ALTER and CHANGE MANAGER for DB2
User Guide: Volume 1

Supporting

Version 12.1.00 of BMC Object Administration for DB2
Version 12.1.00 of BMC Administrative Assistant for DB2
Version 12.1.00 of BMC Database Administration for DB2
Version 12.1.00 of ALTER for DB2
Version 12.1.00 of CHANGE MANAGER for DB2
Version 12.1.00 of BMC Next Generation Technology Database Administration for DB2

December 2016
Contacting BMC Software

Several methods are available for contacting BMC Software.

You can access the BMC Software website at http://www.bmc.com. From this website, you can obtain information about the company, its products, corporate offices, special events, and career opportunities.

United States and Canada

<table>
<thead>
<tr>
<th>Address</th>
<th>Telephone</th>
<th>Fax</th>
</tr>
</thead>
</table>
| BMC SOFTWARE INC  
2103 CITYWEST BLVD  
HOUSTON TX 77042-2827 USA | 1 713 918 8800 or 1 800 841 2031 | 1 713 918 8000 |

Outside United States and Canada

<table>
<thead>
<tr>
<th>Telephone</th>
<th>Fax</th>
</tr>
</thead>
<tbody>
<tr>
<td>+01 713 918 8800</td>
<td>+01 713 918 8000</td>
</tr>
</tbody>
</table>

© Copyright 2007 - 2016 BMC Software, Inc.

BMC, BMC Software, the BMC logo, the BMC Software logo, and other BMC marks are the exclusive properties of BMC Software, Inc. and are registered or may be registered in the U.S. and in other countries. BladeLogic and other BladeLogic marks are the exclusive properties of BladeLogic, Inc. and are registered or may be registered in the U.S. and in other countries. All other trademarks or registered trademarks are the property of their respective owners.

DB2, DB2 Universal Database, DRDA, IBM, OS/390, QMF, RACF, THINK, and z/OS are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

The information included in this documentation is the proprietary and confidential information of BMC Software, Inc., its affiliates, or licensors. Your use of this information is subject to the terms and conditions of the applicable End User License agreement for the product and to the proprietary and restricted rights notices included in the product documentation.

Restricted rights legend

U.S. Government Restricted Rights to Computer Software. UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure of any data and computer software by the U.S. Government is subject to restrictions, as applicable, set forth in FAR Section 52.227-14, DFARS 252.227-7013, DFARS 252.227-7014, DFARS 252.227-7015, and DFARS 252.227-7025, as amended from time to time. Contractor/Manufacturer is BMC SOFTWARE INC, 2103 CITYWEST BLVD, HOUSTON TX 77042-2827, USA. Any contract notices should be sent to this address.
Customer support

Support website

You can obtain technical support from BMC 24 hours a day, 7 days a week at http://www.bmc.com/support. From this website, you can:

■ Read overviews about support services and programs that BMC offers
■ Find the most current information about BMC products
■ Search a database for problems similar to yours and possible solutions
■ Order or download product documentation
■ Download products and maintenance
■ Report a problem or ask a question
■ Subscribe to receive proactive e-mail alerts
■ Find worldwide BMC support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Support by telephone or e-mail

In the United States and Canada, if you need technical support and do not have access to the web, call 1 800 537 1813 or send an e-mail message to customer_support@bmc.com. (In the subject line, enter SupID:yourSupportContractID, such as SupID:12345). Outside the United States and Canada, contact your local support center for assistance.

Before contacting BMC

Have the following information available so that Customer Support can begin working on your issue immediately:

■ Product information
  — Product name
  — Product version (release number)
  — License number and password (trial or permanent)
■ Operating system and environment information
  — Machine type
  — Operating system type, version, and service pack or other maintenance level such as PUT or PTF
  — System hardware configuration
  — Serial numbers
  — Related software (database, application, and communication) including type, version, and service pack or maintenance level
■ Sequence of events leading to the problem
■ Commands and options that you used
■ Messages received (and the time and date that you received them)
  — Product error messages
  — Messages from the operating system
  — Messages from related software
License key and password information

If you have questions about your license key or password, contact Customer Support through one of the following methods:

- Send an e-mail message to customer_support@bmc.com. (In the Subject line, enter **SupID:yourSupportContractID**, such as SupID:12345.)
- In the United States and Canada, call 1 800 537 1813. Outside the United States and Canada, contact your local support center for assistance.
# Contents

**About this book**
- Related publications ................................................................. 9
- Conventions .................................................................................. 10
- Syntax statements ........................................................................ 10
- Summary of changes ...................................................................... 11

**Chapter 1** Overview of ALTER and CHANGE MANAGER 15
- Change management ........................................................................ 15
  - Identification of your environment ........................................... 16
  - Location of your environment .................................................. 16
  - Propagation of your changes ..................................................... 16
- Overview of change management with ALTER and CHANGE MANAGER ........................................................................ 17
  - Overview of ALTER ................................................................. 17
  - Overview of CHANGE MANAGER .............................................. 21
- ALTER architecture ........................................................................ 25
  - ALTER components ................................................................... 27
  - ALTER objects ........................................................................ 30
- CHANGE MANAGER architecture .................................................. 32
  - CHANGE MANAGER components ............................................ 35
  - CHANGE MANAGER objects ................................................... 38

**Chapter 2** Overview of the ALTER and CHANGE MANAGER interface 49
- Before you use the product .......................................................... 49
- ALTER and CHANGE MANAGER interface .................................. 49
  - Main Menu ................................................................................ 50
  - Displaying the current product environment ............................... 51
  - Using Fast Path Navigation ....................................................... 52
  - ISPF commands ........................................................................ 53
  - BMC message information ....................................................... 54
- Achieving your goals with ALTER and CHANGE MANAGER .............. 55
  - Creating, altering, or dropping DB2 objects ................................. 55
Migrating DB2 objects ........................................................................................................90

Chapter 3 Setting up ALTER and CHANGE MANAGER 129

Using options to control your environment ................................................................. 129
   Using the installation options ................................................................. 130
   Using the user options ................................................................. 131
   Using the product options ................................................................. 132
   Product option overrides ................................................................. 133
   Additional installation options module .............................................. 134
   How ALTER or CHANGE MANAGER use options ................................... 135

Data set allocation ........................................................................................................ 136

Object names ................................................................................................................ 136
   Wildcard patterns ........................................................................ 137
   Delimited identifiers ........................................................................ 139

Nonprintable or nonviewable characters ........................................................................... 140

Access to components .................................................................................................. 141
   ALTER and CHANGE MANAGER plans .............................................. 141
   CHANGE MANAGER plans ................................................................. 142
   Execution component plans ................................................................. 143

Authorization switching .......................................................................................... 144
   Execution security exit function codes ................................................... 145
   Execution security exit testing ................................................................. 147

Access to the worklist parallelism feature ................................................................. 148
   Zap to restrict access to worklist parallelism ................................................... 148
   General resource profile for worklist parallelism ........................................ 149

Setting the MEMLIMIT system parameter ................................................................. 149

Modification of ISPF skeletons and variables ................................................................. 150

JCL Generation data sets sizing function ................................................................. 151

Product table maintenance ........................................................................................ 152
   Table size maintenance ........................................................................ 156
   Table security ......................................................................................... 157

Setting options ........................................................................................................... 160
   Setting the attributes for the panel display ................................................. 160
   Setting the options for data set allocation ................................................ 162
   Setting the processing options for Analysis ...................................... 163
   Setting the processing options for Import .............................................. 165
   Setting the processing options for Execution ............................................. 166
   Setting the processing options for Compare ............................................ 167
   Setting the processing options for Baseline ............................................... 168
   Setting the options for the baseline report .................................................. 169
   Setting the user options for CM/PILOT ..................................................... 170
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting the JCL options for job cards</td>
<td>173</td>
</tr>
<tr>
<td>Setting the JCL options for STEPLIBs</td>
<td>175</td>
</tr>
<tr>
<td>Setting the JCL options for static data sets</td>
<td>177</td>
</tr>
<tr>
<td>Setting the JCL options for tapes</td>
<td>182</td>
</tr>
<tr>
<td>Setting the JCL options for temporary work data sets</td>
<td>184</td>
</tr>
<tr>
<td>Setting the JCL options for permanent data sets</td>
<td>186</td>
</tr>
<tr>
<td>Setting the JCL generation data group (GDG) options</td>
<td>196</td>
</tr>
<tr>
<td>Setting the JCL debugging, display, and Execution options</td>
<td>198</td>
</tr>
<tr>
<td>Setting the installation options module names for BMC utilities</td>
<td>201</td>
</tr>
<tr>
<td>Setting the online reorg options</td>
<td>202</td>
</tr>
<tr>
<td>Setting the NGT utility options</td>
<td>204</td>
</tr>
<tr>
<td>Setting the Execution worklist parallelism options</td>
<td>204</td>
</tr>
<tr>
<td>Setting the worklist options</td>
<td>207</td>
</tr>
<tr>
<td>Setting the Analysis options</td>
<td>208</td>
</tr>
<tr>
<td>Setting user variables</td>
<td>209</td>
</tr>
<tr>
<td>Creating a user POF</td>
<td>210</td>
</tr>
<tr>
<td>Updating a user POF</td>
<td>211</td>
</tr>
<tr>
<td>Using multiple POFs</td>
<td>213</td>
</tr>
<tr>
<td>Refreshing the initial POF</td>
<td>214</td>
</tr>
<tr>
<td>Generating POF reports</td>
<td>214</td>
</tr>
<tr>
<td>Reusing a POF in a subsequent installation</td>
<td>215</td>
</tr>
<tr>
<td>Overriding POF values in SLIBs</td>
<td>216</td>
</tr>
<tr>
<td>Creating OUTPUT or TEMPLATE descriptors for copies</td>
<td>218</td>
</tr>
</tbody>
</table>

**Glossary** 221
About this book

This book contains detailed information about the associated product or products. This preface explains the special conventions that the book uses, and how to access related publications.

If applicable, the preface also summarizes the major changes included in the latest release of the product.

Related publications

From the BMC Support Central website, you can use the following methods to access related publications that support your product or solution:


- View Quick Course videos (short overviews of selected product concepts, tasks, or features), which are available from the following locations:
  
  — Documentation Center (primary center and secured center)
  
  — Support Central (at http://www.bmc.com/support/mainframe-demonstrations)
  
  — BMC Mainframe YouTube channel (https://www.youtube.com/user/BMCSoftwareMainframe)


Products with online interfaces also offer online Help via the F1 key or, for graphical user interfaces (GUIs), via a Help button.
If you prefer hardcopy documentation, you can order it from your BMC sales representative or from Support Central. Also, from Support Central you can subscribe to receive proactive e-mail alerts when BMC issues notices.

Conventions

This document uses the following special conventions:

- All syntax, operating system terms, and literal examples are presented in this typeface.

- Variable text in path names, system messages, or syntax is displayed in italic text:
  testsys/instance/fileName

- Menu sequences use a symbol to convey the sequence. For example, Actions => Create Test instructs you to choose the Create Test command from the Actions menu.

Syntax statements

This topic explains conventions for showing syntax statements.

A sample statement follows:

```
COMMAND KEYWORD1 [KEYWORD2 | KEYWORD3] KEYWORD4={YES | NO} fileName...
```

The following table explains conventions for syntax statements and provides examples:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Example</th>
</tr>
</thead>
</table>
| Items in italic type represent variables that you must replace with a name or value. If a variable is represented by two or more words, initial capitals distinguish the second and subsequent words. | alias  
databaseDirectory  
serverHostName |
Convention | Example
---|---
Brackets indicate optional items. Do not type the brackets when you enter the option. A comma means that you can choose one or more of the listed options. You must use a comma to separate the options if you choose more than one option. | `[tableName, columnName, field] [-full, -incremental, -level]`

Braces indicate that at least one of the enclosed items is required. Do not type the braces when you enter the item. | `{DBDName | tableName}
UNLOAD device={disk | tape, fileName | deviceName}
{a | c}`

A vertical bar means that you can choose only one of the listed items. In the example, you would choose either commit or cancel. | `{commit | cancel}`

An ellipsis indicates that you can repeat the previous item or items as many times as necessary. | `columnName...`

---

**Summary of changes**

This topic summarizes product changes and enhancements by version number and release date.

**Version 12.1.00, December 2016**

This release fixes known problems and includes the following enhancements:

- Support for DB2 Version 12.1 features
  - Support for APPLCOMPAT V12R1Mnnn in native stored procedures
  - Support for LOB compression
    DB2 Version 12 lets you compress LOB table spaces where the base table is in a universal table space. CHANGE MANAGER supports this feature, and lets you specify whether a LOB is compressed in the Create Auxiliary Tablespace and Auxiliary Tablespace Detail panels.
  - Support for BUSINESS_TIME enhancement
    IBM DB2 Version 12 supports the INCLUSIVE and EXCLUSIVE option for the BUSINESS_TIME parameter for ALTER TABLE and CREATE TABLE statements. ALTER and CHANGE MANAGER support this option.
Support for online index compression

ALTER and CHANGE MANAGER support the changed behavior of DB2 when you alter the compression value of an index on a universal table space. DB2 Version 12 places the index into Advisory REORG pending (AREOR) status.

PREVENT_NEW_IXCTRL_PART ZPARM

The Analysis component of CHANGE MANAGER now verifies the PREVENT_NEW_IXCTRL_PART ZPARM when you need to LIKE, migrate, or rebuild an index-controlled partitioned table space, or when you need to include the table space in a full-recovery baseline.

ALTER and CHANGE MANAGER support the new value of ALWAYS_PENDING for the DDL_MATERIALIZATION ZPARM.

Version 12.1 of ALTER and CHANGE MANAGER tolerate the following features of IBM DB2 Version 12.

The extension of range-partitioned table spaces with the PAGENUM RELATIVE attribute of the CREATE TABLESPACE and ALTER TABLESPACE statements

Enhanced triggers

The TRANSFER OWNERSHIP command

Foreign keys with the PERIOD BUSINESS_TIME clause

Inserting partitions into a range-partitioned table space

INSERT ALGORITHM clause in CREATE TABLESPACE and ALTER TABLESPACE statements

Unicode columns in an EBCDIC table

Automatic change rules for stored procedures

CHANGE MANAGER can now apply an automatic change rule for the version number of a stored procedure. This enables you to specify stored procedures in a DB2 catalog to DB2 catalog comparison.

Support for spanned records

CHANGE MANAGER can now unload and load existing LOB and XML column data by using spanned records.

Support for NGT Reorg

You can now specify NGT Reorg to reorganize objects on DB2 Version 10 and 11 subsystems.
- Support for PLANMGMT
  This release adds support for the PLANMGMT rebind option. You can now specify the PLANMGMT value that ALTER and CHANGE MANAGER use during a rebind operation.

- Tolerance of IDAA tables
  The products recognize when a table has been added to IDAA and do not change the definition of the table.

- Deprecation of the LOB DATA MOVER program
  The LOB DATA MOVER program is no longer required to move LOB data, and has been deprecated. This was a feature of the BMC Database Administration for DB2 solution.

- RECOVER PLUS for DB2 and COPY PLUS for DB2 name change
  Starting with this release, the name of the RECOVER PLUS for DB2 product has changed to the BMC Next Generation Technology Recover for DB2 for z/OS (or NGT Recover) product. The name of the COPY PLUS for DB2 product has changed to the BMC Next Generation Technology Copy for DB2 for z/OS (or NGT Copy) product.

- Support for IBM DB2
  ALTER and CHANGE MANAGER now support the following versions and modes of DB2:
  
  - DB2 V12 Function level 100
  - DB2 V12 Function level 500
  - DB2 V11 Conversion mode (CM)
  - DB2 V11 Enabling-new-function mode (ENFM)
  - DB2 V11 New-function mode (NFM)
  - DB2 V10 NFM
Overview of ALTER and CHANGE MANAGER

As DB2 applications become more structurally complex and more mission critical, the need to add and modify data structures, as well as the need for change management, increases significantly.

In addition, the growing complexity of the DB2 environment has made the change process itself more difficult and costly. Without an effective change management tool, database administrators (DBAs) find that data structure changes and change management is complex, tedious, resource intensive, and error prone.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Getting Started."

Change management

Every organization, every environment, and every application is unique. Before you can use change management tools to help you manage changes to your data structures and your data, you might need to consider the following questions:

- What does my environment look like?
- Where are my environments located?
- How are my changes propagated?
- When are my changes propagated?
Identification of your environment

Most organizations use at least three environments for each of their applications: development, test, and production.

What does your environment look like? Perhaps you maintain your application code and perform unit tests in a development system. Your test system might be used to perform system and stress tests, as well as simulate production. Your production system might be in single or multiple locations.

Location of your environment

Sometimes multiple environments are located on the same DB2 subsystem.

The same DB2 subsystem can be used for multiple environments if you use different database names and you have different owners. Where are your environments located? Are they located on the same subsystem? Do they share DASD? Are they located in different cities or in different parts of the world?

Propagation of your changes

In some organizations, multiple teams are responsible for developing or maintaining an application in a development environment.

If the changes to the data structures are needed by each team, you might need to synchronize the data structures in the development environment before you migrate the entire structure to the test environment. How will you propagate these changes? Do you manage your change requests based on release cycles or by date?

One way to manage your development, test, and production environments is as follows:

- If you need to create an environment, you can migrate your data structures and data.
- If the environment already exists, you can compare one environment to another to determine what changes need to be made to synchronize the data structures in both of the environments.
- You can create, alter, or drop data structures in the environment, or simply maintain the environment.
For example, you could create your development environment based on your production environment or synchronize your development environment with your production environment. To create your development environment, you could migrate your production environment. To synchronize your development environment with your production environment, you could compare your development environment to the production environment. After your development environment is established, you could modify it or maintain it.

**Overview of change management with ALTER and CHANGE MANAGER**

BMC provides two products to help you manage data structures and data for DB2 subsystems: the ALTER and CHANGE MANAGER products.

**Overview of ALTER**

ALTER provides a powerful solution to the problems of managing a DB2 environment.
By automating and simplifying the change process, ALTER enables you to deal effectively with the demands of a constantly changing DB2 environment. ALTER uses an Interactive System Productivity Facility (ISPF) interface.

ALTER is designed for the single DB2 system that is beginning to encounter complex challenges. As the need for more change management functions develops, as with multiple DB2 subsystems, you can upgrade from ALTER to CHANGE MANAGER.

ALTER enables you to automate the following common tasks:

- Create, modify, and drop data structures within a DB2 subsystem
- Create new data structures by using existing data structures as templates
- Determine the dependencies on changed objects and preserve those dependencies and their associated data
- Migrate data structures and data within a subsystem or from one DB2 subsystem to another subsystem
- Analyze the impact of changes by validating the changes against the DB2 catalog. If conflicts are detected, ALTER notifies you through error messages and warnings.
- Implement a least-cost strategy for performing changes
  You can group your changes to minimize the work that is required to execute them.
- Import data definition language (DDL) files and create the objects
- Perform space estimation for table spaces and indexes
- Provide passive management of authorizations, plans, and packages
- Provide automatic data conversion for changes to column attributes
- Handle requirements for data set allocations
- Specify global structure changes when migrating objects

ALTER provides full management support for the following DB2 data structures:

- Storage groups (stogroups)
- Databases
- Table spaces
When you use ALTER to specify changes for any of these data structures, the product automatically propagates the changes to any dependent objects. For example, if you change the name of a table, ALTER creates a corresponding change in the indexes, synonyms, and other dependent objects that reference the table under its former name.

ALTER uses BMC or IBM utilities in worklists when required. By using installed BMC utilities instead of IBM utilities, you can significantly enhance the performance of executing ALTER worklists. BMC utilities run faster, provide additional features, and might reduce the number of steps in a worklist.

You can use the following BMC utilities with ALTER:

- BMCSTATS utility of DASD MANAGER PLUS for DB2
- BMC Next Generation Technology Copy for DB2 for z/OS
- CHECK PLUS for DB2
- LOADPLUS for DB2
- BMC Next Generation Technology Recover for DB2 for z/OS
- REORG PLUS for DB2
- UNLOAD PLUS for DB2

The utilities must support the same version of DB2 as ALTER or CHANGE MANAGER. For more information, see the utility product’s reference manual.
Data structure changes

Within a DB2 subsystem, an application’s data structures need continual modification and extension because of changes in application requirements, performance tuning, and expanding uses of data.

ALTER enables you to make changes to data structures in an easy, reliable, and automated way. You can efficiently perform changes while preserving data and, if necessary, convert the data to match the changes. ALTER also enables you to rebuild dependent objects and then propagate changes into those dependent objects. Without ALTER, these tasks can be extremely difficult to perform.

WARNING

BMC does not recommend using ALTER to modify the DB2 catalog, directory, and workfile objects.

Data structure migration

If you have multiple DB2 subsystems, you might need to copy data structures from one subsystem to another.

Most DB2 sites have separate subsystems for development, testing, and production. Many sites support multiple production systems, some of which are local and some remote.

ALTER offers the following functionality to simplify data structure migration:

- Generates SQL CREATE statements from the sending subsystem for use on the receiving subsystem
- Applies CREATE statements on the receiving subsystem
- Moves data from the sending subsystem to the receiving subsystem

When you migrate an application’s data structures, ALTER assumes that these data structures do not yet exist on the receiving subsystem. ALTER cannot create new data structures if data structures with the same names already exist on the receiving subsystems.

Data structure importation

You use ALTER to process DDL files that are received from an external subsystem and apply the changes to your subsystem.
The Import component converts these data structure definitions into change requests that are grouped together in a work ID. ALTER stores these change requests in the Change Definition (CD) tables.

**Space estimation**

ALTER provides space estimation capability that enables you to determine the amount of space that a table space or index will require based on the object definitions and their estimated usages.

You can specify estimates of the number of rows in the table, and the average length of the rows. ALTER will use this information to project the number of tracks, cylinders, or blocks that are required to allocate the data set for that object.

---

**Note**

You cannot resize the DB2 catalog or directory tables.

---

**Overview of CHANGE MANAGER**

CHANGE MANAGER enables the DBA to effectively deal with the demands of a constantly changing environment involving multiple DB2 subsystems. Like ALTER, CHANGE MANAGER uses an ISPF interface.

CHANGE MANAGER includes all of the features in ALTER and provides the following additional capabilities:

- Migrates data structure changes across multiple databases and subsystems
- Determines changes to data structures and migrates those changes to one or more copies of the data structures
- Captures and records structure definitions and data within a DB2 subsystem at a point in time (to establish a full-recovery baseline)
- Recovers structures and data to a point in time defined by a full-recovery baseline within a DB2 subsystem
- Compares two versions of structure definitions to:
  - Determine the changes necessary to upgrade one version to another
  - Selectively apply changes to copies of the data structures while preserving the uniqueness of each copy
- Uses data modeling tool outputs to determine the changes to existing DB2 application structures
- Reduces the volume of information that is needed to communicate changes by using a BMC language called Change Definition Language (CDL) to transmit the change information
- Feeds changes that are made on a remote system back to the development system
- Defines and stores reusable rules for making changes
- Uses SQL-like statements to update, delete, and migrate data structures

CHANGE MANAGER provides full management support for the same DB2 data structures as ALTER does.

**Data structure change migration**

*Change migration*, which means to migrate only data structure changes (instead of entire structures) to another copy of the structure on the same DB2 subsystem or on a different DB2 subsystem, enables you to update structures that have already been migrated to another copy.

However, you must also be able to retain any structure modifications that were made locally after the structures were migrated. DB2 users often make local modifications—or variations from the control definition of the application—to meet the following needs:

- **Performance and usage tuning**
  
  Local tuning is a complex process that is based on the specific system that is involved and the performance and use of the application in that environment. Because separate sites often have different amounts of data and different transaction loads, as well as different CPU and DB2 environments, local tuning of application structures is common.

- **Security**
  
  Authorization requirements are associated with object ownership and, in some cases, with object names. As a result, the authorizations for one DB2 subsystem can vary significantly from the authorizations of another subsystem running the same application. In this situation, each subsystem requires local modifications to the application.

- **Additional uses of data**
  
  Often, production systems in a DB2 environment need to use data that is beyond the fixed scope of an application. For example, the data in a payroll application might be needed for reports that the finance department develops. DB2 enables
users to locally generate additional data structures (such as indexes, synonyms, views, aliases, and authorizations) to meet these needs.

Because local modification of applications is so common in DB2, users need an application management strategy that enables them to manage basic elements of the application globally, without compromising the elements that vary locally.

CHANGE MANAGER satisfies the requirement of migrating global changes while retaining local modifications by:

- Determining the changes made to the control version of the application
- Applying only those changes to other versions while preserving all local modifications

CHANGE MANAGER can also determine the accuracy of the proposed changes before applying them. Because changes are allowed at the lowest level of an object’s definition, CHANGE MANAGER preserves and converts data as necessary to accommodate the requested changes.

Additionally, CHANGE MANAGER propagates changes to all dependent structures. For example, when you change a column name, CHANGE MANAGER propagates the change to any index and foreign key definitions that use the column, even though the index and foreign key definitions might be unknown to the sending DB2 subsystem.

---

**WARNING**

BMC does not recommend using CHANGE MANAGER or the CM/PILOT component of CHANGE MANAGER to modify the DB2 catalog, directory, and workfile objects.

---

**Data structure recovery**

Changes to data structures might not produce the effect that you intended.

For example, a change might significantly increase an application’s response time. In this case, you might need to restore the previous data structures. While DB2 has features for logging, backup, and recovery of data, it has no similar features for data structures. If you make an unwanted structure change, or if your changes fail or are unusable, you must fall back to the previous definition. To do so, you must have saved the structure definitions and data of the previous version.

CHANGE MANAGER enables you to capture a set of data structures from the DB2 catalog or DDL file and store the set in a baseline. A baseline can contain only data structures or data structures with their associated data.
Establishing baselines and recovering to them is called data structure recovery. You can reload the data that is stored in a full-recovery baseline after the structures have been recovered. You can also reload the data that exists in those structures at the time of their recovery. In addition, CHANGE MANAGER can convert current data to match the restored data structures.

**Change recording and controlling**

Managing change also requires that you know what changes have been made to an application’s data structures during different periods of time.

CHANGE MANAGER provides a comparison feature that can generate a file that shows the differences between two sets of data structures. Because the file is stored in a sequential data set or partitioned data set (PDS) member, you can view, edit, print, or store it.

**Change feedback**

Data structure changes might not flow in order from the development system, through testing, and into production.

Because changes to the basic application definition can occur at any point in the cycle, you must be able to transmit changes back to the development system or to the control node. Transmitting changes from a remote system to the development system (or to the control node) is called change feedback.

To feed back changes, CHANGE MANAGER enables you to:

- Identify the changes made
- Apply the changes at the development or control system

You do not need to feed back all of the changes to the development system. You should feed back only those changes that are part of the globally managed basic definition of the application.

**Data modeling tools**

In many installations, the data structure design of the application is not controlled in a DB2 subsystem but in a data modeling tool (or Computer-Aided Software Engineering, or CASE, tool), data dictionary system, or other structure design repository.

In this guide, the term data modeling tool refers generically to any external design tool.
Most data modeling tools generate an application’s DDL. Changing an application’s data structures generates new DDL, which you can use to create the data structures in the DB2 catalog. Although data modeling tools can improve the productivity of DB2 developers, the use of these tools creates the following problems:

- If you make an emergency change in a production system, including that change in the data modeling design is difficult and usually requires a manual entry. You can use host-based DB2 tools to make emergency changes (or to define local structure elements not supported by a data modeling tool), but the changed structures will no longer be synchronized with the data modeling design.

- Often, communication between a data modeling tool and DB2 is one way. You can design structures on the data modeling tool and transmit the structures to DB2, but updating the data modeling design after making changes at the DB2 node is difficult.

- Most data modeling tools cannot selectively modify data structures. Because these tools can transmit only a complete application definition, they might be useful only for the initial design of an application. Maintenance and upgrades are then performed at the DB2 level.

- Due to their ease of use and built-in design intelligence, data modeling tools are popular for managing global application data structure definitions. However, the external DB2 specifications that a data modeling tool produces might be incomplete or might fail to reflect local elements of the design.

You can address these problems by using CHANGE MANAGER to migrate changes between the data modeling definition of an application and the DB2 definition. CHANGE MANAGER enables you to:

- Establish control points for the application’s data structures and determine the changes that are made between those control points

- Implement design changes in DB2 without losing either the application’s data or its local data structure modifications

- Transmit changes from DB2 back to the data modeling tool after making global changes to the DB2 version of the application’s data structures

**ALTER architecture**

ALTER provides management of data structures and data for a DB2 subsystem.

The product consists of several components that allow you to create, modify, and drop DB2 data structures while preserving data and dependent objects.
Figure 1 on page 26 illustrates the architecture of the ALTER product. The components and the objects of ALTER are discussed in the sections that follow.

Figure 1: ALTER architecture
Although not depicted in Figure 1 on page 26, the Front End component interacts with other ALTER objects and components. For more information, see “ALTER Front End” on page 27.

ALTER components

Most of the components that ALTER uses to perform its tasks can run in the foreground or as batch jobs.

ALTER consists of the following components:

- Front End
- Specification
- Import (DDL only)
- Analysis
- JCL Generation
- Execution

ALTER Front End

The Front End component is a dialog panel that you use to access and control ALTER.

From the Front End component, you can create, analyze, and execute work IDs, and build JCL in foreground or batch mode.

ALTER Specification

The Specification component enables you to create or edit data structure change or migration requests.

The component stores the change or migration requests in a work ID. With Specification, you can perform the following tasks:

- Create data structures
- Change data structures
Delete data structures

Migrate data structures

Estimate the space required for a table space or index

The Specification component lists objects from the DB2 catalog (or from a copy of that catalog if your site uses catalog indirection). You can change these objects, delete them, or create new objects either from the list of objects or from the Command line of some Specification panels. Changes are not made to the data structures until you execute the change and migration requests.

**ALTER Import**

You can import data structure definitions that are stored in a DDL file.

The Import component converts these data structures into change requests in a work ID. The Import component provides a batch function similar to that provided by the Specification component with limited validation checking.

**ALTER Analysis**

After you create change or migration requests in a work ID, the Analysis component checks the requests for validity with the DB2 catalog, develops an optimal implementation strategy, and generates a worklist.

The worklist contains the DB2 utility commands, AMS commands, security commands, and SQL statements that are necessary for implementing the requests. In addition, the Analysis component propagates changes to dependent structures. The Analysis component can be run in the foreground or in batch using the JCL Generation component.

The Analysis component performs the following validity and compliance checks:

- Verifies that the data structures of changing objects have not changed in the DB2 catalog since the present changes were requested
- Checks the change requests against other requested changes and warns of any conflicts
- Checks the change requests for compliance with DB2 rules and reports conflicts
- Develops the most efficient strategy to implement the requested changes or migrations
- Locates all dependent structures
Binds or rebinds plans and packages

Migrates plan and package authorizations

Checks for objects that are indirectly affected by requested changes

**Note**
The Analysis component warns you if a drop or alter change request will result in the loss of objects, and also warns you of views and plans that might require modifications.

If no errors are found, the Analysis component generates a worklist, stores the worklist or associated data set name with the work ID, and uses it as the default worklist when you run the Execution component. The Analysis component also creates diagnostic output in ALUPRINT and provides authorization switching.

**ALTER JCL Generation**

The JCL Generation component works with the Front End component to construct a job control language (JCL) file for running the Analysis and Import components in batch.

Using symbolic variables, the Front End component resolves all data set names that appear on the interface panels of the components. These names include the names of input files, JCL files, and the diagnostic output files.

**Execution functions**

The Execution component performs the commands that are contained in the worklist that the product generates for change or migrate requests.

Although the tasks that are required to implement a worklist vary depending on the complexity of the actions involved, the Execution component performs the following basic functions:

- Performs worklist control-command processing for DASD MANAGER PLUS
- Unloads DB2 data
- Runs IBM or BMC utilities
- Executes DB2 commands and Access Method Services (AMS) statements
- Establishes synchronization (sync) points to accommodate restart, if needed
- Creates diagnostic output in AEXPRINT
Performs audit and recovery logging, if requested

Performs authorization switching to ensure that DB2 objects are created with the correct owner and creator

The Execution component is the only component that makes physical changes to DB2 data structures.

**ALTER objects**

ALTER uses the following objects to perform its tasks.

This section describes each of the following objects in detail:

- Work IDs
- Worklists
- Unload data sets
- DDL files
- Internal tables

**Work IDs**

Work IDs enable you to identify, control, and track a unit of work.

Work IDs contain change or migration requests that are stored in the CD tables. Creating or selecting a work ID is usually the first step in performing any type of change or migrate process.

ALTER uses the following types of work IDs:

- Alter-type
  
  You use an alter-type work ID to change or modify data structures, drop objects, or create new data structures.

- Migrate-type
  
  You use migrate-type work IDs to specify the data structure definitions that are to be migrated from one DB2 subsystem to another, or to duplicate structures within the same subsystem. Migrate-type work IDs do not modify existing data structures.

  Migrate-type work IDs can contain change rules to modify database object attributes and migrate options to specify the default dependencies to include in a
migration. The change rules compare the object type and attribute to those in the migrated data structures and apply the rule when a match is found.

The migrate process provides a way to create new data structures from data structures that already exist on a subsystem. You can create the new data structures on the same subsystem if the objects are renamed. You can rename the objects through the Specification component or with change rules in the work ID.

■ Receive-type

You use receive-type work IDs to receive a worklist that is generated by a migrate-type work ID that contains the migrated data structures and data from a different subsystem.

You can request any number of changes or migrations with a single work ID. You can enter these requests all in one session or across many sessions. You can also delete or modify requests if you make a mistake or change your mind. Any number of users can request changes or migrations with the same work ID.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Creating Work IDs."

**ALTER worklists**

The worklist data set contains all of the commands and SQL statements that are needed to execute the changes or migrations specified in a work ID.

The Analysis component generates the worklist commands in a specific order. You can edit the worklist, but changing the order of the commands might yield incorrect results.

The worklist can contain any of the following items:

■ SQL statements that perform the change or migration requests, including the dropping and re-creating of all affected objects and the restoring of dependent objects that are lost as a result of a change or that propagate the changes into other objects

■ DB2 and utility commands that implement requested changes and migrations, including:
  — Unloading data
  — Reorganizing table spaces and indexes
  — Reloading data
  — Collecting statistics
— Making image copies
— Rebuilding indexes
— Checking the data
— Rebinding application plans and packages (if an alter-type work ID)
— Binding plan and package authorizations (if a migrate-type work ID)
— Performing other authorizations

■ AMS commands for data set allocation and deletion that support VCAT-defined partitions

Unload data sets

ALTER uses unload data sets to store data while dropping and rebuilding DB2 objects.

ALTER DDL files

ALTER does not generate DDL files. ALTER can use DDL files to import data structures.

