements in the generated jobs

SSID	GrpAttach	Value of the Use DB2 group attach field	Value that is used in the CONNECT statements
V91C	--	Yes	SSID
		No	SSID
V91A	DSG1	Yes	Group attach name
		No	SSID
--	DSGA	Yes	Group attach name
		No	Group attach name

does not support DB2 group attach names. You must specify NO in the **Use DB2 group attach field**.

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 Component Parameters panel for the component 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 component Discover EXEC is stored. Each component has its own Discover EXEC. The Discover EXEC retrieves the component, LPAR, and DB2 parameters from a previously customized component. Write access to this data set is required. Data set names that exceed 42 characters must be enclosed in single quotation marks (').

Data store data set

The name of the data set where Tools Customizer stores information about component, LPAR, and DB2 parameter values. Information about which components 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 (').

User job card settings for customization jobs

The job card information to be inserted into the generated jobs for customizing a component. 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 components in the DB2 Utilities Solution Pack or to change parameter settings.

Related concepts:

Chapter 5, “Customizing components in the DB2 Utilities Solution Pack,” on page 173

Using Tools Customizer to customize components in the DB2 Utilities Solution Pack consists of identifying the pack to customize; selecting components of the pack to customize; defining any required component, LPAR, and DB2 parameters; generating the customization jobs; and submitting the jobs.

Hiding and displaying panel text

After you are familiar with Tools Customizer, you might want to hide the instructions and some of the basic descriptions that are displayed by default on Tools Customizer panels.

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 LPAR Parameters panel (CCQPLPR), and the DB2 Parameters panel (CCQPDB2).

By hiding this information, more data can be displayed on the panels. Later, you can redisplay this information also by using the OPTIONS command.

Procedure

1. On any Tools Customizer panel, issue the OPTIONS command. The Panel Display Options panel (CCQPOPT) is displayed, as shown in the following figure. By default, all options are preselected with a slash (/) to be shown.

```
CCQPOPT                Panel Display Options

Select panel display options and press Enter. To cancel, press End.

Panel Display Options
Specify a slash (/) to select options.
/ Show the panel instructions
/ Show the Product to Customize section
/ Show the Usage Notes section
Command ==>
```

Figure 4. The Panel Display Options panel (CCQPOPT)

2. To hide any of the options, remove the slash, and press Enter.

Chapter 5. Customizing components in the DB2 Utilities Solution Pack

Using Tools Customizer to customize components in the DB2 Utilities Solution Pack consists of identifying the pack to customize; selecting components of the pack to customize; defining any required component, LPAR, and DB2 parameters; generating the customization jobs; and submitting the jobs.

Customization roadmaps describe the steps that you must complete to customize one or more components in the DB2 Utilities Solution Pack. Separate roadmaps are provided for the three most common types of customizations.

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

Table 11. Customization roadmaps

Environment description	Roadmap
You do not have customized versions of the components in the DB2 Utilities Solution Pack, and you need to customize one or more of them for the first time.	"Roadmap: Customizing components for the first time"
You already customized one or more components in the DB2 Utilities Solution Pack as stand-alone products. Now, you have installed the DB2 Utilities Solution Pack, and you want to use the same parameter values to customize these components.	"Roadmap: Customizing new versions of components from a previous customization" on page 174
You have customized versions of the components in the of DB2 Utilities Solution Pack, but you want to change one or more parameter values.	"Roadmap: Recustomizing components" on page 175

Roadmap: Customizing components for the first time

This roadmap lists and describes the steps that are required to customize one or more components in the DB2 Utilities Solution Pack for the first time.

If you are customizing a previous version of one or more components in the DB2 Utilities Solution Pack, see "Roadmap: Customizing new versions of components from a previous customization" on page 174.

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

- All of the pack 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 components for the first time.

Table 12. Steps for customizing for the first time

Step	Description	Instructions
1	Specify the product metadata library for the pack that you want to customize. The name of this library is <i>hlq.SDENU</i> .	"Specifying the metadata library for the pack to customize" on page 177
2	Select the components of the pack that you want to customize.	"Selecting components to customize" on page 178
3	Create new DB2 entries and associate them with components of the pack.	"Creating and associating DB2 entries" on page 181
4	Define the required parameters.	"Defining parameters" on page 183
5	Generate the customization jobs for the component or for the DB2 entries on which the component is ready to be customized.	"Generating customization jobs" on page 188
6	Submit the generated customization jobs.	"Submitting customization jobs" on page 189

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

Table 13. Administrative tasks

Description	Instructions
Browse the different types of parameters.	"Browsing parameters" on page 191
Copy an existing DB2 entry to the list of DB2 entries on which a component can be customized.	"Copying DB2 entries" on page 191
Remove one or more DB2 entries from the associated list.	"Removing DB2 entries" on page 193
Delete one or more DB2 entries from the master list.	"Deleting DB2 entries" on page 193
Display a list of customization jobs that have been previously generated.	"Displaying customization jobs" on page 194
Maintain the customization jobs in the customization library.	"Maintaining customization jobs" on page 194

Roadmap: Customizing new versions of components from a previous customization

This roadmap lists and describes the steps for customizing new versions of components in the DB2 Utilities Solution Pack based on the existing customization values of a previous version of the same components in the pack.

Use this roadmap even if the previous version of DB2 Utilities Solution Pack was not customized by using Tools Customizer.

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

- All of the pack 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 new versions of components in the DB2 Utilities Solution Pack from a previous customization.

Table 14. Steps for customizing new versions of components in the DB2 Utilities Solution Pack from a previous customization

Step	Description	Instructions
1	Specify the product metadata library for the pack that you want to customize. The name of this library is <i>hlq.SDENU</i> .	"Specifying the metadata library for the pack to customize" on page 177
2	Select the components of the pack that you want to customize.	"Selecting components to customize" on page 178
3	Use the component Discover EXEC to discover information about the version of that you previously customized manually.	"Discovering component information automatically" on page 179
4	Define the required parameters.	"Defining parameters" on page 183
5	Generate the customization jobs for the component or for the DB2 entries on which the component is ready to be customized.	"Generating customization jobs" on page 188
6	Submit the generated customization jobs.	"Submitting customization jobs" on page 189

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

Table 15. Administrative tasks

Description	Instructions
Browse the different types of parameters.	"Browsing parameters" on page 191
Copy an existing DB2 entry to the list of DB2 entries on which a component can be customized.	"Copying DB2 entries" on page 191
Remove one or more DB2 entries from the associated list.	"Removing DB2 entries" on page 193
Delete one or more DB2 entries from the master list.	"Deleting DB2 entries" on page 193
Display a list of customization jobs that have been previously generated.	"Displaying customization jobs" on page 194
Maintain the customization jobs in the customization library.	"Maintaining customization jobs" on page 194

Roadmap: Recustomizing components

This roadmap lists and describes the steps to change parameter values and regenerate customization jobs for components in the DB2 Utilities Solution Pack 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 pack 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 components.

Table 16. Required steps for recustomizing components

Step	Description	Instructions
1	Specify the product metadata library for the pack that you want to recustomize. The name of this library is <i>hlq.SDENU</i> .	"Specifying the metadata library for the pack to customize" on page 177
2	Select the components of the pack that you want to recustomize.	"Selecting components to customize" on page 178
3	Edit the specific tasks, steps, or parameters that need to be changed.	<ul style="list-style-type: none"> • "Defining component parameters" on page 183 • "Defining LPAR parameters" on page 185 • "Defining DB2 parameters" on page 186
4	Generate the customization jobs for the component or for the DB2 entries on which the component is ready to be customized.	"Generating customization jobs" on page 188
5	Submit the new generated customization jobs.	"Submitting customization jobs" on page 189

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

Table 17. Administrative tasks

Description	Instructions
Browse the different types of parameters.	"Browsing parameters" on page 191
Copy an existing DB2 entry to the list of DB2 entries on which a component can be customized.	"Copying DB2 entries" on page 191
Remove one or more DB2 entries from the associated list.	"Removing DB2 entries" on page 193
Delete one or more DB2 entries from the master list.	"Deleting DB2 entries" on page 193
Display a list of customization jobs that have been previously generated.	"Displaying customization jobs" on page 194

Table 17. Administrative tasks (continued)

Description	Instructions
Maintain the customization jobs in the customization library.	“Maintaining customization jobs” on page 194

Specifying the metadata library for the pack to customize

You must specify a metadata library for the pack 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 Utilities Solution Pack. This information controls what is displayed on the Component Parameters panel, the LPAR Parameters panel, and the DB2 Parameters panel.

After DB2 Utilities Solution Pack has been SMP/E installed, the default name of the pack metadata library is *high_level_qualifier.SBBYDENU*, 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 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 Metadata Library          13:09:50
Command ==>>>          Scroll ==>> PAGE

Type the name of the metadata library for the pack or the product 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 pack or product has been
SMP/E installed is <hlq>.SxxxDENU, where <hlq> is the high-level qualifier for
the pack or the product, and xxx is the 3-character prefix for the pack or
the product.

Metadata library . BBY.ALIAS.SBBYDENU

Previously Used Metadata Library:

=>
=>
=>
=>

```

Figure 5. The Specify the Metadata Library panel

2. Use one of the following methods to specify the pack metadata library:
 - Type the name of a fully qualified partitioned data set (PDS) or an extended partitioned data set (PDSE) in the **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 Utilities Solution Pack for the first time, you must use this method.

- Place the cursor on the library name in the Recent Metadata Libraries list, and press Enter.

Results

The Select the Components to Customize panel is displayed.

What to do next

- Select the components of the pack that you want to customize.

Selecting components to customize

If you are customizing a solution pack, you must select which components of the pack to customize. If you are customizing a product, skip this step.

Procedure

1. On the Select the Components to Customize panel, specify / next to the names of the components that you want to customize. The following figure shows an example of the Select the Components to Customize panel.

```

CCQPSEC          Select the Components to Customize      Row 1 to 4 of 4
Command ==>>>                                     Scroll ==>>> PAGE

Select one or more components to customize. Press Enter to continue
or End to cancel.

Pack metadata library . : BBY.ALIAS.BBYDENU
Pack to customize . . . : DB2 Utilities Solution Pack   > Version . : 2.1.0

Line commands: / - Select
  Cmd Name                                     Version
----->-----
  IBM DB2 Automation Tool                       4.2.0
  DB2 Utilities Enhancement Tool for z/OS       2.2.0
  IBM DB2 Sort for z/OS                         2.1.0
  IBM DB2 High Performance Unload              4.2.0
----->----- End of components ----->-----

```

Figure 6. The Select the Components to Customize panel

2. Press Enter.

Results

If you are customizing components in the DB2 Utilities Solution Pack 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 components in the DB2 Utilities Solution Pack for the first time

Do not run the 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 component parameters.

Customizing new versions of components in the DB2 Utilities Solution Pack from a previous or current customization

Press Enter to run the Discover EXEC. The Discover Customized Product Information panel is displayed. Specify the required information for running the EXEC.

Discovering component information automatically

You can use the Discover EXEC to discover information from a previous or current customization of the component.

About this task

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

The component provides the Discover EXEC that you will run. Therefore, the information that can be discovered depends on the pack.

Parameter values that are discovered and parameter values that are specified manually are saved in the data store. If parameter values for the component 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 Discover EXEC on the pop-up panel after you specified the component 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 Pack Information panel is displayed, as shown in the following figure:

```

CCQPDC          Discover Customized Component Information          19:57:11

For the component you are customizing, the Discover EXEC retrieves component
information from an already customized component. Specify the required
information. To save your information and run the Discover EXEC, issue the RUN
command. To save your information and stay on this panel, issue the SAVE
command. To verify the syntax of your information without saving it, press
Enter. To save and exit, press End.

Commands: RUN  SAVE

Pack to Customize          Version . . : 1.1.0
Component metadata library : INZ.HINZ420.SINZDENU >LPAR . . : MVS1
Component name . . . . . : DB2 High Performanca >Version . . : 4.2.0

Discover EXEC for Extracting Information from an Already Customized Component
Discover EXEC library . . . INZ.HINZ420.SINZCLST
Discover EXEC name . . . . : INZTDSC
Discover output data set . . INZ.HINZ420.DISCOUT

Information for Discover EXEC
Previous data set of variables to be retrieved
                                INZ.HINZ410.SINZRSAV(INZRSAVE) >
Previous data set with customized file dsnames
                                INZ.HINZ420.SINZSAMP(INZTDSN) >
Verbose mode for procedure . . . . . OFF (OFF,ON)

```

Figure 7. The Discover Customized Component 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

The fully qualified data set name that contains the Discover EXEC.

Discover EXEC name

The name of the Discover EXEC.

Discover output data set

The fully qualified data set where output from the Discover EXEC is stored.

3. Either accept or change the default values in the **Information for Discover EXEC** fields. These fields are generated by the component to be customized. They show the information that is required to run the component Discover EXEC. These fields vary by component.
4. Issue the RUN command to run the component Discover EXEC. Alternatively, save your information without running the component Discover EXEC by issuing the SAVE command. If you issue the RUN command to run the component Discover EXEC, the parameter information is discovered for a component, and the Customizer Workplace panel is displayed.

Results

The discovered parameter values for a component 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 the component on new DB2 entries, create and associate the entries.
- If DB2 entries were discovered and you want to customize the component on only these entries, define the parameters.

Related tasks:

“Creating and associating DB2 entries”

You can create new DB2 entries and associate them with components in the DB2 Utilities Solution Pack.

“Defining parameters” on page 183

To customize a component, you must define component 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 components in the DB2 Utilities Solution Pack.

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 Component panel is displayed, as shown in the following figure:

```
CCQPDA          Associate DB2 Entry for Component          10:07:28
Command ==>                                         Scroll ==> CSR

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 a new DB2 entry

Pack to Customize                                Version . : 1.1.0
Component metadata library : INZ.HINZ420.SINZDE > LPAR . . : MVS1
Component name . . . . . : DB2 High Performan > Version . : 4.2.0

Line commands: A - Associate C - Copy

Cmd SSID GrpAtch
----- End of DB2 entries -----
```

Figure 8. The Associate DB2 Entry for Component 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. The Create DB2 Entries panel is displayed, as shown in the following figure:

```
CCQPDCR          Create a DB2 Entry
Command ==>

Specify a DB2 subsystem ID, a DB2 group attach name, or both for the
new DB2 entry. Press Enter to continue or End to cancel.

New DB2 Entry Information
DB2 subsystem ID . . . . .
DB2 group attach name . .
```

Figure 9. The Create a DB2 Entry 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, such as parameters that define a subsystem as primary, 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 Component 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 Component                10:07:28
Command ===>                Scroll ===> CSR

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 a new DB2 entry

Pack to Customize                Version . . : 1.1.0
Component metadata library : INZ.HINZ420.SINZDE > LPAR . . : MVS1
Component name . . . . . : DB2 High Performan > Version . . : 4.2.0

Line commands: A - Associate C - Copy

Cmd SSID GrpAttch
  DB0  --
----- End of DB2 entries -----

```

Figure 10. The Associate DB2 Entry for Component panel with a new DB2 entry in the master list

- 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 the component, 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 the component.

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” on page 275

Tools Customizer uses several unique terms that you should be familiar with

before you begin to use Tools Customizer.

Defining parameters

To customize a component, you must define component parameters, LPAR parameters, and DB2 parameters, if your customization requires DB2 entries.

About this task

You must define the component parameters first for the following reasons:

- If you ran the component Discover EXEC, you must review the values that were discovered.
- If you select optional tasks and steps on the Component Parameters panel that affect the DB2 entry on which you will customize a component, additional parameters might be displayed on the DB2 Parameters panel.
- If other steps must be completed in a specific sequence, customization notes on the Component Parameters panel will display the correct sequence.