ALTER internal tables

ALTER uses the following types of internal tables to perform its tasks:

■ CD tables, which store change and migration specifications for each work ID
■ CM tables, which store work ID, change rule, and sync data

CHANGE MANAGER architecture

CHANGE MANAGER consists of all of the ALTER components, as well as additional components that allow you to perform tasks across multiple subsystems.
Figure 2 on page 33 illustrates the architecture of the CHANGE MANAGER product. The components and the objects of CHANGE MANAGER are discussed in the sections that follow.

Figure 2: CHANGE MANAGER components and objects
Note

The shaded icons represent the components and objects that are unique to CHANGE MANAGER. The CM/PILOT component of CHANGE MANAGER and the interaction of the Front End component with other CHANGE MANAGER objects and components is not depicted. For an illustration of the CM/PILOT component, see Figure 3 on page 34. For more information about the interaction of the components, see “ALTER components” on page 27.

Figure 3: CM/PILOT components and objects
Note

The shaded icons in Figure 3 on page 34 represent the components and objects that are unique to the CM/PILOT component of CHANGE MANAGER. CM/PILOT also uses some of the components and objects of CHANGE MANAGER. For an illustration of the CHANGE MANAGER architecture, see Figure 2 on page 33.

CHANGE MANAGER components

Most components that CHANGE MANAGER uses to perform its tasks can run in the foreground or as batch jobs.

CHANGE MANAGER uses the following components:

- Front End
- Specification
- Baseline
- Compare
- Import
- Analysis
- JCL Generation
- Execution
- CM/PILOT

The Specification and Analysis components provide the same functionality as they do in ALTER. For a description of these components, see “ALTER components” on page 27.

CHANGE MANAGER Front End

In CHANGE MANAGER, the Front End component can also maintain migrate and baseline profiles, and interact with the Baseline, Compare, and Import components.

Baseline

The Baseline component captures a set of DB2 structure definitions from either the DB2 catalog, a DDL file, or a migrate worklist at a specific point in time.
The captured set of data structure definitions is called a baseline. If the structures are defined in the DB2 catalog, a baseline can also capture the data and authorizations that are associated with those structures. A baseline of both data structures and data is called a full-recovery baseline.

Baselines act as control points during data structure management. Baselines establish a static set of data structures for an application version. If you make a change with unwanted results, you can restore the data structures. If you establish a full-recovery baseline, you can also restore the data. You can also restore the data structure back to a prior baseline and convert the current data to those structures.

You can also use baselines for version comparison. For example, when you first install an application, create a baseline. At a later time, perform a comparison between the baseline and the DB2 catalog. This comparison creates a CDL file that shows any modifications that have been made to the application’s data structures since the initial installation.

**Note**

A baseline is also a CHANGE MANAGER object.

**Compare**

The Compare component determines the differences between two sets of data structures and then generates a CDL file.

The generated CDL file contains all of the changes that the comparison found between the two sets of data structures. You can compare data structures that are stored in a DDL file, work ID, baseline, worklist, or DB2 catalog. The inputs to the Compare component are called the primary and secondary input sources. The CDL file contains the changes that, if applied to the primary input source, would produce the data structures of the secondary input source.

You can use the generated CDL file in any of the following ways:

- Process the file as a set of change requests for the current subsystem
- Save the file as a record of the changes made
- Import the file to a different subsystem in order to update a separate version of the data structures

CDL has advantages over DDL in automating the process of updating data structures. CDL allows more types of modifications to data structures and, unlike DDL, can retain local modifications to those structures.
**CHANGE MANAGER Import**

You can import data structures that are stored in a CDL, DDL, or DML file.

The Import component converts the data structures in the files into change requests in a work ID. CHANGE MANAGER stores these change requests in the CD tables. The Import component provides a batch function similar to that provided by the Specification component with limited validation checking.

**CHANGE MANAGER JCL Generation**

In CHANGE MANAGER, the JCL Generation component also generates JCL for the Compare, Baseline, Baseline Report, and CM/PILOT components.

**CHANGE MANAGER Execution**

When CHANGE MANAGER is used in the BMC Database Administration for DB2 or BMC Object Administration for DB2 solution, the Execution component can enable a worklist to execute in parallel.

That is, the component executes portions of the worklist concurrently. For information about BMC Database Administration for DB2, see “Database Administration for DB2” on page 46.

For information about BMC Object Administration for DB2, see “BMC Object Administration for DB2” on page 47.

**CM PILOT**

The CM/PILOT component automates DB2 change management tasks. With CM/PILOT, you do not need to decide which CHANGE MANAGER processes are required for a task or the sequence in which you need to complete them—CM/PILOT provides scripts to guide you through the process.

The CM/PILOT component enables you to choose from several predefined scripts or to create your own scripts. You can copy scripts, including those that CM/PILOT provides, and modify them to meet your needs. You can also edit, browse, delete, and perform other maintenance tasks on scripts.

CM/PILOT enables you to quickly and easily use the vast change management power of CHANGE MANAGER. By following the dialog panels that are provided in the CM/PILOT scripts or in the scripts that you create, you can specify, analyze, and execute an Analysis worklist to perform the following tasks:

- Change data structures
- Migrate data structures
- Migrate only data
- Receive data structure changes
- Receive DDL to create data structures
- Create full-recovery baselines
- Recover data structures with current or old data
- Select a user-defined script
- Replicate work IDs

When you want to perform a change management task, CM/PILOT determines the components that CHANGE MANAGER will use and ensures that they are run in the correct sequence. CM/PILOT enables you to create tasks that can be done later by someone else or through job scheduling. By reusing these tasks, you can ensure that the change management task is done the same way every time.

The CM/PILOT component enables you to easily prepare for various CHANGE MANAGER tasks that can be processed on demand. CM/PILOT dialog panels are task oriented and easy to use. The panels guide you through the CHANGE MANAGER components that are necessary to accomplish the specific tasks and prompt you for the information that is needed to build worklists that can be processed at any time.

CM/PILOT simplifies the decision-making process and enables you to prepare change management tasks that are performed repeatedly. It also lets you create change management tasks that can be performed by less-experienced DBAs. And with its unique script-locking feature, CM/PILOT lets you create script steps that cannot be modified and can only be run as you have specified.

You can also use two CM/PILOT scripts to help you create SQL-like Data Manipulation Language (DML) statements to update, delete, and migrate data structures. You can also import DML statements into an alter-type work ID. After the statements are imported, CHANGE MANAGER creates entries in the CD tables as if they had been requested in the Specification component.

**CHANGE MANAGER objects**

CHANGE MANAGER uses the following objects to perform its tasks.

This section describes each of the objects in detail.
The work ID and unload data set objects are the same objects as in ALTER. For a description of these objects, see “ALTER objects” on page 30.

**CHANGE MANAGER worklists**

When CHANGE MANAGER is used in the Database Administration or BMC Object Administration for DB2 solution, the worklist can also contain keywords that control the sequencing of parallel processes.

For information about the Database Administration solution, see “Database Administration for DB2” on page 46.

For information about BMC Object Administration for DB2, see “BMC Object Administration for DB2” on page 47.

**Baselines**

A baseline contains a set of data structures and their associated authorizations, taken at a specific point in time, from the DB2 catalog, a DDL file, or a worklist.

CHANGE MANAGER uses structure-only and full-recovery baselines:

- A structure-only baseline contains only data structures.
- A full-recovery baseline contains data structures and data.

The types of input to a baseline are as follows:
- **DB2 catalog**
  A catalog baseline takes its input from a subset of the DB2 catalog. See “Baseline profiles” on page 40.

- **DDL file**
  A DDL baseline takes its input from a file containing SQL DDL statements.

- **Worklist file**
  A worklist baseline takes its input from a worklist that CHANGE MANAGER or another BMC product produces. A worklist baseline is identical to a DDL baseline except for the input source file type. See “Baseline profiles” on page 40.

### Profiles

Profiles are a collection of scope rules, change rules, and locations that enable you to define and control the Analysis, Compare, and Baseline processes.

You use profiles to:

- Select a set of objects
- Customize changes to objects that are migrated to other subsystems
- Create baselines of application data structures

Profiles enable you to automate changes instead of selecting objects individually for migration or making individual changes to object structure definitions. You can create, copy, edit, and reuse profiles.

Two types of profiles are used in CHANGE MANAGER: baseline profiles and migrate profiles.

#### Baseline profiles

You use baseline profiles to create, name, and maintain baselines of application data structures.

You can automatically manage the baselines that you want to keep and those that you want to delete by specifying delete and retain values in the baseline profile. The two types of baseline profiles are catalog and DDL (or worklist) profiles.

- **Catalog baseline profiles**
  The profile for a catalog baseline must contain scope rules that select the DB2 objects to include in the baseline or comparison. The baseline profile can explicitly
specify the scope rules or can reference the scope rules in an outbound migrate profile.
Scope rules are used only when a process is being performed on the DB2 catalog. The scope rules serve to create a subset of the DB2 catalog.

- **DDL or worklist baseline profiles**
  - When defining a DDL baseline profile, you define only header information. The scope is implicitly defined by the set of objects in the DDL or worklist.

**Migrate profiles**

You use migrate profiles to select a set of objects and to customize changes to objects that are migrated to different locations.

The two kinds of migrate profiles are inbound and outbound.

- **Inbound Migrate Profiles**
  - The Import component uses inbound migrate profiles to make input from another system compatible with the receiving subsystem’s version of the application. Inbound migrate profiles contain only change rules that modify imported data structure definitions.
  - The change rules are applied as the CDL file or DDL file is imported. For example, if a change rule specifies changing the database name DEMO* to ACM*, the DDL statement CREATE DATABASE DEMOHRS is read and the CD table entries are generated with a CREATE DATABASE ACMHRS. The profile’s change rules are applied to the imported data structure definitions before CD table entries are created.
  - You cannot specify scope rules or locations with an inbound migrate profile. The file being imported identifies the scope.

- **Outbound Migrate Profiles**
  - You use outbound migrate profiles to generate worklists or CDL files. The outbound migrate profile can contain the following items:
    - A scope for specifying the DB2 objects to include in the operation
    - Locations for defining multiple clones of an application
    - Change rules for defining attribute changes to be automatically applied to objects in each clone
  - You can group locations if they reside in the same DB2 subsystem.
  - When an outbound migrate profile is used as input to Analysis, a worklist is generated for each group specified. One CDL output file is generated for each
group when an outbound migrate profile that specifies locations is used as an input to the comparison.

The Compare process can also use the scope rules of an outbound migrate profile to select a subset of objects from the DB2 catalog to use in a comparison with a DDL file or worklist.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Creating Migrate Profiles."

Scope rules

When you define scope rules for a profile, you are specifying which objects to fetch from the catalog.

You can reference scope rules when you perform a comparison, create a baseline, or analyze a work ID.

In a catalog baseline profile, scope rules specify the DB2 objects to include in the baseline. The scope rules of an outbound migrate profile can specify the objects to migrate or the DB2 objects to include in a comparison. Scope rules can include DB2 objects in the final set of objects to be processed or can exclude them from the final set.

Scope rules can select a specific object or select groups of objects (through the use of wildcard characters) as the set or sets of objects on which to operate. Scope rules can also determine which dependent objects to include and to exclude. For example, a scope rule could include database ABCDEF and its dependent table spaces, tables, indexes, foreign keys, and views, but exclude its dependent synonyms, aliases, or data.

You can define scope rules for both migrate profiles and catalog baseline profiles. You can follow the same procedure to create a set of scope rules for each type of profile. You can also use the same set of scope rules for different profiles. For example, when creating a catalog baseline profile, you can reference the scope rules of a migrate profile. Likewise, a migrate profile can reference the scope rules of a catalog baseline profile. This feature enables you to use the same list to select the same objects for either migration or a baseline.

The Compare component uses the scope rules of a catalog baseline profile when it compares a catalog baseline to the DB2 catalog. The Compare component finds the baseline profile by using the established baseline name, and then uses the scope rules of the catalog baseline profile to select the objects to compare. Compare can also use the scope rules of a specified profile when it compares a DB2 catalog to a DDL file or to a worklist.
**Change rules**

Change rules modify the attributes of existing DB2 object structure definitions.

The change rules compare the object type and attributes to those in the migrated data structures. The change rules are applied when the attributes and the names match. For ALTER, you can define change rules within a migrate-type work ID. For CHANGE MANAGER, you can define change rules within a migrate-type work ID or a migrate profile.

When you define change rules within a migrate profile, the rules are processed for each object that is defined in the scope of the profile. You can streamline and automate changes to sets of object structures by defining change rules for migrate profiles. Because migrate profiles are used in different types of migration processes, special consideration of how change rules work can significantly contribute to the efficient performance of these processes.

Change rules allow you to easily specify and repeat common changes. For example, in many DB2 installations, the owner of a table changes when the table is moved from a test system into production. A migrate profile can perform the changes that are necessary to migrate the structures from the test system into production. This profile can include a change rule that changes all of the tables with owners that match TEST to PROD.

Change rules can perform the following tasks:

- Change the names of objects
- Change the attributes of a specified object or group of objects
- Include specific volumes in a storage group, or exclude volumes from a storage group
- Exclude one or more columns from a table
- Specify that table spaces and indexes that have a STOGROUP attribute be defined to use VCAT on the receiving database
- (Compare only) Suppress changes to altered objects

**Locations for outbound migrate profiles**

Locations, which are defined only in outbound migrate profiles, enable you to create or maintain multiple clones of an application schema.

When an outbound migrate profile contains locations, a worklist or CDL is generated for each location or DB2 subsystem that is specified.
For example, if your installation supports systems in New York, Dallas, and Atlanta, you can create a profile with a location for each system. Each location can have change rules that tailor object names and attributes for that system. Each location can have its own change rules, or can reference the change rules of another location within the profile.

You can also group several locations under a single Group ID to generate just one worklist or one CDL file. Using the Group ID enables you to execute all of the work for a single subsystem in one process. This action is useful when you are migrating or updating several clones of a structure that reside on the same subsystem.

For example, if a development facility has 25 developers, each of whom have their own copy of the application schema, you could use a Group ID to migrate all 25 copies of the schema within a single worklist.

**CDL files**

CDL is a BMC proprietary language that you use to specify changes to DB2 data structures. You can use CDL files to transmit data structure changes between subsystems or to provide a record of changes to data structures.

CDL files specify the changes that are required to transform a primary set of data structures to a secondary set of data structures. You use CDL files in the change, import, and recovery processes.

**CHANGE MANAGER DDL files**

CHANGE MANAGER can generate executable SQL DDL statements (in SPUFI format) when generating a baseline report.

DDL files are used as input by the Import, Baseline, and Compare components.

**CHANGE MANAGER internal tables**

CHANGE MANAGER uses several types of internal tables to perform tasks.

The tables that CHANGE MANAGER uses are as follows:

- BL tables, which store information about baselines (data for full-recovery baselines is stored in external data sets)
- CD tables, which store change and migration specifications for each work ID
- CM tables, which store work ID, change rule, and sync data
CP tables, which store information about CM/PILOT task IDs, applications, scripts, and DML.

Access to most of the internal tables is controlled by the granting of EXECUTE authority on the appropriate plans.

**Task IDs**

A task ID is a unit of work in the CM/PILOT component.

Each task ID has a unique name and contains information that you provide through the panel to perform a CHANGE MANAGER process. The information that you provide is based on the script you select for the task ID. All work in CM/PILOT is accomplished by processing task IDs.

**Scripts**

The panels that the CM/PILOT component displays are based on predefined steps called scripts.

Each script contains ordered steps that prompt you for the information that is required to perform a change management task. CHANGE MANAGER provides you with several scripts; you can also create your own script, or copy, change, and rename any of the scripts that are supplied by CHANGE MANAGER.

**Applications**

In the CM/PILOT component, you can associate a group of CHANGE MANAGER profiles that are used repeatedly for the change management tasks of a specific DB2 application.

This association is called an application.

**DML**

In the CM/PILOT component, you can use DML statements to update, delete, and migrate data structures.

You can change or migrate multiple objects in a single DML statement. The panels for two of the CM/PILOT scripts help you create the DML statements.

**CM/PILOT worklists**

A CM/PILOT worklist is a data set that contains the ordered commands, keywords, and parameters that are needed to process a task ID.
The format of the CM/PILOT worklist is similar to an Analysis worklist. The JCL that processes the CM/PILOT worklist contains the names of the output data sets that an Analysis worklist uses and the associated Execution JCL. When a CM/PILOT worklist is successfully processed, the result is the creation of an Analysis worklist and the Execution JCL to process the worklist.

**Solution integration**

ALTER and CHANGE MANAGER are also components of BMC solutions.

ALTER is a component of the Administrative Assistant solution. CHANGE MANAGER is a component of the BMC Database Administration for DB2, BMC Next Generation Technology Database Administration for DB2, and BMC Object Administration for DB2 solutions.

**Administrative Assistant for DB2**

The Administrative Assistant solution enables users of all experience levels to navigate through the DB2 catalog quickly and to manage a complex DB2 environment easily.

For more information, see these documents:

- BMC Administrative Assistant for DB2 release notes
- *Installation System Reference Manual*
- *Installation System Quick Start*
- *BMC Products and Solutions for DB2 Customization Guide*

**Database Administration for DB2**

You can use the Database Administration solution to manage your DB2 databases quickly, efficiently, and effectively.

For more information, see these documents:

- BMC Database Administration for DB2 release notes
- *Installation System Reference Manual*
You can use the BMC Object Administration for DB2 solution to manage your DB2 databases quickly, efficiently, and effectively.

For more information, see these documents:

- BMC Object Administration for DB2 release notes
- Installation System Reference Manual
- Installation System Quick Start
- BMC Products and Solutions for DB2 Customization Guide

BMC Next Generation Database Administration for DB2

You can use the BMC Next Generation Technology Database Administration for DB2 solution to manage your DB2 databases quickly, efficiently, and effectively.
Overview of the ALTER and CHANGE MANAGER interface

ALTER and CHANGE MANAGER are designed to help you and your organization manage data structures within a DB2 subsystem or between subsystems.

This section introduces you to the interface for ALTER and CHANGE MANAGER, then steps you through the tasks that are necessary to create and migrate DB2 objects.

Before you use the product

Before you use ALTER and CHANGE MANAGER, ensure that you have completed the tasks necessary to install the products.

For information, see:

■ *Installation System Reference Manual*

■ *Installation System Quick Start*

■ *BMC Products and Solutions for DB2 Customization Guide*

ALTER and CHANGE MANAGER interface

The ALTER and CHANGE MANAGER ISPF user interface consists of dialog panels that comply with Common User Access (CUA) conventions.

To call or start ALTER or CHANGE MANAGER, you can use the BMC Software Administrative Products for DB2 panel that BMC provides (see *Figure 4 on page 50*).
Note

If you have CHANGE MANAGER installed, ALTER is not an option on this panel.

Figure 4: Administrative products for DB2 panel

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CATALOG MANAGER for DB2 - Execute DDL or query DB2 Catalog</td>
</tr>
<tr>
<td>2</td>
<td>CHANGE MANAGER for DB2 - Manage changes to DB2 objects/structures</td>
</tr>
<tr>
<td>3</td>
<td>DASD MANAGER for DB2 - Manage or monitor DB2 physical objects</td>
</tr>
</tbody>
</table>

DB2 SSID . . . . . . . DEGA (? = SSID List)
DB2 Catalog Access . . DIRECT (Direct)
Use Shared or Individual product ISPF APPLID? S (S/I - Admin Products only)

The BMC Software Administrative Products for DB2 panel contains the following features:

- The COMMAND line, on which you enter TSO commands
- The product selection list, which includes an input field with which you can specify an option from the list by entering a number
- The DB2 SSID field, in which you specify the DB2 subsystem (SSID) to be used
- The DB2 Catalog Access field, in which you specify whether to use the DB2 catalog data directly or to use a copy or a view of the DB2 catalog
- The Use Shared or Individual product ISPF APPLID field, in which you specify whether to use a shared or individual ISPF profile ID

If you want to specify an ISPF Application ID (APPLID) for each product, use an individual APPLID. If you want to specify an APPLID for each DB2 subsystem, use a shared APPLID.

Main Menu

After you select ALTER or CHANGE MANAGER from the Administrative Products for DB2 panel, the product’s Main Menu is displayed.

You can use the ALTER Main Menu to select options to process work IDs, change installation options, perform specific tasks, and view a list of the latest enhancements to the product.
You can use the CHANGE MANAGER Main Menu (see Figure 5 on page 51) to perform the same functions as ALTER, in addition to building CDL, processing baselines, using baseline and migrate profiles, and accessing the CM/PILOT component of CHANGE MANAGER.

Figure 5: CHANGE MANAGER Main Menu

Select an option. Then press Enter.
- 1. WORKID (Import, Specify, Analyze, Execute)
- 2. Compare (CDL Build)
- 3. Baselines
- 4. Baseline Profiles
- 5. Migrate Profiles
- 6. Environment
- 7. Options
- 8. Task List
- 9. CM/PILOT

? About This Release

Tip: To view a long name value, position the cursor on the value and press PF4 (ZOOM).

Commands: HELP END
(c) Copyright 1992-2017 BMC Software, Inc

Displaying the current product environment

From the Main Menu of ALTER or CHANGE MANAGER, you can list helpful information about the product, data set names, plan names, and applied fixes.

To display information about the environment

1. To display information about the current ALTER or CHANGE MANAGER environment, type ENVI in the Command line on the Main Menu and press Enter.

The BMC Environment panel is displayed, as shown in Figure 6 on page 51.

Figure 6: BMC Environment panel
To obtain a list of the synonyms and tables that are dependent upon each plan, type **SY** in the **Act** column adjacent to the name of the plan.

To display information about fixes (PTFs) that you have applied to the product, type **MAINT** in the **Command** line on the BMC Environment panel and press **Enter**.

The Product Maintenance List panel is displayed.

**Using Fast Path Navigation**

The Installation System for the Administrative products provides a feature called *Fast Path Navigation*. This feature enables you to switch from one product to another, and then return to the original product.

**To initiate Fast Path Navigation**

1. On the Command line of the current product, enter the command corresponding to the product to which you want to switch (see Table 1 on page 52).

**Table 1: Fast Path Navigation Commands**

<table>
<thead>
<tr>
<th>Product</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER</td>
<td>BMACALTER</td>
</tr>
<tr>
<td>CHANGE MANAGER</td>
<td>BMACHG</td>
</tr>
<tr>
<td>CATALOG MANAGER</td>
<td>BMACAT</td>
</tr>
<tr>
<td>DASD MANAGER PLUS for DB2</td>
<td>BMACDASD</td>
</tr>
</tbody>
</table>

For example, if you are currently using ALTER and want to view an object description in CATALOG MANAGER, enter BMACAT on the ALTER Command line. The main menu for the requested product is displayed. In this case, the ALTER session is temporarily suspended and then resumed when you exit CATALOG MANAGER.
For more information about enabling Fast Path Navigation, see:

- Installation System Reference Manual
- Installation System Quick Start
- BMC Products and Solutions for DB2 Customization Guide

**ISPF commands**

Most of the ISPF commands in the product operate in the same manner as they do in other ISPF applications.

Table 2 on page 53 describes the most commonly used ISPF commands.

Table 2: ISPF commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CANCEL</td>
<td>Returns to the previous panel without saving any change that you made on the current panel</td>
</tr>
</tbody>
</table>
| DOWN (or F8)| Scrolls the panel down
More: + on a panel indicates that more information is available below the current line.
Scrolling is available on a Model 2 3270 mainframe terminal, which uses a 24-line by 80-column display. |
| END (or F3) | Validates and processes information, the same as the Enter key
In some panels, pressing END returns to the previous panel. |
| ENTER       | Processes information that is typed on the panel and executes any specified commands
For a sequence of related panels, pressing Enter validates the information on the current panel and displays the next panel in the sequence. |
| HELP (or F1)| Provides panel-level Help |
| LEFT (or F10)| Scrolls the panel to the left
More: < on a panel indicates that more information is available to the left. |
| NEXT        | Displays the next panel in a sequence |
| PFSHOW      | Displays the active function keys
Some panels use every available line to display input variables. To display all variables, enter PFSHOW OFF on the Command line. |
<p>| PREV (or F12)| Displays the previous panel in a sequence |</p>
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RIGHT (or F11)</td>
<td>Scrolls the panel to the right</td>
</tr>
<tr>
<td></td>
<td><strong>More:</strong> &gt; on a panel indicates that more information is available to the right.</td>
</tr>
<tr>
<td>SPLIT (or F2)</td>
<td>Divides the panel and displays the ISPF Primary Option Menu in the new panel</td>
</tr>
<tr>
<td></td>
<td>If you start the product on both panels, ensure that each product is at the same version, release, and maintenance level.</td>
</tr>
<tr>
<td>SSE</td>
<td>Starts the BMC Simple Space Estimation (SSE) feature to estimate space requirements for table space or index objects</td>
</tr>
<tr>
<td>SWAP (or F9)</td>
<td>Switches from one split panel to another</td>
</tr>
<tr>
<td>UP (or F7)</td>
<td>Scrolls the panel up</td>
</tr>
<tr>
<td></td>
<td><strong>More:</strong> - on a panel indicates that more information is available above the current line.</td>
</tr>
<tr>
<td></td>
<td>Scrolling is available on a Model 2 3270 mainframe terminal, which uses a 24-line by 80-column display.</td>
</tr>
<tr>
<td>ZOOM (or F4)</td>
<td>Displays the full value of an object with a long name in either a dialog or a panel</td>
</tr>
<tr>
<td></td>
<td>The cursor must be positioned on the long name value.</td>
</tr>
<tr>
<td></td>
<td>On the following panels, you cannot use the <strong>ZOOM</strong> key to display the entire name of an object:</td>
</tr>
<tr>
<td></td>
<td>- Scope Rules panel (ACMFMPSSL)</td>
</tr>
<tr>
<td></td>
<td>- Change Rules panel (ALUFRULL)</td>
</tr>
<tr>
<td></td>
<td>- Help panels</td>
</tr>
<tr>
<td></td>
<td>For example, when you convert a table space from table-controlled partitioning to index-controlled partitioning, the products might display a Help panel that lists up to three partitioned indexes. If those indexes have long names, the product displays a portion of the names and disables the <strong>ZOOM</strong> key.</td>
</tr>
</tbody>
</table>

**BMC message information**

The BMC Administrative products include informational, warning, and error messages.

Most messages are assigned a message identifier (**messageID**) that consists of an alphabetic prefix and a number. Some message identifiers also include a severity code, such as W, E, or I.
Achieving your goals with ALTER and CHANGE MANAGER

Before products like ALTER or CHANGE MANAGER were developed, changing the data structures in a DB2 environment was, at best, difficult.

The process was also complex, tedious, time consuming, unreliable, and error prone. Whether working in a single DB2 subsystem or in a large, multiple-DB2 subsystem environment, the DBA must manage a large number of objects with complex interrelationships.

This section describes how ALTER and CHANGE MANAGER can automate and simplify the goals of changing data structures and migrating structure changes.

Creating, altering, or dropping DB2 objects

Creating, altering, or dropping DB2 objects in ALTER or CHANGE MANAGER is a three-stage process: specification, analysis, and execution.

1 Specification
   Specify the changes that you want to make to the objects.

2 Analysis
   Analyze the specified changes for accuracy and completeness. The result of the analysis stage is a worklist that contains the SQL and the information that is necessary to change the data structures and preserve all dependent objects.

3 Execution
   Execute the worklist that is generated during the analysis stage.

Note
The changes that you specify in the specification stage and analyze in the analysis stage are actually made to the objects in the execution stage.
The following example illustrates these stages by showing you how to use the Task List feature of CHANGE MANAGER to quickly and easily change the length of a column in a table in your database.

Before you begin a set of tasks to achieve a particular goal, you should allocate your JCL data sets for analysis and execution, ensure that you are using the correct DB2 subsystem, and define your job card with the correct account information.

**To modify objects**

1. Start the Task List feature.

   The Task List feature of CHANGE MANAGER enables you to achieve various goals by taking you step-by-step through a series of tasks.

   a. On the CHANGE MANAGER Main Menu (see Figure 5 on page 51), select **Task List** and press Enter.

   ![Figure 7: Task List Menu](image)

   - Select a task. Then press Enter.
     1. Create, alter, or drop DB2 objects
     2. Migrate DB2 objects using Specification
     3. Migrate DB2 objects using scope rules from a Migrate Profile

   **Commands:** HELP END

   b. On the Task List Menu, select **Create, alter, or drop DB2 objects** and press Enter.

   ![Figure 8: Create, Alter, or Drop DB2 Objects panel](image)

   Specify an Alter WORKID. If the WORKID does not exist, the Create WORKID panel will be displayed.

   To see a list of WORKIDs, type a wildcard pattern. (For example BM*.*)

   Alter WORKID: MVSJXE1.JXE0408A
Type S to select processing options. Then press Enter to continue.

- Create or edit a WORKID
- Specify changes to DB2 objects
- Analyze the changes to create a worklist
- Execute the worklist to process the changes

Commands: HELP END

2 Specify an alter-type work ID.

A work ID enables you to identify, control, and track a unit of work. The alter-type work ID is used to store the changes to the DB2 objects that you specify.

a On the Create, Alter, or Drop DB2 Objects panel, type the name of an alter-type work ID, and type S to select all of the processing options. Then, press Enter.

The Create WORKID panel is displayed.

Figure 9: Create WORKID panel

![Create WORKID panel]

b On the Create WORKID panel, select Alter for the Type. Then, press Enter.

Figure 10: Object Specification panel

![Object Specification panel]
Specify the changes that you want to make to the objects.

The current values of the objects are displayed on panels. You can modify the attributes for objects by specifying new values or by typing new values over the existing values. All of your changes are stored in DB2 tables.

a On the Object Specification panel, type the name of your database, table space, and table. Then, press Enter.

From the Mixed List panel, you begin the process of specifying changes to the objects that are listed.

**Figure 11: Mixed List panel**

```
DEHJ ALTER --------------------------- Mixed List ---------------------------
Command ===>
Scroll. . PAGE
WORKID . . . . : MVSJXE1.JXE0408A
Commands: CANCEL
Type action next to object and press Enter.
E=Edit  L=Like  D=Drop  U=Undo
Objects 1 to 32 of 39
More:     +
Act   Object-Type   Objects
************************************ TOP **************************************
DB . . . . . . R1BASIC
   TSG. . . . R1BASIC  T_T10AIM
   TB . . . . R1BASIC  T_T10AIM
   IX . . . . R1BASIC  I_T10AIM1
TSI. . . . . R1BASIC  T01AICP
   TB . . . . R1BASIC  T_T01AICP
   IX . . . . R1BASIC  I_T01AICP1
IXC. . . . R1BASIC  T02ASSEG
   TS . . . . R1BASIC  T02ASSEG
   IX . . . . R1BASIC  I_T02ASSEG_P
   IX . . . . R1BASIC  I_T02ASSEG_X
```

b On the Mixed List panel, in the Act field, type E next to the table that you want to edit, and press Enter.

**Figure 12: Table Detail panel**

```
DEHJ ALTER -------------------------- Table Detail ----------------------
Command ===>
WORKID . . . . : MVSJXE1.JXE0408A
New Values
TB Owner : R1BASIC
TB Name  : T_T01AICP
Database Name : R1BASIC
Tablespace Name : T01AICP
Audit      . . . : NONE            (NONE,ALL,CHANGES)
Validproc   . . . :                  :
Editproc    . . . :                  :
Data Capture : N    (Y,N)
Restrict Drop : N    (Y,N)
Obid        . . . : 3              (N,Y)
Volatile    . . . : N              (N,Y)
Append      . . . : N              (N,Y)
Label:
Link History TB : N    (N,Y)
Link Archive TB : N    (N,Y)
```
c On the Table Detail panel, type S to select the Table Column List, and press Enter.

The Table Columns List panel displays the current column attributes for the table. In this example, you will change the length of a column.

**Figure 13: Table Columns List panel**

<table>
<thead>
<tr>
<th>DEHJ ALTER</th>
<th>----------------------- Table Columns List -----------------------</th>
<th>Scroll. . PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORKID . . . : MVSJXE1.JXE0408A</td>
<td>Commands: CANCEL</td>
<td></td>
</tr>
<tr>
<td>Table Owner . . : R1BASIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table Name  . . : T_T01AICP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type action next to object and press Enter.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E=Edit L=Like D=Drop U=Undo FP=Fieldproc_Parms CM=Comment ID=Identity_Detail DV=Default_Value AX=Aux_Obj_List CAX=Create_Aux_Obj</td>
<td>Lines 1 to 14 of 14</td>
<td></td>
</tr>
<tr>
<td>More:       &gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Act</td>
<td>Column Name ..+....20...+....30. Column Type</td>
<td>Length N1 DF H1 UDT</td>
</tr>
<tr>
<td>COLC_1</td>
<td>CHAR</td>
<td>20 N Y</td>
</tr>
<tr>
<td>COLSI_2</td>
<td>SMALLINT</td>
<td>N Y</td>
</tr>
<tr>
<td>COLIN_3</td>
<td>INTEGER</td>
<td>N Y</td>
</tr>
<tr>
<td>COLSI_4</td>
<td>SMALLINT</td>
<td>N Y</td>
</tr>
<tr>
<td>COLDC_5</td>
<td>DECIMAL</td>
<td>6.2 N Y</td>
</tr>
<tr>
<td>COLDC_6</td>
<td>DECIMAL</td>
<td>9 N Y</td>
</tr>
<tr>
<td>COLDC_7</td>
<td>DECIMAL</td>
<td>5.2 N Y</td>
</tr>
<tr>
<td>COLVC_8</td>
<td>VARCHAR</td>
<td>15 N Y</td>
</tr>
<tr>
<td>COLF1_9</td>
<td>FLOAT</td>
<td>29 N Y</td>
</tr>
<tr>
<td>COLF2_10</td>
<td>FLOAT</td>
<td>53 N Y</td>
</tr>
<tr>
<td>COLVC_11</td>
<td>VARCHAR</td>
<td>15 N Y</td>
</tr>
<tr>
<td>COLDT_12</td>
<td>DATE</td>
<td>N Y</td>
</tr>
<tr>
<td>COLTS_13</td>
<td>TIMESTAMP</td>
<td>6 N Y</td>
</tr>
<tr>
<td>COLC_14</td>
<td>CHAR</td>
<td>100 N Y</td>
</tr>
</tbody>
</table>

**Figure 13 (continued)**

<table>
<thead>
<tr>
<th>DEHJ ALTER</th>
<th>----------------------- Table Columns List -----------------------</th>
<th>Scroll. . PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORKID . . . : MVSJXE1.JXE0408A</td>
<td>Commands: CANCEL</td>
<td></td>
</tr>
<tr>
<td>Table Owner . . : R1BASIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table Name  . . : T_T01AICP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type action next to object and press Enter.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E=Edit L=Like D=Drop U=Undo FP=Fieldproc_Parms CM=Comment ID=Identity_Detail DV=Default_Value AX=Aux_Obj_List CAX=Create_Aux_Obj</td>
<td>Lines 1 to 14 of 14</td>
<td></td>
</tr>
<tr>
<td>More:       &gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Act</td>
<td>Column Name ..+....20...+....30. Column Type</td>
<td>Length N1 DF H1 UDT</td>
</tr>
<tr>
<td>COLC_1</td>
<td>CHAR</td>
<td>20 N Y</td>
</tr>
<tr>
<td>COLSI_2</td>
<td>SMALLINT</td>
<td>N Y</td>
</tr>
<tr>
<td>COLIN_3</td>
<td>INTEGER</td>
<td>N Y</td>
</tr>
<tr>
<td>COLSI_4</td>
<td>SMALLINT</td>
<td>N Y</td>
</tr>
<tr>
<td>COLDC_5</td>
<td>DECIMAL</td>
<td>6.2 N Y</td>
</tr>
<tr>
<td>COLDC_6</td>
<td>DECIMAL</td>
<td>9 N Y</td>
</tr>
<tr>
<td>COLDC_7</td>
<td>DECIMAL</td>
<td>5.2 N Y</td>
</tr>
<tr>
<td>COLVC_8</td>
<td>VARCHAR</td>
<td>15 N Y</td>
</tr>
<tr>
<td>COLF1_9</td>
<td>FLOAT</td>
<td>29 N Y</td>
</tr>
<tr>
<td>COLF2_10</td>
<td>FLOAT</td>
<td>53 N Y</td>
</tr>
<tr>
<td>COLVC_11</td>
<td>VARCHAR</td>
<td>15 N Y</td>
</tr>
<tr>
<td>COLDT_12</td>
<td>DATE</td>
<td>N Y</td>
</tr>
<tr>
<td>COLTS_13</td>
<td>TIMESTAMP</td>
<td>6 N Y</td>
</tr>
<tr>
<td>COLC_14</td>
<td>CHAR</td>
<td>100 N Y</td>
</tr>
</tbody>
</table>

**Figure 13 (continued)**

<table>
<thead>
<tr>
<th>DEHJ ALTER</th>
<th>----------------------- Table Columns List -----------------------</th>
<th>Scroll. . PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>WORKID . . . : MVSJXE1.JXE0408A</td>
<td>Commands: CANCEL</td>
<td></td>
</tr>
<tr>
<td>Table Owner . . : R1BASIC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Table Name  . . : T_T01AICP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type action next to object and press Enter.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E=Edit L=Like D=Drop U=Undo FP=Fieldproc_Parms CM=Comment ID=Identity_Detail DV=Default_Value AX=Aux_Obj_List CAX=Create_Aux_Obj</td>
<td>Lines 1 to 14 of 14</td>
<td></td>
</tr>
<tr>
<td>More:       &gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Act</td>
<td>Column Name ..+....20...+....30. Column Type</td>
<td>Length N1 DF H1 UDT</td>
</tr>
<tr>
<td>COLC_1</td>
<td>CHAR</td>
<td>20 N Y</td>
</tr>
<tr>
<td>COLSI_2</td>
<td>SMALLINT</td>
<td>N Y</td>
</tr>
<tr>
<td>COLIN_3</td>
<td>INTEGER</td>
<td>N Y</td>
</tr>
<tr>
<td>COLSI_4</td>
<td>SMALLINT</td>
<td>N Y</td>
</tr>
<tr>
<td>COLDC_5</td>
<td>DECIMAL</td>
<td>6.2 N Y</td>
</tr>
<tr>
<td>COLDC_6</td>
<td>DECIMAL</td>
<td>9 N Y</td>
</tr>
<tr>
<td>COLDC_7</td>
<td>DECIMAL</td>
<td>5.2 N Y</td>
</tr>
<tr>
<td>COLVC_8</td>
<td>VARCHAR</td>
<td>15 N Y</td>
</tr>
<tr>
<td>COLF1_9</td>
<td>FLOAT</td>
<td>29 N Y</td>
</tr>
<tr>
<td>COLF2_10</td>
<td>FLOAT</td>
<td>53 N Y</td>
</tr>
<tr>
<td>COLVC_11</td>
<td>VARCHAR</td>
<td>15 N Y</td>
</tr>
<tr>
<td>COLDT_12</td>
<td>DATE</td>
<td>N Y</td>
</tr>
<tr>
<td>COLTS_13</td>
<td>TIMESTAMP</td>
<td>6 N Y</td>
</tr>
<tr>
<td>COLC_14</td>
<td>CHAR</td>
<td>100 N Y</td>
</tr>
</tbody>
</table>

On the Table Columns List panel, tab to the Length field for the column that you are modifying. Type a new value for the length of the column over the existing value. Press Enter.

e Press END to return to the Table Detail panel. Press END again to return to the Mixed List Panel.
The Mixed List panel displays an "*A" next to the table, indicating that you have made a change to the table.

**Figure 14: Mixed List panel showing changes**

```
DEHJ ALTER  --------------------------- Mixed List ---------------------------
Command ===>                                                    Scroll. . PAGE
WORKID . . . . : MVSJXE1.JXE0408A                            Commands: CANCEL
Type action next to object and press Enter.
E=Edit  L=Like  D=Drop  U=Undo
Objects 1 to 32 of 39
More:   +
Act     Object-Type    Objects
************************************ TOP **************************************
DB . . . . RIBASIC TRT10AIM
  TSG. . . . RIBASIC  _T10AIM
  TB . . . . RIBASIC  _T10AIM1
  IX. . . . RIBASIC  _I_T10AIM1
  TSI. . . . RIBASIC  T01AICP
  *A      TB. . . . RIBASIC  _T01AICP
  IXC. . . RIBASIC  _I_T01AICP1
  TS. . . RIBASIC  T02ASSEG
  TB . . . . RIBASIC  T02ASSEG
  IX. . . . RIBASIC  _I_T02ASSEG_P
```

f Press END to return to the Object Specification panel. Press END again.

The Analysis Alter Worklist Interface panel is displayed.

**Figure 15: Analysis Alter Worklist Interface panel**

```
ACMFANL1 ---------------- Analysis Alter Worklist Interface ----------------
Command ==>
WORKID . . . . : MVSJXE1.JXE0408A
Analyze "Alter" WORKID
Select type of analysis.
1. Generate a worklist which will convert and reload current data.
2. Generate a worklist which will convert and reload current data and establish a full recovery baseline.
3. Generate a worklist which will reload data from a previous full recovery baseline.
Select run type. Then press Enter.
Run Type . . . 1. Foreground
            2. Batch
Commands: HELP END
```

4 Analyze the changes that you specified.

The Analysis component of the product examines the changes that you specified to determine the effect that the changes have on the structure and the related or dependent objects.

a On the Analysis Alter Worklist Interface panel, select the option to **Generate a worklist which will convert and reload current data**. Then select to run the analysis in **Foreground**. Press Enter.
When you run Analysis in foreground, your TSO session will be unavailable until the processing has completed.

The Analysis JCL Processing Interface panel is displayed.

**Figure 16: Analysis JCL Processing Interface panel**

```
DEHJ -------------------- Analysis JCL Processing Interface -------------------
Command ===>
WORKID . . . . : MVSJXE1.JXE0408A

Specify Dataset Names
  Worklist . . . . . 'ACM_DB2V11.V121.WLBASE01(JXE0408A)'
  Diagnostics . . . . SYSOUT
  Sysout Class . . . X     (Used only when Diagnostics = SYSOUT)

Select foreground processing options. Then press Enter.
S Override
S Create analysis input
S Edit analysis input
  Run analysis
  Edit Worklist
Commands:  BROWSE PREVIOUS HELP
END
```

b On the Analysis JCL Processing Interface panel, type the name of your Worklist, and type S to select your processing options. Press Enter.

The input for the Analysis process is displayed. The input includes the ALUIN input stream, which provides keywords for analyzing the changes. For more information about the input stream, see the *ALTER and CHANGE MANAGER for DB2 Reference Manual*.

**Figure 17: Analysis input**

```
EDIT       SYS16217.T015555.RA000.MVSJXE1.WLIN.H02         Columns 00001 00072
Command ===>                                                  Scroll ===> PAGE
****** ***************************** Top of Data ****************************
==MSG> -Warning- The UNDO command is not available until you change
==MSG> your edit profile using the command RECOVERY ON.
000001 SSID DEHJ
000002 WORKID MVSJXE1.JXE0408A
000003 SORTDEVT SYSDA
000004 INCLUDE (DATA AMS SQL REBIND )
000005 NOREGENIDENTITY
000006 SMSINCLUDE ()
000007 NODBRMLIB ORDERBY
000008 NOVVALPROP
000009 UNLOADEMPTY NOSTOPCOMMIT TABLEACCESS
000010 NOREORGALT
000011 BMCCHECK BMCREORG REORGONLINE DYNREORG REORGALL REORGREF
000012 BMCREBUILD
000013 SYNCPOINT 10
000014 NOROLEOWN
000015 STANDALONESTATS BMCSTATSUPD TABLEALL
000016 BMCCOPY BMCLOAD BMCULOAD

Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  61
c  Press F8 to scroll down the input. Press END.

d  Press Enter to run Analysis.

The results of Analysis are a diagnostic output file (ALUPRINT) and a worklist.

e  To review the results of Analysis (ALUPRINT), you can split your ISPF session or you can use your normal method to review SYSOUT. The last message in the diagnostic output indicates that the worklist completed successfully.

Figure 18: Analysis diagnostic output
The following figure shows the worklist that was written to the data set that you specified on the Analysis JCL Processing Interface panel.

**Figure 19: CHANGE MANAGER alter-type worklist**

```plaintext
000001 *** CHANGE MANAGER FOR DB2 VERSION 12.01.00(12/05/2016)
000002 *BMC396243I DB2 VERSION 11 NFM
000003 ***********************************************************************
000004 *** PTFS APPLIED TO PRODUCT:
000005 *** NONE
000006 ***********************************************************************
000007 * ALUIN PARMS:
000008 * SSID DEHJ
000009 * WORKID MVSJXE1.JXE0408A
000010 * SORTDEVT SYSDA
000011 * INCLUDE (DATA AMS SQL REBIND )
000012 * NOREGENIDENTITY
000013 * SMSINCLUDE ()
000014 * NODBRMLIB ORDERBY
000015 * LOWALPROP
000016 * UNLOADEMPTY NOSTOPCOMMIT TABLEACCESS
000017 * NOREORGALT
000018 * BMCCHECK BMCREORG REORGONLINE DYNREORG REORGALL REORGREF
000019 * BMCREBUILD
000020 * SYNCPOINT 10
000021 * NOROLEOWN
000022 * UTILSTATS BMCSTATSUPD TABLEALL
000023 * BMCCOPY BMCLOAD BMCUNLOAD
000024 * BMCFASTLOAD
000025 * UNLCOLL
000026 * UNLOADTABLE
000027 * SPANNEDRECS
000028 * NOBINDONIX
000029 * MAXSYSUT 20
000030 * DYNUNLDE
000031 * DYNCOPY
000032 * NOPARTCOPY
000033 * COPYDDN(COPY01 )
000034 * NOUTILCOPY
000035 * NOPARALLEL
000036 * CLONEDATA
000037 * NOUSENGTR
000038 ***********************************************************************
000039 -TIME 000050 '2016-08-04-02.13.19.361929'
000040 -SSID 000100 DEHJ
000041 -WKID 000150 MVSJXE1.JXE0408A
000042 -ZPRM 000200
000043 * PREVENT_NEW_IXCTRL_PART NO
000044 -SYNC 000250 END OF CHECK ZPRM SECTION
000045 ***********************************************************************
```
*** THE FOLLOWING MESSAGES ARE WARNINGS ONLY ***

*BMC45339W* TAPES CANNOT BE STACKED AND MULTIPLE TAPE DRIVES MAY BE REQUIRED IF MULTITASKING IS USED TO UNLOAD AND LOAD PARTITIONED TABLE SPACES. TO DISABLE MULTITASKING AND TO ENABLE TAPE STACKING, ADD THE NOMTASK KEYWORD TO THE ALUIN INPUT STREAM.

*BMC396384I* OPTIONS SPECIFIED ALLOW THESE UTILITIES

*BMC396343W* CREATION OF INDEX-CONTROLLED PARTITIONED OBJECTS MIGHT BE PROHIBITED BY DSNZPARM SETTING PREVENT_NEW_IXCTRL_PART.

*** WORKID MVSJXE1.JXE0208B HAS CHANGES FOR TB RIBASIC.T_T01AICP ***

WORKID MVSJXE1.JXE0208B HAS CHANGES FOR TB RIBASIC.T_T01AICP

- **SYNC 000300** START OF EXECUTION SYNCPOINT

**BMCD 000350**

UNLOAD UNLOADDN (DUMMY)

ACTIVE (YES,NO)

DISCARDS 1

EBCDIC CCSID(37,0,0)

SELECT

COLDC_6 INTO

COLDC_6

DECIMAL(8,0)

FROM RIBASIC.T_T01AICP

-JCLP 000400 BMCD IPPARTS 4

-BMCD 000450

UNLOAD UNLOADDN (SYSR)

ACTIVE (YES,NO)

ON FAILURE ALL TERMINATE UTILITY

DELETEFILES YES

DISCARDS 1

NULLTYPE T1 NULLCHAR ?

EBCDIC CCSID(37,0,0)

SELECT

COLD_1

COLD_2

COLD_3

COLD_4

COLD_5

COLD_6

COLD_7

COLD_8

COLD_9

COLD_10

COLD_11

COLD_12

COLD_13

COLD_14

INTO

COLD_1

COLD_2

COLD_3

COLD_4

COLD_5

COLD_6

COLD_7

COLD_8

COLD_9

COLD_10

COLD_11

COLD_12

COLD_13

COLD_14
000113 , COLTS_13
000114 , COLC_14
000115 FROM RIBASIC.T_TOIAICP
000116 ORDER BY
000117 COLSI_2
000118 , COLVC_8
000119 ;
000120 OUTPUT SYSR
000121 UNIT SYSDA
000122 DSNNAME 'MVSJXE1.DEHJ.JXE0408A.SR000001.P&PART'
000124 -SYNC 000500 UNLOAD TB RIBASIC.T_TOIAICP COMPLETE
000125 -SYNC 000550 END OF UNLOAD TABLES SECTION
000126 ***********************************************************************
000127 -SQL  000600 DROP TABLESPACE RIBASIC.TOIAICP
000128 -SYNC 000650 END OF DROP OBJECTS SECTION
000129 ***********************************************************************
000130 -SETS 000700 SET CURRENT SQLID = 'RDACRJ'
000131 -SQL  000750 CREATE  TABLESPACE TOIAICP IN RIBASIC
000132 NUMPARTS 4
000133 (PART 1
000134 USING STOGROUP SYSDEFLT
000135 PRIQTY 440
000136 SECQTY 220
000137 PCTFREE 0
000138 ,PART 2
000139 USING STOGROUP SYSDEFLT
000140 PRIQTY 440
000141 SECQTY 220
000142 PCTFREE 0
000143 ,PART 3
000144 USING STOGROUP SYSDEFLT
000145 PRIQTY 440
000146 SECQTY 220
000147 PCTFREE 0
000148 ,PART 4
000149 USING STOGROUP SYSDEFLT
000150 PRIQTY 440
000151 SECQTY 220
000152 PCTFREE 0
000153 )
000154 BUFFERPOOL BPO
000155 LOCKSIZE ANY
000156 LOCKMAX 10
000157 CLOSE NO
000158 SEGSIZE 0
000159 -SETS 000800 SET CURRENT SQLID = USER
000160 -SYNC 000850 END OF CREATE TABLESPACE SECTION
000161 ***********************************************************************
000162 -SQL  000900 CREATE TABLE RIBASIC.T_TOIAICP
000163 ( COLC_1
000164 CHAR(20) NOT NULL WITH DEFAULT
000165 , COLSI_2
000166 SMALLINT NOT NULL WITH DEFAULT
000167 , COLIN_3
000168 INTEGER NOT NULL WITH DEFAULT
000169 , COLSI_4
000170 SMALLINT NOT NULL WITH DEFAULT
000171 , COLDC_5
000172 DECIMAL(6,2) NOT NULL WITH DEFAULT
000173 , COLDC_6
000174 DECIMAL(8,0) NOT NULL WITH DEFAULT
000175 , COLDC_7
000176 DECIMAL(5,2) NOT NULL WITH DEFAULT
000177 , COLDC_8
000178 DECIMAL(5,2) NOT NULL WITH DEFAULT
000179 )
```
000247 DISCARDN SYSDS001
000248 DELETEFILES NO
000249 ORDER YES
000250 EBCDIC CCSID(37,0,0)
000251 WORKDDN SYSUT001
000252 LOADDDN SORTP
000253 ENFORCE NO
000254 REDEFINE NO
000255 UNIQUEINTO YES
000256 BMCSTATS YES
000257 INTO TABLE R1BASIC.T_T01AICP
000258 (
000259   COLC_1
000260     POSITION(*) CHAR(20),
000261   COLSI_2
000262     POSITION(*) SMALLINT,
000263   COLIN_3
000264     POSITION(*) INTEGER,
000265   COLSI_4
000266     POSITION(*) SMALLINT,
000267   COLDC_5
000268     POSITION(*) DECIMAL(6,2),
000269   COLDC_6
000270     POSITION(*) DECIMAL(9,0),
000271   COLDC_7
000272     POSITION(*) DECIMAL(5,2),
000273   COLVC_8
000274     POSITION(*) VARCHAR,
000275   COLF1_9
000276     POSITION(*) FLOAT(53),
000277   COLF2_10
000278     POSITION(*) FLOAT(53),
000279   COLVC_11
000280     POSITION(*) VARCHAR,
000281   COLDT_12
000282     POSITION(*) DATE EXTERNAL,
000283   COLTS_13
000284     POSITION(*) TIMESTAMP EXTERNAL(26),
000285   COLC_14
000286     POSITION(*) CHAR(100))
000287
000288 -SYNC 001250 LOAD TB R1BASIC.T_T01AICP COMPLETE
000289 -SYNC 001300 END OF LOAD TABLES SECTION
000290 **********************************************************************
000291 -BMCC 001350
000292     OUTPUT DCPYLOCP
000293       UNIT SYSDA
000294                COPY TABLESPACE R1BASIC.T01AICP
000295                COPYDDN (DCPYLOCP)
000296
000297 -SYNC 001400 COPY OF MULTIPLE TABLESPACES COMPLETE
000298 -SYNC 001450 END OF COPY SECTION
000299 **********************************************************************
000300 -REBD 001500
000301 REBIND TRIGGER PACKAGE ( -
000302 R1BASIC.TRO1ICP1 +
000303 ) -
000304  -
000305  APREUSE(NONE) -
000306  BUSTIMESENSITIVE(YES) -
000307  SYSTIMESENSITIVE(YES) -
000308  ARCHIVESENSITIVE(YES) -
000309  PLANMGMT(EXTENDED) -
000310  EXPLAIN(NO)
000311 -REBD 001550
000312 REBIND TRIGGER PACKAGE ( -
000313 R1BASIC.TRO1ICP2 +
000314 ) -
000315  -
000316  APREUSE(NONE) -
000317  BUSTIMESENSITIVE(YES) -
000318  SYSTIMESENSITIVE(YES) -
000319  ARCHIVESENSITIVE(YES) -
000320  PLANMGMT(EXTENDED) -
000321  EXPLAIN(NO)
```
Press END.

The Execution JCL Build Interface panel is displayed.

**Figure 20: Execution JCL Build Interface panel**

ACMFEXC0 ------------------ Execution JCL Build Interface ------------------
Command ===>

WORKID . . . . : MVSJXE1.JXE0408A
Select JCL and run type. Then press Enter.

JCL Type . . 1 1. Build Initial JCL
2. Build Restart JCL from previous execution JCL
3. Build Startover JCL from previous execution JCL
Run Type . . 1 1. Build JCL in Foreground
2. Build JCL in Batch

Commands: NEXT HELP END

5 Execute the changes that you specified and analyzed.

The Execution component of the product uses the worklist as its input job stream and performs the tasks contained in the worklist. The worklist acts as a detailed procedure for implementing the changes that you defined in the specification stage.

a On the Execution JCL Build Interface panel, press **Enter** to accept the defaults.

The Execution Pre- and Post-Processing Interface panel is displayed.

**Figure 21: Execution Pre- and Post-Processing Interface panel**

ACMFEXC1 ---------- Execution Pre- and Post-Processing Interface ----------
Command ===>

WORKID . . . . : MVSJXE1.JXE0408A
Specify additional steps to be included in this job. Then press Enter.

Pre-Execution Compare
1 1. Do not do a compare before execution
2. Compare two previous baselines
3. Compare a previous baseline to the current catalog

Pre-Execution Baseline
1 1. Do not build a baseline before execution
2. Build a baseline before execution

Post-Execution Compare
1 1. Do not do a compare after execution
2. Compare a previous baseline to the new catalog structures
3. Compare the baseline built during a previous job step to the new
catalog structures

4. Create CDL to fall back to a previous baseline

Post-Execution Baseline
1. Do not build a baseline after execution
2. Build a baseline after execution

Commands: HELP PREVIOUS END

b On the Execution Pre- and Post-Processing Interface Panel, press Enter to accept the defaults.

The Execution JCL Processing Interface panel is displayed.

Figure 22: Execution JCL Processing Interface panel

DEHJ ------------------- Execution JCL Processing Interface -------------------
Command ==>

WORKID . . . . : MVSJXE1.JXE0408A

Specify Dataset Names.
Execution JCL . . 'ACM.DB2V11.V121.EXECJCL(JXE0408A)'
Worklist . . . . . 'ACM.DB2V11.V121.WLBASE01(JXE0408A)'
Diagnostics . . . SYSOUT

JCL Build Options.
- Dataset Sizing : No Sizing        (use Override Defaults to set or change)

Select Processing Options. Then press Enter to continue.
* Override Defaults
* Edit Worklist
* Build Execution JCL
S Edit Execution JCL
- Submit Execution JCL

Commands: BROWSE VIEWX PREVIOUS HELP END

c On the Execution JCL Processing Interface panel, specify the data set name for the Execution JCL, accept the defaults for the JCL build options, and type S to select the processing options. Press Enter.

The Execution JCL is displayed. The JCL includes the //AEXIN DD statement (input stream), which includes keywords that are used in executing the JCL. For more information about the input stream, see the ALTER and CHANGE MANAGER for DB2 Reference Manual.

Figure 23: Execution JCL for an alter-type worklist

000001 //MVSJXE1E JOB (4110),'EXECUTE-JXE0408A',
000002 // CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1),
000003 // NOTIFY=MVSJXE1
000004 /*ROUTE XEQ
BMCPLX1
000005 /*JOBPARM
SYSAFF=DB2B
000006 //
******************************
000007 /* CREATED BY:
MVSJXE1
BMC CHANGE MANAGER
EXECUTION

---

STEP1 EXEC
PGM=AEXEMAIN,
REGION=0M,MEMLIMIT=NOLIMIT,PARM='DMC1QEHJ',
COND=(4,LT)

STEPLIB DD
DSN=CSG.DEHJ.DSNEXIT,DISP=SHR

DD
DSN=CSGI.DB2V11M.DSNLOAD,DISP=SHR

ABNLIGNR DD
DUMMY

SYSUDUMP DD
SYSOUT=* 

---

SAMPLE SYSDUMP JCL

---

PLEASE REPLACE THE DSN AND UNIT WITH ONES THAT FITS

YOUR SHOP STANDARDS AND UNCOMMENT THE JCL

--

SYSTEM DD
SYSOUT=* 

ASUSRPT DD
SYSOUT=* 

DDLOUT DD
SYSOUT=* 

SYSOUT DD
SYSOUT=*

---

STATISTICS WORK DD
STATEMENTS
SYSOUT=* 

STPRNO1 DD
SYSOUT=* 

UTPRINT DD
SYSOUT=* 

RNPRNO1 DD
SYSOUT=* 

DLOUT DD
SYSOUT=* 

SYSDT DD
SYSOUT=* 

AJXERR DD

---
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface    71
+DBNAME(DSNDB04)
000076 +ALLO(KEEPDB2)
000077 //RRGPARMX DD *
000078 +ALLO(KEEPDB2)
000079 //LODPARMS DD *
000080 +ALLO(KEEPDB2)
000081 //**********************************************
000082 //*** RUNTIME ENABLEMENT
CHOSEN
000083 //SYSPROC DD
DSN=AEX.QA120.BMCCLIB,DISP=SHR
000084 //***************************************************************************
000085 //* COMING FROM AJXCLIBU IN
AEX.VC1Q.SLIB
000086 //* UPDATED 04/06/2015
SKH
000087 //***************************************************************************
000088 // DD
DSN=ACT.QA120.CLIST,DISP=SHR
000089 // DD
DSN=ASU.QA120.CLIST,DISP=SHR
000090 //ACT$MSGS DD
DSN=AEX.QA120.BMCMLIB,DISP=SHR
000091 //***************************************************************************
000092 //* COMING FROM AJXCLIBU IN
AEX.VC1Q.SLIB
000093 //* UPDATED 04/06/2015
SKH
000094 //***************************************************************************
000095 // DD
DSN=ACT.QA120.MLIB,DISP=SHR
000096 // DD
DSN=ASU.QA120.MLIB,DISP=SHR
000097 //ISPMLIB DD
DSN=AEX.QA120.BMCMLIB,DISP=SHR
000098 //***************************************************************************
000099 /// SYSTEM ISPF MLIB SKELETON
AJXISPFS
000100 ///</***************************************************************************
000101 // DD
DSN=ISP.SISPMENU,DISP=SHR
000102 ///</***************************************************************************
000103 //* COMING FROM AJXMLIBU IN
AEX.VC1Q.SLIB
000104 //* UPDATED 04/06/2015
SKH
000105 ///</***************************************************************************
000106 // DD
DSN=ACT.QA120.MLIB,DISP=SHR
000107 // DD
DSN=ASU.QA120.MLIB,DISP=SHR
000108 ///</***************************************************************************
After you review the JCL, press END to return to the Execution JCL Processing Interface panel.

From the Execution JCL Processing Interface panel, press Enter to submit the JCL and process your changes.

The result of execution is a worklist execution log (AEXPRINT).

To review the results of execution in the worklist execution log (AEXPRINT), you can split your ISPF session or you can use your normal method to review SYSOUT.

The last message in the diagnostic output indicates that the work ID status is complete and that the worklist completed successfully. The worklist implemented the structure changes that you specified.

Figure 24: Execution results for an alter-type worklist
Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER

Achieving your goals with ALTER and CHANGE MANAGER


ALTER and CHANGE MANAGER for DB2 User Guide
FROM R1BASIC.T_T01AICP
, COLC_14
, COLTS_13
, COLDT_12
, COLVC_11
, COLF2_10
, COLF1_9
, COLVC_8
, COLDC_7
, COLDC_6
, COLDC_5
, COLIN_3
, COLSI_2
, COLC_1
INTO
, COLC_14
, COLTS_13
, COLDT_12
, COLF2_10
, COLF1_9
, COLVC_8
, COLDC_7
, COLDC_6

ORDER BY
DSNAME 'MVSJXE1.DEHJ.JXE0408A.SR000001.P&PART'

;
Achieving your goals with ALTER and CHANGE MANAGER

ALTER and CHANGE MANAGER for DB2 User Guide
**Chapter 2  Overview of the ALTER and CHANGE MANAGER interface**  

**Achieving your goals with ALTER and CHANGE MANAGER**
```
CREATE TABLE R1BASIC.T_T01AICP
(
  COLC_1 CHAR(20) NOT NULL WITH DEFAULT,
  COLSI_2 SMALLINT NOT NULL WITH DEFAULT,
  COLIN_3 INTEGER NOT NULL WITH DEFAULT,
  COLSI_4 SMALLINT NOT NULL WITH DEFAULT,
  COLDC_5 DECIMAL(6,2) NOT NULL WITH DEFAULT,
  COLDC_6 DECIMAL(8,0) NOT NULL WITH DEFAULT,
  COLDC_7 DECIMAL(5,2) NOT NULL WITH DEFAULT,
  COLVC_8 VARCHAR(15) NOT NULL WITH DEFAULT,
  COLF1_9 FLOAT NOT NULL WITH DEFAULT,
  COLF2_10 FLOAT NOT NULL WITH DEFAULT,
  COLVC_11 VARCHAR(15) NOT NULL WITH DEFAULT,
  COLDT_12 DATE NOT NULL WITH DEFAULT,
  COLTS_13 TIMESTAMP NOT NULL WITH DEFAULT,
  COLC_14 CHAR(100) NOT NULL WITH DEFAULT
) IN R1BASIC.T01AICP
```
BUFFEROOL BP1
CLOSI YES

DSNT400I SQLCODE = 000, SUCCESSFUL EXECUTION
0:SYNC 001100 END OF CREATE INDEX SECTION 68706292
0:JCLP 001150 BMCL IPIPARTS 4 36305581
BMCL001I -JCLP IGNORED IN THIS RUN. SKIPPING WORKLIST RECORDS UNTIL NEXT COMMAND

LOAD DATA REPLACE
INDSN('MVSJXE1.DEHJ.JXE0408A.SR000001.P001'
 'MVSJXE1.DEHJ.JXE0408A.SR000001.P002'
 'MVSJXE1.DEHJ.JXE0408A.SR000001.P003'
 'MVSJXE1.DEHJ.JXE0408A.SR000001.P004'
)
ERRORD SYSEX001
DISCARDS 1
DISCARDDN SYSLST001
LOADDDN SORTP
WORKDDN SYSUT001
LOADDN SYSDS001
DELETEFILES NO
ORDER YES
EBCDIC CCSID(37,0,0)
LOAD DDN SYSER001
DISCARDDN SYSDS001
DELETEFILES NO
ORDER YES
UNIQUEINTO YES
BMCSTATS YES
INTO TABLE R1BASIC.T_T01AICP

COLC_1
**** BMC EXECUTION FOR DB2 ****
**** VERSION: V12.01.00
(C) COPYRIGHT 1991 - 2017 BMC SOFTWARE, INC.
LINE PREFIX : DENOTES WORKLIST INPUT STATEMENTS

2016-08-04 02.58.08 ON SYSID DB2B
WORKLIST EXECUTION LOG

BMC50471I z/OS 2.2.0,PID=HBB77A0,DFSMS FOR Z/OS=2.2.0,DB2=11.1.0
BMC50471I REGION=0M,BELOW 16M=8764K,ABOVE 16M=1307952K,IEFUSI=NO,CPUS=3
BMC50471I MEMLIMIT=17592186040320M,AVAILABLE=17592186040300M,MEMLIMIT SET BY:JCL
BMC50471I LOADPLUS FOR DB2--V11.02.00
BMC50471I NO MAINTENANCE TO REPORT

Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface 83
BMC50483I  REMPLCPY   DSNPAT=&UID.&DDNAME.&TS..T&TIME
BMC50483I  REMBLCPY   DSNPAT=&UID.&DDNAME.&TS..T&TIME
BMC50483I  LOCPXCPY   DSNPAT=&UID.&DDNAME.&TS..F&PART..T&TIME
BMC50471I  DB2 DSNHDECP MODULE SETTINGS:

BMC50471I  VERSION                 = 1110
BMC50471I  SUBSYSTEM DEFAULT       = DEHJ
BMC50471I  CHARACTER SET           = ALPHANUM
BMC50471I  DATE FORMAT             = USA
BMC50471I  TIME FORMAT             = USA
BMC50471I  LOCAL DATE LENGTH       = 0
BMC50471I  LOCAL TIME LENGTH       = 0
BMC50471I  DECIMAL POINT           = PERIOD
BMC50471I  DECIMAL ARITHMETIC      = 15
BMC50471I  DELIMITER               = DEFAULT
BMC50471I  SQL DELIMITER           = DEFAULT
BMC50471I  ENCODING SCHEME         = EBCDIC
BMC50471I  APPL. ENCODING SCHEME   = EBCDIC
BMC50471I  MIXED                   = NO
BMC50471I  EBCDIC CCSID            = (37,0,0)
BMC50471I  ASCII CCSID             = (819,0,0)
BMC50471I  UNICODE CCSID           = (367,0,0)
BMC50471I  IMPLICIT TIME ZONE      = CURRENT (-05:00)
BMC50471I  BMC_BMCUTIL       ='BMCUTIL.CMN_BMCUTIL'
BMC50471I  BMC_BMCSYNC       ='BMCUTIL.CMN_BMCSYNC'
BMC50471I  BMC_BMCHIST       ='BMCUTIL.CMN_BMCHIST'
BMC50471I  BMC_BMCDICT       ='BMCUTIL.CMN_BMCDICT'

BMC50102I  LOAD DATA  REPLACE
BMC50102I  INDSN('MVSJXE1.DEHJ.JXE0408A.SR000001.P001'
BMC50102I  ,'MVSJXE1.DEHJ.JXE0408A.SR000001.P002'
BMC50102I  ,'MVSJXE1.DEHJ.JXE0408A.SR000001.P003'
BMC50102I  ,'MVSJXE1.DEHJ.JXE0408A.SR000001.P004'
BMC50102I   )
BMC50102I  ERRDDN SYSER001
BMC50102I  DISCARDS 1
BMC50102I  DISCARDDN SYSDS001
BMC50102I  DELETEFILES NO
BMC50102I  ORDER YES
BMC50102I  EBCDIC CCSID(37,0,0)
BMC50102I  WORKDDN SYSUD001
BMC50102I  LOADDDN SORTP
BMC50102I  ENFORCE NO
BMC50102I  REDEFINE NO
BMC50102I  UNIQUEINTO YES
BMC50102I  BMCSTATS YES
BMC50102I  INTO TABLE R1BASIC.T_T01AICP

**** BMC EXECUTION FOR DB2 ****
**** VERSION: V12.01.00
(C) COPYRIGHT 1991 - 2017 BMC SOFTWARE, INC.
2016-08-04 02.58.12 ON SYSID DB2B
WORKLIST EXECUTION LOG  PAGE 22

LINE PREFIX : DENOTES WORKLIST INPUT STATEMENTS

**** BMC EXECUTION FOR DB2 ****
**** VERSION: V12.01.00
(C) COPYRIGHT 1991 - 2017 BMC SOFTWARE, INC.
2016-08-04 02.58.12 ON SYSID DB2B
WORKLIST EXECUTION LOG  PAGE 22

Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER
READY
DSN SYSTEM(DEHJ)

REBIND TRIGGER PACKAGE ( RIBASIC.TROIICP1 )
APREUSE(NONE) BUSTIMESENSITIVE
SYSTIMESENSITIVE(YES) ARCHIVESENSITIVE(YES) PLANMGMT(EXTENDED)
EXPLAIN(NONE)

DSNT2191 *DEHJ DSNTBRB2 REBIND FOR PACKAGE = DEHJ.RIBASIC.TROIICP1() SUCCEEDED BUT A PREVIOUS AND/OR ORIGINAL PACKAGE COPY WAS NOT CREATED/REPLACED BECAUSE THE CURRENT PACKAGE COPY WAS INVALID.

DSNT254I *DEHJ DSNTBRB2 REBIND OPTIONS FOR PACKAGE = DEHJ.RIBASIC.TROIICP1()
ACTION
OWNER RDACRJ
QUALIFIER RDACRJ
VALIDATE BIND
EXPLAIN ND
ISOLATION CS
RELEASE COMMIT
COPY APREUSE NONE
APCOMPAR
APREAPRIN YES
BUSTIMESENSITIVE YES
SYSTIMESENSITIVE YES
ARCHIVESENSITIVE YES
APPLCOMPAT V11R1

DSNT2551 *DEHJ DSNTBRB2 REBIND OPTIONS FOR PACKAGE = DEHJ.RIBASIC.TROIICP1()
SOLERKOR NOPACKAGE
CURRENTDATA NO
DEGREE
DYNAMICRULES BIND
NODEFER PREPARE
REOPT NONE
KEEPDYNAMIC NO
IMMEDIATENOT
DBPROTOCOL DRDA

** **** B M C   E X E C U T I O N   F O R   D B 2 ****
** **** V E R S I O N: V12.01.00
(C) COPYRIGHT 1991 - 2017 BMC SOFTWARE, INC.
2016-08-04 02.58.18 ON SYSID DB2B WORKLIST EXECUTION LOG PAGE 27

NO PATH
"SYSIBM","SYSPROC","SYSPROC","SYSPROC","RDACRJ"

DSNT275I *DEHJ DSNTBRB2 REBIND OPTIONS FOR PACKAGE = DEHJ.RIBASIC.TROIICP1()
QUERYACCELERATION
GETACCELERATION

DSNT232I *DEHJ SUCCESSFUL REBIND FOR PACKAGE = DEHJ.RIBASIC.TROIICP1()

DSN OEND

: REBD 001550
: REBIND TRIGGER PACKAGE ( -

RIBASIC.TROIICP2 +

APREUSE(NONE) -

BUSTIMESENSITIVE(YES) -

SYSTIMESENSITIVE(YES) -

ARCHIVESENSITIVE(YES) -

PLANMGMT(EXTENDED) -

EXPLAIN(NONE)

03217686
03217686

BMC1682I CONTROL PASSED TO 'IKJEFT01'
BMC16898I DEBUG; CONNECTED WITH USER: MVSJXE1
BMC16898I DEBUG; CONNECTED WITH USER: MVSJXE1
BMC16898I DEBUG; CONNECTED WITH USER: MVSJXE1
BMC16898I DEBUG; CONNECTED WITH USER: MVSJXE1

OREADY
DSN SYSTEM(DEHJ)

REBIND TRIGGER PACKAGE ( RIBASIC.TROIICP2 )
APREUSE(NONE) BUSTIMESENSITIVE
SYSTIMESENSITIVE(YES) ARCHIVESENSITIVE(YES) PLANMGMT(EXTENDED)
EXPLAIN(NONE)

DSNT2191 *DEHJ DSNTBRB2 REBIND FOR PACKAGE = DEHJ.RIBASIC.TROIICP2() SUCCEEDED BUT A PREVIOUS AND/OR ORIGINAL PACKAGE COPY WAS NOT CREATED/REPLACED BECAUSE THE CURRENT PACKAGE COPY WAS INVALID.

DSNT254I *DEHJ DSNTBRB2 REBIND OPTIONS FOR PACKAGE = DEHJ.RIBASIC.TROIICP2()
ACTION
OWNER RDACRJ
QUALIFIER RDACRJ
VALIDATE BIND

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface
This example shows that with CHANGE MANAGER you have the ability to quickly and accurately specify changes to your data structures, perform an in-depth analysis of those changes to determine their effect on your subsystem, and execute SQL and utilities, while collecting statistics on your DB2 objects.

Migrating DB2 objects

Just as changing DB2 objects is a three-stage process, migrating DB2 objects from one environment to another within the same DB2 subsystem in ALTER or CHANGE MANAGER is also a three-stage process.

The three stages are specification, analysis, and execution.
1 Specification
   Identify the objects to be migrated and specify changes to the data structures.

2 Analysis
   Analyze the specified migration changes for accuracy and completeness. The result of the analysis stage is a worklist that contains the SQL and the information that is necessary to generate the data structure in the target DB2 system.

3 Execution
   Execute the worklist that is generated during the analysis stage.

Note
The changes that you specify in the specification stage and analyze in the analysis stage are actually made to the objects in the execution stage.

The following example illustrates these three stages by showing how to use the Task List feature of CHANGE MANAGER to quickly and easily migrate objects from one environment to another environment within the same DB2 subsystem.

Note
Remember that CHANGE MANAGER can also migrate only the changes that you make to a data structure on one DB2 subsystem to other DB2 subsystems.

Before you begin a set of tasks to achieve a particular goal, you should allocate your JCL data sets for analysis and execution, ensure that you are using the correct DB2 subsystem, and define your job card with the correct account information.

For more information, view the Quick Course "CHANGE MANAGER for DB2 - Migrating in Same Subsystem."

To migrate objects

1 Start the Task List feature.

   The Task List feature of CHANGE MANAGER enables you to achieve various goals by taking you step-by-step through a series of tasks.

   a On the CHANGE MANAGER Main Menu (see Figure 5 on page 51), select Task List and press Enter.

      The Task List Menu is displayed, as shown in Figure 25 on page 91.

   Figure 25: Task List Menu

   DEHJ ----------------------------- Task List Menu ----------------------
   Command ===>
Select a task. Then press Enter.

2. Create, alter, or drop DB2 objects
   2. Migrate DB2 objects using Specification
   3. Migrate DB2 objects using scope rules from a Migrate Profile

On the Task List Menu, select **Migrate DB2 objects using Specification** and press Enter.

The Migrate DB2 Objects Using Specification panel is displayed, as shown in Figure 26 on page 92.

**Figure 26: Migrate DB2 Objects Using Specification panel**