Defining component parameters

Component parameters are specific to a component.

About this task

If you ran the component Discover EXEC, you must review the parameters that were discovered.

Procedure

1. Specify E next to the **Component parameters** field on the Customizer Workplace panel, and press Enter. The Component Parameters panel is displayed, as shown in the following figure. If other steps must be completed in a specific sequence before you define the component parameters, a note labeled **Important** will display the correct sequence on this panel.

```

CCQPPRD                               Component Parameters                               13:26:32
Command ==>                           Scroll ==> PAGE

Complete the following tasks to customize the components. The required tasks and
steps are preselected. Ensure that all parameters are specified for each
selected step within a task. Press End to save and exit.

Commands: SAVE - Save parameter values
Line Commands: / - Select

Pack to Customize                       Version . : 1.1.0
Component metadata library : INZ.ALIAS.SINZDENU > LPAR. . : MVS1
Component name . . . . . : DB2 High Performance Unloa> Version . : 4.2.0

Component customization library : INZ.PRODUCT.CUST.$MVS1$.INZ420
More:      +

Required parameters
HLQ of DB2 Tool installation . . . . . INZ.INZ410

/ 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

    Create JCL for later use
    High level qualifier of DB2 Path Checker

Create generic plan tables

/ Create new DSN3@ATH

    Create new DSN3@SGN

```

Figure 11. The Component Parameters 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.

- Press End to save your changes and exit, or issue the SAVE command to save your changes and stay on the Component 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 or Discovered, edit these parameters.

Related tasks:

“Defining LPAR parameters”

LPAR parameters are parameters on the local LPAR that are required to customize a component.

“Defining DB2 parameters” on page 186

DB2 parameters are parameters for a DB2 entry.

Defining LPAR parameters

LPAR parameters are parameters on the local LPAR that are required to customize a component.

Procedure

- 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

Pack to Customize                                Version . . : 1.1.0
Component metadata library : INZ.HINZ420.SINZDENU > LPAR. . . : MVS1
Component name . . . . . : DB2 High Performance Unlo > Version . . : 4.2.0

ISPF Libraries
Message library . . . . . ISPF.ISPPLIB                > Add...
Panel library . . . . . ISPF.ISPPLIB                > Add...
Skeleton library . . . . . ISPF.ISPSLIB              > Add...
ISPF table input library . . . . . ISPF.ISPTLIB       > Add...
ISPF user profile library . . . . . ISPF.PROFILE       > Add...
Command procedures library . . . . . SYS1.PROCLIB     > Add...

Language Environment Libraries
Language environment high-level qualifier
                                     abc                >

```

Figure 12. The LPAR Parameters panel

- 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 or Discovered, edit these parameters.

Related tasks:

“Defining component parameters” on page 183

Component parameters are specific to a component.

“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 component 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 181.

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:

```

CCQPDB2                DB2 Parameters                13:36:01
Enter values for all of the DB2 parameters. Press End to save and exit.

Commands: SAVE - Save parameter values

Pack to Customize                      Version . . : 1.1.0
Component metadata library : INZ.HINZ420.SINZDENU > LPAR. . . : MVS1
Component name . . . . . : DB2 High Performance UnTo > Version . . : 4.2.0

More:      +

DB2 subsystem ID . . . . . : DB01
Group attach name . . . . . :

General DB2 Information
Mode . . . . . NFM      (CM,CM8,CM9,NFM)
Level Number . . . . . 101 (810,910,101)

DB2 Libraries
Load Library . . . . . DSN.SDSNLOAD      >  Add...
Run Library . . . . . DSN.SDSNRLIB      >  Add...
Exit Library . . . . . DSN.SDSNEXIT     >  Add...
Sample library . . . . . DSN.SDSNSAMP   >  Add...
Macro library . . . . . DSN.SDSNMACS    >  Add...

DB2 Bufferpools
Name of the 4 KB bufferpool . . . . . BP0
Name of the 8 KB bufferpool . . . . . BP8K0

DB2 Utilities
Plan name for the DSNTIAD utility . . . .

DB2 Tools Objects
Database name . . . . . SYSTOOLS
Storage group name . . . . . SYSTOOLS      >

```

Figure 13. The DB2 Parameters 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 or Discovered, edit these parameters.

Related tasks:

“Defining component parameters” on page 183

Component parameters are specific to a component.

“Defining LPAR parameters” on page 185

LPAR parameters are parameters on the local LPAR that are required to customize a component.

Generating customization jobs

To generate customization jobs for a component and any associated DB2 entries, issue the GENERATEALL command, or select one or more DB2 entries on which to customize a component.

Procedure

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

- If you want to generate customization jobs at the component 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 component 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 Component Customization panel, which you can use to submit the jobs.

Submitting customization jobs

Submit the customization jobs to customize a component.

Before you begin

Ensure that the correct jobs are generated.

About this task

The following figure shows part of the Finish Component 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 Component 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.

Pack to Customize                       Version . : 1.1.0
Component metadata library : XYZ.HINZ420.SINZDENU >      LPAR . . : MVS1
Component name . . . . . : DB2 High Performance Unloa > Version . : 4.2.0

Line Commands: E - Edit  B - Browse

Component customization library . : XYZ.HINZ420.$ZPS1$.INZ420          >

Cmd Member  SSID GrpAttch Template Date      Description
-----
A0TVAR     --  --      INZTVAR  2012/06/06  Build the member of parameters
A1PARM     --  --      INZPARM  2012/06/06  Define DSNAME of PARMLIB to loa
A2HPUCL    --  --      INZHPUCL 2012/06/06  Generate CLIST to start HPU
A3LAUNC    --  --      INZLAUNC 2012/06/06  Integrate DB2 HPU to DB2 Tools
A4ADT00    --  --      INZADT00 2012/06/06  Integrate DB2 HPU into DB2 Admi
A5BIAAAA   --  DG93    INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAD   --  DB0A    INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAG   DZ7M  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAH   DZ8A  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAI   DZ8C  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAJ   DZ8D  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAM   DZ8E  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAN   DZ9E  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAO   DZAA  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAR   DZAJ  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAS   DG93  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAT   DZ8B  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAV   DB0A  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BIAAAW   DZAE  --      INZBIND  2012/03/19  BIND and GRANT for HPU
A5BINDAA   --  DG93    INZBIND  2012/06/06  BIND and GRANT for HPU
A5BINDAD   --  DB0A    INZBIND  2012/06/06  BIND and GRANT for HPU
A5BINDAG   DZ7M  --      INZBIND  2012/06/06  BIND and GRANT for HPU
----- End of customized jobs -----

```

Figure 14. The Finish Component Customization panel

The member-naming conventions depend on whether the customization jobs are for DB2 entries, and LPAR, or the component.

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. The component to be customized 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 component

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. The component to be customized 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: A0BNDGR, 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 against a customization job or the customization library, and press Enter. An ISPF browsing 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

The component is customized, and the Customizer Workplace panel is displayed. The status is Customized for the DB2 entries on which the component 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 a component.

Browsing parameters

You can browse the 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 **Component 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” on page 275

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 Component 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 Component panel, and the customization jobs are deleted.

Related concepts:

“Tools Customizer terminology” on page 275

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 components 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 Component 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 components, 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 components in the table on the Delete Associated DB2 Entry panel, any associations and all customization jobs for the components 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 Component 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 a component for another DB2 entry later.

Procedure

On the Customizer Workplace panel, issue the JOBLIST command. The Finish Component 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 a component 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 Component 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 Component 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 the component 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.

1. Associate all the DB2 entries in this one instance of Tools Customizer. The LPARs on which the DB2 subsystems reside do not matter.
2. Generate the customization jobs for each DB2 entry.
3. 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.

Chapter 6. Post-customization tasks

After you customize the components of the solution pack, some components have post-customization tasks that you might need to perform depending on the environment that you want to set up.

Post-customization tasks: DB2 Automation Tool

After you customize DB2 Automation Tool, you might want to complete the task in this section.

Creating multiple configurations of DB2 Automation Tool

DB2 Automation Tool V4.2 allows multiple configurations of itself to be installed on the same subsystem. Each configuration has its own CLIST, data repository, and set of plans and packages. The configurations share the same set of DB2 Automation Tool execution libraries; each configuration is identifiable by a three-character user indicator in the CLIST. This allows an installation to, for example, have a test version of DB2 Automation Tool in the same subsystem as a production version.

About this task

If multiple configurations are desired, you must completely customize one configuration using IBM Tools Customizer and then repeat the customization process using Tools Customizer for each subsequent configuration. Each configuration must have a unique HAA configuration value as specified on the Tools Customizer Product Parameters panel; this value becomes the user indicator parameter (USERIND) in the product CLISTs.

The following procedure illustrates the process for two configurations.

Procedure

1. Make a separate copy of the product metadata data set (*hilevel.SHAADENU*) for each configuration that will be created. For product configuration number 1, create a copy of the product metadata data set; for example, it could be named `DB2TOOL.PRD.CFG01.SHAADENU`. For product configuration number 2, create a copy of the product metadata data set called (for example) `DB2TOOL.PRD.CFG02.SHAADENU`.
2. Customize product configuration number 1:
 - a. Start Tools Customizer.
 - b. On the Tools Customizer main menu, select Option 0 (Settings) and enter the desired customization library qualifier. This value should be different for each configuration. For example, it could be called `DB2TOOL.PRD.CFG01`, following the naming convention in this example. Save and exit.
 - c. On the Tools Customizer main menu, select option 1.
 - d. On the Specify the Product to Customize panel, enter `DB2TOOL.PRD.CFG01.SHAADENU` in the Product metadata library field.

- e. If you want to discover information from a previous installation of DB2 Automation Tool, run Discover. The HAA configuration value will be input as a Discover parameter. If you do not run Discover, the HAA configuration value defaults to "HAA".
 - f. On the Product Parameters panel, take note of the HAA configuration value for this configuration. Subsequent configurations will require a different value.
 - g. Generate customized jobs against each SSID. Tools Customizer will generate the jobs in DB2TOOL.PRD.CFG01.\$lpar\$.HAA420.
3. Customize product configuration number 2:
 - a. Navigate to the Tools Customizer main menu (CCQPHME).
 - b. On the Tools Customizer main menu, select option 0 (Settings) and enter a different customization library qualifier; for example, DB2TOOL.PRD.CFG02 (continuing the example naming convention). Save and exit.
 - c. On the Tools Customizer main menu, select option 1.
 - d. On Specify the Product to Customize panel, enter the data set name for the second configuration's copy of the metadata, DB2TOOL.PRD.CFG02.SHAADENU, in the Product metadata library field.
 - e. If you want to discover information from a previous installation of DB2 Automation Tool, run Discover. The HAA configuration value will be input as a Discover parameter. If you do not run Discover, the HAA configuration value defaults to "HAA".
 - f. On the Product Parameters panel, ensure that the HAA configuration value is different from the configuration value specified in product configuration number 1.
 - g. Generate customized jobs against each SSID. Tools Customizer will generate the jobs in DB2TOOL.PRD.CFG02.\$lpar\$.HAA420.

What to do next

You can repeat this procedure for as many configurations as you required. Each configuration must have its own set of metadata files and a unique HAA configuration value.

Post-customization tasks: DB2 High Performance Unload

After you customize DB2 High Performance Unload, you might want to complete the tasks in this section.

Optional: Integrating DB2 HPU into DB2 Tools Launchpad

Optionally, you can integrate DB2 HPU into DB2 Tools Launchpad. Tools Customizer will create the necessary JCL, but you must manually complete some steps after you submit the customization job.

Before you begin

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

- The DB2 Tools Launchpad is installed.
- Tools Customizer generated the job from the INZLAUNC template, and you submitted the job.

About this task

The DB2 Tools Launchpad is a centralized panel from which you can launch integrated DB2 Tools. After you integrate DB2 HPU, you can launch DB2 HPU from the DB2 Tools Launchpad.

Procedure

1. Run the CLIST INZADBI in SINZCLST. The CLIST uses the high-level qualifier that you specify for the DB2 Admin data sets and the name of the library that contains the ADBDMTI EXEC. The DB2 Tools Table – ADD An Entry panel is displayed, as shown in the following figure:

```
----- DB2 Tools Table - ADD An Entry ----- 08:57
Command ==>

Tool Name : HPU High Performance Unload
Code      : HPU      (User-defined code, for shortcut tool identifier)
Prog No.  : 5655-AA1 (IBM program product number or equivalent)
Release   : 420      (Product release number)
Group     : 2        (Tool category, as follows:
                    1 - Administration Tools
                    2 - Application Management Tools
                    3 - Performance Management Tools
                    4 - Recovery and Replication Management)
Installed : Y        (Y - yes, N - no)
Command   : SELECT CMD(INZHPU D LP )
```

Figure 15. Adding an entry to the DB2 Tools Launchpad panel

2. Press Enter to confirm the new DB2 HPU command.

Results

When INZADBI completes successfully, a new line, HPU, is added to the DB2 Tools Launchpad.

Optional: Integrating DB2 HPU into DB2 Administration Tool

Optionally, you can integrate DB2 HPU into DB2 Administration Tool (DB2 Admin). Tools Customizer will create the necessary JCL, but you must manually complete some steps after you submit the customization job.

Before you begin

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

- DB2 Admin is installed.
- Tools Customizer generated the job from the INZADTOO template, and you submitted the job.

About this task

DB2 Admin helps you manage DB2 environments efficiently and effectively. After you have integrated DB2 HPU, you can run DB2 HPU unload operations by using DB2 Admin.

Procedure

1. Follow the instructions that are described in INZDB21X in the SINZCLIST library. You can customize these instructions by changing the name of the library which contains DB2 Admin commands tables and the name of the library which contains the ADBDMTI EXEC.

Requirement: Before you can use the INZDB21X member, you must generate it with Tools Customizer whenever you change the library names.

2. Run the ADB21S and ADB21T procedures that you have modified. These procedures are used to re-create the DB2 Admin Tool command tables.

Results

When ADB21S and ADB21T complete successfully, you can start DB2 HPU interactively by using the DB2 Admin.

Post-customization tasks: DB2 Sort

After you customize DB2 Sort, you must create access to the DB2 Sort libraries. You can then use the Installation Verification Program to test your settings before moving DB2 Sort to a production environment.

Accessing DB2 Sort libraries

After activating and customizing DB2 Sort, you must create access to the following DB2 Sort libraries: SCNKLINK and SCNKLPA. If access to these two libraries was established before you began this customization, then ignore the procedure that follows and refresh the library by using the following MVS system command:
MODIFY LLA,REFRESH

Procedure

Decide how you want to access the DB2 Sort libraries, SCNKLINK and SCNKLPA. To make these decisions, consider the following information about the attributes of the DB2 Sort libraries:

- The SCNKLINK and SCNKLPA libraries can be in the LNKLST.
- The SCNKLPA library can be in the LPA, but it is not required to be in the LPA.
- If the SCNKLPA library is not in the LPA, it must be APF-authorized.
- The SCNKLINK library must be APF-authorized.

What to do next

You can submit the Installation Verification Program job that you generated with Tools Customizer to test your product configuration before moving to a production environment. After the IVP job runs to completion, refer to the procedure to generate an IVP report in “Generating an IVP report” on page 202.

After the product has been successfully installed and configured and the load library is APF-authorized, you may begin using DB2 Sort. Refer to the sample job in *DB2 Sort for z/OS User's Guide*.

Installation Verification Program: Overview

Installation Verification Program (IVP) is a program you can run in a test environment to verify your DB2 Sort installation prior to moving to a production environment.