```
DEHJ ------------------------ Migrate DB2 Objects Using Specification -----
Command   ===>
Specify a Migrate WORKID.
If the WORKID does not exist, the Create WORKID panel will be displayed.
To see a list of WORKIDs, type a wildcard pattern. (For example BM*.*)
Migrate WORKID: MVSJXE1.JXE0308A

Optionally specify a Migrate Profile.
If the profile does not exist, the Create Profile panel will be displayed.
To see a list of profiles, type a wildcard pattern. For example BM*.*
Migrate Profile:

Type S to select processing options. Then press Enter to continue.
  - Create or edit a Migrate Profile
  - Create or edit a WORKID
  - Specify changes to DB2 objects
  - Analyze the changes to create a worklist
  - Execute the worklist to process the changes

Commands: HELP END
```

2. On the Migrate DB2 Objects Using Specification panel, specify a migrate-type work ID.

A work ID enables you to identify, control, and track a unit of work. The migrate-type work ID is used to specify data structure definitions to be migrated from one DB2 subsystem to another, or to duplicate structures within the same subsystem.

a. On the Migrate DB2 Objects Using Specification panel, type the name of a migrate-type work ID, and type S to select all of the processing options except **Create or edit a Migrate Profile**. Then, press Enter.
The Create WORKID panel is displayed, as shown in Figure 27 on page 93.

**Figure 27: Create WORKID panel**

DEHJ ------------------------------ Create WORKID ----------- WORKID NOT FOUND  
Command ===>  
WORKID . . . . : MVSJXE1.JXE0308A  
Type . . . . . . 2 1. Alter  
2. Migrate  
3. Receive  
Comment . . . . .  

Select additional panels to display for Migrate type WORKID. Then press Enter.  
WORKID Migrate Options  
WORKID Change Rules  
Commands: HELP END CANCEL

b On the Create WORKID panel, select Migrate for the Type. Type S to select the WORKID Change Rules. Then, press Enter.

The Change Rules panel is displayed, as shown in Figure 28 on page 93.

**Figure 28: Change Rules panel**

DEHJ ----------------------   Change Rules ------------------------------------  
Command ===>                                                  SCROLL ===> PAGE  
WORKID . . . . : MVSJXE1.JXE0308A  
Type action next to existing rule or overtype existing values.  
Rules 1 to 1 of 1  
E=Edit I=Insert D=Delete L=Like C=Create                         More:       >  
Act Opt Obj Attribute: Current Attribute Value     New Attribute Value  
************************************** TOP ************************************  
************************************* BOTTOM **********************************

Change rules enable you to specify global changes to the name, owner, creator, or other attributes of the objects. The new attributes that you specify apply to all objects in the migration that possess the attributes.

c On the Change Rules panel, type E in the Act field to select Edit. Press Enter.

The Edit Change Rules panel is displayed, as shown in Figure 29 on page 93.

**Figure 29: Edit Change Rules panel**

DEHJ -------------------------- Edit Change Rules -----------------------------  
Command ===>  
d On the Edit Change Rules panel, select Change as the Option for the attribute, select All as the Object, and type S to select Display Attribute List. Press Enter.

By selecting All objects, you can specify one change rule that changes an attribute for all of the tables, indexes, views, and so on. You can change an attribute that applies to all of the migrated objects that have that attribute.

The Edit Rules Attribute Selection panel is displayed, as shown in Figure 30 on page 94.

Figure 30: Edit Rules Attribute Selection panel

<table>
<thead>
<tr>
<th>Option</th>
<th>Change Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>All</td>
</tr>
<tr>
<td>Attribute Name</td>
<td></td>
</tr>
</tbody>
</table>

Select from attribute list below.

<table>
<thead>
<tr>
<th>Attribute Number</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Activate</td>
</tr>
<tr>
<td>2</td>
<td>Asutime</td>
</tr>
<tr>
<td>3</td>
<td>Audit</td>
</tr>
<tr>
<td>4</td>
<td>BuffpoolIX</td>
</tr>
<tr>
<td>5</td>
<td>BuffpoolTS</td>
</tr>
<tr>
<td>6</td>
<td>Close</td>
</tr>
<tr>
<td>7</td>
<td>Collection</td>
</tr>
<tr>
<td>8</td>
<td>Compress</td>
</tr>
<tr>
<td>9</td>
<td>Createdby</td>
</tr>
<tr>
<td>10</td>
<td>Creator</td>
</tr>
<tr>
<td>11</td>
<td>Database</td>
</tr>
<tr>
<td>12</td>
<td>DataCapture</td>
</tr>
<tr>
<td>13</td>
<td>DataCapture</td>
</tr>
<tr>
<td>14</td>
<td>Dbrmlib</td>
</tr>
<tr>
<td>15</td>
<td>Debug_Mode</td>
</tr>
<tr>
<td>16</td>
<td>Define</td>
</tr>
<tr>
<td>17</td>
<td>Devtype</td>
</tr>
<tr>
<td>18</td>
<td>Editproc</td>
</tr>
<tr>
<td>19</td>
<td>Editproc</td>
</tr>
<tr>
<td>20</td>
<td>Erase</td>
</tr>
<tr>
<td>21</td>
<td>Fieldproc</td>
</tr>
<tr>
<td>22</td>
<td>Freepage</td>
</tr>
<tr>
<td>23</td>
<td>Grantee</td>
</tr>
<tr>
<td>24</td>
<td>Grantor</td>
</tr>
<tr>
<td>25</td>
<td>* . InLine</td>
</tr>
<tr>
<td>26</td>
<td>Location</td>
</tr>
<tr>
<td>27</td>
<td>Lockmax</td>
</tr>
<tr>
<td>28</td>
<td>Locksize</td>
</tr>
<tr>
<td>29</td>
<td>Maxparts</td>
</tr>
<tr>
<td>30</td>
<td>Name</td>
</tr>
<tr>
<td>31</td>
<td>Owner</td>
</tr>
<tr>
<td>32</td>
<td>Pctfree</td>
</tr>
<tr>
<td>33</td>
<td>Piecesize</td>
</tr>
<tr>
<td>34</td>
<td>Priqty</td>
</tr>
<tr>
<td>35</td>
<td>Qualifier</td>
</tr>
<tr>
<td>36</td>
<td>Resident</td>
</tr>
<tr>
<td>37</td>
<td>Resident</td>
</tr>
<tr>
<td>38</td>
<td>Schema</td>
</tr>
<tr>
<td>39</td>
<td>Secqty</td>
</tr>
<tr>
<td>40</td>
<td>Segsize</td>
</tr>
<tr>
<td>41</td>
<td>SPAlter</td>
</tr>
<tr>
<td>42</td>
<td>SPType</td>
</tr>
<tr>
<td>43</td>
<td>Stog_VCAT</td>
</tr>
<tr>
<td>44</td>
<td>* . Log</td>
</tr>
<tr>
<td>45</td>
<td>Tablespace</td>
</tr>
<tr>
<td>46</td>
<td>Trackmod</td>
</tr>
<tr>
<td>47</td>
<td>Validdproc</td>
</tr>
<tr>
<td>48</td>
<td>VCAT</td>
</tr>
<tr>
<td>49</td>
<td>VCAT_Stog</td>
</tr>
<tr>
<td>50</td>
<td>Version</td>
</tr>
<tr>
<td>51</td>
<td>Volorder</td>
</tr>
<tr>
<td>52</td>
<td>Volume</td>
</tr>
<tr>
<td>53</td>
<td>WLM_Env</td>
</tr>
</tbody>
</table>

On the Edit Rules Attribute Selection panel, select the number for the Name attribute. Press Enter, and then press END.
The Edit Change Rules panel is displayed, as shown in Figure 31 on page 95.

**Figure 31: Edit Change Rules panel—NAME attribute**

```plaintext
DEHJ -------------------------- Edit Change Rules ----------------------------
                  4. Table             10. Trigger            16. Sequences
                  5. Check Constraint 11. Alias              17. All
                  6. Foreign Key       12. Synonym

Attribute. . NAME                   _  Display Attribute List
Current Attribute Value. .
New Attribute Value. . . .
Specify optional object name(s).
Part1                                                  (SG, DB, owner, schema)
Part2                                                  (TS, name)
Part3                                                  (SP, version)
Commands: HELP END CANCEL PF4=ZOOM
```

f On the Edit Change Rules panel, type the **Current Attribute Value** and the **New Attribute Value** for the database name. Press END.

The change rules are displayed in the Change Rules panel, as shown in Figure 32 on page 95.

**Figure 32: Change Rules panel—NAME attribute**

```plaintext
DEHJ ----------------------   Change Rules ------------------------------------
Command ===>                                                  SCROLL ===> PAGE
WORKID . . . . : MVSJXE1.JXE0308A
Type action next to existing rule or overtype existing values.
E=Edit I=Insert D=Delete L=Like C=Create                         More:       >
Act Opt Obj Attribute  Current Attribute Value     New Attribute Value
***********************************************************************
C ** NAME       RIBASIC                     J1BASIC
***********************************************************************
```

g On the Change Rules panel, type **L** in the **Act** field for the name rule. The **L** enables you to create a new rule like the existing rule. Press **Enter**.

h On the second line, type **OWNER** over **NAME** in the **Attribute** field, type ***** for the **Current Attribute Value** (if you do not know the current owner), and type your TSO logon ID (or other desired value) for the **New Attribute Value**. Press **Enter**.

The Change Rules panel is displayed, as shown in Figure 33 on page 95.

**Figure 33: Change Rules panel—OWNER attribute**

```plaintext
DEHJ ------------------   Change Rules ------------------------------------
Command ===>                                                  SCROLL ===> PAGE
```
i  Press END.

The Edit WORKID Migration Options panel is displayed.

j  Press END to return to the Object Specification panel, as shown in Figure 34 on page 96.

Figure 34: Object Specification panel

DEHJ MIGRATE ---------------------- Object Specification ----------------------
Command ===> 
WORKID . . . : MVSJXE1.JXE0308A                        _ Changed Objects List
Specify the object type(s) to be included in a list.
Stogroup. ..                                                      (SG     Name)
Database. . RIBASIC                                             (DB     Name)
Tablespace *                                                    (DB.TS  Name)
Table . . . *                                                    (Own.TB Name)
Check Cnst.                                                      (Own.TB.Chk )
Foreign Key                                                      (Own.TB.Rel )
Index . . . *                                                    (Own.IX Name)
Unique Cnst                                                      (Own.TB.Uniq)
View. . . .                                                      (Own.VW Name)
Trigger . .                                                      (Sch.TR Name)
Alias . . .                                                      (Own.AL Name)
Synonym . .                                                      (Own.SY Name)

The following objects are retrieved independently of all other DB2 objects.
Sequence .                                                      (Sch.SQ Name)
Stored Proc *                                                    (Sch.SP Name)

3  Identify the objects and dependent objects that you want to migrate.

a  On the Object Specification panel, type the name of a Database. Press Enter.

The Mixed List panel is displayed with the dependent objects of the database, as shown in Figure 35 on page 96.

Figure 35: Mixed List panel with dependent objects
b On the Mixed List panel, type M in the Act field next to the database that you want to migrate. Press Enter.

c Type MO in the Act field next to the database that you selected. Press Enter.

The Migrate Options Overrides panel is displayed, as shown in Figure 36 on page 97.

Figure 36: Edit WORKID Migration Options panel

```
DEHJ MIGRATE ------------------------------ Migrate Options Overrides ------------------------------
Command ===>  

WORKID . . . : MVSJXE1.JXE0308A  
Database . . . : RIBASIC  

Specify dependencies that should also be migrated.  
(Y=include them; N=omit them)  

_  Migrate Data  
_  Migrate Tablespaces  
_  Migrate Tables  
_  Migrate Unique Constraints  
_  Migrate Check Constraints  
_  Migrate Foreign Keys  
_  Migrate Indexes  
_  Migrate Views  
_  Migrate Authorizations  
_  Migrate Synonyms  
_  Migrate Aliases  
_  Migrate Trigger  
_  Migrate Auxiliary Objects  

Commands: HELP END CANCEL
```

d On the Migrate Options Overrides panel, type Y to select the dependent objects that you want to migrate. Press Enter, and then press END.

In this example, data was selected. No dependent objects were selected.

The Mixed List panel is displayed, as shown in Figure 37 on page 97. An "*M" is displayed next to the database that you selected to migrate.

Figure 37: Mixed List panel—database selected for migration

```
DEHJ MIGRATE ------------------------------ Mixed List ------------------------------  
Command ===>  

WORKID . . . : MVSJXE1.JXE0308A  

Type action next to object and press Enter.  
E=Edit  L=Like  D=Drop  U=Undo  M=Migrate  MO=Migrate options  

Objects 1 to 32 of 39  
More: +  

Act  Object-Type  Objects  
**********************************************  TOP  **********************************************  

*M  DB . . . : RIBASIC  
TB . . . : RIBASIC  
IX . . . : RIBASIC  
TSI . . . : RIBASIC  
TB . . . : RIBASIC  
IXC . . . : RIBASIC  

Chapter 2 Overview of the ALTER and CHANGE MANAGER interface  97
e On the Mixed List panel, type M in the Act field next to the table space that you want to migrate. Press Enter.

f Type MO in the Act field next to the table space that you selected. Press Enter. The Migrate Options Overrides panel is displayed.

g On the Migrate Options Overrides panel, type Y to select the dependent objects that you want to migrate. Press Enter, and then press END.

In this example, data and all of the dependent objects were selected.

The Mixed List panel is displayed, as shown in Figure 38 on page 98. An "*M" is displayed next to the table space that you selected to migrate.

Figure 38: Mixed List panel—objects selected for migration

<table>
<thead>
<tr>
<th>Act</th>
<th>Object-Type</th>
<th>Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>*M</td>
<td>DB</td>
<td>RIBASIC T_T10AIM</td>
</tr>
<tr>
<td></td>
<td>TSG.</td>
<td>RIBASIC T_T10AIM1</td>
</tr>
<tr>
<td>*M</td>
<td>TSI.</td>
<td>RIBASIC T01AICP</td>
</tr>
<tr>
<td></td>
<td>TB.</td>
<td>RIBASIC T01AICP1</td>
</tr>
<tr>
<td></td>
<td>IX.</td>
<td>RIBASIC I_T01AICP1</td>
</tr>
<tr>
<td></td>
<td>TS.</td>
<td>RIBASIC T02ASSEG</td>
</tr>
<tr>
<td></td>
<td>TB.</td>
<td>RIBASIC T02ASSEG</td>
</tr>
</tbody>
</table>

h Press END to return to the Object Specification panel. Press END again.

The Analysis Migrate Worklist Interface panel is displayed, as shown in Figure 39 on page 98.

Figure 39: Analysis Migrate Worklist Interface panel

<table>
<thead>
<tr>
<th>Act</th>
<th>Object-Type</th>
<th>Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DB</td>
<td>RIBASIC T_T10AIM</td>
</tr>
<tr>
<td></td>
<td>TSG.</td>
<td>RIBASIC T_T10AIM1</td>
</tr>
<tr>
<td></td>
<td>TSI.</td>
<td>RIBASIC T01AICP</td>
</tr>
<tr>
<td></td>
<td>TB.</td>
<td>RIBASIC T01AICP1</td>
</tr>
<tr>
<td></td>
<td>IX.</td>
<td>RIBASIC I_T01AICP1</td>
</tr>
<tr>
<td></td>
<td>TS.</td>
<td>RIBASIC T02ASSEG</td>
</tr>
<tr>
<td></td>
<td>TB.</td>
<td>RIBASIC T02ASSEG</td>
</tr>
</tbody>
</table>

Select type of analysis.
1. Analyze Specification entries (use the WORKID’s change rules)
2. Analyze Specification entries (use a Migrate Profile’s locations and...
4 Analyze the migration changes that you specified.

The Analysis component of the product examines the changes that you specified and generates a worklist that contains the instructions that are necessary to implement the actions chosen during the specification stage.

a On the Analysis Migrate Worklist Interface panel, select the **Analyze Specification entries (use the WORKID’s change rules)** option. Then select to run the Analysis in **Batch**. Press **Enter**.

*Note*

If you run Analysis in foreground, your TSO session is unavailable until the processing has completed.

The Analysis JCL Processing Interface panel is displayed, as shown in Figure 40 on page 99.

**Figure 40: Analysis JCL Processing Interface panel**

```
DEHJ ---------- Analysis JCL Processing Interface ---------
Command   ===>  
WORKID . . . : MVSJXE1.JXE0308A

Specify Dataset Names
  JCL . . . . . . 'ACM.DB2V11.V121.ANLYJCL(JXE0308A)'
  Worklist . . . . 'ACM.DB2V11.V121.WLBASE01(JXE0308A)'
  Diagnostics . . . SYSOUT

Select batch processing options. Then press Enter.
S Override
S Create JCL
S Edit JCL
  Submit JCL
  Edit Worklist
Commands:  BROWSE PREVIOUS HELP END
```

b On the Analysis JCL Processing Interface panel, specify your data set names, and type S to select the processing options. In this example, one of the options for analysis is overridden. Press **Enter**.
The Analysis Worklist Options panel is displayed, as shown in **Figure 41 on page 100.**

**Figure 41: Analysis Worklist Options panel**

| ALUFOOA1  | Analysis Worklist Options | Command ===>
|-----------|---------------------------|-------------
|           | Type information. Then press Enter to continue. |             |
|           | Specify allocation parameters for new datasets. |             |
| Dataset   | Unitname                  | Volume | Priqty | Secqty | Alloc Unit |             |
| Worklist  | SYSDA                     | 15     | 5      | Tracks  |             |
|           | Specify optional Global AUTHID. |             |
| Global AUTHID | . .  |             |         |
|           | Specify sync point frequency. |             |
| Frequency | . . . 10                  |             |
|           | Analyze for ROLE as OWNER. |             |
| ROLEOWN   | . . . N                   |             |
|           | Use spanned records.      |             |
| SPANNEDRECS | . . Y                |             |
|           | Commands: HELP END        |             |

On the Analysis Worklist Options panel, press **Enter** to accept the defaults.

**Tip**

To change the default values, see the section explaining overrides in the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

The Analysis Worklist Command Options panel is displayed, as shown in **Figure 42 on page 100.**

**Figure 42: Analysis Worklist Command Options panel**

| ALUFOAN2  | Analysis Worklist Command Options | Command ===>
|-----------|-----------------------------------|-------------
|           | Type information. Then press Enter to continue or PF12 for previous panel. |             |
|           | Select the elements to include in this worklist. |             |
| S SQL     | S Data | S Clone Data | S AMS | S STORCLAS | S MGMTCLAS | S DATACLAS | S Bind | S DBRMLIB | S Bindauth | S All Packages on Index Create or Alter | Commands: HELP END PREVIOUS |
d. On the Analysis Worklist Command Options panel, press **Enter** to accept the
defaults.

The Analysis Input Stream Parameters panel is displayed, as shown in **Figure 43** on page 101.

**Figure 43: Analysis Input Stream Parameters panel**

```
ALUFOA39 ---------------- Analysis Input Stream Parameters ------------------------
Command ===>>
Type information. Then press Enter to continue or PF12 for previous panel.
Select any of the following analysis parameters.
  _ ENV             Display installation environment
  _ PKEYPROP        Propagate parent key changes to foreign keys
  _ VALWARN         Display errors as warnings for conditions in text
  _ VVALPROP        Propagate table column changes to base views
  _ NOORDERBY       Omit the ORDER BY clause for unloads in the worklist
  _ NOUNLOADEMPTY   Do not unload tables that IBM RUNSTATS indicates are empty
  _ STOPCOMMIT      Generate an AT(COMMIT) statement for every STOP command
  _ DEFERUNIQUEIX   Create unique indexes with DEFER YES
  _ CISIZE4K        Create all VSAM data sets with a CISIZE of 4 KB
  _ PARALLEL        Enable worklist parallelism
  _ SINGLEPHASE     Omit the -STOP command between phases of a migrate worklist
```

Commands:  HELP END PREVIOUS

e. On the Analysis Input Stream Parameters panel, type **S** to select the option
**SINGLEPHASE**. When you migrate within a single subsystem, you normally
do not want to stop between phases in a migrate worklist. (For more
information, see the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.*

f. Press **END**.

The JCL Generation component is invoked and Batch Analysis JCL Generation
is displayed, as shown in **Figure 44** on page 101. The JCL includes the
//ALUIN DD and //AJXPOFIN DD statements (input streams), which
include keywords for processing the worklist. For more information about the
input streams, see the *ALTER and CHANGE MANAGER for DB2 Reference Manual.*

**Figure 44: Batch Analysis JCL**

```
000001 //MVSJXE1A JOB (4110), 'ANALYSIS-JXE030BA',
000002 //  CLASS=A, MSGCLASS=X, MSGLEVEL=(1,1),
000003 //  NOTIFY=MVSJXE1
000004 //***************************************************
000005 //*  CREATED BY :  MVSJXE1
000006 //*  TIMESTAMP  :  08/03/2016.04.52.58
000007 //*  ENVIRONMENT:  ISPF 7.2MVS     TSO
000008 //*  RELEASE    :  V12.01.00 12/05/2016
```
Achieving your goals with ALTER and CHANGE MANAGER

ALTER and CHANGE MANAGER for DB2 User Guide
Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface    105
Press END to return to the Analysis JCL Processing Interface panel, as shown in Figure 45 on page 106.

Figure 45: Analysis JCL Processing Interface panel

DEHJ ------------------------ Analysis JCL Processing Interface -----
Command ===> 

WORKID . . . . : MVSJXE1.JXE0308A

Specify Dataset Names
JCL . . . . . . . . 'ACM.DB2V11.V121.ANLYJCL(JXE0308A)'
Worklist . . . . . 'ACM.DB2V11.V121.WLBASE01(JXE0308A)'
Diagnostics . . . . SYSOUT

Select batch processing options. Then press Enter.
* Override
* Create JCL
* Edit JCL
S Submit JCL
_ Edit Worklist

Commands: BROWSE PREVIOUS HELP END
h. On the Analysis JCL Processing Interface panel, press Enter to submit the JCL.

i. After you submit the job, press Enter to return to the Analysis JCL Processing Interface panel.

The results of Analysis are a diagnostic output file (ALUPRINT) and a worklist.

j. To review the results of the analysis of the work ID (ALUPRINT), you can split your ISPF session or you can use your normal method to review SYSOUT. The last message in the diagnostic output (Figure 46 on page 107) indicates that the worklist completed successfully.

**Figure 46: Analysis diagnostic output**

```plaintext
SSID DEHJ
WORKID MVSJXE1.JXE0308A
SORTDEVT SYSDA
INCLUDE (DATA AMS SQL )
NOREGENIDENTITY
SMSINCLUDE ()
NODBMRMLIB ORDERBY
NOVVALPROP SINGLEPHASE
UNLOADEMPTY NOSTOPCOMMIT
BMCCHECK BMCREORG REORGONLINE DYNREORG REORGALL REORGREF
BMCREBUILD
SYNCPOINT 10
NOROLEOWN
STANDALONESTATS BMCSTATSUPD TABLEALL
BMCCOPY BMCLOAD BMCCUNLOAD
BMCFASTLOAD
NOUNLDCOLL
UNLOADTABLE
SPANNEDRECS
NOBINDONIX
MAXSYSUT 20
DYNUNLD
DYNCOPY
NOPARTCOPY
COPYDDN(COPY01 )
NOUTILCOPY
NOPARALLEL
CLONEDATA
NOUNSENGTR
*BMC396243I DB2 VERSION 11 NFM
*** PTFs APPLIED TO PRODUCT:
*** NONE
***********************************************************************
*** WORKLIST GENERATION 2016-08-03-06.19.36.886624
*** CHANGE MANAGER FOR DB2 VERSION 12.01.00(12/05/2016)
WORKLIST GENERATION EXECUTION FOR WORKID MVSJXE1.JXE0308A ON SSID DEHJ
***********************************************************************
*BMC396243I DB2 VERSION 11 NFM
***********************************************************************
*** THE FOLLOWING MESSAGES ARE WARNINGS ONLY
***********************************************************************
*BMC396343W CREATION OF INDEX-CONTROLLED PARTITIONED OBJECTS MIGHT BE PROHIBITED BY DSNZPARM SETTING PREVENT_NEW_IXCTRL_PART.
***********************************************************************
*** END OF WARNING MESSAGES - WARNINGS WERE ISSUED
***********************************************************************
WORKLIST WAS WRITTEN TO DATASET 'ACM.DB2V11.V121.WLBASE01(JXE0308A)'
WORKLIST GENERATION COMPLETE WORKID MVSJXE1.JXE0308A
```
Figure 47 on page 108 shows the worklist that was written to the data set that you specified on the Analysis JCL Processing Interface panel.

Figure 47: CHANGE MANAGER migrate-type worklist

```
000001 *** CHANGE MANAGER FOR DB2  VERSION 12.01.00(12/05/2016)
000002 *BMC3962431 DB2 VERSION 11 NFM
000003 ************************************************************
000004 *** PTFs APPLIED TO PRODUCT:
000005 *** NONE
000006 ************************************************************
000007 * ALUIN PARMS:
000008 * SSID DEHJ
000009 * WORKID MVSJXE1.JXE0308A
000010 * SORTDEVT SYSDA
000011 * INCLUDE (DATA AMS SQL )
000012 * NOREGENIDENTITY
000013 * SMSINCLDE ()
000014 * NODBRMLIB ORDERBY
000015 * NOVALPROP SINGLEPHASE
000016 * UNLOADEMPTY NOSTOPCOMMIT
000017 * BMCCHECK BMCREORG REORGONLINE DYNREORG REORGALL REORGREF
000018 * BMCREBUILD
000019 * SYNCPOINT 10
000020 * NOROLEOWN
000021 * STANDALONESTATS BMCSTATSUPD TABLEALL
000022 * BMCCOPY BMCLLOAD BMCTUNLOAD
000023 * BMCFASTLOAD
000024 * NOUNLCOLL
000025 * UNLOADTABLE
000026 * SPANNEDRECS
000027 * NOBINDONIX
000028 * MAXSYSUT 20
000029 * DYNUNLD
000030 * DYNCOPY
000031 * NOPARTCOPY
000032 * COPYDDN(COPY01 )
000033 * NOUNITCOPY
000034 * NOPARALLEL
000035 * CLONEDATA
000036 * NOUSENGTR
000037 ************************************************************
000038 -TIME 000050 '2016-08-03-06.19.36.886624'
000039 -SSID 000100 DEHJ
000040 -JCLP 000150 MIGR MSSID DEHJ
000041 -WKID 000200 MVSJXE1.JXE0308A
000042 -ZPRM 000250 PREVENT_NEW_IXCTRL_PART NO
000043 -SYNC 000300 END OF CHECK ZPARM SECTION
000044 -TIME 000050
000045 -SYNC 000300 END OF CHECK ZPARM SECTION
000046 ************************************************************
000047 ************************************************************
000048 -SYNC 000300
000049 *** THE FOLLOWING MESSAGES ARE WARNINGS ONLY
000050 ************************************************************
000051 *** ORIGINATOR OF WORKLIST WAS WORKID MVSJXE1.JXE0308A ON SSID DEHJ
000052 -MIGR 000350 PHASE-1 START OF SECTION TO BE RUN ON SENDING SYSTEM
```

Achieving your goals with ALTER and CHANGE MANAGER
START OF EXECUTION SYNCPPOINT

JCLP BMCD IPARTS 4

BMCD
UNLOAD UNLOADDN (SYSR)
ACTIVE (YES,NO)
ON FAILURE ALL TERMINATE UTILITY
DELETEFILES YES
DISCARDS 1
NULLTYPE T1 NULLCHAR ?
FORMAT BMCLOAD
SELECT *
FROM R1BASIC.T_T01AICP
ORDER BY COLSI_2, COLVC_8
;
OUTPUT SYSR
UNIT SYSDA
DSNAME 'MVSJXE1.DEHJ.JXE0308A.SR000001.P&PART'

UNLOAD TB RIBASIC.T_T01AICP COMPLETE

END OF UNLOAD TABLES SECTION

SET CURRENT SQLID = 'MVSJXE1'
CREATE DATABASE J1BASIC STOGROUP SYSDEFLT BUFFERPOOL BP8K0

SET CURRENT SQLID = USER
END OF DATABASE SECTION

SET CURRENT SQLID = 'MVSJXE1'
CREATE TABLESPACE T01AICP IN J1BASIC NUMPARTS 4
(PART 1 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
(PART 2 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
(PART 3 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
(PART 4 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
BUFFERPOOL BP0 LOCKSIZE ANY LOCKMAX 10 CLOSE NO SEGSIZE 0

SET CURRENT SQLID = USER
CREATE TABLE MVSJXE1.T_T01AICP IN J1BASIC NUMPARTS 4
(PART 1 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
(PART 2 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
(PART 3 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
(PART 4 USING STOGROUP SYSDEFLT PRIQTY 440 SECQTY 220 PCTFREE 0)
BUFFERPOOL BP0 LOCKSIZE ANY LOCKMAX 10 CLOSE NO SEGSIZE 0

SET CURRENT SQLID = USER
END OF CREATE TABLESPACE SECTION
k  Press END.

The Execution JCL Build Interface panel is displayed, as shown in Figure 48 on page 111.

Figure 48: Execution JCL Build Interface panel

ACMFEXC0 ------------------ Execution JCL Build Interface ------------------
Command ===>

WORKID . . . . : MVSJXE1.JXE0308A
Select JCL and run type. Then press Enter.
5 Execute the worklist.

The Execution component executes the migrate worklist in a single phase. First, the data is unloaded from the objects. Then, the objects are created, the data is loaded, and the utilities are executed.

The Execution component of the product uses the worklist as its input job stream and performs the tasks contained in the worklist. The worklist acts as a detailed procedure for implementing the changes that you defined in the specification stage.

a On the Execution JCL Build Interface panel, press Enter to accept the defaults.

The Execution Pre- and Post-Processing Interface panel is displayed, as shown in Figure 49 on page 112.

**Figure 49: Execution Pre- and Post-Processing Interface panel**

```
ACMFEXC1 ---------- Execution Pre- and Post-Processing Interface ---------------
Command ===> 
WORKID . . . . : MVSJXE1.JXE0308A
Specify additional steps to be included in this job. Then press Enter.
Pre-Execution Compare
  1. Do not do a compare before execution
  2. Compare two previous baselines
  3. Compare a previous baseline to the current catalog
Pre-Execution Baseline
  1. Do not build a baseline before execution
  2. Build a baseline before execution
Post-Execution Compare
  1. Do not do a compare after execution
  2. Compare a previous baseline to the new catalog structures
  3. Compare the baseline built during a previous job step to the new catalog structures
  4. Create CDL to fall back to a previous baseline
Post-Execution Baseline
  1. Do not build a baseline after execution
  2. Build a baseline after execution
Commands: HELP PREVIOUS END
```

b On the Execution Pre- and Post-Processing Interface panel, press Enter to accept the defaults.
The Execution JCL Processing Interface panel is displayed, as shown in Figure 50 on page 113.

**Figure 50: Execution JCL Processing Interface panel**

```
DEHJ ------------------- Execution JCL Processing Interface -------------------
Command ===> 

WORKID . . . . : MVSJXE1.JXE0308A

Specify Dataset Names.
Execution JCL . . 'ACM.DB2V11.V121.EXECJCL(JXE0308A)'
Worklist . . . . 'ACM.DB2V11.V121.WLBASE01(JXE0308A)'
Diagnostics . . . SYSOUT

JCL Build Options.
Dataset Sizing : No Sizing        (use Override Defaults to set or change)

Select Processing Options. Then press Enter to continue.
- Override Defaults
- Edit Worklist
S Build Execution JCL
S Edit Execution JCL
- Submit Execution JCL

Commands: BROWSE VIEWX PREVIOUS HELP END
```

c On the Execution JCL Processing Interface panel, specify your data set names, specify SYSOUT for diagnostics, accept the defaults for your build option, and type S to select your processing options. Press Enter.

The Execution JCL is displayed. The JCL includes the //AEXIN DD statement (input stream), which includes keywords that are used in executing the JCL. For more information about the input stream, see the ALTER and CHANGE MANAGER for DB2 Reference Manual.

**Figure 51: Execution JCL for a migrate-type worklist**

```
000001 //MVSJXE1E JOB (4110), 'EXECUTE-JXE0308A'.
000002 //  CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1).
000003 //  NOTIFY=MVSJXE1
000004 /*ROUTE XEQ BMCPLX1
000005 /*JOBPARM SYSAFF=DB2B
000006 //***************************************************************************
000007 /*  CREATED BY :  MVSJXE1
000008 /*  TIMESTAMP  :  08/03/2016.07.06.21
000009 /*  ENVIRONMENT:  ISPF 7.2MVS   TSO
000010 /*  RELEASE    :  V12.01.00 12/05/2016
000011 /*  DB2 VERSION:  11015
000012 //***************************************************************************
000013 //---------------------------------------------------------------------
000014 //        BMC CHANGE MANAGER EXECUTION
000015 //---------------------------------------------------------------------
000016 //STEP1 EXEC PGM=AEXEMAIN,
000017 //  REGION=OM,MEMLIMIT=NOLIMIT,PARM='DMC1QEHJ',
000018 //  COND=(4,LT)
000019 //STEPLIB DD DSN=CSG.DEHJ.DSNEXIT,DISP=SHR
000020 // DD DSN=CSGI.DB2V11M.DSNLOAD,DISP=SHR
000021 //ABNLIBGR DD DUMMY
000022 //SYSDUMP DD SYSOUT=* 
000023 //***************************************************************************
```
SAMPLE SYSMDUMP JCL

Please replace the DSN and UNIT with ones that fit your shop standards and uncomment the JCL.

SYSMDUMP DD DSN=CUSTOMER.SYSMDUMP,DISP=(,CATLG,CATLG),UNIT=XXXXXXX,
SPACE=(CYL,(200,50),RLSE),
DCB=(DSORG=PS,RECFM=FBS,LRECL=4160,BLKSIZE=24960)

**********************************************************************

SYSTEMR DD SYSOUT=*  
ASUSRPRT DD SYSOUT=*  
*--------------------------------------------------------------------

* STATISTICS WORK DD STATEMENTS
*--------------------------------------------------------------------

STPRIN01 DD SYSOUT=*  
UTPRINT  DD SYSOUT=*  
RNPRIN01 DD SYSOUT=*  
DDLOUT  DD SYSOUT=*  
SYSOUT  DD SYSOUT=*  
AJXERR  DD SYSOUT=*  
AJXIN   DD *
ACM
SSID DEHJ WORKID MVSJXE1.JXE0308A  
DYNWORKUNIT SYSDA
DASDDOPT DSA2QDOP  
CATDOPT DC112SMP
STOPWAIT 3  
STOPWTSECS 10

LOCK EXCLUSIVE

SYSSIN DD SPACE=(CYL,(15,15)),UNIT=SYSDA,DISP=(NEW,DELETE),
DSORG=PS,LRECL=80,BLKSIZE=3200,RECFM=FB

SYSTSN DD SPACE=CTRK,(1,1)),UNIT=SYSDA,DISP=(NEW,DELETE),
DSORG=PS,LRECL=80,BLKSIZE=3200,RECFM=FB

SYSPRINT DD SPACE=(CYL,(15,15)),UNIT=SYSDA,DISP=(NEW,DELETE)

SYSERROR DD SYSOUT=*  
OBJREPRT DD SYSOUT=*  
SUMMARY  DD SYSOUT=*  
RUNSTATS DD SYSOUT=*  
UTLPARMS DD *
+HRECALL(YES)  
+SVRDISK(2)
+OVERRIDEOUTPUT(XDSN)

RRGPARMS DD *
+RUNSTATS(NO)
+LOB(YES)
+DBNAME(DSNDB04)
+ALLO(KEEPDB2)

RRGPARMX DD *
+ALLO(KEEPDB2)

LODPARMS DD *
+ALLO(KEEPDB2)

*** RUNTIME ENABLEMENT CHOSEN

SYSPROC DD DSN=AEX.QA1120.BMCCLIB,DISP=SHR

**********************************************************************

COMING FROM AJXCLIBU IN AEX.VC1Q.SLIB
UPDATED 04/06/2015 SKH

**********************************************************************
Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  115
d After you review the JCL, press END to return to the Execution JCL Processing Interface panel.

e From the Execution JCL Processing Interface panel, press Enter to submit the JCL and migrate your objects.

The result of execution is a worklist execution log (AEXPRINT).

f To review the results of execution in the worklist execution log (AEXPRINT), you can split your ISPF session or you can use your normal method to review SYSOUT. Figure 52 on page 116 shows the worklist execution log.

The last message in the worklist indicates that the work ID status is complete and that the worklist completed successfully. The worklist migrated the objects that you specified.

Figure 52: Execution results for a migrate-type worklist
Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  117

Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER

ALTER and CHANGE MANAGER for DB2 User Guide

***** B M C U N L O A D   P L U S   F O R   D B 2   V11R2.00 *****

1

***** B M C   E X E C U T I O N   F O R   D B 2 ****

( C ) COPYRIGHT 1991 - 2017 BMC SOFTWARE, INC.

2016-08-03 07.16.55 ON SYSID DB2B

WORKLIST EXECUTION LOG

PAGE 5

---

2016-08-03 07.16.55 ON SYSID DB2B

WORKLIST EXECUTION LOG

PAGE 6

---
Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER

Chapter 2  Overview of the ALTER and CHANGE MANAGER interface  121
BMC50470I MAXEXTSZ = ((0,K),(0,K))                      ((0,K),(0,K))                        ((0,K),(0,K))
BMC50470I EXPDT =
BMC50470I RETPD =
BMC50470I EXPDT =
BMC50470I GDGEMPTY = NO                                 NO                                   NO
BMC50470I GDGLIMIT = 5                                  5                                    5
BMC50470I DDTYPE   = LOCBXCPY                           REMPXCPY                             REMBXCPY
BMC50470I ACTIVE   = NO                                 NO                                   NO
BMC50470I IFALLOC  = USE                                USE                                  USE
BMC50470I SMS      = NO                                 NO                                   NO
BMC50470I SMSUNIT  = NO                                 NO                                   NO
BMC50470I SIZEPCT  = (100,100)                          (100,100)                            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)                (SYSALLDA,SYSALLDA)                  (SYSALLDA,SYSALLDA)
BMC50470I UNITCNT  = (0,0)                              (0,0)                                (0,0)
BMC50470I AVGVOLSP = ((30000,TRK),(30000,TRK))          ((30000,TRK),(30000,TRK))            ((30000,TRK),(30000,TRK))
BMC50470I DSNTYPE  = (NONE,NONE)                        (NONE,NONE)                          (NONE,NONE)
BMC50470I DATACLAS = (NONE,NONE)                        (NONE,NONE)                          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)                        (NONE,NONE)                          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)                        (NONE,NONE)                          (NONE,NONE)
BMC50470I THRESHLD = 0                                  0                                    0
BMC50470I MAXEXTSZ = ((0,K),(0,K))                      ((0,K),(0,K))                        ((0,K),(0,K))

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SYSMAP     DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCPLCPY   DSNPAT=&UID.&DDNAME.&TS..T&TIME
BMC50483I LOCBLCPY   DSNPAT=&UID.&DDNAME.&TS..T&TIME
BMC50483I REMPLCPY   DSNPAT=&UID.&DDNAME.&TS..T&TIME
BMC50483I LOCPXCPY   DSNPAT=&UID.&DDNAME.&TS..F&PART..T&TIME

BMC50471I DB2 DSMHDCEP MODULE SETTINGS:
BMC50471I VERSION = 1110
BMC50471I SUBSYSTEM DEFAULT = DEHJ
BMC50471I DATE FORMAT = USA
BMC50471I TIME FORMAT = USA
BMC50471I LOCAL DATE LENGTH = 0
BMC50471I LOCAL TIME LENGTH = 0
BMC50471I DECIMAL POINT = PERIOD
BMC50471I DECIMAL ARITHMETIC = 15
BMC50471I DELIMITER = DEFAULT
BMC50471I ENCODING SCHEME = EBCDIC
BMC50471I MIXED = NO
BMC50471I EBCDIC CCSID = (37,65534,65534)
BMC50471I ASCII CCSID = (819,65534,65534)
BMC50471I UNICODE CCSID = (367,1208,1200)
BMC50471I IMPLICIT TIME ZONE = CURRENT (-05:00)
BMC50482I DB2 MODE = NFM
BMC50471I BMC_BMCUTIL = 'BMCUTIL.CMN_BMCUTIL'
BMC50471I BMC_BMCSYNC = 'BMCUTIL.CMN_BMCSYNC'
BMC50471I BMC_BMCDICT = 'BMCUTIL.CMN_BMCDICT'
BMC50471I BMC_BMCUTIL = 'BMCUTIL.CMN_BMCUTIL'
BMC50471I BMC_BMCSYNC = 'BMCUTIL.CMN_BMCSYNC'
BMC50471I BMC_BMCDICT = 'BMCUTIL.CMN_BMCDICT'
BMC50102I  LOAD DATA  REPLACE
BMC50102I  INDSN('MVSJXE1.DEHJ.JXE0308A.SR000001.P001'
BMC50102I  ,'MVSJXE1.DEHJ.JXE0308A.SR000001.P002'
BMC50102I  ,'MVSJXE1.DEHJ.JXE0308A.SR000001.P003'
BMC50102I  ,'MVSJXE1.DEHJ.JXE0308A.SR000001.P004'
BMC50102I  )
BMC50102I  DISCARDDN SYSDS001
BMC50102I  DELETEFILES NO
BMC50102I  ORDER YES
BMC50102I  LOAD DATA REPLACE
BMC50102I  INTO TABLE MVSJXE1.T_TOLAIACP
BMC50102I  WHEN TABLE = 1
BMC50156I  CALCULATED NUMBER OF INPUT RECORDS = 814 USING A SCALING FACTOR OF 100%
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

1                                         **** B M C   E X E C U T I O N   F O R   D B 2 ****
1                                         **** V E R S I O N:  V12.01.00
2016-08-03 07.17.06 ON SYSID DB2B                      WORKLIST EXECUTION LOG                                               PAGE  16
2016-08-03 07.17.06 ON SYSID DB2B                      WORKLIST EXECUTION LOG                                               PAGE  16

**** B M C   E X E C U T I O N   F O R   D B 2 ****
**** V E R S I O N:  V12.01.00
(C) COPYRIGHT 1991 - 2017 BMC SOFTWARE, INC.
Chapter 2  Overview of the ALTER and CHANGE MANAGER interface   125

Achieving your goals with ALTER and CHANGE MANAGER
Achieving your goals with ALTER and CHANGE MANAGER

**ALTER and CHANGE MANAGER for DB2 User Guide**

---

126 ALTER and CHANGE MANAGER for DB2 User Guide
Press END to return to the Migrate DB2 Objects Using Specification panel.

Press END to return to the Task List Menu panel.

Press END to return to the CHANGE MANAGER Main Menu.

This example shows that with CHANGE MANAGER you have the ability to quickly and accurately specify objects that you want to migrate to another environment, perform an in-depth analysis of the objects selected to determine their effect on your subsystem, and execute SQL statements, utilities, and other commands.
Setting up ALTER and CHANGE MANAGER

This section describes how to set options for how ALTER and CHANGE MANAGER operate, restrict access to the components of the products, and maintain the tables used by the products.

Before you set up the ALTER or CHANGE MANAGER product for the ISPF interface, ensure that you have completed the tasks necessary to install the products. For information, see:

- *BMC Products and Solutions for DB2 Customization Guide*
- *Installation System Quick Start*
- *Installation System Reference Manual*

In addition, configure a TSO region size of 32 MB or greater, for each user who signs on.

Using options to control your environment

For your production requirements, you should reevaluate the option settings for ALTER or CHANGE MANAGER. For most situations, the installation default values are sufficient. However, you might want to customize the operating environment and panels.

ALTER or CHANGE MANAGER uses default, user, and product options to define the operating environment and to specify how the product’s components work. The options also contain default values for data set names and allocations, job control language (JCL) generation information, and component plan names.

These options provide you with the ability to:

- Tailor the interface
Set up defaults for generating job statements

Specify defaults for parameters, names, and prefixes for allocated data sets

Specify processing options for the Analysis, JCL Generation, Execution, Import, Baseline, and CM/PILOT components

Typically, the person who installs ALTER or CHANGE MANAGER sets default values for user options. For information about establishing installation option values at installation, see:

- BMC Products and Solutions for DB2 Customization Guide
- Installation System Quick Start
- Installation System Reference Manual

Using the installation options

The default operating environment is controlled by a number of option values that are defined in the installation options module.

The components of ALTER or CHANGE MANAGER use the global values that are stored in the installation options module to determine how to process information. The Installation System generates the installation options module when you install ALTER or CHANGE MANAGER. The module contains an assembly-language program with an options macro.

You can customize ALTER or CHANGE MANAGER for all users by editing the default values in the installation options module. The default name of the module is ACMDOP1. The source of the installation options module is located in the HLQ.UBMCCNTL data set. HLQ identifies the high-level qualifier that you specify when you install the products.

**Note**
ALTER or CHANGE MANAGER uses plan names directly. If you need to specify different plan name values for each DB2 subsystem, you must have multiple installation option modules.

To display the installation options

1. From the Main Menu panel, enter **DOPTS** on the **Command** line to display the installation options that are in effect for the current CHANGE MANAGER session.
Using the user options

The first time that you run ALTER or CHANGE MANAGER, the product copies the values from the installation options module into your ISPF profile.

The values in this profile are referred to as your user options. The product uses these user options to generate JCL and to generate keywords for an input stream for each user’s subsystem. The product uses the following input streams:

- AEXIN input stream, which is used by the Execution component
- AEXPIN, which is used by the Execution component for the worklist parallelism feature in the Database Administration and BMC Object Administration for DB2 solutions
- ALUIN, which is used by the Analysis, Baseline, Baseline Report, Compare, CM/Pilot, and Import components
- AJXIN and AJXPOFIN, which are used by the Batch Execution JCL Generation component

Note
AJXPOFIN is also used by Batch Analysis JCL Generation.

You can refresh the values in your user options by editing and reassembling the installation options module.

Storing values in the ISPF profile

The ISPF profile resides in the data set member prdxPROF, where prdx is the value of the application ID in the BMCDB2 CLIST. To define or modify the values in this profile, you can use the Options panels of ALTER or CHANGE MANAGER, JCL Generation, or CM/PILOT. You can use literal characters or symbolic variables to specify the values on the Options panels. (For more information about symbolic variables, see the ALTER and CHANGE MANAGER for DB2 Reference Manual.)

1 To save your user options in your profile, exit the product.

Note
If your ISPF session abnormally ends (abends), the user options that you modified are not saved.

To refresh an option value in all existing ISPF profiles

You can refresh the values in your user options by editing and reassembling the installation options module.
1 Type a comma and an R after the option value and then enclose the value in parentheses, as shown in the following example:

\[
\text{DBCS=(N,R),} \\
\ast
\]

*Note*

Do not remove the comma after the right parenthesis or the continuation character (\*) in column 72, except for the last option value.

The next time that you run the product, the new global value replaces the old local value in the user options. You can modify the local value through the options panels. If you need to change the installation options after installation, you must reassemble the installation options module.

For more information about refreshing user options, see:

- *Installation System Reference Manual*
- *Installation System Quick Start*
- *BMC Products and Solutions for DB2 Customization Guide*

**Using the product options**

The POFDS keyword (in the installation options module) specifies an 80-character sequential file. This product options file (POF) is built during product installation and contains parameters and values for the JCL Generation, ALTER, and CHANGE MANAGER options. The file is located in the \texttt{HLQ.UBMCCNTL} data set. The POF does not require assembly and linkage and does not need to reside in an APF-authorized data set.

The Batch Execution JCL Generation component uses the AJXIN keyword, POFDS, to specify the POF data set. Batch Execution JCL Generation can use an input stream with the ddname of AJXPOFIN to override keywords in the data set that the POFDS keyword specifies. For more information, see *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

When you install the products, only one POF is created. This initial POF is initialized and populated with the default ISPF variables and values from the installation panels. This POF is shared among several products, if those products are installed at the same time.

In addition, the Installation System uses the same application ID (or profile) for the products in the BMCDB2 CLIST. This single application ID enables the JCL Generation options to be shared with other products, such as the CATALOG...
MANAGER product. Thus, when you specify an option for generating JCL in one product, your selection applies to all of the products.

Note
Although BMC recommends that you use a single application ID, you can choose individual product application IDs on the BMCDB2PR panel.

JCL Generation also handles user POFs, which are POFs that can be written from the ISPF variables that are set in the Front End component of ALTER or CHANGE MANAGER or edited. You can use a user POF to reset all of the options that you will use in the current session to create JCL. You can also use the user POFs to set options for different sets of applications, particularly if the applications have different naming standards.

Note
If a POF keyword in your user POF uses a library from an earlier version of the product, update the keyword to use a library for the most recently installed version of the product. For example, assume that the value of the BMC_COPY_LOAD keyword is BMC1120.ACM.D11.LOAD and you install version 12.1.00 of the product. Update the value of the keyword to a version 12.1.00 LINK library (BMC1210.ACM.D12.BMCLINK).

- The first time that the product is invoked, all of the values in the ISPF profile are set to the values that are in the initial POF. If a POF is not specified, default values are assigned to the variables in the profile.

- If the POFODATE parameter in the initial POF is greater than the value of the POF date that is stored in the ISPF profile, the values in the POF that are marked with refresh ,(R) are used to reset the ISPF variables.

- If you specify a new initial POF in the POFDS installation option, the values in the POF that are marked with refresh ,(R) are used to reset the ISPF variables. The value of the POFODATE keyword in the new initial POF is saved in the ISPF profile.

For more information about POFs, see the *ALTER and CHANGE MANAGER for DB2 Reference Manual*.

Product option overrides

You can override the user options for an ALTER or CHANGE MANAGER component by using the Override Options panels for that component.

For example, you can override the product options for JCL Generation by using the Override Options panels for Execution. (You can also override the options that are
related to dynamic allocation for JCL Generation in the same panels for Analysis.)

The override options remain in effect only for the current ALTER or CHANGE
MANAGER session.

The Front End uses the override options, as well as the user options, to generate the
input streams for the other components. The local keywords in an input stream can
override some of the values in the POF and ISPF profile. For example, when
foreground processing is selected, the Analysis component’s ALUIN input stream is
generated when you select Create Input from the Analysis JCL Processing Interface
panel.

---

**Note**

If you do not use the Front End component to run other components, you must
explicitly specify (in each component’s input stream in the JCL) any local values that
you want to use that are different from the global installation option values that are
defined at installation.

---

### Additional installation options module

When you issue the DOPTS command on the **Command** line of any ALTER or
CHANGE MANAGER panel, the products invoke an ISPF browse session.

In this browse session, the installation option keywords and values are displayed.

---

**Note**

This list of installation option keywords does not include the keywords that the
keywords in the POF replace.

---

You can copy the keywords in the browse session to a permanent data set. In your
stand-alone installation options job in the installation JCL data set, specify the name
of this permanent data set as the data set name in the //ASM.SYSIN DD card.
How ALTER or CHANGE MANAGER use options

Figure 53: How ALTER or CHANGE MANAGER use options

* The only input stream that you can edit is the ALUIN input stream in foreground Analysis
Data set allocation

When you use the Front End to run components of ALTER or CHANGE MANAGER, you specify data sets for JCL, diagnostic output, worklist files, and report files.

For CHANGE MANAGER, you also specify data sets for Change Definition Language (CDL) files.

When you specify these data sets, use the guidelines in Table 3 on page 136.

**Note**

You must predefine partitioned data sets (PDSs). If a sequential file or member of a PDS does not exist, the product dynamically creates or allocates it.

### Table 3: Guidelines for data sets

<table>
<thead>
<tr>
<th>Data set</th>
<th>Type</th>
<th>Record format</th>
<th>Record length</th>
</tr>
</thead>
<tbody>
<tr>
<td>JCL</td>
<td>Sequential file</td>
<td>Fixed length (FB)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>PDS member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diagnostic output</td>
<td>Sequential file</td>
<td>Variable length (VB)</td>
<td>137 characters</td>
</tr>
<tr>
<td>Worklist</td>
<td>Sequential file</td>
<td>Fixed length (FB)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>PDS member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CDL file</td>
<td>Sequential file</td>
<td>Fixed length (FB)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>PDS member</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Report</td>
<td>Sequential file</td>
<td>Fixed length (FBA)</td>
<td>80 characters</td>
</tr>
<tr>
<td></td>
<td>PDS member</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Object names

You must name an ALTER or CHANGE MANAGER object when you create it.

Work IDs, profiles, and baselines have a two-part name in the following format:

```
owner.identifier
```

The *owner* can be from one to eight characters, and the *identifier* can be from one to 18 characters. The two parts are subject to the same restrictions as DB2 identifiers: you cannot use delimited identifiers or double-byte character set (DBCS) characters.
For information about delimited identifiers, see “Delimited identifiers” on page 139. If you do not explicitly specify the owner, the owner defaults to your user ID.

Note
The implicit owner for a baseline is the owner of the profile that was used to create the baseline.

Wildcard patterns

Wildcard patterns are used throughout the product to specify a partial match in the name of an object.

Table 4 on page 137 describes how wildcard patterns can be used in various components of ALTER and CHANGE MANAGER.

Table 4: Using wildcards

<table>
<thead>
<tr>
<th>Component</th>
<th>Use of wildcards</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specification</td>
<td>To produce lists of objects</td>
</tr>
<tr>
<td>Scope rules</td>
<td>In object names, to include lists of objects under a single scope rule</td>
</tr>
<tr>
<td>Change rules</td>
<td>To find object names that changes are applied to and to supply old and new change rule values</td>
</tr>
</tbody>
</table>

Table 5 on page 137 lists the wildcard characters that are supported for the Specification filters.

Table 5: Wildcard characters for Specification filters

<table>
<thead>
<tr>
<th>Wildcard character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>? (question mark)</td>
<td>Matches any single character, if the data type of the object is fixed length or variable length</td>
</tr>
<tr>
<td>_ (underscore)</td>
<td>For example, a value of AB?D or AB_D matches ABCD or AB1D, but not ABD.</td>
</tr>
</tbody>
</table>
Wildcard character | Description
--- | ---
% (percent sign) | Matches any string of zero or more characters, depending on the data type of the object. If the data type of the object is fixed length, an embedded wildcard matches any single character. For example, a value of %DB or *DB for a database name matches ADB or BDB, but not DB or XYZDB. A trailing wildcard matches a string of zero or more characters. For example, a value of %DB* for a database name matches ADB or ADB123, but does not match DB123. If the data type of the object is variable length, an embedded or a trailing wildcard matches a string of zero or more characters. For example, a value of *TB or %TB matches TB, ATB, or ABTB, but not ABTBZ.
* (asterisk) | Matches any string of zero or more characters. For example, a value of AB*D or AB%D matches ABCD, AB123D, or ABD, but not AB.

Table 6 on page 138 lists the wildcard characters that are supported for scope rules.

Table 6: Wildcard characters for scope rules

<table>
<thead>
<tr>
<th>Wildcard character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>? (question mark)</td>
<td>Matches any single character. For example, a value of AB?D or AB_D matches ABCD, AB1D, but not ABD.</td>
</tr>
<tr>
<td>_ (underscore)</td>
<td>Matches any single character. For example, a value of AB?D or AB_D matches ABCD, AB1D, but not ABD. You can create an underscore (_) as a literal instead of a wildcard by preceding it with a back slash ().</td>
</tr>
<tr>
<td>% (percent sign)</td>
<td>Matches any string of zero or more characters. For example, a value of AB*D or AB%D matches ABCD, AB123D, or ABD, but not AB.</td>
</tr>
<tr>
<td>* (asterisk)</td>
<td>Matches any string of zero or more characters. For example, a value of AB*D or AB%D matches ABCD, AB123D, or ABD, but not AB.</td>
</tr>
</tbody>
</table>

Table 7 on page 138 lists the wildcard characters that are supported for change rules.

Table 7: Wildcard characters for change rules

<table>
<thead>
<tr>
<th>Wildcard character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>? (question mark)</td>
<td>Matches any single character. For example, a current value of AB?D or AB_D matches ABCD, AB1D, but not ABD. You can create an underscore (_) as a literal instead of a wildcard by preceding it with a back slash ().</td>
</tr>
<tr>
<td>_ (underscore)</td>
<td>Matches any single character. For example, a current value of AB?D or AB_D matches ABCD, AB1D, but not ABD. You can create an underscore (_) as a literal instead of a wildcard by preceding it with a back slash ().</td>
</tr>
<tr>
<td>% (percent sign)</td>
<td>Matches any string of zero or more characters. For example, a current value of AB*D or AB%D matches ABCD, AB123D, or ABD, but not AB.</td>
</tr>
<tr>
<td>* (asterisk)</td>
<td>Matches any string of zero or more characters. For example, a current value of AB*D or AB%D matches ABCD, AB123D, or ABD, but not AB.</td>
</tr>
</tbody>
</table>

Table 8 on page 139 shows some examples of valid and invalid wildcard uses.
### Table 8: Examples of valid and invalid wildcard use for change rules

<table>
<thead>
<tr>
<th>Current value</th>
<th>New value</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB*DE</td>
<td>X*ABC</td>
<td>Valid; wildcards match</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: Any characters found in the current value between the B and D characters are inserted between the X and A characters of the new value.</td>
</tr>
<tr>
<td>AB*DE</td>
<td>X<em>A</em></td>
<td>Invalid; number of wildcards do not match</td>
</tr>
<tr>
<td>AB<em>DE</em></td>
<td>X*ABC</td>
<td>Invalid; number of wildcards do not match</td>
</tr>
<tr>
<td>AB*DE</td>
<td>ABPROD</td>
<td>Valid; no wildcards in new value</td>
</tr>
<tr>
<td>DBTEST</td>
<td>AB*DE</td>
<td>Invalid; no wildcards in original value</td>
</tr>
<tr>
<td>A*B_</td>
<td>X_Y*</td>
<td>Invalid; wildcards out of order</td>
</tr>
</tbody>
</table>

### Delimited identifiers

ALTER or CHANGE MANAGER support the use of delimited identifiers for DB2 objects, but not for BMC objects, such as work IDs, baselines, baseline profiles, and migrate profiles.

The double quotation mark ("" is the only escape character that is supported.

Delimited identifiers can also be used in data definition language (DDL), CDL, worklists, Data Manipulation Language (DML), and for reserved words. ALTER and CHANGE MANAGER evaluate the identifiers to determine whether delimiters are required. If delimiters are not required, they are removed.

Delimited identifiers are supported to specify owners of DB2 objects and to specify the names of certain types of DB2 objects. You can use delimited identifiers to specify DB2 object names that use long identifiers. These objects include tables, table columns, indexes, aliases, synonyms, views, view columns, triggers, and trigger columns. Names for databases and table spaces cannot be named using delimited identifiers. Owners of these DB2 objects, however, can be named using delimited identifiers. The following example illustrates the use of a delimited identifier:

```
000008 -SQL  000450 CREATE TABLE NCH.TBLARGE
000009                      ("SYNONYM"      CHAR(15) NOT NULL,
```

**Note**

Twenty bytes are allowed for object names and ten bytes for owner names specified with delimited identifiers; two bytes are used for the escape characters. If you use delimited identifiers to specify names, embedded double quotation marks within a name might cause truncation. Two consecutive double quotation marks are required to represent one double quotation mark within a delimited identifier.
Wildcard characters that are specified within delimiters are considered wildcard characters, not literal characters. The asterisk (*) specified for a STOGROUP volume ID is supported for users of the IBM Storage Management Subsystem (SMS).

For additional information about using delimited identifiers, see the IBM documentation.

Nonprintable or nonviewable characters

You must specify nonprintable and nonviewable characters (such as null and control characters) in limit keys, view text, trigger text, or check constraint text in an external hexadecimal format.

These characters are not indicated in literal strings. For example, the EBCDIC string '2 1' does not indicate that the second character is null; the string appears to be blank. The string should be specified as X'F200F1'.

Table 9 on page 140 provides the hexadecimal formats for common nonprintable or nonviewable characters.

Table 9: Nonprintable or nonviewable characters

<table>
<thead>
<tr>
<th>Character</th>
<th>Hexadecimal format</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null</td>
<td>X'00'</td>
</tr>
<tr>
<td>Horizontal tab ('HT')</td>
<td>X'05'</td>
</tr>
<tr>
<td>Form feed</td>
<td>X'0C'</td>
</tr>
<tr>
<td>Carriage return</td>
<td>X'0D'</td>
</tr>
<tr>
<td>New line ('NL')</td>
<td>X'15'</td>
</tr>
<tr>
<td>Line feed ('LF')</td>
<td>X'25'</td>
</tr>
</tbody>
</table>

Note the following items about nonviewable characters:

- If a value in a limit key field in a partitioned index contains a nonviewable character, you cannot edit that value in the Specification component.
- If null characters exist in the internal format of a value for an object, the Baseline, Import, and Compare components issue a warning message.
Access to components

Maintaining security over the components is an important consideration.

Executing a worklist can change the definitions of your DB2 objects as they are defined in the DB2 system catalog tables. To control access to the components, you control the authorization that is granted to plans for the components.

Plans that are included with ALTER and CHANGE MANAGER provide access to the Front End, Import, Specification, Analysis, Baseline, Compare, and Execution components. You can control whether users can access ALTER and CHANGE MANAGER by controlling the authorization granted to each of these plans.

ALTER and CHANGE MANAGER plan names have the format prdvryz, where prd is the product code, vr is the version and release, y is the access type, and z is the component.

The Execution component plan names have the format prdvrnn, where prd is the product code, vr is the version and release, and nn is a unique plan identifier.

ALTER and CHANGE MANAGER plans

ALTER and CHANGE MANAGER use plans to provide access to the Front End, Import, Specification, and Analysis components.

Table 10 on page 141 lists the plans that the components of ALTER and CHANGE MANAGER use.

Table 10: ALTER and CHANGE MANAGER plans

<table>
<thead>
<tr>
<th>Plan name</th>
<th>Component name</th>
<th>Plan description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACMvrYa</td>
<td>Analysis</td>
<td>Enables users to access the Analysis component to analyze changes and generate worklists. Users cannot make changes to DB2 objects and data by creating a worklist. However, the Specification and Analysis plans provide a way for users to request and analyze changes. For this reason, administrators can usually place minimum restrictions on using Specification and Analysis.</td>
</tr>
<tr>
<td>ACMvrYF</td>
<td>Environment</td>
<td>Enables users to use the ENVI command to review the system environment. Because this plan is not accessed outside of the other components, administrators can usually grant PUBLIC access to this plan. This plan does not control the use of the ENV keyword with batch components.</td>
</tr>
<tr>
<td>Plan name</td>
<td>Component name</td>
<td>Plan description</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------</td>
<td>------------------</td>
</tr>
<tr>
<td>ACMvryF</td>
<td>Front End</td>
<td>Controls access to the ISPF interface. Authority to use this plan enables users to create and maintain the product’s objects and perform other Front End functions. If the Distributed Data Facility (DDF) is installed, the -BIND statement for this plan adds a reference to the remote package list.</td>
</tr>
<tr>
<td>ACMvryI</td>
<td>Import</td>
<td>Enables users to import files into the Change Definition (CD) tables. This plan accesses the Import component. For ALTER, the plan enables users to import data definition language (DDL) into the CD tables. For CHANGE MANAGER, it enables users to import DDL, Change Definition Language (CDL), and worklists into the CD tables.</td>
</tr>
<tr>
<td>ACMvryS</td>
<td>Specification</td>
<td>Enables users to request changes to database objects and data. This plan accesses the Specification component. With access to Specification, users can also search and review groups of objects. <strong>Note:</strong> While Specification does not enable users to actually perform changes, it does enable them to specify changes and to view existing data structures. If the Distributed Data Facility (DDF) is installed, the -BIND statement for this plan adds a reference to the remote package list.</td>
</tr>
</tbody>
</table>

### CHANGE MANAGER plans

CHANGE MANAGER uses additional plans to provide access to the Baseline, Compare, and Baseline Report components.