DB2 Sort IVP is a self-contained evaluation and testing methodology you can use to determine that DB2 Sort is installed correctly and performing properly.

When you run IVP, it performs the following tasks:

- Creates a storage group and database for the IVP objects and then drops them at the end of the job
- Creates tables with different sizes and row lengths, and multiple indexes and index lengths
- Ensures data is not in clustering sequence
- Executes the LOAD, REBUILD INDEX, and REORG utilities with and without DB2 Sort

When you are customizing DB2 Sort with , you can choose the cases for IVP. There are four table sizes: 8 billion bytes, 10 billion bytes, 25 billion bytes and 40 billion bytes. Base your decision on the amount of disk space and time you want the job to use.

To ensure that the IVP results are meaningful, you must run IVP in an appropriate test environment that represents your production environment as much as possible.

Use the following general requirements when setting up your test environment:

- **CPU:** 2 general purpose processors minimum, preferably 4 or more; 1 or more zIIP processors
- **DASD:** Minimum of 6 volumes (PAV-capable)
- **Channels:** Minimum of 6 (quiet)
- **Memory:** Minimum of 10 GB
- **LPAR:** Stand-alone or quiet system
- **Traces:** Turn off all DB2 Utility traces that impact performance times

Using Installation Verification Program

Use the Installation Verification Program (IVP) in a test environment to verify that your DB2 Sort installation and any customizations you made are functioning correctly before moving to a production environment.

Before you begin

Make sure that DB2 Sort has been installed and you have an appropriate test environment. For details about setting up the test environment, see "Installation Verification Program: Overview."

Since information about the IVP job will be recorded in SMF records and performance trace data, you must ensure that your system is set up to generate the performance trace data, using the following requirement:

- Issue the following DB2 command to collect performance trace data:
-START TRACE(P) DEST(SMF) CLASS(30) IFCID(25) TDATA(COR,CPU)

Note: Issue a DISPLAY TRACE(*) first. If another active trace with CLASS(30) has started, you will need to issue a START TRACE(P) CLASS(31) or (32). If all three classes are in use, add IFCID(25) using a MODIFY TRACE command. Alternatively, you can issue the STOP TRACE(P) command and reissue the START TRACE(P), combining the different IFCIDs into the single START TRACE(P) command.

If you have generated the IVP job with Tools Customizer, submit the job after you have set up your environment to generate the performance trace data and after your DB2 Sort libraries have been made accessible. For instructions on making your libraries accessible, see “Accessing DB2 Sort libraries” on page 200.

Generating an IVP report

After you run one of the IVP jobs, you can generate a report to display the results.

About this task

The report program uses SMF type 30 subtype 4 and DB2 SMF 102 (IFCID 25) records as input.

To display the spreadsheet located in the SCNKBENU library, a PC/workstation with a program that can edit spreadsheets from Microsoft Office Excel 2003 (XLS format) or later is required.

Procedure

1. Extract the SMF type 30 subtype 4 records and the SMF 102 records from your SMF data sets and put them into a single data set. You can use the IFASMFDP program to create the data set.
2. If DB2 SMF records are in compressed format, you must decompress the SMF records prior to using the reporter job with the SMF decompression program DSNTSMFD.

Note: For more information about the SMF decompression program DSNTSMFD, refer to:

- *DB2 11 for z/OS Utility Guide and Reference (SC19-4067)*
- *DB2 10 for z/OS Utility Guide and Reference (SC19-2984)*

For DB2 10 the SMF decompression program is provided with PTF UK64597.

3. Open the CNKIVRPT member which is located in the SCNKXSAMP library.
4. Make the editing changes according to the instructions that are included in the JCL and submit the job. This job creates an output file named *userid.CNKDATA.TXT*.
5. Transfer your *userid.CNKDATA.TXT* file to a PC/workstation in ASCII mode, renaming it to *c:\temp\cnkdata.txt*.

Note: If your transfer facility offers a carriage return/ line feed option (CRLF), it should be selected.

6. Transfer the member named CNKIVXLS which is located in the SCNKBENU library to a PC/workstation in binary mode, renaming it to *cnkivxls.xls*.
7. On the PC/workstation, open the *cnkivxls.xls* spreadsheet. Enable content in response to the security warning. Select the Worksheet tab and follow the directions within the worksheet.

Results

The IVP report program produces the following two reports:

- Summary
- Details

After the worksheet has been filled in, choose the Summary and Details tabs to view the reports.

What to do next

After the product has been successfully installed and configured and the load library is APF-authorized, you may begin running DB2 utilities and tools with DB2 Sort.

Post-customization tasks: DB2 Utilities Enhancement Tool

After you customize DB2 Utilities Enhancement Tool, complete the post-customization tasks. Some of these tasks are required, and some are optional.

Table 18 lists the tasks.

Table 18. Required and optional post-customization tasks

Task	Required or optional
“Setting up the initialization options member (optional)”	Optional
“APF-authorizing the load library” on page 205	Required
“Copying the started task PROC” on page 205	Required
“Customizing DSNUTILB intercept parameters (optional)” on page 206	Optional (default values are available)
“Copying the DSNUTILF module” on page 210	Required if using the DSNUTILB intercept
“Creating a security exit (optional)” on page 210	Optional
“Creating a pre- or post-cancel exit (optional)” on page 211	Optional
“Starting the started task” on page 212	Required

Setting up the initialization options member (optional)

The initialization options member, called *abpidOPTS*, is used by the DB2 UET started task upon initialization.

About this task

Tools Customizer already customized the options member for the ABPID that you specified in the Tools Customizer panels, if you selected the task **Create the Options Module**. If you plan to run multiple started tasks to monitor different DB2 subsystems, you are required to create separate initialization options members for each one. This customization step is required if you will use multiple started tasks.

This member resides in the *hlq.mlq.SABPSAMP* data set and is identified in the started task's ABPOPTS DD statement of the PROC (*abpidPROC*). The following example of the *abpidOPTS* member specifies the following options in XML:

```
<?XML VERSION="1.0" ENCODING="UTF-8"?>
<!DOCTYPE OPTIONS SYSTEM "DD:DTD(ABPDTDOP)">
<!-- ***** -->
<OPTIONS
  ABPBMMAIN_CANCEL_MEMBER="abpidBCAN"
  ABPBMMAIN_GLOBAL_MEMBER="abpidBGLB"
  ABPID="abpid"
  AUDIT_ACTIVE="YES"
  AUDIT_MAX_AGE="0"
  CANCEL_ESCALATION_ACTIVE="NO"
  DB2_CONNECT_TO_ALL_SUBSYSTEMS="YES"
  DB2_CONNECTION_IDLE_TIMEOUT="300"
  DB2_PLAN_NAME="plan_name"
  DB2_SSID="subsystem_identifier"
  DB2_TASK_COUNT="2"
  DB2_TASK_IDLE_TIMEOUT="900"
  DYNAMIC_SYSOUT_CLASS="*"
  LOGGING_ACTIVE="YES"
  LOGGING_MAX_AGE="0"
  WORK_DATABASE_BUFFERPOOL="bufferpool_name"
  WORK_DATABASE_NAME="database_name"
  WORK_DATABASE_STOGROUP="stogroup_name"
  OVERRIDE_MAPPING_TABLE="value"
  POST_CANCEL_EXIT="NONE"
  PRE_CANCEL_EXIT="NONE"
  SECURITY_EXIT="NONE"
  SVC_NUMBER="svc_number"
  TRACE_ACTIVE="YES"
  TRACE_SIZE="1"
  WORKFILE_DATACLAS="data_class"
  WORKFILE_MGMTCLAS="management_class"
  WORKFILE_STORCLAS="storage_class"
  WORKFILE_UNIT="unit_name"
  WORKLIST_ERROR_MAX_AGE="0"
  WTO_ROUTCDE="11"
/>
```

Tools Customizer set values for many of these options based on the values that you specified in the customization dialogs. You can edit these values, if necessary. Make sure that you edit the *abpidOPTS* member that was generated for your started task configuration rather than the original initialization options member (*ABPOPTS*), which also resides in the *hlq.mlq.SABPSAMP* library.

When editing the initialization options member, consider whether an option is required or optional. Only the following options are required and must be present in the started task initialization options member:

- ABPID
- DB2_PLAN_NAME
- DB2_SSID
- WORK_DATABASE_NAME
- SVC_NUMBER

All other options are optional because they have default values. If you omit an “optional” option from the initialization options member, DB2 UET uses the default value for that option.

Tip: You will be able to temporarily override the values for the following options from the Control System panel in the ISPF interface: `AUDIT_ACTIVE`, `LOGGING_ACTIVE`, `TRACE_ACTIVE`, `WORKFILE_DATACLAS`, `WORKFILE_MGMTCLAS`, `WORKFILE_STORCLAS`, `WORKFILE_UNIT`, and `WTO_ROUTCDE`. Any changes that you make from this panel are retained only until you restart the started task. The changes are not saved to the initialization options member.

What to do next

If you would like to prevent users from being able to cancel threads using DB2 UET, you may create a security exit that is invoked each time a thread cancel command is issued. See task “Creating a security exit (optional)” on page 210.

APF-authorizing the load library

Ensure that the product load library is APF-authorized so that it will be available for use. This customization step is required.

About this task

DB2 UET requires that the product load library (*hlq.mlq.SABPLOAD*) be authorized by the z/OS Authorized Program Facility (APF). To include this load library in your system APF-authorized list, issue the following z/OS operator command:

```
SETPROG APF,ADD,DSNAME=hlq.mlq.SABPLOAD,VOLUME=volser
```

Where

- *hlq* is the high-level qualifier.
- *mlq* is the mid-level qualifier that you specified when you ran Tools Customizer.
- *volser* is the volume serial number of the DASD device where the SABPLOAD library resides.

Copying the started task PROC

Copy the DB2 UET started task PROC to your system PROCLIB to make the started task address space available to the user interfaces for the product. This customization step is required.

About this task

Run the A6MODCR job located in the data set that is specified in the **Product Customization Library** field on the Tools Customizer Finish Product Customization panel. This job will create the *abpidPROC* member in your *hlq.mlq.SABPSAMP* library. This job was only created if you selected the Tools Customizer subtask **Create BCAN, BGLB, PROC, PLCY Members**.

Procedure

1. Copy the *abpid* PROC member that gets created in the *hlq.mlq.SABPSAMP* to a member in your system PROCLIB.
2. In the EXEC statement, ensure that `REGION=0M` is specified to avoid storage problems. Also, ensure that `TIME=1440` is specified to allow DB2 UET to run for an unlimited amount of time.
3. Ensure that the STEPLIB and SVCLIB data sets are APF-authorized.
4. If you plan to monitor threads across multiple DB2 subsystems that have different DB2 versions, ensure that the STEPLIB concatenation specifies the

lowest (earliest) of these DB2 versions as the DSNLOAD library. Otherwise, connection problems might occur when you attempt to manage threads on DB2 subsystems other than the primary subsystem that contains the audit and logging tables.

What to do next

Copy the DSNUTILF module from the *hlq.mlq.SABPLOAD* product library to a STEPLIB or JOBLIB DD that is used in your DB2 utility jobs. See the task “Copying the DSNUTILF module” on page 210.

Customizing DSNUTILB intercept parameters (optional)

If you plan to use the DSNUTILB intercept, you can customize the cancel parameters and global parameters that the DSNUTILB intercept will use for all thread blocking and cancelation activities. These parameters are specified in the SABPSAMP members *abpidBCAN* and *abpidBGLB*. This customization step is optional.

About this task

Tools Customizer created a job called A6MODCR that in turn creates the *abpidBCAN* and *abpidBGLB* members (where *abpid* is the value that you specified in the Tools Customizer field **DB2 UET Configuration Name**) in the *hlq.mlq.SABPSAMP* library for your started task configuration. The DB2 UET provides default values for all of the parameters in these members. Normally, you customize these members only if you want to override the default behavior.

If you omit a parameter from one of these members, DB2 UET will use the default value for that parameter. If you specify an invalid value for a parameter, any thread blocking and cancelation processing that the DSNUTILB intercept attempts to perform will fail.

Tip: You can edit these members while the started task is running and performing interception to correct any syntax errors or change parameter settings. However, for your revisions to take effect for an intercepted DB2 utility, you will need to re-run the utility.

Procedure

1. Edit the values for any of the following parameters in the *abpidBCAN* member, or accept the default values. These parameters pertain to each cancel request. You cannot add comments to this member or specify additional parameters.

CANCEL_TYPE BACKOUT|NOBACKOUT|NO_UR_CHECKING

One of the following types of cancelation processing:

Important: The NOBACKOUT and NO_UR_CHECKING cancel types are not supported for threads on a DB2 subsystem other than the local subsystem to which a connection is established based on the <DB2SYSTEM> element value in the intercept policy. In a DB2 data sharing environment, these options do not apply to other subsystems in the data sharing group.

- **BACKOUT:** Specify this option to have DB2 back out (roll back) any uncommitted work that exists in the current unit of recovery when a thread is canceled. This backout processing ensures that the data

integrity of the updated DB2 objects is maintained. Any updates that occurred after the last commit operation are backed out. This option is the default option.

- **NOBACKOUT: Do not specify this option if possible.** This option causes DB2 UET to check for any outstanding units of recovery just prior to canceling threads but prevents DB2 from performing backout processing. If DB2 UET finds an outstanding unit of recovery for a thread, the thread is *not* canceled and the return code RC12 is issued. Depending on how you set the ON_FAILURE parameter in the *abpidBGLB* member, the cancel request either terminates or continues to the next thread. By default, the request will continue and cancel the other threads. If DB2 UET does not find an outstanding unit of recovery, the thread is canceled. In this situation, data integrity problems can occur if an outstanding unit of recovery is created for a thread after DB2 UET has determined that no outstanding units of recovery exist for the thread and before the thread is canceled. Therefore, use this option with caution.
- **NO_UR_CHECKING: Do not specify this option unless necessary.** This option causes threads to be canceled without backout processing or any checking for outstanding units of recovery prior to cancellation. All threads that match your thread-filtering criteria are canceled, even if outstanding units of recovery exist for them. This situation is likely to result in data corruption in your database. (This option is equivalent to using the DB2 NOBACKOUT option on the -CANCEL THREAD command. For more information about the DB2 NOBACKOUT option, see the *IBM DB2 Version 10 for z/OS Command Reference*.)

ESCALATE YES|NO

Whether DB2 UET issues the z/OS Cancel command to cancel a thread when the DB2 -CANCEL THREAD command fails to do so. If you set this parameter to YES, DB2 UET can issue the z/OS Cancel command to terminate the TSO user, batch job, or started task from which the thread originated. This escalated cancellation processing is supported only for the following connection types: CAF, IMSDLIBT, RRSFAF, and TSO. Also, in a DB2 data sharing environment, escalated cancellation is supported only for threads on a local subsystem that you specify in a DB2SYSTEM element of the intercept policy. If you set this parameter to NO, DB2 UET issues the DB2 -CANCEL THREAD command but does not escalate cancellation processing to the z/OS Cancel command when the DB2 -CANCEL THREAD command fails to terminate a thread; any threads that were not canceled remain active. The default value is NO.

Important: DB2 UET will not perform an escalated cancellation, even if you set the ESCALATE parameter to YES, if a thread has a connection type that is not supported for escalated cancellation processing or if the CANCEL_ESCALATION_ACTIVE option in the started task initialization options member is set to NO.

CHECK_THDTERM_RETRY_COUNT *integer*

After issuing the DB2 -CANCEL THREAD command, DB2 UET checks the status of the canceled threads to determine whether DB2 has actually terminated them. If some threads are not yet terminated, DB2 UET continues to check the thread status until all of the canceled threads are actually terminated or until the maximum retry count that you set with this parameter is reached. If the maximum retry count is reached and a thread has still not been terminated, DB2 UET issues an error message with the appropriate return code. If you also set the ESCALATE parameter to YES,

DB2 UET will then issue the z/OS Cancel command to terminate the thread. Valid values are from 1 through 32,767. The default value is 20.

Tip: Set the CHECK_THDTERM_RETRY_INTERVAL parameter to define how frequently thread-status checking occurs.

CHECK_THDTERM_RETRY_INTERVAL *seconds*

How often (in seconds) DB2 UET checks the status of the threads that have been canceled to determine whether they are actually terminated in DB2. Valid values are from 1 through 32,767. The default value is 3.

Tip: Set the CHECK_THDTERM_RETRY_COUNT parameter to define the maximum number of times that DB2 UET will attempt thread-status checking.

2. Edit the values for any of the following parameters in the *abpidBGBL* member, or accept the default values. These global control parameters pertain to all thread blocking and cancellation activities that the DSNUTILB intercept will perform. You cannot add other parameters to this member.

ESCAPE (*character*)

The escape character that you want to use to delimit a wildcard character in a intercept policy rule for selecting threads for cancellation when that character is used as an actual part of the data value and not as a wildcard. If you do not specify the escape character immediately before such a character, DB2 treats the character as a wildcard when performing pattern matching during thread filtering. (For more information about escape expressions and pattern matching, see the *IBM DB2 Version 10 for z/OS SQL Reference*.) The default escape character is the plus sign (+). This parameter is ignored in environments that use a double-byte character set (DBCS).

EXEC_TYPE (CHECKPARMS|EXECUTE|SIMULATE)

One of the following execution modes for thread-cancellation processing:

- CHECKPARMS: Specify this option to have DB2 UET validate the syntax of all of the input parameters that are specified without actually blocking and canceling any threads. The DB2 utility will still run although no threads are canceled.
- EXECUTE: Specify this option to actually perform thread blocking and cancellation. All threads that are selected based on the EXCLUDE and INCLUDE rules that you define in the DSNUTILB intercept policy will be blocked and canceled. This option is the default option.
- SIMULATE: Specify this option to perform a trial run of thread-cancellation processing for a utility. DB2 UET simulates the cancellation of threads that are selected based on the EXCLUDE and INCLUDE rules that you define in the DSNUTILB intercept policy. The processing flow, messaging, and reporting are the same as if you specified the EXECUTE mode, but no threads are actually blocked and canceled. You can use this option to pre-check which threads will be canceled and to troubleshoot potential errors. The DB2 utility will still run although no threads canceled.

ON_FAILURE (CONTINUE|TERMINATE [RESET_OBJECT_STATUS YES|NO])

Whether thread-cancellation processing continues or terminates when an error occurs. By default, when an error occurs, the thread-cancellation processing continues. However, if you specify TERMINATE for this parameter, the job will terminate and no additional cancel processing will occur.

You can add a subparameter on the TERMINATE option to specify whether DB2 UET reinstates the original statuses of the DB2 objects when cancelation processing fails or one or more of the canceled threads do not actually terminate. During thread blocking, DB2 UET changes the statuses of the DB2 objects to either RO (read only) or UT (only the utility for which threads are being canceled has access). You can control whether DB2 UET leaves the object statuses in this state after a cancel failure by adding the RESET_OBJECT_STATUS subparameter, as follows:

```
ON_FAILURE (TERMINATE RESET_OBJECT_STATUS YES|NO)
```

If you accept the default value of YES for this subparameter, DB2 UET reinstates the original statuses of the objects. If you specify NO, DB2 UET retains the object statuses that were in effect during thread blocking. This subparameter applies only to thread-cancel operations that include thread blocking.

REPORT_TYPE (SUMMARY|DETAIL)

The level of reporting that DB2 UET provides for actual or simulated runs of thread-cancelation processing. DB2 UET can generate up to five reports for each cancel request for a utility. Valid options are:

- **SUMMARY:** Specify this option to print the Threads Canceled report and the Threads Canceled Unit of Recovery report for each cancel request for which active threads are selected for cancelation. This option is the default option.
- **DETAIL:** Specify this option to print the following five reports for each cancel request: the Threads Canceled report, the Threads Canceled Unit of Recovery report, the All Active Threads report, the All Active Threads Unit of Recovery report, and the All Active Threads Objects Referenced report. If no threads are selected for cancelation processing for a cancel request, DB2 UET generates only the three reports on all active threads.

THREAD_QUIESCE_TIME (seconds)

For thread-blocking actions, you can specify the number of seconds that DB2 UET waits between initiating thread blocking and canceling the active threads that are identified by EXCLUDE and INCLUDE rules that you define in the DSNUTILB intercept policy. This interval is intended to allow the applications that are using the existing threads to complete units of work or quiesce prior to thread cancelation. If you do not specify this parameter, the default value of 0 is used, which causes DB2 UET to cancel existing threads immediately after initiating thread blocking.

3. If you copied the *abpidBCAN* and *abpidBGLB* members under other names before editing them, make sure that you specify the new member names in the started task initialization options member (*abpidOPTS*). The *ABPBMAIN_CANCEL_MEMBER* and *ABPBMAIN_GLOBAL_MEMBER* options in the *abpidOPTS* member must correctly reference the *abpidBCAN* and *abpidBGLB* members for the started task to use these parameters for thread-cancelation processing for intercepted DB2 utilities.

What to do next

Optionally, make any necessary edits to the options module created by Tools Customizer. See task “Setting up the initialization options member (optional)” on page 203.

Copying the DSNUTILF module

If you plan to use the DSNUTILB intercept, copy the DB2 UET DSNUTILF load module to a data set in the STEPLIB or JOBLIB concatenation for your DSNUTILB utility jobs.

About this task

This customization step is required for the DB2 UET started task to perform DSNUTILB interception services.

Procedure

1. Copy the DSNUTILF module into one of the APF-authorized libraries in the STEPLIB/JOBLIB concatenation.
2. Optionally, you can copy the DSNUTILF module to a data set other than RUNLIB.LOAD, SDSNEXIT or SDSNLOAD. If you choose this method, ensure that the data set is APF-authorized and is contained in the JOBLIB of your utility job or in the STEPLIB of the DSNUTILB jobsteps.
3. Depending on how you invoke DSNUTILB, you might also need to specify the target data set at the following locations:
 - If you plan to invoke DSNUTILB by using the DSNUTILS or DSNUTILU stored procedure, concatenate the DB2 UET LOAD library to the STEPLIB/JOBLIB concatenation.
 - If you plan to invoke DSNUTILB from a TSO or ISPF session, specify the target data set name in the STEPLIB or ISPLLIB concatenation for your TSO or ISPF sessions.

Results

Note that with the DSNUTILF module in the load library concatenation, the DSNUTILB program will still be able to operate normally even if the DB2 UET STC becomes unavailable or if the DB2 UET intercept policy prevents DSNUTILB interception.

What to do next

Optionally customize the DSNUTILB interception parameters. See task “Customizing DSNUTILB intercept parameters (optional)” on page 206.

Creating a security exit (optional)

If you want to control user access to product commands for intercept processing and to certain ISPF panels, you can create a security user exit. This customization step is optional.

Before you begin

Before you begin, coordinate with your security administrator to determine the appropriate security scheme for your DB2 UET users. Also, review the reference information on the security exit. See the *DB2 Utilities Enhancement Tool User's Guide*.

About this task

The security exit must be written in assembler language. A sample exit (ABPXSE00) and corresponding DSECT (ABPAPISE) are provided in the *hlq.mlq.SABPSAMP* library, where *hlq* is the high-level qualifier and *mlq* is the

mid-level qualifier that you specified when you ran Tools Customizer.

Procedure

1. Copy the sample exit ABPXSE00 and the DSECT ABPAPISE to the library where you want to store your security exit.

Important: Do *not* change the copy of the DSECT ABPAPISE. Maintain the DSECT in its original state to ensure that it properly defines the layout of the memory for the API control block.

2. Edit the copy of the exit ABPXSE00, as needed, to implement your security scheme. The sample exit includes comments to guide you through this process. Depending on your security requirements, you might need to edit the exit significantly or make only minor changes.
3. Assemble and link the exit code to create an executable exit routine.
4. Ensure that the load module for the exit is available to the started task at runtime either by including it in the APF-authorized DB2 UET load library or by putting it in its own APF-authorized load library and then including that library in the STEPLIB concatenation.
5. Specify the exit name in the SECURITY_EXIT option of the started task initialization options member.

What to do next

Optionally, create pre- or post-cancel user exits to perform actions either before or after a thread is canceled. See the *DB2 Utilities Enhancement Tool User's Guide*.

Creating a pre- or post-cancel exit (optional)

If you need to perform additional processing prior to or after thread cancelations, you can create a pre-cancel exit, post-cancel exit, or both. This customization step is optional.

Before you begin

Before you begin, work with your production control coordinator, DB2 DBA, or other appropriate personnel at your site to determine your pre-cancel or post-cancel processing requirements. For example, you might need a pre-cancel exit to prevent the cancelation of threads that are running under a specific DB2 plan or a post-cancel exit to notify users when cancelations complete. Also, review the reference information on the pre-cancel and post-cancel exits. For more information about pre-cancel and post-cancel exits, see the *DB2 Utilities Enhancement Tool User's Guide*.

About this task

These exits must be written in assembler language. Sample exits and corresponding DSECTs are provided in the *hlq.mlq.SABPSAMP* library, where *hlq* is the high-level qualifier and *mlq* is the mid-level qualifier that you specified when you ran Tools Customizer.

Procedure

1. Copy the sample exit and the DSECT for the pre- or post-cancel exit to the library where you want to store the exit.
 - For a pre-cancel exit, copy the sample exit ABPXPR00 and the DSECT ABPAPIPR.

- For a post-cancel exit, copy the sample exit ABPXPO00 and the DSECT ABPAPIPO.

Important: Do *not* change the copy of the DSECT. Maintain the DSECT in its original state to ensure that it properly defines the layout of the memory for the API control block.

2. Edit the copy of the sample exit, as needed, to implement your pre- or post-cancel processing requirements. The sample exit includes comments to guide you in this process. Depending on your requirements, you might need to significantly edit the exit or make only minor changes.
3. Assemble and link the exit code to create an executable exit routine.
4. Ensure that the load module for the exit is available to the started task at runtime either by including it in the APF-authorized DB2 UET load library or by putting it in its own APF-authorized load library and then including that library in the STEPLIB concatenation.
5. Specify the exit name in the PRE_CANCEL_EXIT option or POST_CANCEL_EXIT option of the started task initialization options member.

What to do next

Start the started task to begin using the product. See “Starting the started task.”

Starting the started task

Start the DB2 UET started task so that you can begin using the product interfaces to block and cancel threads or implement the DB2 utility enhancements. This customization step is required.

About this task

To start the started task, issue the following operator command from the z/OS console:

```
S abpstc
```

Where *abpstc* is the member name of the DB2 UET PROC in the system PROCLIB.

If you issue the command from SDSF instead, begin with a forward slash as follows:

```
/S abpstc
```

The Tools Customizer generated the started task name based on the value that you specified in the Tools Customizer field **DB2 UET Configuration Name**, and then inserted that name in the started task PROC. If you changed the started task name in the PROC, make sure that you use that new name in Start command.

What to do next

Start the ISPF interface to view all active threads on the system, or to cancel specific threads. See the task “Starting the ISPF interface.”

Starting the ISPF interface

Start the ISPF interface so that you can use it to view or cancel DB2 threads, view the product status, or override selected started task initialization options.

About this task

If you copied the REXX EXEC for running the interface to another data set or data set member, make sure that you specify the name of that data set or member in the command statement.

The variables in the command statement are as follows:

- *hlq* is the is the high-level qualifier that you specified during customization.
- *mlq* is the is the mid-level qualifier that you specified during customization.

Procedure

1. Issue the following operator command from the z/OS console:

```
TSO ex 'hlq.mlq.SABPCLST(ABPF)'
```

2. Press Enter.

Results

- If you are starting the interface for the first time, DB2 UET automatically displays the Set ABPID panel so that you can select the DB2 UET configuration to use. After you specify a configuration, DB2 UET automatically displays the Set DB2 System panel so that you can select the DB2 subsystem to which to connect.
- If you previously selected a product configuration and DB2 subsystem, DB2 UET displays the Main Menu first. If the selected product configuration is not running when you start the interface, the Set ABPID panel is displayed and lists any product configurations that are currently running. You can either select one of the active product configurations or ask the product administrator to start the product configuration that you usually use.

Chapter 7. Troubleshooting and messages

Use this information to diagnose and correct problems that you might experience when you customize the components of the DB2 Utilities Solution Pack.

Important: This section includes only the messages that you might encounter during the Tools Customizer customization process. It does not include messages that can be issued by a solution pack component. For the complete set of messages that are associated with any solution pack component, see the component user's guide.

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.

Tools Customizer messages

Use the information in these messages to help you diagnose and solve Tools Customizer problems.

CCQB000I The component parameter data was saved in the data store.

Explanation: Changes that were made to the component parameters were saved in the data store.

System action: None.

User response: No action is required.

CCQB001I The DB2 parameter data was saved in the data store.

Explanation: Changes that were made to the DB2 parameters were saved in the data store.

System action: None.

User response: No action is required.

CCQB002I The LPAR parameter data was saved in the data store.

Explanation: Changes that were made to the LPAR parameters were saved in the data store.

System action: None.

User response: No action is required.

CCQB003E At least one step must be selected in a selected task. The selected task is *task_description*.

Explanation: When a task is selected, at least one step must be selected. A selected step is missing from the specified task.

System action: Processing stops.

User response: Select a step in the specified task or deselect the task.

CCQB004I The required information to run the Discover EXEC was saved in the data store.

Explanation: The data store contains all the information that is required to run the Discover EXEC.

System action: None.

User response: No action is required.

CCQB005E The conflicting values for the *parameter_name* parameter must be resolved before the information can be saved.

Explanation: Two values for one parameter conflict

with each other, and they must be resolved to save the information.

System action: Processing stops.

User response: Resolve the conflicting values for the parameter.

CCQB006E One row must be selected.

Explanation: One row in the table must be selected.

System action: Processing stops.

User response: Select one row.

CCQB007E Only one row can be selected.

Explanation: Multiple rows in the table are selected, but only one row is allowed to 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 component 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 component 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 component, 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 component, 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 "Gathering diagnostic information" on page 215. Contact IBM Software Support.

CCQC005S The *template_name* template exceeds the number of allowed templates for a customization sequence. The template is in the *library_name* 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 "Gathering diagnostic information" on page 215. 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 215. 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.

User response: See "Gathering diagnostic information" on page 215. 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 215. 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 215. 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 component 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 component 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 *template_name* template in the *library_name* metadata library contains the following file-tailoring 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 215. 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 "Gathering diagnostic information" on page 215. 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 215. 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 215. 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 215. Contact IBM Software Support.

CCQC021S The *template_name* template in the *library_name* metadata library is not valid because the template must start with the *parameter_name* job card parameter.

Explanation: The specified template must start with the specified job card parameter.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQC022S The 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: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQC023S The)DOT file-tailoring control statement must include the *table-name* table name in the *template_name* template. 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 215. 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 ISPFTRC command, and press Enter.

3. Issue the Tools Customizer command that issues the error.
4. Specify the ISPFTRC command, and press Enter. The ISPF file tailoring trace data set is created. It adheres the following naming convention: *TSO_ID*.ISPF.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 component has not been customized for any DB2 entries.