**Table 11 on page 142** lists additional plans that the components of CHANGE MANAGER use.

#### Table 11: CHANGE MANAGER plans

<table>
<thead>
<tr>
<th>Plan name</th>
<th>Component name</th>
<th>Plan description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACMvryB</td>
<td>Baseline</td>
<td>Enables users to establish and delete baselines. Because baselines (especially full-recovery baselines) contain critical information that is used for database recovery, administrators should allow only informed users access to this plan.</td>
</tr>
</tbody>
</table>
### Plan name | Component name | Plan description
---|---|---
ACMvryC | Compare | Enables users to compare data structure definitions. Because no DB2 data structures are modified, administrators can usually grant unlimited access to this plan. If the DDF is installed, the -BIND statement for this plan adds a reference to the remote package list.

ACMvryR | Baseline Report | Enables users to create baseline reports. Grant unlimited access to this plan.

### Execution component plans

You can restrict access to functions of the Execution component by using PLAN authorizations.

Table 12 on page 143 lists the plans that the Execution component uses.

#### Table 12: Execution plans

<table>
<thead>
<tr>
<th>Plan name</th>
<th>Function name</th>
<th>Plan description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AEXvHA (ALTER and CHANGE MANAGER) AEXvDA (DASD MANAGER PLUS)</td>
<td>Execution Monitor Entry (Authorization)</td>
<td>Enables users to execute a worklist when EXECUTE authority is granted. You should carefully consider who receives authorization to use this plan.</td>
</tr>
<tr>
<td>Plan name</td>
<td>Function name</td>
<td>Plan description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AEXvrHM (ALTER and CHANGE MANAGER)</td>
<td>Execution Monitor</td>
<td>Enables users to attach to DB2 with alternate authorization IDs for the -AUTH commands. This plan does not control who has authorization to execute a worklist. Because this plan does not affect who can run Execution, you can grant PUBLIC authority to this plan. The Execution plan contains some packages that use dynamic SQL. Some of these packages cause long-running SQL and might need to be added to your resource limit specification table (RLST). The packages are as follows:</td>
</tr>
<tr>
<td>AEXvrDM (DASD MANAGER PLUS)</td>
<td></td>
<td>▪ AEXAUNLD unloads data from tables.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ AEXSQLIO performs all worklist -SQL commands, including deletions before a -LOAD or -BMCL command that migrates data only.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>▪ AEXESTDL performs some of the restart logic before restarting a -LOAD command, including deleting previously loaded rows.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>By restricting authorization to run the Execution plans, you can control what change and migrate functions users can perform. For example, by granting unlimited access to Specification and Analysis while controlling access to the Execution Monitor Entry, you can allow your users to run ALTER for training purposes or use it as a system dictionary. The Execution Security Exit provides further control over the Execution component’s authorization switching function.</td>
</tr>
</tbody>
</table>

### Authorization switching

The Execution component performs an authorization switching function to ensure that DB2 structures are created with the proper owner and creator.

To secure this function, you can perform one of the following activities:

- Restrict the EXECUTE authority on the Execution Monitor Entry plan. This restriction will prevent unauthorized users from being able to run Execution. For the highest level of security, restrict access to the Execution Monitor Entry plan.

- Create an Execution security exit to add site-specific restrictions to authorization switching.
Execution calls the security exit at various points to allow installation security checking and option enforcement.

Typical uses of this exit include:

- Enforcing hash checking so that users are not allowed to modify worklists
- Switching the authorization ID of the execution run user at initialization
- Preventing or restricting user authorization ID switching
- Preventing user modification of SQL GRANT commands
- Inspecting and modifying SQL commands before they are executed

The assembler source for the security exit is distributed in the DBSAMP library (member ALUEUSX1). To enable the security exit, modify ALUEUSX1, then assemble and link the member. This module is loaded during initialization.

ALUEUSX1 communicates via Registers 0, 1, and 15. When ALUEUSX1 is called, Register 0 contains a function code. Register 1 contains the address of a user exit control block that AEXUSERX DSECT (in the sample program) describes. ALUEUSX1 returns a value in Register 15 that informs Execution how to proceed.

For more information about switching authorization, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

### Execution security exit function codes

The table in this topic lists and describes the Register 0 function codes.

The sample exit includes detailed documentation about register usage, DSECT mapping, and call conventions. For a list of return codes, see the messages library in the BMC Documentation Center (http://www.bmc.com/available/documentation-center.html).
### Table 13: Execution security exit function codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Call</th>
<th>Occurs</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>X'01'</td>
<td>Initialization</td>
<td>During Execution initialization</td>
<td>At this point, you can inspect the user ID that is running the job, change it if necessary, or halt processing. You can also enable hash checking to prevent a user from making any modifications to a worklist (other than deleting entire commands) throughout the execution run. If you need storage for local routines, you should perform a GETMAIN at this time. Store the address in the user exit control block AEXUSERX, as described in the code sample.</td>
</tr>
<tr>
<td>X'02'</td>
<td>-AUTH command switch</td>
<td>Immediately before an authorization ID switch</td>
<td>The authorization ID might be inspected or changed, or processing might be halted. If the AEXEFAUH bit of flag ASXEFLAG in the user exit DSECT is set, hash checking has failed on the current -AUTH command. If the return code from this exit is an odd number (that is, the low order 1 bit is set), the command that executed under the switch is verified to ensure that the hash code is correct. This verification prevents you from changing commands that are executed under a switch. If this bit is off (return code 0 or 4), hash checking is done as specified for the main run. If this bit is on (return code 1 or 5), then hash checking is forced for the duration of the switch. In this case, Execution halts when it encounters a modified or new statement in the worklist. If the return code low order 2 bit is on, the command that is executed under the switch is allowed to fail hash checking provided that the next command is not an SQL GRANT. This restriction means that you can modify the worklist as needed but cannot modify any GRANT statements. The low order 2 bit in the return code indicates whether to enforce hash failures when an SQL GRANT command is processed. Hash checking is enforced for the duration of the switch. In this case, Execution halts when it encounters a modified or new SQL GRANT statement in the worklist.</td>
</tr>
<tr>
<td>X'03'</td>
<td>termination/ cleanup</td>
<td>Immediately before termination</td>
<td>The call enables you to perform any necessary cleanup functions. For example, if you used another security product to perform security validations, you might want to sign off. Remember to perform a FREEMAIN to free any storage that you allocated. Return codes are ignored.</td>
</tr>
</tbody>
</table>
Authorization switching

### Execution security exit testing

Before you enable your security exit, you should test the exit.

Following are some suggestions to help you develop and test your security exit:

- Test the exit initially with simple logic and test cases to verify that you are correctly processing the function codes and return codes.

- Add the DEBUG keyword to the AEXIN parameter input.
  
  Message BMC16990I is issued to AEXPRINT and includes the following information:
— Function code with which the exit is called
— WORKID or SQLID that is about to be switched
— USERID that is associated with the job itself
— Return code (Register 15) that is returned from the exit
— WORKID or SQLID that is returned from the exit

Contact BMC Customer Support if you have questions about the Execution security exit.

Access to the worklist parallelism feature

With the Database Administration or BMC Object Administration *for DB2* solution, you can use the worklist parallelism feature to execute portions of a worklist concurrently.

CHANGE MANAGER uses the BMC Cross-System Image Manager (XIM) technology to provide sysplex performance improvements by enabling the distribution and management of discrete units of work (UOW) across one or more IBM OS/390 and z/OS images.

By default, user access to execute portions of a worklist concurrently and to dynamically start XIM is not restricted. You can control access to these functions for a user or a group of users by performing the following tasks:

1. Apply a zap.

2. If you are using the IBM Resource Access Control Facility (RACF), specify a general resource profile.

*Note*

If you are using another security package that is compatible with the System Authorization Facility (SAF), contact BMC Customer Support.

Zap to restrict access to worklist parallelism

To enable the restriction of access to worklist parallelism functions, you must apply a zap.
Apply the following zap to the Execution function of CHANGE MANAGER:

NAME AEXPMAIN MAINRACC
VER 003E 47F0,C1D8
REP 003E 4700,0000
CHECKSUM 0916482E

General resource profile for worklist parallelism

In RACF, general resource profiles are used to protect the resources that are defined in the class descriptor table, such as programs.

To restrict a user’s or group’s access to each of the worklist parallelism functions, you must add general resource profiles with the following profile information:

- CLASS => FACILITY
- PROFILE => BMCACM.ssid.PARALLEL.object

The profile definition contains the following values:

- BMCACM specifies that the profile is for CHANGE MANAGER.
- ssid represents the name of the DB2 subsystem or a DB2 group attachment name (wildcard characters can be used to match one or more characters).
- PARALLEL represents the function that is secured.
- object represents the object or resource name that is secured.
  - For executing a worklist, the object is EXECUTE.
  - For starting XIM dynamically, the object is DYNSTART.

Each user or group that is given access to a resource profile must have an access level of CONTROL or higher.

Setting the MEMLIMIT system parameter

1. Use any of the following methods to override the default value of the MEMLIMIT value:
   - Specify the MEMLIMIT parameter in the JCL
Specify REGION=0M in the JCL

Use the SMF IEFUSI exit

---

**Note**

The default value for the System Management Facility (SMF) MEMLIMIT parameter is 2 GB. This value is set in member SMFPRMxx in SYS1.PARMLIB.

---

## Modification of ISPF skeletons and variables

The JCL Generation component generates JCL by using a method that improves upon standard ISPF file-tailoring services. The method used is based on ISPF skeletons.

The $AJXDOC member in the $HLQ.BMCSLIB data set lists and briefly describes each non-DD and DD statement skeletons that JCL Generation uses. The member also lists the variables that JCL Generation uses to construct the default names for permanent data sets.

Each skeleton name in the $AJXDOC member has a corresponding member in the $HLQ.BMCSLIB data set. Some of the members that contain skeletons also provide comments, tips, and suggestions for using the skeletons.

You can specify up to five user-defined variables in the POF. The JCL Generation User Defined Variables panel allows you to specify variable names (up to eight characters). Each variable has a corresponding symbolic variable, as shown in Table 14 on page 150. You can use the symbolic variables in your job cards or data set prefixes. For information about specifying the variables, see “Setting user variables” on page 209.

### Table 14: User-defined variables

<table>
<thead>
<tr>
<th>POF variable</th>
<th>Symbolic variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER_VAR1_CHAR</td>
<td>&amp;UVR1</td>
</tr>
<tr>
<td>USER_VAR2_CHAR</td>
<td>&amp;UVR2</td>
</tr>
<tr>
<td>USER_VAR3_CHAR</td>
<td>&amp;UVR3</td>
</tr>
<tr>
<td>USER_VAR4_CHAR</td>
<td>&amp;UVR4</td>
</tr>
<tr>
<td>USER_VAR5_CHAR</td>
<td>&amp;UVR5</td>
</tr>
</tbody>
</table>

The *ALTER and CHANGE MANAGER for DB2 Reference Manual* describes the symbolic and SLIB variables that JCL Generation uses and specifies the length of each variable.

---

150  *ALTER and CHANGE MANAGER for DB2 User Guide*
To improve the performance of the JCL construction phase of JCL Generation, BMC uses an SLIB compiler. Consequently, *if you edit SLIBs after installation, you must recompile them*. For information, see *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

**JCL Generation data sets sizing function**

You can use the JCL Generation data set sizing function to tailor the data set sizes when the JCL is built.

The function gathers information from one of the following sources (shown in general order of accuracy, from most accurate to least accurate):

1. The BMC Statistics repository (statistics that the BMCSTATS or NGT Stats utility gathers)
2. DB2 system catalog (statistics that the IBM RUNSTATS utility gathers)
3. Results of VSAM object sampling
4. Default data set allocation parameters that are set from the JCL Generation Individual Data Set Options panel

Whether using the BMC Statistics repository, object sampling, or no sizing, JCL Generation obtains some information from the DB2 system catalog.

The formulas for estimating data set size are taken from the documentation for the IBM DB2 utilities and from the documentation for the BMC utilities. Table 15 on page 151 shows the statistics that JCL Generation uses for space estimation and the source of the statistics.

<table>
<thead>
<tr>
<th>Source</th>
<th>BMCSTATS</th>
<th>IBM RUNSTATS</th>
<th>VSAM sampling</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Value</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of active pages</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Number of modified pages</td>
<td>X</td>
<td>NA</td>
<td>X</td>
</tr>
<tr>
<td>Page size</td>
<td>NA</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Maximum row length</td>
<td>NA</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Average row length</td>
<td>X</td>
<td>NA</td>
<td>X</td>
</tr>
<tr>
<td>Number of rows</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 15: Data set sizing values and sources
Table: Source

<table>
<thead>
<tr>
<th>Value</th>
<th>BMCSTATS</th>
<th>IBM RUNSTATS</th>
<th>VSAM sampling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of non-clustering indexes</td>
<td>NA</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Longest key</td>
<td>X</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Number of foreign keys</td>
<td>NA</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Number of indexes</td>
<td>X</td>
<td>X</td>
<td>NA</td>
</tr>
<tr>
<td>Longest foreign key</td>
<td>NA</td>
<td>X</td>
<td>NA</td>
</tr>
</tbody>
</table>

**Tip**

To specify the data sizing method, see “Setting the JCL options for static data sets” on page 177.

### Product table maintenance

You are responsible for maintaining the tables that are used by ALTER or CHANGE MANAGER.

In addition to installing the tables, maintenance tasks for the tables include:

- Sizing the tables to accommodate growth requirements
- Purging the tables of old or unneeded data
- Updating the rename table
- Reorganizing the tables to improve performance
- Backing up for recovery

For more information about the administrative tasks associated with managing the application databases, see the IBM documentation.

CHANGE MANAGER uses the tables that are listed in Table 17 on page 153. ALTER uses the all of the CD tables and some of the CM tables. For versions 10.1 and later, these tables store the version and release of the installed product. Earlier versions prefixed the table names with V vr, where the vr reflected the product version and release.

The products use the prefixes listed in the default table name (Table 16 on page 153).
Table 16: Table prefixes

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>BL</td>
<td>Baseline component of CHANGE MANAGER</td>
</tr>
<tr>
<td>CD</td>
<td>Specification and Analysis components of ALTER and CHANGE MANAGER</td>
</tr>
<tr>
<td>CM</td>
<td>Objects for ALTER and CHANGE MANAGER</td>
</tr>
<tr>
<td>CP</td>
<td>CM/PILOT component of CHANGE MANAGER</td>
</tr>
</tbody>
</table>

Table 17: ALTER and CHANGE MANAGER tables

<table>
<thead>
<tr>
<th>Default table name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BL_ALIAS</td>
<td>Alias definitions</td>
</tr>
<tr>
<td>BL_AUXTABLE</td>
<td>Auxiliary object (such as LOB table space, auxiliary table, and auxiliary index) definitions</td>
</tr>
<tr>
<td>BL_CHECK</td>
<td>Check constraint definitions</td>
</tr>
<tr>
<td>BL_CHECK2</td>
<td>Path schema for check constraints</td>
</tr>
<tr>
<td>BL_COLAUTH</td>
<td>Column authorizations</td>
</tr>
<tr>
<td>BL_COLUMN</td>
<td>Table and view column definitions</td>
</tr>
<tr>
<td>BL_DATABASE</td>
<td>Database definitions</td>
</tr>
<tr>
<td>BL_DBAUTH</td>
<td>Security information for databases in a baseline</td>
</tr>
<tr>
<td>BL_INDEX</td>
<td>Index definitions</td>
</tr>
<tr>
<td>BL_KEY</td>
<td>Index key definitions</td>
</tr>
<tr>
<td>BL_KEYTARGETS</td>
<td>Key-targets for extended index definitions</td>
</tr>
<tr>
<td>BL_PACKAGE</td>
<td>Package definitions</td>
</tr>
<tr>
<td>BL_PARMS</td>
<td>Stored procedure parameter definitions</td>
</tr>
<tr>
<td>BL_REL</td>
<td>Foreign key definitions</td>
</tr>
<tr>
<td>BL_RELKEY</td>
<td>Foreign key column definitions</td>
</tr>
<tr>
<td>BL_RESAUTH</td>
<td>Security information for table spaces and storage groups in this baseline</td>
</tr>
<tr>
<td>BL_ROUTINES</td>
<td><em>(ALTER and CHANGE MANAGER version 11.1 and later)</em> Stored procedure definitions</td>
</tr>
<tr>
<td>BL_ROUTINESTEXT</td>
<td>LOB table for baseline stored procedure text</td>
</tr>
<tr>
<td>BL_RTAUTH</td>
<td>Privileges granted on stored procedures in a baseline</td>
</tr>
<tr>
<td>BL_SEQUENCES</td>
<td>Identity column definitions</td>
</tr>
<tr>
<td>BL_SQAUTH</td>
<td>Privileges granted on sequence objects in this baseline</td>
</tr>
<tr>
<td>Default table name</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BL_STOGROUP</td>
<td>Storage group definitions</td>
</tr>
<tr>
<td>BL_SYNONYM</td>
<td>Synonym definitions</td>
</tr>
<tr>
<td>BL_TABAUTH</td>
<td>Security information for tables and views in this baseline</td>
</tr>
<tr>
<td>BL_TABLE</td>
<td>Table definitions</td>
</tr>
<tr>
<td>BL_TABLEPART</td>
<td>Table and index space partition definitions</td>
</tr>
<tr>
<td>BL_TABLESPACE</td>
<td>Table space definitions</td>
</tr>
<tr>
<td>BL_TRIG_COL</td>
<td>Trigger column definitions</td>
</tr>
<tr>
<td>BL_TRIG_TEXT</td>
<td>Trigger text definitions</td>
</tr>
<tr>
<td>BL_TRIG_TEXT_AUX</td>
<td>Trigger text definitions for auxiliary tables</td>
</tr>
<tr>
<td>BL_TRIGGER</td>
<td>Trigger definitions</td>
</tr>
<tr>
<td>BL_UNQCNST</td>
<td>Unique constraint definitions</td>
</tr>
<tr>
<td>BL_UNQCNST_COL</td>
<td>Unique constraint column definitions</td>
</tr>
<tr>
<td>BL_VIEW</td>
<td>View definitions</td>
</tr>
<tr>
<td>BL_VIEW_TEXT</td>
<td>View text definitions</td>
</tr>
<tr>
<td>BL_VIEW_TEXT_AUX</td>
<td>View text definitions for auxiliary tables</td>
</tr>
<tr>
<td>BL_VOLS</td>
<td>Additional volume information for storage groups and table partitions</td>
</tr>
<tr>
<td>CD_ALIAS</td>
<td>Alias definitions</td>
</tr>
<tr>
<td>CD_AUXTABLE</td>
<td>Auxiliary object (such as LOB table space, auxiliary table, and auxiliary index) definitions</td>
</tr>
<tr>
<td>CD_CHECK</td>
<td>Check constraint definitions</td>
</tr>
<tr>
<td>CD_CHECK2</td>
<td>Path schema for check constraints</td>
</tr>
<tr>
<td>CD_COLUMN</td>
<td>Table and view column definitions</td>
</tr>
<tr>
<td>CD_DATABASE</td>
<td>Database definitions</td>
</tr>
<tr>
<td>CD_INDEX</td>
<td>Index definitions</td>
</tr>
<tr>
<td>CD_KEY</td>
<td>Index key definitions</td>
</tr>
<tr>
<td>CD_KEYTARGETS</td>
<td>Key-targets for extended index definitions</td>
</tr>
<tr>
<td>CD_PACKAGE</td>
<td>Package definitions</td>
</tr>
<tr>
<td>CD_PARMS</td>
<td>Stored procedure parameter definitions</td>
</tr>
<tr>
<td>CD_REL</td>
<td>Foreign key definitions</td>
</tr>
<tr>
<td>CD_RELKEY</td>
<td>Foreign key column definitions</td>
</tr>
<tr>
<td>CD_ROUTINES</td>
<td>Stored procedure definitions</td>
</tr>
<tr>
<td>Default table name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>CD_ROUTINES2</td>
<td><em>(ALTER and CHANGE MANAGER version 11.1 and later)</em> Stored procedure definitions</td>
</tr>
<tr>
<td>CD_ROUTINES2NEWTEXT</td>
<td>New text of stored procedure</td>
</tr>
<tr>
<td>CD_ROUTINES2TEXT</td>
<td>Text of stored procedure</td>
</tr>
<tr>
<td>CD_SEQUENCES</td>
<td>Identity column definitions</td>
</tr>
<tr>
<td>CD_STOGROUP</td>
<td>Storage group definitions</td>
</tr>
<tr>
<td>CD_SYNONYM</td>
<td>Synonym definitions</td>
</tr>
<tr>
<td>CD_TABLE</td>
<td>Table definitions</td>
</tr>
<tr>
<td>CD_TABLEPART</td>
<td>Table and index space partition definitions</td>
</tr>
<tr>
<td>CD_TABLESPACE</td>
<td>Table space definitions</td>
</tr>
<tr>
<td>CD_TRIG_COL</td>
<td>Trigger column definitions</td>
</tr>
<tr>
<td>CD_TRIG_TEXT</td>
<td>Trigger text definitions</td>
</tr>
<tr>
<td>CD_TRIG_TEXT_AUX</td>
<td>Trigger text definitions for auxiliary tables</td>
</tr>
<tr>
<td>CD_TRIGGER</td>
<td>Trigger definitions</td>
</tr>
<tr>
<td>CD_UNQCNST</td>
<td>Unique constraint definitions</td>
</tr>
<tr>
<td>CD_UNQCNST_COL</td>
<td>Unique constraint column definitions</td>
</tr>
<tr>
<td>CD_VIEW</td>
<td>View definitions</td>
</tr>
<tr>
<td>CD_VIEW_TEXT</td>
<td>View text definitions</td>
</tr>
<tr>
<td>CD_VIEW_TEXT_AUX</td>
<td>View text definitions for auxiliary tables</td>
</tr>
<tr>
<td>CD_VOLS</td>
<td>Additional volume information for storage groups and table parts</td>
</tr>
<tr>
<td>CM_BASELINE</td>
<td>Baseline header information</td>
</tr>
<tr>
<td>CM_BLPROFILE</td>
<td>Baseline profile information</td>
</tr>
<tr>
<td>CM_HISTORY</td>
<td>Comparison statistical information</td>
</tr>
<tr>
<td>CM_LOCATION</td>
<td>Location information for migrate profiles</td>
</tr>
<tr>
<td>CM_MIGPROFILE</td>
<td>Migrate profile information</td>
</tr>
<tr>
<td>CM_RENAME</td>
<td>Old and new names of renamed objects</td>
</tr>
<tr>
<td>CM_RULES</td>
<td>Rules for migrate profiles</td>
</tr>
<tr>
<td>CM_SCOPE</td>
<td>Scope for migrate and baseline profiles</td>
</tr>
<tr>
<td>CM_SECURITY</td>
<td>Security table</td>
</tr>
<tr>
<td>CM_SYNC</td>
<td>Execution sync table</td>
</tr>
<tr>
<td>CM_UNLOADDSN</td>
<td>Information about unloaded data sets used in a full-recovery baseline</td>
</tr>
</tbody>
</table>
The baseline tables (those named BL_ name) of CATALOG MANAGER require more ongoing maintenance than other tables.

Because the baseline tables hold critical information used for database recovery, information in them should be retained as long as the baselines are needed. If your installation regularly establishes baselines, these tables grow over time. You must ensure that these tables are reorganized and that the size of the tables is expanded on a regular basis. To help control the baseline table, you can delete any unwanted baselines. For more information about deleting baselines, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

The sync tables of ALTER and CATALOG MANAGER (CM_SYNC) and the CM/PILOT component (CP_SYNC) also require more ongoing maintenance. The records in the sync table for a particular work ID or task ID are used as progress indicators during execution of a worklist. These tables grow over time. You will need to ensure that these tables are reorganized and expanded on a regular basis. You will also need to delete any work IDs and task IDs that are no longer needed.

**Tip**
You can also use member ALUDSYNC in the HLQUBMCCNTL data set to maintain the CM_SYNC and CP_SYNC tables.

To help control the size of the sync tables, you can delete any unwanted work IDs or task IDs. When you delete a work ID, all of the change and migration requests for that work ID are deleted from the product's CD tables. If the status of the work ID is Exec Strt (for Execution started) or Exec Comp (for Execution complete), all of the rows that are written to the product's work ID tables (such as the sync table) are also...
Deleting the work ID from the product tables does not affect committed changes that were made to the DB2 catalog during execution of the work ID. (For information about viewing the status of the worklist, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.)

**Note**

Deleting a work ID with Exec Strt status is not recommended because the work ID cannot be restarted for completion.

**WARNING**

Full-recovery baselines are associated with alter-type work IDs. If you delete a work ID that has an associated full-recovery baseline, the work ID for that baseline will become null.

**Tip**

To delete work IDs or task IDs, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

---

**Table security**

Only those users who need to perform diagnostic functions on ALTER or CHANGE MANAGER should be given access to the ALTER or CHANGE MANAGER database objects.

Access to the database through ALTER and CHANGE MANAGER functions is achieved through EXECUTE authority on plans, except on the rename table. The Compare component uses the CHANGE MANAGER rename table (CM_RENAME) when a baseline is used as input to resolve the names of renamed objects. If users change the names of DB2 objects outside of CHANGE MANAGER, they should update the CHANGE MANAGER rename table so that the Compare component can correctly resolve renamed objects. To update the table, the users must submit SQL INSERT statements through a DB2 access tool, and must therefore have authority to update the rename table. Table 18 on page 158 shows the SQL INSERT statements for manually updating the CHANGE MANAGER rename table. Access to the rename table is available by granting DB2 authorization to INSERT, UPDATE, DELETE and SELECT that table. For more information about renamed object resolution, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2.

**WARNING**

Failure to properly update the rename table could result in data loss, because Compare generates DROP and CREATE statements rather than an ALTER statement for renamed objects.
### Table 18: SQL for updating rename table

<table>
<thead>
<tr>
<th>DB2 object</th>
<th>INSERT statement</th>
</tr>
</thead>
</table>
| Storage group       | `INSERT INTO CM_RENAME
  (OBJECT, NAME_PART1, NEWNAME_PART1, TIMESTMP)
  VALUES
  ('SG', 'oldName', 'newName', 'timeStamp')`                                                                                                  |
| Database            | `INSERT INTO CM_RENAME
  (OBJECT, NAME_PART1, NEWNAME_PART1, TIMESTMP)
  VALUES
  ('DB', 'oldName', 'newName', 'timeStamp')`                                                                                               |
| Table space         | `INSERT INTO CM_RENAME
  (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP)
  VALUES
  ('TS', 'oldDatabase', 'oldTableSpace', 'newDatabase', 'newTableSpace', 'timeStamp')`                                                   |
| Table               | `INSERT INTO CM_RENAME
  (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP)
  VALUES
  ('TB', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')`                                                                    |
| Table column        | `INSERT INTO CM_RENAME
  (OBJECT, NAME_PART1, NAME_PART2, CONAME, NEWNAME_PART1, NEWNAME_PART2, NEWCONAME, TIMESTMP)
  VALUES
  ('CO', 'oldCreator', 'oldTable', 'oldColumn', 'newCreator', 'newTable', 'newColumn', 'timeStamp')`                                           |
| Check constraint    | `INSERT INTO CM_RENAME
  (OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTMP)
  VALUES
  ('CK', 'oldCreator', 'oldTable', 'oldName', 'newCreator', 'newTable', 'newName', 'timeStamp')`                                           |
| Index               | `INSERT INTO CM_RENAME
  (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTMP)
  VALUES
  ('IX', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')`                                                                    |
<table>
<thead>
<tr>
<th>DB2 object</th>
<th>INSERT statement</th>
</tr>
</thead>
</table>
| Unique constraint| `INSERT INTO CM_RENAME
(OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTAMP)
VALUES
('UC', 'oldCreator', 'oldTable', 'oldName', 'newCreator', 'newTable', 'newName', 'newTimeStamp')` |
| Trigger          | `INSERT INTO CM_RENAME
(OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTAMP)
VALUES
('TR', 'oldSchema', 'oldName', 'newSchema', 'newName', 'timeStamp')`                      |
| Foreign key      | `INSERT INTO CM_RENAME
(OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTAMP)
VALUES
('FK', 'oldCreator', 'oldTable', 'oldName', 'newCreator', 'newTable', 'newName', 'timeStamp')` |
| View             | `INSERT INTO CM_RENAME
(OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTAMP)
VALUES
('VW', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')`                     |
| View column      | `INSERT INTO CM_RENAME
(OBJECT, NAME_PART1, NAME_PART2, CONAME, NEWNAME_PART1, NEWNAME_PART2, NEWCONAME, TIMESTAMP)
VALUES
('VC', 'oldCreator', 'oldView', 'oldColumn', 'newCreator', 'newView', 'newColumn', 'timeStamp')` |
| Stored procedure  | `INSERT INTO CM_RENAME
(OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTAMP)
VALUES
('SP', 'oldSchema', 'oldName', 'newSchema', 'newName', 'newTimeStamp')`                     |
| Stored procedure  | `INSERT INTO CM_RENAME
(OBJECT, NAME_PART1, NAME_PART2, NAME_PART3, NEWNAME_PART1, NEWNAME_PART2, NEWNAME_PART3, TIMESTAMP)
VALUES
('SP', 'oldSchema', 'oldName', 'newSchema', 'newName', 'newTimeStamp')`                     |
| (external)       |                                                                                                                                                 |
| (native SQL)     |                                                                                                                                                 |
### Setting options

You can set user and product options in ALTER and CHANGE MANAGER, CM/PILOT, and JCL Generation.

### Setting the attributes for the panel display

Use the Panel Attribute Options panel to specify your preferences for long names, entry field and error delimiters, and screen colors.

#### To set the attributes for the panel display

1. From the ALTER or CHANGE MANAGER Main Menu, select **Options** and press **Enter**.

2. From the ALTER or CHANGE MANAGER Main Options Menu panel, select **Panel Attribute Options** and press **Enter**.

<table>
<thead>
<tr>
<th>DB2 object</th>
<th>INSERT statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Synonym</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTAMP) VALUES ('SY', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>Alias</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTAMP) VALUES ('AL', 'oldCreator', 'oldName', 'newCreator', 'newName', 'timeStamp')</td>
</tr>
<tr>
<td>Sequence</td>
<td>INSERT INTO CM_RENAME (OBJECT, NAME_PART1, NAME_PART2, NEWNAME_PART1, NEWNAME_PART2, TIMESTAMP) VALUES ('SQ', 'oldSchema', 'oldName', 'newSchema', 'newName', 'timeStamp')</td>
</tr>
</tbody>
</table>
The ALTER or CHANGE MANAGER Panel Attribute Options panel is displayed (Figure 54 on page 161).

**Figure 54: Panel Attribute Options panel**

```
ALUFOPT1 ---------- CHANGE MANAGER Panel Attribute Options ----------
Command =>
Type information. Then press Enter to continue or PF12 for previous panel.
Zoom/Truncation Attributes.
  Allow popup displays . . . Y  (Y/N)
  Omission Characters . . . <>
  Autotab Character . . . . +
  Location . . . . . . . . . M  (Beginning, Middle, End)

More: +
Select your preference for an entry field delimiter.
  3 1. Underscore(USCORE)  2. Reverse video  3. None
Specify color/delimiter to be used when an error is detected.
  Color of field . . . . . . .
  Delimiter . . . . . . . . .
  2 1. Underscore  2. Reverse video  3. None
ISPF Color Attributes.
  Type any color preference for the color attribute you wish to substitute
  ISPF Color Attributes
  Color Attribute You Wish to Substitute
  Blue . . . . . . . . .
  Green . . . . . . . . .

Commands:  RESET HELP END
```

3 Specify the values that should be used when displaying long names:

**Note**
The product stores the values for each of the zoom and truncation attributes in your ISPF profile. However, you can also write the values to a user POF.

a In the **Allow popup displays** field, specify whether to display a dialog or a panel when the **ZOOM (F4)** key is pressed on an object name:

- If you specify **Y**, the product displays the object name in a dialog. If the name is too long to be displayed in a dialog, the product displays the name in a panel.
- If you specify **N**, the product displays the object name in a panel.

b In the **Omission Characters** field, type the characters that replace the beginning and end of a truncated string in an object name that is too long to be displayed.

c In the **Autotab Character** field, specify whether to display an autotab character in front of an object name that is too long to be displayed:

- If you specify **N**, an autotab character is not displayed and you cannot tab to the object name. To zoom on the object name, position the cursor on the object name and press the **ZOOM (F4)** key.
If you specify any character other than N, the character is displayed and you can tab to the object name. To zoom on the object name, press the ZOOM (PF4) key.

Tip

The autotab character indicates which fields must be zoomed to view the entire object name.

d In the Location field, type B, M, or E to specify the location of characters to be omitted in object names that are too long to be displayed.

<table>
<thead>
<tr>
<th>To replace characters</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>At the left end (beginning) of the name</td>
<td>B</td>
</tr>
<tr>
<td>In the middle of the name</td>
<td>M</td>
</tr>
<tr>
<td>At the right end (end) of the name</td>
<td>E</td>
</tr>
</tbody>
</table>

4 Type a value for your entry field delimiter.

5 Type new values for your error message color and delimiter.

6 Type new values for your color preferences.

7 Press END to save your changes and to return to the Main Menu.

Setting the options for data set allocation

Use the Dataset Allocation Options and the Image Copy and Unload Options panels to specify the parameters for allocating your data sets. The worklist and CDL data sets are the only data sets that ALTER and CHANGE MANAGER allocate. All other data sets, such as JCL, Compare, Baseline, Report, and DDL, must be preallocated.

To set the options for data set allocation

1 From the ALTER or CHANGE MANAGER Main Menu, select Options and press Enter.

The ALTER or CHANGE MANAGER Main Options Menu panel is displayed.

2 Select Dataset Allocation Options and press Enter.
The ALTER or CHANGE MANAGER Dataset Allocation Options panel is displayed, as shown in Figure 55 on page 163.

**Figure 55: Dataset Allocation Options panel**

```
ALUFOPT2  -------  CHANGE MANAGER Dataset Allocation Options  -----------
Command ===>
```

Type information. Then press Enter to continue or PF12 for previous panel.

```
DB2 Subsystem ID . .  DEGA
```

Specify Dataset Allocation Parameters.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Unitname</th>
<th>Volume</th>
<th>Priqty</th>
<th>Secqty</th>
<th>Alloc Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Worklist/BL Report</td>
<td>SYSDA</td>
<td>15</td>
<td>5</td>
<td></td>
<td>Tracks</td>
</tr>
<tr>
<td>CDL</td>
<td>SYSDA</td>
<td>15</td>
<td>5</td>
<td></td>
<td>Tracks</td>
</tr>
</tbody>
</table>

Commands:  HELP END

3 Specify the DB2 subsystem ID (SSID).

4 Specify the data set allocation parameters for a worklist and a baseline report by typing new values for the **Unitname**, **Volume**, **Priqty**, and **Secqty** fields.