System action: None.

User response: Complete the steps to customize a component. Customized jobs are generated when all required component, LPAR parameters, and DB2 parameters are defined and at least one DB2 entry on which to customize the component 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 215. Contact IBM Software Support.

CCQC027S The *job_name* customization job was not found in the *library_name* customization library.

Explanation: The selected customization job does not exist in the customization library.

System action: Processing stops.

User response: See “Gathering diagnostic

information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQC029I The customization jobs were generated for *component_name*.

Explanation: The customization jobs were generated for the specific component.

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 component.

Explanation: The component 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 component 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 component.

System action: None.

User response: No action is required.

CCQC032S The customization jobs were not generated for *component_name*.

Explanation: A severe error occurred while the jobs were being generated for the specified component.

System action: None.

User response: See “Gathering diagnostic information” on page 215. 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.

CCQC034S The specified operation is not allowed.

Explanation: Issuing commands against customization jobs from the customization library from an ISPF browse or edit session that was started on the Finish Component 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 component 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 component 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 Component 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 Component 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD005S The XML structure of the *member_name* environment 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 “Gathering diagnostic information” on page 215. 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 215. Contact IBM Software Support.

CCQD007S The XML structure of the *member_name* environment index 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 environment index member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD012S The XML structure of the *member_name* environment index member is not valid. The content length for the *element_name* 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD013S The XML structure of the *member_name* environment index member is not valid. The *attribute_name* 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 “Gathering diagnostic information” on page 215. 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: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD100W The *member_name* component index member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation: While determining if the component index 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.

CCQD101S The *member_name* component index member is not valid. The PL/I XML parser issued the following exception error code: *code_number*.

Explanation: While determining if the component 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* component index member is not valid. The *element_name* element is unknown.

Explanation: The specified component index member contains an unknown element.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD103S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD104S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD105S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD106S The XML structure of the *member_name* component 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 component index member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD107S The XML structure of the *member_name* component index 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 component index member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD108S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD109S The XML structure of the *member_name* component 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 component index member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD110S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD111S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD112S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD113S The XML structure of the *member_name* component index member is not valid. The *attribute_name* attribute in the *element_name* element is unknown.

Explanation: The specified attribute in the component index member is unknown.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD118S The content of the *member_name* component 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: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD121I The specified pack contains the *component_name*, which was previously specified as a stand-alone 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 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* component environment member is not valid. The PL/I XML parser issued the following exception warning code: *code_number*.

Explanation: While determining if the component 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 *member_name* component environment member is not valid. The PL/I XML parser issued the following exception error code: *code_number*.

Explanation: While determining if the component 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* component environment member is not valid. The *element_name* element is unknown.

Explanation: The specified component environment member contains an unknown element.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD303S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD304S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD305S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD306S The XML structure of the *member_name* component 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 component environment member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD307S The XML structure of the *member_name* component 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 component environment member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD308S The XML structure of the *member_name* component environment 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. 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 215. Contact IBM Software Support.

CCQD309S The XML structure of the *member_name* component 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 component environment member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD310S The XML structure of the *member_name* component environment 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. 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 215. Contact IBM Software Support.

CCQD311S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD312S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD313S The XML structure of the *member_name* component environment member is not valid. The *attribute_name* attribute in the *element_name* element is unknown.

Explanation: The specified attribute in the component environment member is unknown.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD350I The *subsystem_ID* DB2 subsystem is associated with this component.

Explanation: The specified DB2 subsystem was added and saved in the Tools Customizer data store for the component 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 component.

Explanation: The specified DB2 member for the group attach name was added and saved in the Tools

Customizer data store for the component 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 component.

Explanation: The specified DB2 group attach name was added and saved in the Tools Customizer data store for the component to be customized.

System action: Processing continues.

User response: No action is required.

CCQD353E The *subsystem_ID* DB2 subsystem is already associated with this component.

Explanation: The specified DB2 subsystem cannot be added for the component to be customized because it already exists in the component 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 component.

Explanation: The specified DB2 member for the group attach name cannot be added for the component to be customized because it already exists in the component 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 component.

Explanation: The specified DB2 group attach name cannot be added for the component to be customized because it already exists in the component 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 component.

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 component.

Explanation: The specified DB2 SSID was unassociated with the component 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 component.

Explanation: The specified DB2 member for the DB2 group attach name was unassociated with the component 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 component.

Explanation: The specified DB2 group attach name was unassociated with the component 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 *library_name* metadata library is not associated with the specified DB2 data sharing group member *member_name* 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 component 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 component 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 *data_set_name* 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 *data_set_name* 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 The *data_set_name* data store data set could not be created.

Explanation: Tools Customizer could not create the specified data set.

System action: Processing stops.

User response: Ensure that you have the authority to

create data sets and that the DASD is not full.

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 "Gathering diagnostic information" on page 215. Contact IBM Software Support.

CCQD513E The specified DB2 entry already exists and is associated with the current component on the Customizer Workplace panel.

Explanation: The DB2 entry cannot be added because it already exists, and it is already associated with the component 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 component.

Explanation: The DB2 entry has already been created and associated with the component 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 Component panel but is not associated with the current component.

Explanation: The DB2 entry exists, but it must be associated with the component to be customized.

System action: None.

User response: On the Customizer Workplace panel, issue the ASSOCIATE command to associate the DB2 entry with the component.

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 215. 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 component is already associated with the allowed number of DB2 entries.

Explanation: The DB2 entry was not completely copied because a component 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 *member_name* 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 *metadata_library* 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 *subsystem_ID* DB2 subsystem already exists and is associated with the current component on the Customizer Workplace panel.

Explanation: The specified DB2 subsystem exists and is associated with the component that you are customizing.

System action: None.

User response: Specify another DB2 subsystem.

CCQD561E The *member_name* DB2 member for the *group_attach_name* DB2 group attach name already exists and is associated with the current component on the Customizer Workplace panel.

Explanation: The specified DB2 data sharing group for the DB2 group attach namer exists and is associated with the component 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 component on the Customizer Workplace panel.

Explanation: The specified DB2 group attach name exists and is associated with the component 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 component.

Explanation: The specified subsystem is already associated.

System action: None.

User response: Specify a different DB2 subsystem.

CCQD566E The *member_name* 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 component.

Explanation: The specified DB2 member 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 component.

Explanation: The specified DB2 group attach name is already associated.

System action: None.

User response: Specify another DB2 group attach name.

CCQD568I To customize *product_name*, at least one DB2 entry must be associated with this product.

Explanation: The specified product requires at least one associated DB2 entry.

System action: None.

User response: To continue the customization process for the specified product, associate one or more DB2 entries with it.

CCQD569I To customize the *product_name* product configuration, at least one DB2 entry must be associated with this configuration.

Explanation: The configuration for the specified product requires at least one associated DB2 entry.

System action: None.

User response: To continue the customization process for the configuration of the specified product, 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.

CCQD580S The *subsystem_ID* DB2 subsystem was copied to the list of DB2 entries but was not associated because the component is already associated with the allowed number of DB2 entries.

Explanation: The copied DB2 subsystem was not associated with the component because the component is associated with the maximum number of DB2 entries.

System action: None.

User response: Remove an associated DB2 entry and associate the component with the copied DB2 subsystem.

CCQD581S The *member_name* 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 component 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 component because the component is associated with the maximum number of DB2 entries.

System action: None.

User response: Remove an associated DB2 entry and associate the component 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 component is already associated with the allowed number of DB2 entries.

Explanation: The copied DB2 group attach name was not associated with the component because the component is associated with the maximum number of DB2 entries.

System action: None.

User response: Remove an associated DB2 entry and associate the component with the copied DB2 group attach name.

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 component.

Explanation: The LPAR that is stored in the data store data set must be used to customize the component.

System action: Processing stops.

User response: Use the LPAR that is stored in the data store data set.

CCQD587W The *level_number* DB2 level of the *subsystem_name* DB2 subsystem is not supported by the product.

Explanation: The product does not support the specified DB2 level.

System action: Processing continues.

User response: Specify a supported 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 does not support the specified DB2 level.

System action: Processing continues.

User response: Specify a supported level of DB2.

CCQD589W The *level_number* DB2 level of the *group_name* DB2 group attach name is not supported by the product.

Explanation: The product does not support the specified DB2 level.

System action: Processing continues.

User response: Specify a supported level of DB2.

CCQD593I The *subsystem_ID* DB2 subsystem was deleted.

User response: No action is required.

CCQD594I The *member_name* DB2 for the *group_attach_name* DB2 group attach name was deleted.

User response: No action is required.

CCQD595I The *group_attach_name* DB2 group attach name was deleted.

User response: No action is required.

CCQD596E The *subsystem_ID* DB2 subsystem was not deleted.

Explanation: An internal error occurred while the specified DB2 subsystem was being deleted.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 215. 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 215. Contact IBM Software Support.

CCQD600W The *member_name* component 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 component 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* component 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 component 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* component 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 215. Contact IBM Software Support.

CCQD603S The XML structure of the *member_name* component customization 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 215. Contact IBM Software Support.

CCQD604S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD605S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD606S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD607S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD608S The XML structure of the *member_name* component customization 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 215. Contact IBM Software Support.

CCQD609S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD610S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD611S The XML structure of the *member_name* component customization 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQD612S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD613S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQD614S The content of the *member_name* component 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 215. 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 *member_name* 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.

CCQD800W The *member_name* 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 *member_name* 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 *subsystem_ID* 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 *member_name* 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* component 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 component 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* component 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 component 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 component 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 component parameter.

System action: Processing continues.

User response: No action is required. To stop this message from being issued, remove the extra values from the component 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 *member_name* DB2 member for the *group_attach_name* DB2 group attach name was changed to the *subsystem_ID* 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 215. 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 component, verify your user settings.

Explanation: The user settings must be verified before a component 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 component 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 component can be customized.

System action: Tools Customizer stops.

User response: Change the user settings.

CCQI000W The XML structure of the *member_name* 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 *member_name* 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 215. 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 215. 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 215. 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 215. 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 215. Contact IBM Software Support.

CCQI013S The XML structure of the *member_name* DB2 parameter metadata member is not valid. The *attribute_name* 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 215. 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 215. Contact IBM Software Support.

CCQI015S The content of the 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 *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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI052S The *parameter_name* component parameter in the *template_name* template does not have associated metadata in the *member_name* component parameter metadata member.

Explanation: The specified template does not contain metadata for a component parameter. The name of the component parameter metadata member, the name of the component parameter, and the name of the template are indicated in the message text.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI053E The following metadata data set was not found: *data_set_name*.

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 *data_set_name* 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI057S The *member_name* component parameter metadata member was not found in the *data_set_name* data set.

Explanation: The component parameter metadata member was not found in the specified data set.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI058I *component_name* does not have any DB2 parameters.

Explanation: DB2 parameters are not required to customize the specified component.

System action: Processing continues.

User response: No action is required.

CCQI059I *component_name* does not have any LPAR parameters.

Explanation: LPAR parameters are not required to customize the specified component.

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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI062S The *parameter_name* component parameter in the *task_description* task condition does not have associated metadata in the *member_name* component parameter metadata member.

Explanation: Associated metadata is missing for the specified component parameter in a task.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 215. 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 215. Contact IBM Software Support.

CCQI065S The *parameter_name* component 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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI068S The *parameter_name* component parameter in the *task_description* task, *step_description* step, and *template_name* template condition does not have associated metadata in the *member_name* component parameter metadata member.

Explanation: Associated metadata is missing for the specified component parameter in a task, step, and template.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI069S Component metadata does not support multiple configurations, but the *template_name* component template contains the *parameter_name* parameter. Enable multiple configurations support for this component, and try again.

Explanation: The specified template contains a parameter for multiple configurations, but the component 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 *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 215. 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 215. Contact IBM Software Support.

CCQI072E The *parameter_name* component 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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI075S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. 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 215. 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 215. Contact IBM Software Support.

CCQI078S The XML structure of the *member_name* component parameter metadata member is not valid. The *parameter_name* parameter refers to the *section-name* section. This section was not found in the component parameter metadata member.

Explanation: The specified parameter refers to a section that is not in the component parameter metadata member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI080S The content of the *member_name* 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 *value_name*.

Explanation: The specified value for an attribute in the DB2 parameter metadata member is not valid.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI082S The content of the *member_name* component 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 component parameter metadata member is not valid.

System action: Processing stops.

User response: See “Gathering diagnostic

information” on page 215. Contact IBM Software Support.

CCQI090S The component-defined DB2 parameter *parameter_name* in the *member_name* 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 215. Contact IBM Software Support.

CCQI091S The component-defined LPAR parameter in the *member_name* 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 *member_name* 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI108S The XML structure of the *member_name* LPAR 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI110S The XML structure of the *member_name* LPAR 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 215. 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 215. Contact IBM Software Support.

CCQI112S 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 215. Contact IBM Software Support.

CCQI115S 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI117S The content of the *member_name* LPAR 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 LPAR parameter metadata member is not valid.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

information” on page 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI124S The XML structure of the *member_name* component customization 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 215. Contact IBM Software Support.

CCQI200W The XML structure of the *member_name* 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.

CCQI201S The XML structure of the *member_name* 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 215. Contact IBM Software Support.

CCQI203S 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 215. Contact IBM Software Support.

CCQI204S The XML structure of the *member_name* information 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI206S The XML structure of the *member_name* information 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 215. 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI214S The content of the *member_name* information 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 information metadata member is not valid.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI215S The content of the *member_name* information 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 information metadata member is not valid.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 the *element_name* element exceed *maximum_number* characters.

Explanation: The specified attribute contains too many characters.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 *member_name* 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 *data_set_name* 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*.SDENU.

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: Specify the correct metadata library.

CCQI253E 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.

CCQI300W 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI304S The XML structure of the *member_name* sequence 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 “Gathering diagnostic information” on page 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI308S The XML structure of the *member_name* sequence 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 215. Contact IBM Software Support.

CCQI309S The XML structure of the *member_name* sequence 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 215. Contact IBM Software Support.

CCQI310S The XML structure of the *member_name* sequence 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 contain content.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI311S The XML structure of the *member_name* 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 “Gathering diagnostic information” on page 215. 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI316S The content of the *member_name* 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 215. Contact IBM Software Support.

CCQI317S The content of the *member_name* sequence 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 sequence metadata member is not valid.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 215. 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 215. Contact IBM Software Support.

CCQI352S The *template_name* component template was not found in the *data_set_name* metadata data set.

Explanation: Tools Customizer could not find the specified component template in the data set.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 215. 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 *element_name* element already exists.

Explanation: The specified attribute contains a value that already exists.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 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: See “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI404S The XML structure of the *member_name* 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 215. 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 215. Contact IBM Software Support.

CCQI406S The XML structure of the *member_name* 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 215. 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 “Gathering diagnostic information” on page 215. 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 215. 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 215. Contact IBM Software Support.

CCQI410S The XML structure of the *member_name* 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 215. Contact IBM Software Support.

CCQI411S The XML structure of the *member_name* 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 215. Contact IBM Software Support.

CCQI412S 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 215. Contact IBM Software Support.

CCQI413S The XML structure of the *member_name* parameter metadata member is not valid. The *attribute_name* 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 “Gathering diagnostic information” on page 215. 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 215. 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 215. Contact IBM Software Support.

CCQI422S 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 DB2 parameter.

Explanation:

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 215. Contact IBM Software Support.

CCQI450S The *member_name* component parameter metadata member was not found in the *data_set_name* data set.

Explanation: Tools Customizer could not find the specified component parameter metadata member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 215. Contact IBM Software Support.

CCQI600W The XML structure of the *member_name* component 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 component 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* component 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 component

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.