**Note**
The product stores the values for the data set allocation parameters in your ISPF profile. However, you can also write the values to a user POF.

5 *(CHANGE MANAGER only)* Specify the data set allocation parameters for the CDL data set by typing new values for the **Unitname**, **Volume**, **Priqty**, and **Secqty** fields.

**Note**
The product stores the values for the data set allocation parameters in your ISPF profile. However, you can also write the values to a user POF.

6 Press END to save your changes and to return to the ALTER or CHANGE MANAGER Main Menu.

**Setting the processing options for Analysis**

Use the Analysis Options panel to specify the data set names, processing options, and job parameters for the Analysis component.

**To set the processing options for Analysis**

1 From the ALTER or CHANGE MANAGER Main Menu, select **Options** and press **Enter**.
2 On the ALTER or CHANGE MANAGER Main Options Menu panel, select Analysis Options and press Enter.

The ALTER or CHANGE MANAGER Analysis Options panel is displayed, as shown in Figure 56 on page 164.

**Figure 56: Analysis Options panel**

```
ALUFOPT3 --------------- CHANGE MANAGER Analysis Options ----------------------
Command ===> 

Type information. Then press Enter to continue or PF12 for previous panel.
Specify ANALYSIS dataset names.
  JCL ....... 'RDACRJ.V11.ANLYJCL(&WORKID8)'
  Worklist . . . 'RDACRJ.V11.WLBASE01(&WORKID8)'
  Diagnostics .. SYSOUT

Specify default ANALYSIS processing options. Run Type . . 2 1. Foreground
  Override Options
  $ Build JCL/Parms
  $ Edit JCL/Parms
  _ Submit/Run
  _ Save Last Used Options

Specify default ANALYSIS job parameters.
  _ Image copy before DROPs or REORGs $ Include AMS commands

Global AUTHID ..
Commands: HELP END
```

3 Specify the Analysis data set names for the JCL, worklist, and diagnostics.

**Note**

The data sets for the JCL and worklist can be either sequential files or PDS members. The data set for the diagnostics must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

The values for these options are stored in your ISPF profile. However, the values can also be written to a user POF.

4 Specify the default processing options for Analysis by typing S to select an option.

5 Select whether to run Analysis in foreground or in batch.

6 Specify the default job parameters for Analysis.

**Note**

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

7 *(optional)* Type a name for a default global AUTHID. For more information about global authorization IDs, see the ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2
8 Press END to save your changes and to return to the ALTER or CHANGE MANAGER Main Menu.

Setting the processing options for Import

To define or modify the values in your ISPF profile, you can use the Options panels of ALTER or CHANGE MANAGER.

Use the Import Options panel to specify the data set names and the processing options for the Import component of ALTER or CHANGE MANAGER.

To set the processing options for Import

1 From the ALTER or CHANGE MANAGER Main Menu, select Options and press Enter.

2 On the ALTER or CHANGE MANAGER Main Options Menu panel, select Import Options and press Enter.

The ALTER or CHANGE MANAGER Import Options panel is displayed, as shown in Figure 57 on page 165.

Figure 57: Import Options panel

<table>
<thead>
<tr>
<th>Command</th>
<th>Type information. Then press Enter to continue or PF12 for previous panel.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Specify IMPORT dataset names.</td>
</tr>
<tr>
<td>JCL</td>
<td>‘RDACRJ.V11.IMPJCL(&amp;WORKID8)’</td>
</tr>
<tr>
<td></td>
<td>Diagnostics . . SYSOUT</td>
</tr>
<tr>
<td></td>
<td>Select default IMPORT processing options.</td>
</tr>
<tr>
<td>Run Type</td>
<td>2 1. Foreground</td>
</tr>
<tr>
<td></td>
<td>2. Batch</td>
</tr>
<tr>
<td></td>
<td>Edit CDL</td>
</tr>
<tr>
<td></td>
<td>$ Build JCL/Parms</td>
</tr>
<tr>
<td></td>
<td>$ Edit JCL/Parms</td>
</tr>
<tr>
<td></td>
<td>Submit/Run</td>
</tr>
<tr>
<td></td>
<td>Save Last Used Options</td>
</tr>
<tr>
<td></td>
<td>Commands: HELP END</td>
</tr>
</tbody>
</table>

3 Specify the Import data set names for the JCL and diagnostics.
The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4 Specify the default processing options for Import by typing S to select an option.

Note

In ALTER, you can edit the DDL. In CHANGE MANAGER, you can edit the CDL or DDL.

5 Specify whether to run Import in foreground or in batch.

6 Press END to save your changes and to return to the ALTER or CHANGE MANAGER Main Menu.

Setting the processing options for Execution

Use the Execution Options panel to specify the data set names and the method of generating JCL for the Execution component.

To set the processing options for Execution

1 From the ALTER or CHANGE MANAGER Main Menu, select Options and press Enter.

2 On the ALTER or CHANGE MANAGER Main Options Menu panel, select Execution Options and press Enter.

The Execution Options panel is displayed, as shown in Figure 58 on page 166.
3 Specify the Execution default data set names for the JCL and diagnostics.

The Execution JCL contains the JCL to run Execution. The batch JCL job contains the JCL to run Batch Execution JCL Generation, which builds the JCL to run Execution.

---

**Note**

The data set for JCL can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file, unless parallelism is enabled for the Database Administration or BMC Object Administration for DB2 solution. If you are running a worklist in parallel, the Diagnostics file must be allocated to SYSOUT. AEXPRINT is the ddname of the diagnostic output. The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

---

4 *(optional)* Specify up to three Database Request Module (DBRM) libraries that can be concatenated in the Execution JCL for binds in a worklist.

---

**Note**

The product stores the values for these options in your ISPF profile. However, you can write the values to a user POF.

---

5 Specify the default processing options for Execution by typing S to select an option.

6 Select whether to generate JCL in foreground or in batch.

7 Press END to save your changes and to return to the ALTER or CHANGE MANAGER Main Menu.

---

**Setting the processing options for Compare**

Use the Compare Options panel to specify the data set names and the processing options for the Compare component of CHANGE MANAGER.

**To set the processing options for Compare**

1 From the CHANGE MANAGER Main Menu, select **Options** and press **Enter**.
2 On the CHANGE MANAGER Main Options Menu panel, select **Compare Options** and press **Enter**.

The CHANGE MANAGER Compare Options panel is displayed, as shown in Figure 59 on page 168.

**Figure 59: Compare Options panel**

```
ALUFOPT5 --------------- CHANGE MANAGER Compare Options -----------------------
Command ===> Type information. Then press Enter to continue or PF12 for previous panel.
Specify COMPARE dataset names.
  CDL . . . . . . 'ACM.DB2V10.V11.CDLBAS01(&WORKID8)'
  JCL . . . . . . 'ACM.DB2V10.V11.CMPJCL(&WORKID8)'
  Diagnostics . . SYSOUT
  _ Select to generate Compare report information as comments in CDL
Select default COMPARE processing options.     Run Type . . 2 1. Foreground
  S Override Compare defaults 2. Batch
  _ Build JCL/Parms
  _ Edit JCL/Parms
  _ Submit/Run
  _ Save Last Used Options
Commands:  HELP END
```

3 Specify the Compare data set names for the CDL, JCL, and diagnostics.

**Note**

The data sets for JCL and the CDL file can be either sequential files or PDS members. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

The product stores the values for the CDL, JCL, and diagnostics data sets in your ISPF profile. However, you can also write the values to a user POF.

4 Specify whether to generate comparison report information as comments in the CDL file.

5 Specify the default processing options for Compare by typing **S** to select an option.

6 Specify whether to run Compare in foreground or in batch.

7 Press **END** to save your changes and to return to the CHANGE MANAGER Main Menu.

**Setting the processing options for Baseline**

Use the Baseline Options panel to specify the data set names and the processing options for the Baseline component of CHANGE MANAGER.
To set the processing options for Baseline

1. From the CHANGE MANAGER Main Menu, select Options and press Enter.

2. On the CHANGE MANAGER Main Options Menu panel, select Baseline Options and press Enter.

   The CHANGE MANAGER Baseline Options panel is displayed, as shown in Figure 60 on page 169.

**Figure 60: Baseline Options panel**

```
ALUFOPT6 --------------- CHANGE MANAGER Baseline Options ----------------------
Command ===> 

Type information. Then press Enter to continue or PF12 for previous panel.

Specify BASELINE dataset names.
  JCL . . . . . . 'RDACRJ.V11.BLJCL(&WORKID8)'
  Diagnostics . . SYSOUT

Select default BASELINE processing options.           Run Type . . 2 1. Foreground
S Build JCL/Parms
S Edit JCL/Parms
  Submit/Run
  Save Last Used Options

Commands: HELP END
```

3. Specify the Baseline data set names for the JCL and diagnostics.

   **Note**

   The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground). The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4. Specify the default processing options for Baseline by typing S to select an option.

5. Specify whether to run Baseline in foreground or in batch.

6. Press END to save your changes and to return to the CHANGE MANAGER Main Menu.

---

**Setting the options for the baseline report**

Use the Baseline Report Options panel to specify the data set names and the processing options for the baseline report for CHANGE MANAGER.
To set the options for the baseline report

1. From the CHANGE MANAGER Main Menu, select Options and press Enter.

2. On the CHANGE MANAGER Main Options Menu panel, select Baseline Report Options and press Enter.

The CHANGE MANAGER Baseline Report Options panel is displayed, as shown in Figure 61 on page 170.

Figure 61: Baseline Report Options panel

<table>
<thead>
<tr>
<th>ALUFOPT7</th>
<th>CHANGE MANAGER Baseline Report Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command ==&gt;</td>
<td>Type information. Then press Enter to continue or PF12 for previous panel.</td>
</tr>
<tr>
<td>Specify BASELINE REPORT dataset names.</td>
<td></td>
</tr>
<tr>
<td>JCL . . . . . 'RDACRJ.V11.BLRJCL(&amp;WORKID8)'</td>
<td></td>
</tr>
<tr>
<td>Report . . . . 'RDACRJ.V11.BLREPS(&amp;WORKID8)'</td>
<td></td>
</tr>
<tr>
<td>Diagnostics . . SYSOUT</td>
<td></td>
</tr>
<tr>
<td>Select default REPORT processing options.</td>
<td></td>
</tr>
<tr>
<td>Run Type . . 1 1. Foreground</td>
<td></td>
</tr>
<tr>
<td>2. Batch</td>
<td></td>
</tr>
<tr>
<td>S Build JCL/Parms</td>
<td></td>
</tr>
<tr>
<td>S Edit JCL/Parms</td>
<td></td>
</tr>
<tr>
<td>Submit/Run</td>
<td></td>
</tr>
<tr>
<td>S Edit Report</td>
<td></td>
</tr>
<tr>
<td>_ Save Last Used Options</td>
<td></td>
</tr>
<tr>
<td>Commands: HELP END</td>
<td></td>
</tr>
</tbody>
</table>

3. Specify the baseline report data set names for the JCL, report, and diagnostics.

   **Note**
   The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).
   The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4. Specify the default processing options for the baseline report by typing S to select an option.

5. Specify whether to run the baseline report in foreground or in batch.

6. Press END to save your changes and to return to the CHANGE MANAGER Main Menu.

Setting the user options for CM/PILOT

The first time that you run CHANGE MANAGER, the Front End copies the values from the installation options module into your ISPF profile.
To define or modify the values in your ISPF profile, you can use the Options panels of the CM/PILOT component of CHANGE MANAGER. Use the panels to specify the task ID data set names, worklist processing options, application profile names, and CHANGE MANAGER data set names.

**To set the user options for CM/PILOT**

1. From the CHANGE MANAGER Main Menu, select CM/PILOT and press Enter.

2. From the CM/PILOT Main Menu, select Options and press Enter. The CM/PILOT TASKID Options panel is displayed, as shown in Figure 62 on page 171.

   **Figure 62: CM/PILOT TASKID Options panel**

   AUCF0PT1  --------------------- CM/PILOT TASKID Options  ------------------------
   Command ===>

   Type information. Then press Enter to continue.

   Specify TASKID options.

   Specify TASKID dataset names.
   - Worklist . . . . 'RDACRJ.V11.TASKWL(&TASKID)'
   - JCL . . . . . . . 'RDACRJ.V11.TASKJCL(&TASKID)'
   - Diagnostics . . . SYSOUT

   Select default Worklist processing options.
   - Override run options
   - Create Worklist
   - Edit Worklist
   - Create JCL
   - Edit JCL
   - Submit JCL
   - Save Last Used Options

   Commands: HELP END CANCEL

3. Specify the task ID data set names for the worklist, JCL, and diagnostics.

   These data set names appear as defaults on the TASKID Interface panel.

   **Note**

   The JCL data set can be either a sequential file or a PDS member. The data set for the Diagnostics file must be a sequential file. Diagnostics might also be sent to SYSOUT or to TERM (in foreground).

   The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

4. Specify the default processing options for the CM/PILOT worklist by typing S to select an option.

5. Press Enter to display the CM/PILOT Application Options panel, which is shown in Figure 63 on page 171.

   **Figure 63: CM/PILOT Application Options panel**

   AUCF0PT2  --------------------- CM/PILOT Application Options  ------------------------
   Command ===>

   Chapter 3 Setting up ALTER and CHANGE MANAGER  171
6 Specify the profile names and a name template for the work ID. These names appear as defaults on the Create Application panel.
   a Type the name of the **Inbound Migrate Profile** that is used by the Import component of CHANGE MANAGER for applying change rules to attributes of data structures that are imported to the local DB2 subsystem.
   b Type the name of the **Outbound Migrate Profile** that is used by the Analysis and Compare components of CHANGE MANAGER for generating worklists and CDL files. Outbound migrate profiles can contain change rules, scope rules, and locations.
   c Type the name of the **DDL Baseline Profile** that is used by the Baseline component of CHANGE MANAGER for creating baselines of DDL files and worklists.
   d Type the name of the **Catalog Baseline Profile** that is used by the Analysis, Baseline, and Compare components of CHANGE MANAGER for specifying the data structures from the DB2 catalog to include in a baseline.

   **Note**
   You can specify wildcard patterns for the profiles on the CM/PILOT Application Options panel. Then, when you create an application, the wildcard patterns appear on the Create Application panel. When you press **Enter** on the Create Application panel, a list of profiles is displayed.

   e Type the name of the **WORKID Name Template** that enables you to standardize naming conventions for new work IDs that CM/PILOT creates.

7 Press **Enter** to display the CM/PILOT Output Dataset Options panel, which is shown in Figure 64 on page 172.

**Figure 64: CM/PILOT Output Dataset Options panel**

Type information. Then press Enter to continue or PF12 for previous panel.
Specify CHANGE MANAGER dataset names.

JCL . . . . . . 'RDACRJ.V11.EXECJCL(&WORKID8)'
Worklist . . . 'RDACRJ.V11.WLBASE01(&WORKID8)'
CDL . . . . . . 'RDACRJ.V11.CDLBAS01(&WORKID8)'

Commands: HELP END CANCEL

8 Specify the CHANGE MANAGER data set names. These data set names appear as CHANGE MANAGER defaults on the CHANGE MANAGER Datasets panel.

**Note**
You cannot modify the CM/PILOT user options for the CHANGE MANAGER data set names by changing the user options in CHANGE MANAGER. You can modify these user options only from the CM/PILOT Output Dataset Options panel or by refreshing the user options.

The product stores the values for these options in your ISPF profile. However, you can also write the values to a user POF.

a Type the name of the JCL data set that contains the JCL to process the Analysis worklist.

b Type the name of the worklist data set that contains the Analysis worklist.

c Type the name of the CDL data set that contains the CDL which results from a comparison.

9 Press END to save your changes and to return to the CM/PILOT Main Menu.

10 Press END to return to the CHANGE MANAGER Main Menu.

---

**Setting the JCL options for job cards**

Use the Options panels of the JCL Generation component to define or modify the values in your ISPF profile and a user POF. Use the Jobcard Options panel to specify information about the job cards that the JCL uses.

**To set the JCL options for job cards**

1 Use the following menu selections to display the JCL Generation Jobcard Options Update panel (Figure 65 on page 174):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN Options</td>
</tr>
<tr>
<td>From this menu</td>
<td>Select this item and press Enter</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Jobcard Options</td>
</tr>
</tbody>
</table>

### Figure 65: JCL Generation Jobcard Options Update panel

![JCL Generation Jobcard Options Update panel](image)

**COMMAND ===>**

Type data and press Enter.

Is a TSO submit exit used to generate jobcards? . N (Y/N)

Enter jobcards below:

```plaintext
//MVSCAL2I JOB ('&ZACCTNUM,'&PGMR',
// CLASS=A,MSGLEVEL=(1,1),NOTIFY=&SYSUID
/***

Jcllib . . . . .
Sysexec. . . . . ADM.INST1110.XXXREXX
Region size . . OM (See JCL Reference for valid options)
Memlimit... . . NOLIMIT (See JCL Reference for valid options)
Time parameter . . (See JCL Reference for valid options)
System MLIB... . . SYS1.0000.ISPMENU
Runtime HLQ... . . ADM.INSTxxxx
User HLQ . . . .
LLQ. . . . . . .
ULLQ. . . . . . . (Leave blank if using runtime enablement)
```

2. In the **Is a TSO submit exit used to generate jobcards?** field, type **Y** or **N**.

3. In the **Enter Jobcards below:** field, type the job statement information that you want to add to the JCL.

4. In the **Sysexec** field, type the name of the PDS in which a REXX EXEC is a member.

**Tip**

To indicate the data set name for a different SSID, append the &SSID or &MSSID symbolic variable to the name.

5. In the **Region size** field, type the amount of memory to allocate for each step so that your job can run.

6. In the **Memlimit** field, type the limit on the above-the-bar memory for an address space.

7. In the **Time parameter** field, type the value for the TIME limit for each step in a batch job stream.

8. In the **System MLIB** field, type the name of the system ISPF message library.

9. In the **Runtime HLQ** field, type the high-level qualifier (HLQ) for ISPF data sets for the installation environment.

During installation, if you chose to use the runtime enablement (RTE) feature, the Installation System set this value to an HLQ for user runtime libraries. If you
chose not to use RTE, the Installation System set the value to an HLQ for Execution.

This value supports the following symbolic variables:

- \&DB2V2 and \&DB2V3, which resolve to the version of DB2
- \&SSID, which resolves to the DB2 subsystem ID

When you include the \&SSID symbolic variable, the product can use a single POF with multiple subsystems.

10 In the **User HLQ** field, type the high-level qualifier used for the user-defined data sets for the installation environment.

---

**Note**
The **User HLQ** field is used only if the **LLQ** field is blank.

11 In the **LLQ** field, type the low-level qualifier for ISPF data sets for the installation environment.

During installation, if you chose to use the runtime enablement feature, the Installation System set this value to BMC. If you chose not to use the feature, the Installation System set the value to DB.

12 In the **ULLQ** field, type the low-level qualifier for user-defined data sets for the installation environment.

---

**Note**
If the **User HLQ** field contains a value, the **ULLQ** value is ignored.

13 Press **END** to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

### Setting the JCL options for STEPLIBs

Use the Options panels of the JCL Generation component to define or modify the values in your ISPF profile and a user POF.

**To set the JCL options**

1 Use the following menu selections to display the JCL Generation STEPLIB Options Update panel (*Figure 66 on page 176*):
From this menu | Select this item and press Enter
---|---
ALTER or CHANGE MANAGER Main Menu | Options
ALTER or CHANGE MANAGER Main Options Menu | JCLGEN options
JCL Generation Update Main Menu | Steplib Options

**Figure 66: JCL Generation STEPLIB Options Update panel**

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>----------------- JCL GENERATION STEPLIB OPTIONS UPDATE -----------------</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Type data and press Enter.</td>
</tr>
<tr>
<td>DSNEXIT</td>
<td>SYS3.&amp;SSID..DSNEXIT</td>
</tr>
<tr>
<td>DB2 DSNLOAD</td>
<td>CSG1.DB2V&amp;DB2V2.M.DSNLOAD</td>
</tr>
<tr>
<td>Override lib</td>
<td>ADM.INST1120.UDBLINK</td>
</tr>
<tr>
<td>CATALOG MANAGER</td>
<td>ADM.INST1120.BMCLINK</td>
</tr>
<tr>
<td>ALTER/CHANGE MANAGER</td>
<td>ADM.INST1120.BMCLINK</td>
</tr>
<tr>
<td>DASD MANAGER PLUS</td>
<td>ADM.INST1120.BMCLINK</td>
</tr>
<tr>
<td>EXECUTION</td>
<td>ADM.INST1120.BMCLINK</td>
</tr>
<tr>
<td>REORG PLUS</td>
<td>D2U.INST1120.BMCLINK</td>
</tr>
<tr>
<td>LOADPLUS</td>
<td>D2U.INST1120.BMCLINK</td>
</tr>
<tr>
<td>UNLOAD PLUS</td>
<td>D2U.INST1120.BMCLINK</td>
</tr>
<tr>
<td>NQT RECOVER</td>
<td>RMD.INST1120.BMCLINK</td>
</tr>
<tr>
<td>CHECK PLUS</td>
<td>D2U.INST1120.BMCLINK</td>
</tr>
<tr>
<td>SQL EXPLORER</td>
<td>PRF.INST1120.BMCLINK</td>
</tr>
<tr>
<td>Additional lib</td>
<td>SCC.INST1120.BMCLINK</td>
</tr>
<tr>
<td>IOA LOAD 1</td>
<td></td>
</tr>
<tr>
<td>IOA LOAD 2</td>
<td></td>
</tr>
</tbody>
</table>

2 Specify the data set names for the libraries:

**Tip**
To indicate the data set name for a different SSID, append the &SSID or &MSSID symbolic variable to the name.

a In the DSNEXIT field, type the data set name of the DB2 EXIT (DSNEXIT) library.

**Tip**
To indicate the version of DB2, append the &DB2V2 or &DB2V3 symbolic variable to the name.

b In the DB2 DSNLOAD field, type the data set name of the library in which the DB2 DSN Command Processor load modules are stored.

c In the Override lib field, type the data set name for the override LINK library that should appear first in the STEPLIB statement.

d In the field for the relevant product, type the data set names of the LINK libraries for the BMC products.

e In the Additional lib field, type the data set name for the additional LINK library that should appear last in the STEPLIB statement.
3 Press **END** to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

### Setting the JCL options for static data sets

To define or modify the values in your ISPF profile and a user POF, use the Options panels of the JCL Generation component.

Use the Static Data Set Options panel to specify the options for sizing and cleaning up your data sets.

**To set the JCL options for static data sets**

1. Use the following menu selections to display the JCL Generation Static Data Set Options Update panel:

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Static Data Set Options</td>
</tr>
</tbody>
</table>

   **Figure 67: JCL Generation Static Data Set Options Update panel**

   ![JCL Generation Static Data Set Options Update panel](image)

   **COMMAND ===>
   Type data and press Enter.

   - **Data set sizing option**...
     
     **N** (No Sizing), **B** (Bmcstats), **C** (DB2 Catalog), **O** (Object Sampling)
   - **Data set sizing device**...
     
     **3390** (3380/3390)
   - **Max cylinders**...
     
     **99999** (Do not exceed this primary value in JCL.)
   - **If max cylinders are exceeded, use the following for DASD data sets**
     
     - **Max primary quantity**...
       
       **200** (Cylinders, 1 - 9999)
     - **Max secondary quantity**...
       
       **20** (Cylinders, 1 - 9999)
     - **Max unit count**...
       
       **(Blank or 1 - 59 volumes)**
   - **Include data set cleanup step**...
     
     **Y** (Y/N)
   - **Return code for cleanup step**...
     
     **0** (04)
   - **Temporary unit**...
     
     **SYSDA** (SYSDA, SYSALLDA, etc.)
   - **Include SYSPRINT DD**...
     
     **N** (Y/N)

   Note: See Debugging, Display and Execution Options to display sizing options in the JCL.

2. In the **Data set sizing option** field, type **N**, **B**, **C**, or **O** to specify the sizing method, as shown in **Table 19 on page 178**.
Whether or not data set sizing is performed, DB2 catalog access is required to resolve any symbolic variables. For more information about data set sizing, see “JCL Generation data sets sizing function” on page 151.

### Table 19: Data set sizing options

<table>
<thead>
<tr>
<th>For this method</th>
<th>Type</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>No data set sizing</td>
<td>N</td>
<td>The product uses the default primary and secondary quantities for the permanent data sets. To modify the quantities for the primary and secondary space for a permanent data set, see Step 4 on page 179.</td>
</tr>
</tbody>
</table>
| Sizing by using statistics from the DASD MANAGER PLUS tables | B    | Use BMCSTATS.  
  For ALTER and CHANGE MANAGER:  
  — You must have installed the DASD MANAGER PLUS product.  
  — Both ALTER and CHANGE MANAGER must be interacting with DASD MANAGER PLUS.  
  ■ If a column is defined as a LONG VARCHAR, this option averages the row size.  
  ■ If you select this option for Batch Execution JCL Generation, the DATASETSIZING B keyword is inserted into the AJXIN input stream.  
  ■ Current BMCSTATS statistics should be available for objects in the worklist before you select this option. |
| Sizing by using statistics from the DB2 catalog | C    | ■ If a column is defined as a LONG VARCHAR, this option does not average the row size.  
  ■ If you select this option for Batch Execution JCL Generation, the DATASETSIZING C keyword is inserted into the AJXIN input stream.  
  ■ Current IBM RUNSTATS catalog statistics should be available for objects in the worklist before you select this option. |
<table>
<thead>
<tr>
<th>For this method</th>
<th>Type</th>
<th>Considerations</th>
</tr>
</thead>
</table>
| Estimating sizes based on physical, random sampling of VSAM data sets | O | ▪ If a column is defined as a LONG VARCHAR, this option averages the row size.  
▪ If you select this option for Batch Execution JCL Generation, the DATASETSIZING O keyword is inserted into the AJXIN input stream.  
▪ You can use this option if the statistical information in the DB2 catalog or in the DASD MANAGER PLUS tables is not current. However, JCL generation might take additional time to complete. |

You can specify whether to include comments in the Execution JCL that show statistics for determining data set sizes. For more information, see “Setting the JCL debugging, display, and Execution options” on page 198.

3 In the **Data set sizing device** field, specify the type of DASD to use in calculating the sizes of data sets.

4 In the **Max cylinders** field, specify the maximum number of cylinders for a data set.

5 Specify the values that should be used for the following options when the maximum number of cylinders are exceeded:

   a In the **Max primary quantity** field, type the number of cylinders for the maximum primary quantity.

   b In the **Max secondary quantity** field, type the number of cylinders for the maximum secondary quantity.

   c In the **Max unit count** field, type the maximum number of volumes.

      If you want to use multiple data sets on DASD, specify a value greater than 1 for **Max unit count**. On the JCL Generation Individual Data Set Options Update panels (see “Setting the JCL options for permanent data sets” on page 186), specify the name of a DASD unit.

6 In the **Include data set cleanup step** field, type Y or N to specify whether to generate a step in the JCL to delete the permanent work data sets. The product deletes the data sets at the end of Execution. For Batch Execution JCL Generation, selecting this option inserts the JCLCLEANUP YES keyword into the AJXIN input stream.

You can generate the JCL for a job step that automatically deletes many of the permanent work data sets that Execution creates. Execution creates these data sets...
during worklist processing and sets the disposition of the data sets to new, catalog, catalog (DISP=(NEW,CATLG,CATLG)).

The cleanup job step is performed only if the condition code returned from any previous job step is less than or equal to the number that is specified in the Return code for cleanup step field. Table 20 on page 180 lists the types of work data sets that are included in the cleanup job step. These data sets are automatically deleted unless otherwise noted.

Table 20: Work data sets in the JCL cleanup job step

<table>
<thead>
<tr>
<th>Work data set</th>
<th>ddname</th>
<th>Used in JCL cleanup by</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discard (SYSDISC) a</td>
<td>SYSDnnnn</td>
<td>LOADPLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBM LOAD</td>
</tr>
<tr>
<td>Error</td>
<td>SYSERnnn</td>
<td>CHECK PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LOADPLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBM CHECK DATA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBM LOAD</td>
</tr>
<tr>
<td>Map</td>
<td>SYSMAP</td>
<td>IBM LOAD</td>
</tr>
<tr>
<td>Punch a</td>
<td>SYSPUNCH</td>
<td>REORG PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBM REORG</td>
</tr>
<tr>
<td>Unload (SYSREC) b</td>
<td>SYSRnnnn or Rnnnnyyyy</td>
<td>REORG PLUS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBM REORG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The unload data sets that are used by REORG PLUS and IBM REORG are deleted automatically.</td>
</tr>
<tr>
<td>Work</td>
<td>Not applicable</td>
<td>Utilities that are listed in Table 21 on page 181.</td>
</tr>
</tbody>
</table>

a These data sets are specified in the cleanup job step but are commented out. You must edit the Execution JCL and remove the comment delimiters to delete these data sets automatically.

b Other unload data sets that are used by the LOADPLUS and UNLOAD PLUS utilities and IBM LOAD utility are specified in the cleanup job step but are commented out. You must edit the Execution JCL and remove the comment delimiters to delete the other unload data sets automatically.

Table 21 on page 181 lists the work data sets that are used by the corresponding utilities.
Table 21: Work data sets used by utilities

<table>
<thead>
<tr>
<th>Work data set</th>
<th>Utility</th>
</tr>
</thead>
<tbody>
<tr>
<td>SORTOUT</td>
<td>CHECK PLUS</td>
</tr>
<tr>
<td>SORTPnnn</td>
<td>LOADPLUS</td>
</tr>
<tr>
<td>SORTOnnn</td>
<td>REORG PLUS</td>
</tr>
<tr>
<td>IBM CHECK DATA</td>
<td></td>
</tr>
<tr>
<td>IBM LOAD</td>
<td></td>
</tr>
<tr>
<td>IBM REORG</td>
<td></td>
</tr>
<tr>
<td>SYSUTnnn</td>
<td>CHECK PLUS</td>
</tr>
<tr>
<td>SUTnnn</td>
<td>LOADPLUS</td>
</tr>
<tr>
<td>WRKnnn</td>
<td>REORG PLUS</td>
</tr>
<tr>
<td>NGT Recover</td>
<td></td>
</tr>
<tr>
<td>IBM CHECK DATA</td>
<td></td>
</tr>
<tr>
<td>IBM LOAD</td>
<td></td>
</tr>
<tr>
<td>IBM REORG</td>
<td>RECOVER INDEX</td>
</tr>
<tr>
<td>IBM REBUILD INDEX</td>
<td></td>
</tr>
</tbody>
</table>

7 In the **Return code for cleanup step** field, specify the value that should be compared against the condition code that is returned from any previous job step.

You can specify any two-digit value for the return code; however, BMC recommends that you specify 4.

8 In the **Temporary unit** field, type the name of the unit that is used to allocate temporary files when JCL is generated.

The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

9 In the **Include SYSPRIN2 DD** field, type **Y** or **N** to specify whether to generate the //SYSPRIN2 DD SYSOUT=* DD in the JCL.

The SYSPRIN2 output data set contains SYSPRINT output messages for versions 10.2 and later of the following BMC utilities:

- CHECK PLUS
- LOADPLUS
- REORG PLUS
- UNLOAD PLUS
If you type Y, you can view the SYSPRINT output from a utility while an execution job runs the utility or when an execution job cancels during the running of the utility.

---

**Note**
SYSPRINT2 data sets have the following restrictions:

- When you specify BMCSTATS YES or UPDATEDB2STATS YES for LOADPLUS or REORG PLUS, SYSPRINT2 does not contain the statistics report from the Common Statistics component.

- When invoking the IBM DSNUTILB utility, REORG PLUS and LOADPLUS ignore the SYSPRINT2 DD statement.

10 Press **END** to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

---

**Setting the JCL options for tapes**

To define or modify the values in your ISPF profile and a user POF, use the Options panels of the JCL Generation component.

Use the Tape Options panel to specify information about tape units and stacking options.

**To set the JCL options for tapes**

1 Use the following menu selections to display the JCL Generation Tape Options Update panel (Figure 68 on page 182):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update - Main Menu</td>
<td>Tape Options</td>
</tr>
</tbody>
</table>

**Figure 68: JCL Generation Tape Options Update panel**

```plaintext
----------------------- JCL GENERATION TAPE OPTIONS UPDATE -----------------------
COMMAND ===> 
Type data and press Enter.
Tape Unit 1 ........... CART (blank or TAPE, CART, etc)
Tape Unit 2 ........... TAPE (blank or TAPE, CART, etc)
Tape Unit 3 ........... TAPE (blank or TAPE, CART, etc)
Tape Volume count .... 99 (0 - 255)
Tape EXPDT .......... (Blank or YYDDD or YYYY/DDD)
```
2. In the **Tape Unit 1**, **Tape Unit 2**, and **Tape Unit 3** fields, type the names of valid tape units for your installation.

3. In the **Tape Volume count** field, type the maximum number of tape volumes.

4. In the **Tape EXPDT** field, type the expiration date for a tape.

5. In the **Tape RETPD** field, type the retention date for a tape.

6. In the **Tape TRTCH** field, type the parity, data conversion, translation, and compression value for 7-track tape drives as shown in Table 22 on page 183.

**Table 22: Values for 7-track tape drives**

<table>
<thead>
<tr>
<th>To choose</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not to use seven-track tape drives</td>
<td>(Leave the field blank)</td>
</tr>
<tr>
<td>Odd parity, conversion, and no translation</td>
<td>C</td>
</tr>
<tr>
<td>Even parity, no conversion, and no translation</td>
<td>E</td>
</tr>
<tr>
<td>Odd parity, no conversion, and translation</td>
<td>T</td>
</tr>
<tr>
<td>Even parity, no conversion, and translation</td>
<td>ET</td>
</tr>
<tr>
<td>Data compression</td>
<td>COMP</td>
</tr>
<tr>
<td>No data compression</td>
<td>NOCOMP</td>
</tr>
</tbody>
</table>

7. For each type of copy or product data set, type **Y** or **N** to specify whether the data sets should be stacked on a tape with data sets of the same type.

Consider the following if you choose tape stacking:

- Tape stacking options for baseline recovery data sets apply to CHANGE MANAGER only.

- Tape stacking options for backup SYSREC and archive data sets apply to CATALOG MANAGER only.