CCQI602S The XML structure of the *member_name* component customization parameter metadata member is not valid. The *element_name* element is unknown.

Explanation: The specified component customization parameter metadata member contains an unknown element.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI603S The XML structure of the *member_name* component customization parameter 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 215. Contact IBM Software Support.

CCQI604S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI605S The XML structure of the *member_name* component customization 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 215. Contact IBM Software Support.

CCQI606S The XML structure of the *member_name* component customization 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 in the component customization parameter metadata member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI607S The XML structure of the *member_name* component 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 component customization parameter metadata member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI608S The XML structure of the *member_name* component customization 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 in the component customization parameter metadata member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI609S The XML structure of the *member_name* component customization 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 in the component customization parameter metadata member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI610S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQI611S The XML structure of the *member_name* component 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI612S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQI613S The XML structure of the *member_name* component customization parameter metadata member is not valid. The *attribute_name* attribute in the *element_name* element is unknown.

Explanation: The specified component customization parameter metadata member contains an unknown attribute.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI614S The XML structure of the *member_name* component customization parameter metadata member is not valid. The value of the *element_name* 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI615S The XML structure of the *member_name* component customization parameter metadata member is not valid. The value of the *attribute_name* attribute for the *element_name* 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI616S The XML structure of the *member_name* component customization parameter metadata member is not valid. The data type of the *element_name* element is 'not valid. The value of the element 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 215. Contact IBM Software Support.

CCQI617S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQI650S The XML structure of the *member_name* component 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 215. Contact IBM Software Support.

CCQI651S The XML structure of the *member_name* component customization parameter metadata member is not valid. The *parameter_name* parameter refers to the following section, which was not found in the *member_name* component customization parameter metadata member: *section-name*.

Explanation: The specified section is not in the component customization parameter metadata member.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI652S The *member_name* component customization metadata member not valid. 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 specified length cannot be shorter than the default length.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI653S The content of the *member_name* component customization parameter metadata member is not valid. The value of the *attribute_name* attribute in the *element_name* element is not valid. 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 215. Contact IBM Software Support.

CCQI700W The XML structure of the *member_name* 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 *member_name* 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. 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 215. Contact IBM Software Support.

CCQI708S The XML structure of the *member_name* solution pack 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 215. 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 215. 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI714S The XML structure of the *member_name* solution pack 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 215. 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 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI717S The 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 215. 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 “Gathering diagnostic information” on page 215. 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQI751S The version in the *library_name* solution pack metadata library is different than the version 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 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 215. 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 215. 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 215. 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.

CCQO000W 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.

CCQO001S 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.

CCQO002S The XML structure of the *member_name* discover parameter metadata member is not valid. The *element_name* element is unknown.

Explanation: The specified element in the discover parameter metadata member is unknown.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO003S The XML structure of the *member_name* discover 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 215. Contact IBM Software Support.

CCQO004S 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO005S The XML structure of the *member_name* discover 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 215. Contact IBM Software Support.

CCQO006S 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO007S 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 215. Contact IBM Software Support.

CCQO008S 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 215. Contact IBM Software Support.

CCQO009S 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO010S The XML structure of the *member_name* discover 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 contain content.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO011S The XML structure of the *member_name* discover 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 requires content.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO012S The XML structure of the *member_name* discover parameter metadata member is not valid. The content length for the *attribute_name* attribute in the *element_name* 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 215. Contact IBM Software Support.

CCQO013S The XML structure of the *member_name* discover parameter metadata member is not valid. The *attribute_name* attribute in the *element_name* element is unknown.

Explanation: The specified attribute is unknown.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO014S 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 215. Contact IBM Software Support.

CCQO015S 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 215. Contact IBM Software Support.

CCQO016S 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO017S The content of the *member_name* component 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 component parameter metadata member is not valid.

System action: Processing stops.

User response: See “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO050S 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.

CCQO051W 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.

CCQO052S 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.

CCQO053W 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.

CCQO054W The *subsystem_ID* 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.

CCQO055W 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 Component Information panel.

CCQO056W 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 Component Information panel.

CCQO057W The Discover EXEC could not find the *parameter_name* parameter in the metadata for the component to be customized. The record was not processed.

Explanation: The specified parameter could not be found in the metadata for the component to be customized.

System action: Processing continues.

User response: Ensure that information is specified correctly on the Discover Customized Component Information panel.

CCQO058W The *parameter_name* component 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 215. Contact IBM Software Support.

CCQO059W The *parameter_name* component parameter cannot contain more than 72 characters. The record was not processed.

Explanation: The specified component parameter contains too many characters.

System action: Processing continues.

User response: Ensure that the specified component parameter does not exceed 72 characters.

CCQO060W The *record_name* Discover record from the REXX EXEC output must start with the following record type: *record_type*. 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 "Gathering diagnostic information" on page 215. Contact IBM Software Support.

CCQO061I If you do not have a previously customized version of the component, do not run the Discover EXEC. Press END to go to the Customizer Workplace panel.

Explanation: This message is issued when you customize a component for a the first time. It prompts you to use the Discover EXEC to discover data from a previous customization of the specified component.

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 Component Information panel. Otherwise, press End to continue without discovering data from a previous customization of the component.

CCQO062W 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.

CCQO064W 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 215. Contact IBM Software Support.

CCQO065W 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 “Gathering diagnostic information” on page 215. Contact IBM Software Support.

CCQO066W 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 Component Information panel.

CCQO067W 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 Component Information panel.

CCQO068W The value of the *parameter_name* 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 Component Information panel.

CCQO069W 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 Component Information panel.

CCQO070W 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.

CCQO071W 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.

CCQO072S 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 215. Contact IBM Software Support.

CCQO073E 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.

CCQO074S The content of the *member_name* discover metadata member is not valid. The value of the *attribute_name* attribute in the *element_name* 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 215. Contact IBM Software Support.

CCQO075W 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.

CCQO076W 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.

CCQO077S 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 215. Contact IBM Software Support.

CCQO080I *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 “Gathering diagnostic information” on page 215. 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.SDENU*.

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 component.

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 *library_name* metadata library for the component that is part of the *library_name* pack was not found in the catalog. The name of the pack is *pack_name*, 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 *library_name* metadata library for the component that is part of the *library_name* 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 *data_set_name* 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.

CCQS007E 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 An error occurred while the *data_set_name* Discover data set was being created.

Explanation: While the specified data set was being created, an error occurred.

System action: Processing continues.

User response: Ensure that you have WRITE authority access to this data set.

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.

CCQS020E 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.

CCQS021E The 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.

CCQS022E 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.

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 *nm*, where *nm* 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 *nm*, where *nm* is 1 - 99.

CCQT000I The component 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 component configuration ID *configuration_ID* was created.

Explanation: The specified configuration ID was created.

System action: None.

User response: No action is required.

CCQT004I The component configuration ID *configuration_ID* was removed.

Explanation: The specified configuration ID was removed.

System action: None.

User response: No action is required.

CCQT005E The component configuration ID *configuration_ID* is not valid. The component 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 .

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 *configuration_ID* 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.

CCQT017I The *configuration_ID* configuration was not copied.

Explanation: The specified configuration was not copied.

System action: Processing stops.

User response: No action is required.

CCQT018E Specify Y or N, and press Enter.

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 component configuration was successfully updated.

Explanation: The configuration was updated.

System action: Processing continue.

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 *component_name* was not found.

Explanation: The specified component was not found.

System action: Processing stops.

User response: Specify another component.

Chapter 8. Reference

Use DB2 Automation Tool, DB2 High Performance Unload, DB2 Sort, and DB2 Utility Enhancement reference information when you need more information about the customization of the components. You can find details about the tasks, steps, and parameters that are displayed on the Component Parameters panel, and information about the generated jobs.

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 a component 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.

does not support DB2 group attach names.

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 a component. If the component to be customized requires DB2 entries, you can customize the component only on DB2 entries that are in the associated list. When you customize a component, 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 the component.

Master list

The list of all DB2 entries that are defined but are not associated with a component. Tools Customizer obtains information about these DB2 entries either from entries that were created manually or from the customizations of other components 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 a component, 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 Component panel.

If the associated list does not have the DB2 entries on which you want to customize a component, 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.

Component parameters

Parameters that are specific to the component. These parameters are defined by the component and are stored in a data member that is defined by the component.

LPAR parameters

Parameters on the local LPAR that are required to customize . 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

Component, LPAR, and DB2 entry status type

After you specify the component that you want to customize, the component, the LPAR, and the DB2 entries have a status. The status is partly based on whether required parameters are defined. For some components, LPAR parameters or DB2 parameters might not be required. In these cases, the status is Not Required.

To customize a component, all of the required parameters must be defined.

If required parameters for the the component 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 component parameters, LPAR parameters, and DB2 parameters for the DB2 entries on which a component will be customized.

The following table shows the meaning of the status types. Each status is defined differently for each type of parameter.

Table 19. Status types for the component, the LPAR, and the DB2 entries

Status	Component	LPAR	DB2 entries
Incomplete	The required component parameters are not defined, or the required component parameters are defined but LPAR parameters, DB2 parameters, or both are not defined.	The required parameters are not defined.	The required parameters are not defined.
Discovered	The component parameter definitions were discovered by using the component Discover EXEC.	N/A	N/A
Ready to Customize	The required component, LPAR, and DB2 parameters are defined, the status 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.
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.

Component status type

Each of the components in a solution pack has a status, and the status is partly based on whether the component is customized.

The following table shows the meaning of the status types.

Table 20. Status types for the component, the LPAR, and the DB2 entries

Status	Meaning
Customized	The customization process is finished, and the component is customized.
Pending Customization	The customization process is not finished, and the component is not customized.
Not Found	The metadata library for the component was not found.
Not Applicable	The component cannot be customized by using Tools Customizer.

Related tasks:

“Creating and associating DB2 entries” on page 181

You can create new DB2 entries and associate them with components in the DB2 Utilities Solution Pack.

“Copying DB2 entries” on page 191

You can copy associated and not associated DB2 entries to other DB2 entries or to new DB2 entries.

“Removing DB2 entries” on page 193

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 component with Tools Customizer. These data sets are supplied by component, supplied by Tools Customizer, or allocated by Tools Customizer.

The pack provides the following data sets:

Metadata library

Contains the metadata for the pack to be customized. Tools Customizer uses the metadata to determine which tasks, steps, and parameters to display on the Component 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*.SBBYDENU, 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

Contains the component Discover EXEC. Each component in the pack has a unique Discover EXEC. When you customize a component, you can use the Discover EXEC to automatically retrieve and store component information, such as parameter values from an already customized component. 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 example, the lowest-level qualifier for most components is SXYZDENU, where XYZ is the three-character

component prefix. You can change the default value on the Discover Customized Component 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 components in the DB2 Utilities Solution Pack. 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 components, 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 component Discover EXEC. The component Discover EXEC retrieves the metadata and values for the parameters from a previous customization of the component.

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 Component Information panel. WRITE access to this data set is required.

Data store data set

Contains component, 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 component, 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 the component.

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 a component, submit the members of the data set in the order in which they are displayed on the Finish Component Customization panel.

The data set naming convention is *hlq.\$LPAR_name\$.xyzorm*, 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
- *xyzorm* is the three-letter pack 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.
- You cannot share the data store data set across multiple LPARs with shared DASD or copy the data store data set to another LPAR. Tools Customizer creates many cross-references between product and DB2 associations. Therefore, if you share or copy the data store data set, member names that are empty or that do not exist might be generated.

How to read syntax diagrams

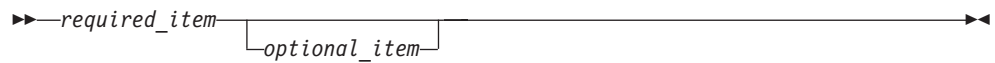
The following rules apply to the syntax diagrams that are used in this information:

- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
 - The >>--- symbol indicates the beginning of a syntax diagram.
 - The ---> symbol indicates that the syntax diagram is continued on the next line.

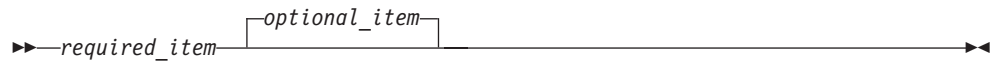
- The >--- symbol indicates that a syntax diagram is continued from the previous line.
- The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).



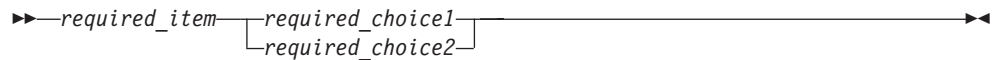
- Optional items appear below the main path.



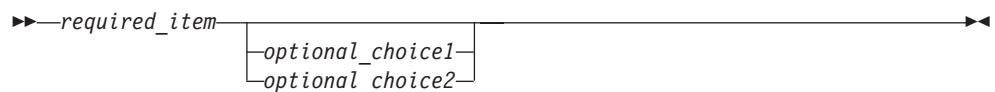
If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.



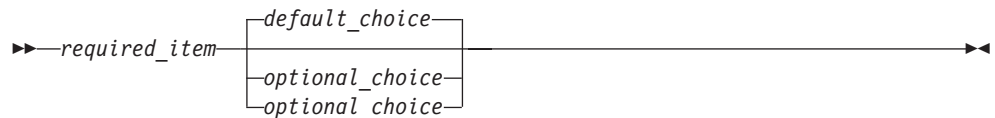
- If you can choose from two or more items, they appear vertically, in a stack. If you *must* choose one of the items, one item of the stack appears on the main path.



If choosing one of the items is optional, the entire stack appears below the main path.



If one of the items is the default, it appears above the main path, and the remaining choices are shown below.



- An arrow returning to the left, above the main line, indicates an item that can be repeated.



If the repeat arrow contains a comma, you must separate repeated items with a comma.



A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown. Variables appear in all lowercase italic letters (for example, *column-name*). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses; for example, (1).

Notices

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

This material may 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 give you any license to these patents. You can send license inquiries, in writing, to:

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

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

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

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

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

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

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

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

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

IBM Corporation
J64A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

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

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

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

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.

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

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

Programming interface information

This publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of DB2 Utilities Solution Pack.

This publication documents information that is NOT intended to be used as Programming Interfaces of DB2 Utilities Solution Pack.

This publication primarily documents intended Programming Interfaces that allow the customer to write programs to obtain the services of DB2 Utilities Solution Pack.

This publication also documents information that is NOT intended to be used as Programming Interfaces of DB2 Utilities Solution Pack. This information is identified where it occurs by an introductory statement to a topic or section.

This publication primarily documents information that is NOT intended to be used as Programming Interfaces of DB2 Utilities Solution Pack.

This publication also documents intended Programming Interfaces that allow the customer to write programs to obtain the services of DB2 Utilities Solution Pack. This information is identified where it occurs by an introductory statement to a topic or section.

Trademarks

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

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

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

Other company, product, and service names may be trademarks or service marks of others.

Privacy policy considerations

IBM Software products, including software as a service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's Privacy Policy at <http://www.ibm.com/privacy> and the section titled "Cookies, Web Beacons, and Other Technologies" in IBM's Online Privacy Statement at <http://www.ibm.com/privacy/details>. Also, see the "IBM Software Products and Software-as-a-Service Privacy Statement" at <http://www.ibm.com/software/info/product-privacy>.

Index

Special characters

69

A

ABPID
 setting in initialization options
 member 203
abpidBCAN member 206
abpidBGLB member 206
abpidOPTS member 203
abpidPROC member 205
accessibility
 overview 12
accessing DB2 Sort libraries 200
ACCPREP parameter 123
ACF2 requirements 19
action and alert resolution
 in Management Console 4
activating DB2 Sort 162
ACTLGUSR parameter 69
APF LOAD modules library 68
APF LOAD modules parameter 68
APF-authorizing the load library 205
authorization 161
Autonomics Director for DB2
 automating sensor data collection 4
 autonomics strategy 4
autonomics policies 10
autonomics strategy 4

B

batch utility
 enabling 161
 generating jobs 161
batch windows
 scheduling maintenance jobs
 automatically 10
benefits
 DB2 Utilities Solution Pack 1

C

cancel parameters, DSNUTILB
 intercept 206
CANCEL_TYPE parameter
 for DSNUTILB intercept 206
catalog navigation
 in Management Console 4
changing display options 171
CHECK_THDTERM_RETRY_COUNT
 parameter
 for DSNUTILB intercept 206
CHECK_THDTERM_RETRY_INTERVAL
 parameter
 for DSNUTILB intercept 206
COLLOBJT parameter 69

compatibilities of DB2 Automation Tool
 product versions 19
component
 post-customization tasks 197
configuration
 authorizing 161
 enabling batch utility 161
configuration ID, product
 setting in initialization options
 member 203
cookie policy 283, 286
Copy DB2 Entries panel 191
creating user exits
 post-cancel exit 211
 pre-cancel exit 211
 security exit 210
CTRLLIBM parameter 83
customization
 associated list
 adding DB2 entries 181
 overview 275
 associating DB2 entries 181
 browsing parameters 191
 changing display options 171
 changing parameters 173
 component 275
 component parameters
 changing 176
 defining 183
 editing 176, 183
 modifying 176
 Component Parameters panel 183
 copying DB2 entries 191
 Create a DB2 Entry panel 181
 creating DB2 entries 181
 customization jobs
 deleting 194
 displaying 194
 generating 188
 maintaining 194
 regenerating 188
 renaming 194
 sort sequence 189
 submitting 189, 194
 customization library
 deleting jobs 194
 maintaining 194
 overview 278
 recustomizing 194
 renaming jobs 194
 customization library qualifier
 specifying 168
 Customized status 275
 Customizer Workplace panel 188
 customizing a component for the first
 time 173
 customizing a new version of a
 component 173
 customizing settings 168
data sets
 customization library 278

customization (*continued*)
 data sets (*continued*)
 data store 163, 278
 DATASTOR 163
 Discover EXEC library 278
 metadata library 163, 278
 data store
 overview 278
 data store data set
 specifying 168
 DB2 Administration Tool
 parameters 114
 DB2 data sharing members
 adding 181
 associating 181
 copying 191
 creating 181
 deleting 193
 generating jobs for 188
 removing 193
 selecting 188
 specifying 188
 unassociating 193
 DB2 group attach field
 specifying 168
 DB2 group attach names
 adding 181
 associating 181
 copying 191
 creating 181
 DB2 HPU conversion parameters 115
 DB2 HPU DB2 parameters 69
 DB2 HPU file management
 parameters 116
 DB2 HPU library dsnames 68
 DB2 HPU output data parameters
 output data 83
 DB2 HPU sort parameters 129
 DB2 HPU tuning parameters 123
 DB2 parameters
 defining 186
 editing 186
 DB2 Parameters panel 186
 DB2 subsystems
 adding 181
 associating 181
 copying 191
 creating 181
 defining component parameters 183
 defining DB2 parameters 186
 defining LPAR parameters 185
 defining parameters 183, 188
 deleting DB2 entries 193
 deleting jobs 176

- customization (*continued*)
 - Discover Customized Component Information panel 179
 - Discover EXEC
 - customizing a new version of a component 173, 174
 - overview 278
 - retrieving component information automatically 179
 - Discover process parameters 67
 - Discovered status 275
 - discovering component information 179
 - discovering previous versions 174
 - display options 171
 - displaying jobs 194
 - displaying panel text 171
 - editing component parameters 183
 - editing LPAR parameters 185
 - editing parameters 173
 - Errors in Customization status 275
 - finding trace data set 215
 - Finish Component Customization panel 189
 - first-time 173
 - first-time customization 173
 - generating jobs 188
 - hiding panel text 171
 - high-level qualifier 275
 - Incomplete status 275
 - integrating DB2 HPU into DB2 Admin 114, 199
 - integrating DB2 HPU into DB2 Launchpad 198
 - job sort order 189
 - jobs
 - deleting 194
 - displaying 194
 - maintaining 194
 - renaming 194
 - sort order 189
 - submitting 189, 194
 - LPAR parameters
 - defining 185
 - editing 185
 - LPAR Parameters panel 185
 - LPARs 194
 - maintaining jobs 194
 - master list
 - adding DB2 entries 181
 - Associate DB2 Entry for Component panel 181
 - overview 275
 - maximizing information panels 171
 - metadata libraries
 - specifying 177
 - metadata library
 - maintenance, best practices 163
 - overview 278
 - specifying 168
 - modifying parameters 173
 - modifying settings 168
 - multiple instances 168
 - multiple-LPAR environment 194
 - Not Required status 275
 - options 171
 - panel display options 171

- customization (*continued*)
 - panels
 - Associate DB2 Entry for Component 181
 - Component Parameters 183
 - Create a DB2 Entry 181
 - Customizer Workplace 188
 - DB2 Parameters 186
 - Discover Customized Component Information 179
 - Finish Component Customization 189
 - LPAR Parameters 185
 - Specify the Metadata Library 177
 - parameter values 40, 139
 - parameters
 - browsing 191
 - defining 183, 188
 - viewing 191
 - preparing to use Tools Customizer 168
 - product 275
 - Ready to Customize status 275
 - recustomization 173, 176
 - recustomizing 176
 - recustomizing a component 173
 - removing DB2 entries 193
 - roadmaps 173
 - customizing for the first time 173
 - first-time customization 173
 - reustomizing 176
 - selecting components 178
 - Specify the Metadata Library panel 177
 - specifying data sets 168
 - specifying metadata libraries 177
 - starting Tools Customizer 167
 - status types
 - Customized 275
 - Discovered 275
 - Errors in Customization 275
 - Incomplete 275
 - Not Required 275
 - Ready to Customize 275
 - submitting jobs 189
 - terminology 275
 - trace data set 215
 - troubleshooting 215
 - finding trace data set 215
 - user job card settings
 - specifying 168
 - viewing parameters 191
 - customization library
 - overview 278
 - customization library qualifier
 - specifying 168
 - customizing settings 168
 - customizing the product
 - APF-authorizing the load library 205
 - copying the DSNUTILF module 210
 - copying the started task PROC 205
 - creating a pre- or post-cancel user exit 211
 - creating a security exit 210
 - setting DSNUTILB intercept parameters 206
 - setting initialization options 203

D

- data set names
 - gathering 38, 136
- data sharing environments 31
- data store
 - maintenance, best practices 163
 - overview 278
- data store data set
 - specifying 168
- database administration
 - in Management Console 4
- DATEDDEL parameter 83
- DB2 Admin
 - integrating DB2 HPU 199
- DB2 Administration Tool parameters
 - customizing 114
- DB2 Analytics Accelerator
 - automating decisions to include DB2 tables 10
- DB2 Automation Tool
 - overview 1
 - post-customization tasks 197
- DB2 autonomies strategy 4
- DB2 data sharing environments 31
- DB2 group attach field
 - specifying 168
- DB2 High Performance Unload
 - overview 1
 - post-customization tasks 198
 - pre-customization tasks 161
- DB2 HPU conversion parameters
 - customizing 115
- DB2 HPU DB2 parameters
 - customizing 69
- DB2 HPU file management parameters
 - customizing 116
- DB2 HPU library dsnames
 - customizing 68
- DB2 HPU output data parameters
 - customizing 83
- DB2 HPU sort parameters
 - customizing 129
- DB2 HPU tuning parameters
 - customizing 123
- DB2 Launchpad
 - integrating DB2 HPU 198
- DB2 sensor data
 - scheduling collection and evaluation 4
- DB2 Sort
 - overview 1
 - post-customization tasks 200
 - pre-customization tasks 162
- DB2 Sort - accessing libraries 200
- DB2 Sort - activating 162
- DB2 Sort libraries
 - SCNKLINK 200
 - SCNKLPA 200
- DB2 Thread Expert, migrating from 36
- DB2 Utilities Enhancement Tool (UET)
 - overview 1
- DB2 Utilities Solution Pack
 - components 1
 - overview 1
 - pre-customization tasks 161
 - what is it? 1

- DB2 Utility Enhancement Tool
 - post-customization tasks 203
- DB2 version support 19
- DB2 versions, migrating and fallback 25
- DB2 versions, migration and fallback 25
- DB2DBRM 68
- DB2PARAL parameter 69
- DB2SORT subsystem parameter 28, 162
- DB2SORTI 28
- DBRM library 68
- DBRM parameter 68
- default installation options 26
- DFSIGDCB parameter 83
- diagnostic information
 - gathering 215
- Discover EXEC
 - overview 278
- Discover process parameters
 - customizing 67
- dispatching priority 35
- display options 171
- displaying panel text 171
- DISPLUSR parameter 69
- documentation
 - accessing 11
 - sending feedback 11
- dsnames
 - installation libraries 68
- DSNUTILB intercept
 - customizing cancel and global parameters 206
- DSNUTILF module 210
- DTDELIM parameter 83
- DTNULDLM parameter 83
- DYNALLOC runtime parameter 28
- dynamic sort work allocation
 - installation options 26
 - runtime parameter 28

E

- ESCALATE parameter
 - for DSNUTILB intercept 206
- ESCAPE parameter
 - for DSNUTILB intercept 206
- EXEC_TYPE parameter
 - for DSNUTILB intercept 206

F

- fallback, DB2 version 25
- first-time customization 173

G

- GBLPARAL parameter 123
- global parameters, DSNUTILB intercept 206

H

- HAARSTAT 26
- hardware requirements 22
- hiding panel text 171

I

- IEFACTRT exit 19
- INFVSMIG parameter 116
- initialization options member
 - setting up at customization 203
- installation library dsnames 68
- Installation Verification Program (IVP)
 - overview 201
 - report generation 202
 - using 201
- interactive component
 - generating
 - batch utility jobs 161
- intercept policy
 - using to improve performance 7
- intercept, DSNUTILB
 - customizing cancel and global parameters 206
- INZADBI member 198
- INZDB21X 199
- INZLINK 68
- INZLOAD 68
- INZPLIB library 68
- ISPF interface
 - starting 213
- ISPF messages library 68
- ISPF messages parameter 68
- ISPF panels library 68
- ISPF panels parameter 68
- ISPF skeletons library 68
- ISPF skeletons parameter 68
- ISPF tables library 68
- ISPF tables parameter 68

L

- Launchpad
 - integrating DB2 HPU 198
- legal notices
 - cookie policy 283, 286
 - notices 283
 - programming interface information 283, 285
 - trademarks 283, 285
- libraries
 - APF LOAD modules 68
 - authorizing SINZLINK 161
 - DBRM 68
 - dsnames 68
 - INZPLIB 68
 - ISPF messages 68
 - ISPF panels 68
 - ISPF skeletons 68
 - ISPF tables 68
 - LIBUTIL 68
 - LOAD modules 68
 - PARMLIB 68
 - SINZDBRM 68
 - SINZLINK 68
 - SINZLOAD 68
 - LIBUTIL library 68
 - LIMUNIT parameter 116
 - LOAD modules library 68
 - LOAD modules parameter 68
 - LOWMEM parameter 123

M

- Management Console
 - automating REORG operations 10
 - modernization in 10
 - monitoring DB2 10
 - troubleshooting DB2 10
 - with DB2 Utilities Solution Pack 4
 - maximizing information on panels 171
- MAXSORT parameter 129
- MAXTUNIT parameter 116
- members
 - INZADBI 198
- MEMLIMIT parameter 19
- metadata library
 - maintenance, best practices 163
 - overview 278
 - specifying 168
- migrating DB2 versions 25
- migrating from DB Thread Expert 36
- modifying settings 168

N

- notices 283, 285
- NULLPOS parameter 83

O

- ON_FAILURE parameter
 - for DSNUTILB intercept 206
- OPALLFMT parameter 83
- optimization mode option 26
- options 171
- OPTMODE
 - installation option 26
 - runtime parameter 28
- output formats
 - specifying parameters 83
- overview
 - DB2 Utilities Solution Pack 1
 - of DB2 Utility Solution Pack components 1

P

- PAGEMON
 - installation option 26
- panel display options 171
- panels
 - Copy DB2 Entries 191
 - Select the Components to Customize 178
- PARALLEL parameter 123
- parameters 67
 - ACCPREP 123
 - ACTLGUSR 69
 - APF LOAD modules 68
 - COLLOBJT 69
 - CTRLLIBM 83
 - customization 40, 139
 - customizing DB2 69
 - DATEDEL 83
 - DB2PARAL 69
 - DBRM 68
 - DFSIGDCB 83

parameters (continued)

DISPLUSR 69
 DTDELIM 83
 DTNULDLM 83
 GBLPARAL 123
 INFVSMIG 116
 ISPF messages 68
 ISPF panels 68
 ISPF skeletons 68
 ISPF tables 68
 LIMUNIT 116
 LOAD modules 68
 LOWMEM 123
 MAXSORT 129
 MAXTUNIT 116
 NULLPOS 83
 OPALLFMT 83
 PARALLEL 123
 PARMLIB 68
 PGDFIN 69
 PLANOBJT 69
 PLANOWN 69
 QSBUFNO 116
 QUIESCAT 69
 QUIESUSR 69
 QUITUSR 69
 RETRYMOD 123
 RETRYNB 123
 RETRYW 123
 RTSESTIM 69
 SCUNMOD 115
 SCUNSUB 115
 SCUNTSO 115
 SIZE 123
 SORTUTIL 129
 SQLACCES 69
 SRTVNBRE 129
 SRTVSMAX 129
 SRTVSMIN 129
 STRNGCST 83
 SUBTKSOF 123
 TAPEUNIT 116
 TIMEDEL 83
 TEMPLDSCB 116
 ULACCTRL 69
 ULAUTAG 83
 ULCHKCPY 69
 ULCNVTRC 69
 ULDATE 83
 ULDEGREE 69
 ULFORMAT 83
 ULHIDDEN 69
 ULIDENT 83
 ULLIKE 69
 ULLOCK 69
 ULMASK 83
 ULMAXPAR 123
 ULNULL 83
 ULOCSIZE 123
 ULOPTLDP 83
 ULOPTLDT 83
 ULOPTLX 69
 ULOPTNS 69
 ULPIC 83
 ULQSC 69
 ULQSCB 69
 ULROWSET 123

parameters (continued)

ULRTSID 69
 ULRTTST 69
 ULSEDB2 69
 ULSIGDTP 83
 ULTIME 83
 ULTMSTP 83
 ULTR32K 83
 ULTRIM 83
 UNLDDDN 83
 UNLLDER 69
 UNLSCHEM 69
 UTLPARAL 123
 VBUFSIZE 123
 VIZ003 68
 VIZ004 68
 VIZ007 68
 VIZ012 68
 VIZ013 68
 VIZ015 68
 VIZ016 68
 VIZ017 68
 VIZ018 68
 VSBUFND 116
 VUA007 116
 VUM011 69
 VUM012 69
 VUM014 69
 VUM018 116
 VUM020 69
 VUM022 116
 VUM023 116
 VUM024 129
 VUM025 129
 VUM026 129
 VUM027 69
 VUM028 69
 VUM029 83
 VUM030 69
 VUM031 69
 VUM032 69
 VUU011 69
 VUU012 69
 VUU013 69
 VUU014 83
 VUU015 83
 VUU016 83
 VUU017 83
 VUU018 83
 VUU019 83
 VUU020 83
 VUU021 69
 VUU022 83
 VUU023 69
 VUU025 114
 VUU026 114
 VUU027 114
 VUU028 69
 VUU029 83
 VUU030 69
 VUU031 83
 VUU032 83
 VUU033 69
 VUU034 69
 VUU035 123
 VUU036 123
 VUU038 83

parameters (continued)

VUU039 69
 VUU040 83
 VUU041 123
 VUU042 69
 VUU043 83
 VUU044 123
 VUU045 83
 VUU046 83
 VUU047 123
 VUU048 123
 VUU049 123
 VUU054 83
 VUU055 83
 VUU056 83
 VUU057 83
 VUU058 83
 VUU059 83
 VUU060 123
 VUU061 123
 VUU062 69
 VUU063 69
 VUU064 83
 VUU065 69
 VUU066 69
 VUU067 69
 VUX003 123
 VUX004 123
 VUX005 129
 VUX006 129
 VUX007 123
 VUX008 123
 VUX009 123
 VUX010 116
 VUX011 69
 VUX016 116
 VUX017 116
 VUX019 116
 VUX022 123
 VUX023 69
 VUX025 123
 VUX030 123
 VUX031 69
 VUX032 116
 VUX033 116
 VUX036 69
 VUX037 129
 VZM005 116
 VZM006 115
 VZM007 115
 VZM008 115
 VZM010 116
 WAITQTY 123
 WAITQTYM 123
 WAITUNIT 123
 WRKMXPR 116
 WRKSPACE 129
 WRKTUNIT 116
 WRKTVCNT 116
 WRKUNTSW 116
 WRKVOL 116
 PARMLIB library 68
 PARMLIB parameter 68
 performance
 improving sort processing
 performance 7
 improving utility performance 7

- PGDFIN parameter 69
- PLANOBJT parameter 69
- PLANOWN parameter 69
- policies
 - autonomics 10
- policy management
 - in Management Console 4
- post-cancel exit
 - creating 211
- post-customization tasks 197
 - DB2 Automation Tool 197
 - DB2 High Performance Unload 198
 - DB2 Sort 200
 - DB2 Utility Enhancement Tool 203
- pre-cancel exit
 - creating 211
- pre-customization tasks
 - DB2 High Performance Unload 161
 - DB2 Sort 162
 - DB2 Utilities Solution Pack 161
- preparing to use Tools Customizer 168
- problems
 - diagnostic information about 215
- Processor resource consumption
 - automating routine maintenance 9
- programming interface information 283, 285

Q

- QSBUFNO parameter 116
- QUIESCAT parameter 69
- QUIESUSR parameter 69
- QUITSUSR parameter 69

R

- reader comment form 11
- real-time statistics
 - automating tasks based on 4
- REGION parameter 28
- REPORT_TYPE parameter
 - for DSNUTILB intercept 206
- repository, porting statistics from 26
- requirements
 - DFSMSdss 23
 - hardware 22, 23
 - mainframe 23
 - operating system 23
 - software 22, 23
 - storage 23
- RETRYMOD parameter 123
- RETRYNB parameter 123
- RETRYW parameter 123
- roadmaps
 - customizing for the first time 173
 - first-time customization 173
- RTSESTIM parameter 69
- running different versions of DB2
 - Automation Tool 19
- RUNSTATS
 - automatic scheduling based on
 - criteria 10
- runtime considerations 28
- runtime parameters
 - DYNALLOC 28

- runtime parameters (*continued*)
 - OPTMODE 28
 - TUNEMSG 28
- RVT issue 36

S

- sample parameters
 - unload 83
- scenario
 - improving the performance of a DB2 utility by using a DB2 Utilities Enhancement Tool intercept
 - policy 7
 - improving the sort processing
 - performance of the DB2 utilities 7
 - reducing processor consumption by
 - automating routine maintenance 9
- SCNKLINK - DB2 Sort library 200
- SCNKLPA - DB2 Sort library 200
- screen readers and magnifiers 12
- SCUNMOD parameter 115
- SCUNSUB parameter 115
- SCUNTSO parameter 115
- security 33
- security exit
 - creating 210
- Select the Components to Customize
 - panel 178
- sensitivity to paging activity option 26
- sensor data
 - scheduling collection and
 - evaluation 4
- service information 11
- service level agreements
 - compliance using Management Console 4
- SINZDBRM library 68
- SINZLINK library
 - authorizing 161
 - description 68
- SINZLOAD library 68
- SIZE parameter 123
- SMFPRMxx member 19
- SMFPRMxx member of
 - SYS1.PARMLIB 19
- software requirements 19, 22
- SORTUTIL parameter 129
- specifying data sets 168
- SQLACCES parameter 69
- SRTVNBRE parameter 129
- SRTVSMAX parameter 129
- SRTVSMIN parameter 129
- started task
 - initialization options 203
 - running multiple started tasks 33
 - starting 212
- starting the ISPF interface 213
- starting the started task 212
- storage requirements 30
- STRNGCST parameter 83
- SUBTKSOF parameter 123
- support
 - required information 215
- support information 11
- syntax diagrams
 - how to read 280

- SYS1.PARMLIB 19

T

- TAPEUNIT parameter 116
- technotes 11
- THREAD_QUIESCE_TIME parameter
 - for DSNUTILB intercept 206
- TIMEDEL parameter 83
- TMPLDSCB parameter 116
- Tools Customizer
 - associated list
 - adding DB2 entries 181
 - overview 275
 - associating DB2 entries 181
 - browsing parameters 191
 - changing display options 171
 - component 275
 - component parameters
 - changing 176
 - editing 176
 - modifying 176
 - Component Parameters panel 183
 - Copy DB2 Entries panel 191
 - copying DB2 entries 191
 - Create a DB2 Entry panel 181
 - creating DB2 entries 181
 - customization jobs
 - deleting 194
 - displaying 194
 - generating 188
 - maintaining 194
 - renaming 194
 - sort sequence 189
 - submitting 189, 194
 - customization library
 - deleting jobs 194
 - maintaining 194
 - recustomizing 194
 - renaming jobs 194
 - customization library qualifier
 - specifying 168
 - customization templates 131
 - Customized status 275
 - Customizer Workplace panel 188
 - customizing a component for the first time 173
 - customizing a new version of a component 173
 - customizing components for the first time 173
 - customizing conversion
 - parameters 115
 - customizing DB2 HPU DB2
 - parameters 69
 - customizing file management
 - parameters 116
 - customizing new versions of components 174
 - customizing output data
 - parameters 83
 - customizing sort parameters 129
 - customizing tuning parameters 123
 - data sets
 - customization library 278
 - data store 163, 278
 - DATASOR 163

- Tools Customizer (*continued*)
 - data sets (*continued*)
 - Discover EXEC library 278
 - metadata library 163, 278
 - data store data set
 - specifying 168
 - DB2 data sharing members
 - adding 181
 - associating 181
 - copying 191
 - creating 181
 - DB2 entries 275
 - adding 181
 - associating 181
 - copying 191
 - creating 181
 - defining 188
 - deleting 193
 - generating jobs for 188
 - removing 193
 - selecting 188
 - specifying 188
 - unassociating 193
 - DB2 group attach field
 - specifying 168
 - DB2 group attach names
 - adding 181
 - associating 181
 - copying 191
 - creating 181
 - DB2 parameters
 - defining 186
 - editing 186
 - DB2 Parameters panel 186
 - DB2 subsystems
 - adding 181
 - associating 181
 - copying 191
 - creating 181
 - defining component parameters 183
 - defining DB2 parameters 186
 - defining LPAR parameters 185
 - defining parameters 183, 188
 - deleting DB2 entries 193
 - deleting jobs 176
 - Discover Customized Component Information panel 179
 - Discover Customized Product Information panel
 - customizing parameters 67
 - Discover EXEC
 - customizing a new version of a component 173
 - customizing new versions of components 174
 - retrieving component information automatically 179
 - Discovered status 275
 - discovering component information 179
 - displaying jobs 194
 - displaying panel text 171
 - editing component parameters 183
 - editing LPAR parameters 185
 - Errors in Customization status 275
 - finding trace data set 215
 - Tools Customizer (*continued*)
 - Finish Component Customization panel 189
 - first-time customization 173
 - generating jobs 188
 - hiding panel text 171
 - high-level qualifier 275
 - Incomplete status 275
 - integrating DB2 HPU into DB2 Admin 114
 - job sort order 189
 - jobs
 - deleting 194
 - displaying 194
 - maintaining 194
 - renaming 194
 - submitting 194
 - LPAR Parameters panel 185
 - maintaining jobs 194
 - master list
 - adding DB2 entries 181
 - Associate DB2 Entry for Component panel 181
 - overview 275
 - maximizing information on panels 171
 - metadata libraries 177
 - specifying 177
 - metadata library
 - specifying 168
 - multiple instances 168
 - multiple-LPAR environment 194
 - Not Required status 275
 - panels
 - Associate DB2 Entry for Component 181
 - Component Parameters 183
 - Copy DB2 Entries 191
 - Create a DB2 Entry 181
 - Customizer Workplace 188
 - DB2 Parameters 186
 - Discover Customized Component Information 179
 - Finish Component Customization 189
 - LPAR Parameters 185
 - Select the Components to Customize 178
 - Specify the Metadata Library 177
 - parameters
 - browsing 191
 - viewing 191
 - preparing to use 168
 - product 275
 - Product Parameters panel
 - customizing conversion parameters 115
 - customizing DB2 Admin parameters 114
 - customizing DB2 HPU DB2 parameters 69
 - customizing file management parameters 116
 - customizing library dsnames 68
 - customizing output data parameters 83
 - customizing sort parameters 129
 - Tools Customizer (*continued*)
 - Product Parameters panel (*continued*)
 - customizing tuning parameters 123
 - Ready to Customize status 275
 - recustomization 173
 - recustomizing a component 173
 - recustomizing components 176
 - removing DB2 entries 193
 - roadmaps
 - customizing a new version of a component 174
 - customizing new versions of components 174
 - recustomizing components 176
 - using the Discover EXEC 174
 - Select the Components to Customize panel 178
 - selecting components 178
 - Specify the Metadata Library panel 177
 - specifying metadata libraries 177
 - starting 167
 - status types
 - Customized 275
 - Discovered 275
 - Errors in Customization 275
 - Incomplete 275
 - Not Required 275
 - Ready to Customize 275
 - submitting jobs 189
 - terminology 275
 - trace data set 215
 - troubleshooting 215
 - user job card settings
 - specifying 168
 - using the Discover EXEC 174
 - viewing parameters 191
 - trace data set
 - finding 215
 - trademarks 283, 285
 - TUNEMSG
 - installation option 26
 - runtime parameter 28
 - tuning messages option 26
- U**
 - ULACCTRL parameter 69
 - ULAUTAG parameter 83
 - ULCHKCPY parameter 69
 - ULCNVTRC parameter 69
 - ULDATE parameter 83
 - ULDEGREE parameter 69
 - ULFORMAT parameter 83
 - ULHIDDEN parameter 69
 - ULIDENT parameter 83
 - ULLIKE parameter 69
 - ULLOCK parameter 69
 - ULMASK parameter 83
 - ULMAXPAR parameter 123
 - ULNULL parameter 83
 - ULOCSIZE parameter 123
 - ULOPTLDP parameter 83
 - ULOPTLDT parameter 83
 - ULOPTLX parameter 69
 - ULOPTNS parameter 69

- ULPIC parameter 83
- ULQSCE parameter 69
- ULQSCEBH parameter 69
- ULROWSET parameter 123
- ULRTSID parameter 69
- ULRTTST parameter 69
- ULSEDB2 parameter 69
- ULSIGDTP parameter 83
- ULTIME parameter 83
- ULTMSTP parameter 83
- ULTR32K parameter 83
- ULTRIM parameter 83
- UNLDDDN parameter 83
- UNLLDER parameter 69
- unload parameters 83
- UNLSCHEM parameter 69
- user job card settings
 - specifying 168
- UTLPARAL parameter 123

V

- VBUFSIZE parameter 123
- vector table issue 36
- VIZ003 parameter 68
- VIZ004 parameter 68
- VIZ007 parameter 68
- VIZ012 parameter 68
- VIZ013 parameter 68
- VIZ015 parameter 68
- VIZ016 parameter 68
- VIZ017 parameter 68
- VIZ018 parameter 68
- VSBUFND parameter 116
- VUA007 parameter 116
- VUM011 parameter 69
- VUM012 parameter 69
- VUM014 parameter 69
- VUM018 parameter 116
- VUM020 parameter 69
- VUM022 parameter 116
- VUM023 parameter 116
- VUM024 parameter 129
- VUM025 parameter 129
- VUM026 parameter 129
- VUM027 parameter 69
- VUM028 parameter 69
- VUM029 parameter 83
- VUM030 parameter 69
- VUM031 parameter 69
- VUM032 parameter 69
- VUU011 parameter 69
- VUU012 parameter 69
- VUU013 parameter 69
- VUU014 parameter 83
- VUU015 parameter 83
- VUU016 parameter 83
- VUU017 parameter 83
- VUU018 parameter 83
- VUU019 parameter 83
- VUU020 parameter 83
- VUU021 parameter 69
- VUU022 parameter 83
- VUU023 parameter 69
- VUU025 parameter 114
- VUU026 parameter 114
- VUU027 parameter 114

- VUU028 parameter 69
- VUU029 parameter 83
- VUU030 parameter 69
- VUU031 parameter 83
- VUU032 parameter 83
- VUU033 parameter 69
- VUU034 parameter 69
- VUU035 parameter 123
- VUU036 parameter 123
- VUU038 parameter 83
- VUU039 parameter 69
- VUU040 parameter 83
- VUU041 parameter 123
- VUU042 parameter 69
- VUU043 parameter 83
- VUU044 parameter 123
- VUU045 parameter 83
- VUU046 parameter 83
- VUU047 parameter 123
- VUU048 parameter 123
- VUU049 parameter 123
- VUU054 parameter 83
- VUU055 parameter 83
- VUU056 parameter 83
- VUU057 parameter 83
- VUU058 parameter 83
- VUU059 parameter 83
- VUU060 parameter 123
- VUU061 parameter 123
- VUU062 parameter 69
- VUU063 parameter 69
- VUU064 parameter 83
- VUU065 parameter 69
- VUU066 parameter 69
- VUU067 parameter 69
- VUX003 parameter 123
- VUX004 parameter 123
- VUX005 parameter 129
- VUX006 parameter 129
- VUX007 parameter 123
- VUX008 parameter 123
- VUX009 parameter 123
- VUX010 parameter 116
- VUX011 parameter 69
- VUX016 parameter 116
- VUX017 parameter 116
- VUX019 parameter 116
- VUX022 parameter 123
- VUX023 parameter 69
- VUX025 parameter 123
- VUX030 parameter 123
- VUX031 parameter 69
- VUX032 parameter 116
- VUX033 parameter 116
- VUX036 parameter 69
- VUX037 parameter 129
- VZM005 parameter 116
- VZM006 parameter 115
- VZM007 parameter 115
- VZM008 parameter 115
- VZM010 parameter 116

W

- WAITQTY parameter 123
- WAITQTYM parameter 123
- WAITUNIT parameter 123

- WRKMXPR parameter 116
- WRKSPACE parameter 129
- WRKTUNIT parameter 116
- WRKTVCNT parameter 116
- WRKUNTSW parameter 116
- WRKVOL parameter 116
- WTO messages, use in automated
 - operations 36



Product Number: 5697-DUP

Printed in USA

SC27-6563-00