- Tape stacking is not applicable if you choose to dynamically allocate copy or unload data sets, unless you are using the NGT Copy utility to copy explicitly created table spaces.
Tape stacking is not applicable if you use the Database Administration or BMC Object Administration for DB2 solutions to execute a worklist in parallel.

Tape stacking is disabled in the JCL for a worklist if all of the following conditions exist:

— You use tape for unload (SYSREC) data sets, or you use tape because the maximum threshold value for DASD for a data set is exceeded.

— You use tape stacking for unload (SYSREC) data sets.

— You use the UNLOAD PLUS and LOADPLUS utilities and you include partitioned table spaces in the scope.

If you are using the UNLOAD PLUS and LOADPLUS utilities to multitask the unloading and loading of data, you cannot use tape stacking for unload (SYSREC) data sets. If you run the JCL, you might exceed the number of tape drives at your site.

Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

Setting the JCL options for temporary work data sets

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component. Use the Options For Sort Files panel to specify information about the temporary work data sets:

- SORTWORK (SORTWK)
- DATAWORK (DATAWK)
- LOGSORT (LOGSWK)

The temporary work data sets (such as SORTWORK) are defined by using DISP=(,PASS) in the JCL.

To set the JCL options for temporary work data sets

1 Use the following menu selections to display the JCL Generation Options For Sort Files Update panel (Figure 69 on page 185):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Individual Data Set Options</td>
</tr>
</tbody>
</table>
2 Specify the options for SORTWORK data sets:

a In the **Number of SORTWORK Data Sets** field, type the number of SORTWORK data sets.

b In the **SORTWORK unit name** field, type the name of the unit for SORTWORK data sets.

The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify **NONE**.

3 Specify the options for DATAWORK data sets:

a In the **Number of DATAWORK Data Sets** field, type the number of DATAWORK data sets.

b In the **DATAWORK unit name** field, type the name of the unit for DATAWORK data sets.

The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify **NONE**.

4 Specify the options for LOGSORT data sets:

a In the **Number of LOGSORT Data Sets** field, type the number of LOGSORT data sets.

b In the **LOGSORT unit name** field, type the name of the unit for LOGSORT data sets.
The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

5 If you typed N in the Data set sizing option field in Step 2 on page 177, specify the default primary and secondary quantities:
   a In the Default Primary Quantity field, type the value for the primary quantity in cylinders.
   b In the Default Secondary Quantity field, type the value for the secondary quantity in cylinders.

6 Specify the IBM Storage Management Subsystem (SMS) definitions for the optional SORTOUT data set classes:
   a In the SMS Data Class field, type the name of the data class.
   b In the SMS Storage Class field, type the name of the storage class.
   c In the SMS Management Class field, type the name of the management class.

7 In the $ORTPARM data set name field, type the name of the data set that provides parameters for SyncSort.

8 Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

Setting the JCL options for permanent data sets

To define or modify the values in your ISPF profile and a user POF, use the Options panels of the JCL Generation Update - Main Menu component. Use the individual data set options panels to specify information about the following permanent work data sets and image copy data sets:

- Sortout (WORKDDN or LOADDN)
- Sysut (WORKDDN)
- Copy (COPYDDN, RECOVERYDDN, RECOVERDDN, ICDDN, RECOVERYICDDN, OUTCOPYDDN, FCOPYDD, or EXPORTDDN)
- Sysrec (UNLDDN, INDDN, or UNLOADDDN)
- Archive (ARCHDDN)
- Cntl file (CNTLDDN)
- *(CHANGE MANAGER only)* Baseline recovery
- Discard (DISCADDRN)
- Error (ERRDNN)
- Map (MAPDNN)
- Report
- Punch (PUNCHDDN)
- Filter (FILTERDDN)

The permanent work data sets, which contain data, allow restarts. They are defined by using DISP=(NEW,CATLG) or DISP=SHR for restart or startover JCL. Examples include input (SYSUT), output (SORTOUT), discard (SYSDISC), map (SYSMAP), error (SYSERR), and punch (SYSPUNCH).

Other permanent data sets are used for restart and recover purposes. They use the same dispositions as the permanent work data sets. Examples include unload (SYSREC), copy (SYSCOPY), and baseline recovery (BLRP) (for CHANGE MANAGER only).

The LOB SYSREC data sets is used only by the UNLOAD PLUS utility to unload and load data contained in LOB columns.

The BMC Next Generation Technology Reorg for DB2 for z/OS (NGT Reorg) product uses only the copy, punch, and discard data sets.

**To set the JCL options for permanent data sets**

1. Use the following menu selections to display the panels for permanent work data sets:

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update - Main Menu</td>
<td>Individual Data Set Options</td>
</tr>
<tr>
<td>JCL Generation Individual Data Set Options</td>
<td>Type of data set (SORTOUT, SYSUT, or COPY)</td>
</tr>
</tbody>
</table>
Not all of the options are available on all of the data set options panels. If you choose to dynamically allocate your copy or unload data sets, any changes that you make to the copy (SYSCOPY), unload (SYSREC), and baseline recovery (BLRP – for CHANGE MANAGER only) data set options in the JCL Generation override panels do not take effect.

Figure 70: JCL Generation Data Set Options For Sortout Update panel

--- JCL GENERATION DATA SET OPTIONS FOR SORTOUT UPDATE ---

Type data and press Enter. Press PF3 or END to return to the main panel.

Enter Data Set Prefix below:
.. &PREFIX..&WKID..&STEPN
Unit Name . . . . . . . . . . . SYSDA   (SYSDA, TAPE, etc)
Primary Space . . . . . . . . 10       (Cylinders)
Secondary Space . . . . . . . . 2        (Cylinders)
Tape EXPDT. . . . . . . . . . .          (Blank or YYDDD or YYYY/DDD)
Tape RETPD. . . . . . . . . . .          (Blank or 1 - 9999 days)
SMS Data Class. . . . . . . .          (Blank or Data Class)
SMS Storage Class . . . . . .          (Blank or Storage Class)
SMS Management Class  . . . .          (Blank or Management Class)
Threshold Value . . . . . . . . 0        (Cylinders, 0 means no Threshold)
Alternate Unit Name . . . . . .        (SYSDA, TAPE, etc)
Alternate SMS Data Class. . . .       (Blank or Data Class Name)
Alternate SMS Storage Class . .       (Blank or Storage Class Name)
Alternate SMS Management Class       (Blank or Management Class Name)

2 Specify the prefix for the data set.

Consider the following items when you specify the prefix:

- The suffix in the name of an unload (SYSREC) or image copy data set varies, depending on whether the data set is dynamically allocated:
  - For data sets that are not dynamically allocated, JCL Generation appends the ddname to the prefix to create the name of the data set.
  - To suppress the ddname, specify Y for the appropriate keyword in the POF.

<table>
<thead>
<tr>
<th>Data set</th>
<th>POF keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local primary copy</td>
<td>PCPY1_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>PCPY2_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>RCPY1_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>RCPY2_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>File reference</td>
<td>UNLD_FREF_SUPPR_SUFF</td>
</tr>
<tr>
<td>Primary SYSREC</td>
<td>UNLD1_SUPPRESS_SUFF</td>
</tr>
<tr>
<td>Backup SYSREC</td>
<td>UNLD2_SUPPRESS_SUFF</td>
</tr>
</tbody>
</table>
For data sets that are dynamically allocated, Analysis appends a period, a two-character abbreviation, and a six-digit number to the prefix to create the name of the data set.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline recovery</td>
<td>&amp;PREFIX..&amp;OBNOD.BLnnnnnn</td>
</tr>
<tr>
<td>Discard</td>
<td>&amp;PREFIX..&amp;OBNOD.SDnnnnnn</td>
</tr>
<tr>
<td>Local primary copy</td>
<td>&amp;PREFIX..&amp;OBNOD.LPnnnnnn</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>&amp;PREFIX..&amp;OBNOD.LBnnnnnn</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>&amp;PREFIX..&amp;OBNOD..P&amp;PART.RPnnnnnn</td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>&amp;PREFIX..&amp;OBNOD..P&amp;PART.RBnnnnnn</td>
</tr>
<tr>
<td>EXPORT data set a</td>
<td>&amp;PREFIX..&amp;WKID.EXnnnnnn</td>
</tr>
<tr>
<td>Punch</td>
<td>&amp;PREFIX..&amp;WKID..&amp;STEPN.PUnnnnnnn</td>
</tr>
<tr>
<td>SYSUT</td>
<td>&amp;PREFIX..&amp;WKID..&amp;STEPN.SUnnnnnnn</td>
</tr>
<tr>
<td>Primary SYSREC b</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn</td>
</tr>
<tr>
<td>LOB SYSREC c</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn</td>
</tr>
<tr>
<td>XML file reference d</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn.Xnn</td>
</tr>
<tr>
<td>LOB file reference e</td>
<td>&amp;USERID..&amp;MSSID..&amp;WORKID8.SRnnnnnn.Lnn</td>
</tr>
</tbody>
</table>
The EXPORT data set is used by the Copy Migration feature of the Recovery Management, BMC Recovery for DB2, BMC Next Generation Technology Database Administration for DB2, BMC Object Administration for DB2, and Database Administration solutions. The Copy Migration feature includes the NGT Copy EXPORT command and NGT Recover IMPORT command.

If you are using the UNLOAD PLUS and LOADPLUS utilities to separate data sets for each table space partition and multitask the unloading and loading of data, Analysis also appends the P&PART variable to the prefix to create the name of the primary SYSREC data set (&USERID..&MSSID..&WORKID8.SRnnnnnn.P&PART).

For the LOB SYSREC data set (LOBDSN), Analysis also appends .Cnnn to the prefix, where nnn is the number of the LOB column in the table (&USERID..&MSSID..&WORKID8.SRnnnnnn.Cnnn).

The file reference data set is used in the SYXCnn OUTPUT descriptor for the BMC UNLOAD PLUS utility and in the SYXCnn TEMPLATE descriptor for the IBM UNLOAD utility. The utilities dynamically allocate the XML file reference data set. Assume that you are using the unload and load utilities to separate data sets for each table space partition and to multitask unloading and loading data. In that case, Analysis also appends the P&PART variable to the prefix to create the name of the primary SYSREC data set (&USERID..&MSSID..&WORKID8.SRnnnnnn.Xnn.P&PART).

The file reference data set is used in the SYXBnn OUTPUT descriptor for the BMC UNLOAD PLUS utility and in the SYLBnn TEMPLATE descriptor for the IBM UNLOAD utility. The utilities dynamically allocate the LOB file reference data set. Assume that you are using the unload and load utilities to separate data sets for each table space partition and to multitask unloading and loading data. In that case, Analysis also appends the P&PART variable to the prefix to create the name of the primary SYSREC data set (&USERID..&MSSID..&WORKID8.SRnnnnnn.Lnn.P&PART).

You must ensure the uniqueness of the name of the data set.

In a worklist, NGT Reorg always dynamically allocates data sets. Analysis adds the following suffixes to image copies that NGT Reorg creates. You can suppress the suffix by specifying *_SUPRESS = Y in the POF.

Table 23: Nonpartitioned objects without LOBs

<table>
<thead>
<tr>
<th>Data set</th>
<th>Suffix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local primary copy</td>
<td>.LPnnnn</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>.LBnnnn</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>.RPnnnn</td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>.RBnnnn</td>
</tr>
</tbody>
</table>

Note

You must ensure the uniqueness of the name of the data set.
### Table 24: Partitioned objects without LOBs

<table>
<thead>
<tr>
<th>Data set</th>
<th>Suffix</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local primary copy</td>
<td>.LPnnnn</td>
<td>Suppressing the suffix omits .LPnnnn, .LBnnnn, .RPnnnn, and .RBnnnn.</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>.LBnnnn</td>
<td>Using the &amp;GDG symbolic variable (Figure 71 on page 192) omits .LPnnnn, .LBnnnn, .RPnnnn, and .RBnnnn.</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>.RPnnnn</td>
<td>The P&amp;PART variable element of the suffix is included.</td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>.RBnnnn</td>
<td>If you use the &amp;GDG symbolic variable (Figure 71 on page 192), then .LPnnnn, .LBnnnn, .RPnnnn, and .RBnnnn are omitted. The P&amp;PART and (+1) symbolic variable are included.</td>
</tr>
</tbody>
</table>

### Table 25: Nonpartitioned objects with LOBs and AUX YES in reorganization

<table>
<thead>
<tr>
<th>Data set</th>
<th>Suffix</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local primary copy</td>
<td>.&amp;UNIQ</td>
<td>The value of *_SUPRESS_SUFF does not affect the suffix.</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>.&amp;UNIQ</td>
<td>When you use the &amp;GDG symbolic variable, the suffix is always .&amp;UNIQ.(+1)</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>.&amp;UNIQ</td>
<td></td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>.&amp;UNIQ</td>
<td></td>
</tr>
</tbody>
</table>

### Table 26: Partitioned objects with LOBs and AUX YES in reorganization

<table>
<thead>
<tr>
<th>Data set</th>
<th>Suffix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local primary copy</td>
<td>.&amp;UNIQ..P&amp;PART</td>
</tr>
<tr>
<td>Local backup copy</td>
<td>.&amp;UNIQ..P&amp;PART</td>
</tr>
<tr>
<td>Recovery primary copy</td>
<td>.&amp;UNIQ..P&amp;PART</td>
</tr>
<tr>
<td>Recovery backup copy</td>
<td>.&amp;UNIQ..P&amp;PART</td>
</tr>
</tbody>
</table>
Data set | Suffix | Details
--- | --- | ---
| | a | The value of \*_SUPRESS_SUFF does not affect the suffix.
| | b | When you use the &GDG symbolic variable, the suffix is always .&UNIQ..P&PART. (+1)

The following symbolic variables can be added by Analysis:

<table>
<thead>
<tr>
<th>Symbolic variable</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>&amp;UNIQ</td>
<td>Resolves to a unique eight-character string</td>
</tr>
<tr>
<td>P.&amp;PART</td>
<td>Resolves to a number that is from two to five digits long, depending on the length of the partition number</td>
</tr>
<tr>
<td>(+1)</td>
<td>Resolves to an eight-character string in the format G000nVooo</td>
</tr>
</tbody>
</table>

- To specify a GDG for the local and recovery image copy data sets, add the &GDG symbolic variable to the end of the data set prefix (Figure 71 on page 192).

**Figure 71: Using the &GDG symbolic variable**

```
COMMAND ===>
```

```plaintext
Type data and press Enter. Press PF3 or END to return to the main panel.
```

<table>
<thead>
<tr>
<th>Enter Data Set Prefix below:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>. &amp;PREFIX..&amp;OBNOD&amp;GDG</td>
<td></td>
</tr>
<tr>
<td>Unit Name</td>
<td>SYSDA (SYSDA, TAPE, etc)</td>
</tr>
<tr>
<td>Primary Space</td>
<td>10 (Cylinders)</td>
</tr>
<tr>
<td>Secondary Space</td>
<td>2 (Cylinders)</td>
</tr>
<tr>
<td>Tape EXPDT</td>
<td>Blank or YYDDD or YYYY/ddd</td>
</tr>
<tr>
<td>SMS Data Class</td>
<td>Blank or Data Class</td>
</tr>
<tr>
<td>SMS Storage Class</td>
<td>Blank or Storage Class</td>
</tr>
<tr>
<td>SMS Management Class</td>
<td>Blank or Management Class</td>
</tr>
<tr>
<td>Threshold Value</td>
<td>0 (Cylinders, 0 means no Threshold)</td>
</tr>
<tr>
<td>Alternate Unit Name</td>
<td>SYSDA, TAPE, etc</td>
</tr>
<tr>
<td>Alternate SMS Data Class</td>
<td>Blank or Data Class Name</td>
</tr>
<tr>
<td>Alternate SMS Storage Class</td>
<td>Blank or Storage Class Name</td>
</tr>
<tr>
<td>Alternate SMS Management Class</td>
<td>Blank or Management Class Name</td>
</tr>
</tbody>
</table>

When you use the &GDG variable, JCL Generation resolves the data set name using the symbolic variable, and the name includes the GDG number (Figure 72 on page 192).

**Figure 72: Data set names resolved with the &GDG symbolic variable**

```plaintext
/* Utility Copy DD Statements */
"SYCL0001 DD DSN=RDACRJ.DEMOCJ.S9(+1),
DCB=(SYS1.MODEL),
DISP=(NEW,CATLG,CATLG),
SPACE=(CYL,(10,2),RLSE),
UNIT=SYSDA
"SYCL0002 DD DSN=RDACRJ.DEMOCJ.S3(+1),
DCB=(SYS1.MODEL),
DISP=(NEW,CATLG,CATLG),
SPACE=(CYL,(10,2),RLSE),
```
If you specify a GDG for a dynamically allocated data set, Analysis does not append the two-character abbreviation and the six-digit number to the prefix.

**Note**

ALTER and CHANGE MANAGER use data set prefixes for unload (SYSREC) and image copy data sets. These prefixes ensure the uniqueness of the name of the data set when the data set is dynamically allocated. The POF that is generated in the HLQ.UBMCCNTL data set includes the new prefixes, unless you copy an existing POF during installation. If you copy an existing POF during installation to the HLQ.UBMCCNTL data set, you might need to modify the data set prefixes.

- The `SRnnnnnn` unload data set can contain the &MSSID symbolic variable, which identifies the DB2 subsystem ID (SSID) on the sending subsystem for a two-phase migration. The Analysis component also generates the -JCLP worklist command with the &MSSID variable and a value that refers to the sending SSID.

- The prefixes for copy data sets can contain the &TSSID symbolic variable, which identifies the receiving (or target) DB2 subsystem ID (SSID). If you specify the name of the DB2 SSID as the target SSID for a migrate-type work ID, the Analysis component generates a -JCLP worklist command with the TSSID parameter. The TSSID parameter specifies the name of the receiving (or target) SSID. JCL Generation uses the name of the receiving SSID to build the Execution JCL. For example, if the following command is in the worklist, JCL...
Generation Update - Main Menu uses the DEBA SSID for the receiving subsystem:

```
-JCLP 000700 MIGR TSSID DEBA
```

If you do not specify the name of an SSID as the target, or you specify an asterisk, JCL Generation Update - Main Menu uses the value of the &SSID symbolic variable as the value for the &TSSID variable.

**Note**

If the copy data set is dynamically allocated, JCL Generation Update - Main Menu substitutes the &SSID variable for the &TSSID variable in the COPY OUTPUT descriptor. The copy utility resolves the &SSID variable.

**Tip**

To specify the target SSID, see the task for defining migrate options for a migrate-type work ID in the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

3 In the **Unit Name** field, type the name of the unit.

Consider the following items when you specify the unit:

- The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

- If you use a tape unit, JCL Generation Update - Main Menu allocates the data sets before it opens them.

- For the Error data set (SYSERnnn) and Map data set (SYSMAP), specify SYSDA.

  If you specify TAPE and need to restart the IBM LOAD utility, you must uncatalog the existing data set. Then, you must change the DD statements to a disposition of (NEW,CATLG,CATLG).

- If you have large data sets and want to avoid extents or multiple data sets on DASD, specify a tape, virtual tape, or cartridge unit.

- If you are processing a worklist in parallel, you must specify a DASD unit for your permanent work data sets.

- Tape stacking is disabled if either of the following conditions exist. As a result, specifying tape for the unit might cause you to exceed the number of tape drives at your site.

  - You use tape for the sort input (SYSUT, SUT, WRK) or sort output (SORTOUT, SORTO, SORTP) data sets, or you use tape because the maximum threshold value for DASD for a data set is exceeded. Additionally, you specify a value for the maximum number of SYSUT temporary work
data sets in Analysis that the LOADPLUS and REORG PLUS utilities use to build nonclustering indexes.

—You use the UNLOAD PLUS and LOADPLUS utilities in Analysis, and you include partitioned table spaces in the scope.

■ As an alternative to specifying tape for the unit, you can specify a DASD unit that will span multiple data sets. Return to the JCL Generation Static Data Set Options panel (see “Setting the JCL options for static data sets” on page 177) and specify a value for **Max cylinders** and a value greater than 1 for **Max unit count**.

■ BMC recommends that you specify **SYSDA** for the unit name for ROWID SYSREC unload data sets. (The ROWID SYSREC data set is used only by the UNLOAD PLUS utility and the LOB DATA MOVER program in the Database Administration solution.) If the ROWID SYSREC is set to SYSDA, the performance of the worklist commands for unloading the ROWID data set can be improved when the worklist is executed in parallel.

4 If you typed **N** in the **Data set sizing option** field in Step 2 on page 177, specify the default primary and secondary quantities:

a In the **Primary Space** field, type the value for the primary quantity in cylinders.

b In the **Secondary Space** field, type the value for the secondary quantity in cylinders.

This option is not applicable if you choose to dynamically allocate copy or unload data sets.

5 In the **Tape EXPDT** field, type the expiration date for a tape.

6 In the **Tape RETPD** field, type the retention date for a tape.

7 Specify the SMS definitions for the data set classes:

a In the **SMS Data Class** field, type the name of the data class.

b In the **SMS Storage Class** field, type the name of the storage class.

c In the **SMS Management Class** field, type the name of the management class.

8 In the **Threshold Value** field, type, in cylinders, the primary quantity for the data set.

If this value is exceeded, JCL Generation Update - Main Menu uses the alternate unit and the alternate SMS parameters. Zero indicates that a threshold is not
specified for the unit. If you specify zero, JCL Generation Update - Main Menu does not use an alternate unit and the alternate SMS parameters.

When ALTER or CHANGE MANAGER dynamically allocates the copy data sets for IBM COPY, it uses TEMPLATE descriptors in the worklist for local and recovery primary and backup copies. If you specify a value for Threshold Value, the product provides templates for a primary set of data set attributes and an alternate set of data set attributes. If the threshold is exceeded, the utility can perform template switching and use the alternate data set.

For more information about the TEMPLATE descriptors, see ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2 and the -COPY worklist command in the ALTER and CHANGE MANAGER for DB2 Reference Manual.

**Note**
If you are processing a worklist in parallel, do not specify a threshold value.

9 In the **Alternate Unit Name** field, type the alternate name of the unit to be used if the threshold value specified in Step 8 on page 195 is exceeded.

**Note**
The alternate unit name must be a valid tape unit name. For more information, see “Setting the JCL options for tapes” on page 182.

10 Specify the SMS definitions for the alternate data set classes:

   a In the **Alternate SMS Data Class** field, type the name of the data class.
   
   b In the **Alternate SMS Storage Class** field, type the name of the storage class.
   
   c In the **Alternate SMS Management Class** field, type the name of the management class.

11 Press **END** to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

---

**Setting the JCL generation data group (GDG) options**

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Generation Data Group Options panel to specify information about GDGs.
To set the JCL options for generation data groups

1 Use the following menu selections to display the JCL Generation Generation Data Group Options Update panel (Figure 73 on page 197):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Generation Data Group Options (GDGDs)</td>
</tr>
</tbody>
</table>

Figure 73: JCL Generation Generation Data Group Options Update panel

```
---------- JCL GENERATION GENERATION DATA GROUP OPTIONS UPDATE ----------
COMMAND ===> Type data and press Enter.
  Define GDG base at JCL generation?.. N (Y/N)
  Specify NSCR on GDG definition?.. N (Y/N)
  Number of primary copy GDG entries.. 10 (1-255)
  Number of recovery copy GDG entries . 10 (1-255)
  Type GDG Model data set below:
  ... SYS1.MODEL
NOTE: GDGs only apply to copy data sets.
```

2 In the Define GDG base at JCL generation? field, type Y or N to specify whether JCL Generation creates the base of the GDG.

**Note**

JCL Generation cannot create the base of the GDG if you use IBM COPY to generate image copies for implicitly created objects.

3 In the Specify NSCR on GDG definition? field, type Y if the base of a GDG is defined in the IDCAMS DEFINE command as EMPTY (NSCR), or N if the base is defined as SCRATCH (SCR):

- SCR (the default) indicates to scratch (delete) the generation data set when it is uncataloged.
- NSCR indicates to uncatalog the generation data set when the maximum number of generation data sets to keep is reached.

4 In the Number of primary copy GDG entries field, type the maximum number of generation data sets to keep for primary copies.

5 In the Number of recovery copy GDG entries field, type the maximum number of generation data sets to keep for recovery copies.

6 At Type GDG Model data set below, type the name of the GDG model data set.
Note

If you type NONE, the DCB=model.dataSetName is omitted from the JCL for the data set.

7 Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

Setting the JCL debugging, display, and Execution options

To define or modify the values in your ISPF profile and a user POF, use the Options panels of the JCL Generation component.

Use the Debugging, Display and Execution Options panel to specify information about how comments are handled in the JCL.

To set the JCL options for debugging, display, and Execution

1 Use the following menu selections to display the JCL Generation Debugging, Display And Execution Options Update panel:

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Debugging, Display and Execution Options</td>
</tr>
</tbody>
</table>

**Figure 74: JCL Generation Debugging, Display And Execution Options Update panel**

AJX0DBG JCL GENERATION DEBUGGING,DISPLAY AND EXECUTION OPTIONS UPDATE --------

COMMAND ===>

Type data and press Enter.

Include data set sizing comments in JCL . . . . N (Y/N)
Include variable substitution comments in JCL . . N (Y/N)
Suppress comments in JCL . . . . . . N (Y/N)
NOTE: Do not set suppress comments to Y if you have specified either sizing or variable substitutions.
Specify an Alternate Program for IKJEFT01 . . . .
Specify a Plan name to run DSNTIAD. . . . .
Pre Job Step JCL INCLUDE member name . . . .
Post Step JCL INCLUDE member name . . . .
Post Job JCL INCLUDE member name . . . .
Include in AEXIN parameters:
SYNDELETE . . N (Y/N)  BINDFAIL . . N (Y/N)
HASHFAIL . . N (Y/N)  HASHMARNRC . . (NUMERIC)
REBINDFAIL . . N (Y/N)  REBINDRC . . (NUMERIC)
2MEGSQ . . N (Y/N)  NOFAILNOIMAGECPY N (Y/N)
STOPWAIT . . 3 (NUMERIC)
STOPWAIT SECS . . 10 (NUMERIC)

2 Specify whether to include debugging comments in the generated JCL:
Note

BMC recommends that you include the comments if you suspect that the JCL was generated incorrectly and you need to send documentation to Customer Support. If you want to reduce the number of lines of JCL, do not include the comments.

a In the Include data set sizing comments in JCL field, type Y or N to specify whether to include comments in the generated JCL that show statistics for determining data set sizes.

Comments are shown as dsso/cc, where dsso is the data set sizing option and cc is a comment code. Table 27 on page 199 lists the comment codes that Execution generates in the JCL.

Table 27: Comment codes for data set sizing

<table>
<thead>
<tr>
<th>Data set sizing options</th>
<th>Comment code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B, C, or O</td>
<td>C</td>
<td>Uses statistics from the DB2 catalog</td>
</tr>
<tr>
<td></td>
<td>H</td>
<td>Uses the high relative-byte address (RBA)</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>Uses multiple objects to size one data set (for example, SYSUTs)</td>
</tr>
<tr>
<td></td>
<td>N</td>
<td>Indicates that the data set could not be sized because statistics could not be found</td>
</tr>
<tr>
<td></td>
<td>O</td>
<td>Uses VSAM object sampling</td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>Uses the following formula to calculate the SORTWK size: (work space * 2) / number of SORTWK data sets</td>
</tr>
<tr>
<td></td>
<td>W</td>
<td>Warns that the sizing might be inaccurate</td>
</tr>
</tbody>
</table>

Figure 75 on page 199 shows example comments in the Execution JCL.

Figure 75: Execution JCL with comments

```
b In the **Include variable substitution comments in JCL** field, type **Y** or **N** to specify whether to include comments that show the SLIB variables and their assigned values.

JCL Generation uses these variables to resolve the names of the data sets in the generated JCL.

3 In the **Suppress comments in JCL** field, type **Y** or **N** to specify whether to suppress all comments in the generated JCL.

**Note**
If you chose to include either the statistics comments or the variable comments in Step 2 on page 198, you cannot select to suppress all comments in the JCL.

4 In the **Specify an Alternate Program for IKJEFT01** field, type the name of a program to be used instead of IKJEFT01.

**Note**
The alternate program is only used for nonworklist JCL.

5 In the **Specify a Plan name to run DSNTIAD** field, type the name of the DB2 plan to run the IBM DSNTIAD program.

6 In the **Pre Job Step JCL INCLUDE member name** field, type the name of a JCL member to be included before each step in the JCL.

7 In the **Post Step JCL INCLUDE member name** field, type the name of a JCL member to be included after each step in the JCL.

8 In the **Post Job JCL INCLUDE member name** field, type the name of a JCL member to be included at the end of a job.

9 For each AEXIN keyword, type **Y** or **N** to specify whether to include the keyword in the AEXIN input stream.

For more information about the keywords, see the list of keywords in the *ALTER and CHANGE MANAGER for DB2 Reference Manual*.

10 Press **END** to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.
Setting the installation options module names for BMC utilities

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the BMC Utility Option Module Names panel to specify the name of the installation options module for the BMC utilities.

To set the JCL options for utility installation options module names

1. Use the following menu selections to display the JCL Generation BMC Utility Option Module Names Update panel (Figure 76 on page 201):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Update Main Menu</td>
<td>Utility Options</td>
</tr>
<tr>
<td>Utility Options</td>
<td>BMC Utility Option Module Names</td>
</tr>
</tbody>
</table>

Figure 76: JCL Generation BMC Utility Option Module Names Update panel

--------------- JCL GENERATION BMC UTILITY OPTION MODULE NAMES UPDATE -------------
COMMAND ===>  
Type module names and press Enter.
Press PF3 or END to return to the main panel.

<table>
<thead>
<tr>
<th>Command</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NGT COPY . . . .</td>
<td>Default ACP$OPTS</td>
</tr>
<tr>
<td>REORG PLUS . . .</td>
<td>Default ARU$OPTS</td>
</tr>
<tr>
<td>LOADPLUS . . .</td>
<td>Default AMU$OPTS</td>
</tr>
<tr>
<td>UNLOAD PLUS . .</td>
<td>Default ADU$OPTS</td>
</tr>
<tr>
<td>NGT RECOVER . .</td>
<td>Default AFR$OPTS</td>
</tr>
<tr>
<td>CHECK PLUS . . .</td>
<td>Default ACK$OPTS</td>
</tr>
<tr>
<td>DASD MANAGER PLUS</td>
<td>Default blank</td>
</tr>
</tbody>
</table>

2. Specify the name of the installation options module for the BMC utilities.

If you use any of the BMC utilities, the installation options module names are listed in the AEXIN input stream in the JCL.

3. Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.
Setting the online reorg options

Use the Online Reorg Utility Options panel to specify the options for reorganizing table spaces.

--- Note ---
CATALOG MANAGER and DASD MANAGER PLUS currently do not use the values on the Online Reorg Utility Options panel.

To set the JCL options for online reorg options

1. Use the following menu selections to display the Online Reorg Utility Options panel (Figure 77 on page 202):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>Utility Options</td>
</tr>
<tr>
<td>Utility Options</td>
<td>Online Reorg Options</td>
</tr>
</tbody>
</table>

**Figure 77: Online Reorg Utility Options panel**

```
COMMAND ===> BMCREORG XBMID: XBMB
REORG MAPPING TABLE: J234.JFLTBMAB
REORG MAPPING DATABASE: XXXX

NOTE: The mapping table full length cannot exceed 72 characters
NOTE: The mapping database full length cannot exceed 8 characters
```

2. In the **BMCREORG XBMID** field, specify the EXTENDED BUFFER MANAGER (XBM) subsystem (SSID) that the REORG PLUS utility accesses.

REORG PLUS uses XBM or its XBM SNAPSHOT UPGRADE FEATURE (SUF) technology to create a snapshot of the data sets to be reorganized. ALTER and CHANGE MANAGER use this value when reorganizing a table space by using an online reorganization (SHRLEVEL CHANGE).

The value of the SSID can be from 1 to 8 characters long.

3. In the **REORG MAPPING TABLE** field, specify the name of the mapping table that the IBM REORG utility uses to map the row IDs (RIDs) in the source table to the RIDs in the target table.
Note

*(DB2 Version 11 and later)* If you want to use the system default values of IBM REORG, (not define a mapping table or a mapping database) for online reorg (SHRLEVEL CHANGE), perform the following steps:

1. On the Online Reorg Utility Options panel, do not specify a mapping database name.

2. On the Online Reorg Utility Options panel, do not specify a mapping table name.

3. On the ALTER and CHANGE MANAGER Analysis Options panel, set the Require Mapping Table for IBM field to N. See “Setting the Analysis options” on page 208.

The name can be from 1 to 72 characters long, and can contain the &ZUSER or &USERID symbolic variable.

Note

On the ALTER and CHANGE MANAGER Analysis Options panel, you can indicate whether to include the name of the mapping table in the syntax for the IBM REORG utility.

The REORG PLUS utility invokes the IBM DSNUTILB utility control program to enable certain features. If you have specified to use the REORG PLUS utility, you still need to specify mapping table information. For information about the features for which REORG PLUS invokes DSNUTILB, see the *REORG PLUS for DB2 Reference Manual*.

If you are specifying a mapping table for DB2 Version 11 and later, ensure that the length of the LRSN column is appropriate for the tablespace that is being reorganized.

4. *(DB2 Version 11 and later)* In the REORG MAPPING DATABASE field, specify the name of the default database that the IBM REORG utility uses when it implicitly creates a mapping table.

Note

On the ALTER and CHANGE MANAGER Analysis Options panel, the Require Mapping Table for IBM field must be set to N. See “Setting the Analysis options” on page 208.

The full length of the mapping database can be from 1 to 8 characters long.

You cannot define both the mapping database and the mapping table.

5. Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.
Setting the NGT utility options

Use the NGT Utility Options panel to specify NGT utility options.

**Note**
The CATALOG MANAGER and DASD MANAGER PLUS products currently do not use the values on the NGT Utility Options panel.

1. Use the following menu selections to display the NGT Utility Options panel (Figure 78 on page 204):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>ALTER or CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>Utility Options</td>
</tr>
<tr>
<td>Utility Options</td>
<td>NGT Utility Options</td>
</tr>
</tbody>
</table>

**Figure 78: NGT Utility Options panel**

AJX00CC --------------------- NGT Utility Options -----------------------------
COMMAND ===>
Type a value for Database name and press Enter.

DBNAME. . . DSNDB04 Database Name

2. In the **DBNAME** field, enter the name of a database to override the NGT_UTILDB value in the product options file (POF).

When the -NGTU command is included in a worklist, the NGT_UTILDB keyword controls the behavior of JCL generation and the NGT utility parameter +DBNAME (dbName) in the RRGPARMS DD in the execution JCL.

3. Press **END** to save your changes.

Setting the Execution worklist parallelism options

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the Execution Worklist Parallelism Options panel to specify the options for worklist parallelism in the CHANGE MANAGER component of the Database Administration and BMC Object Administration for DB2 solutions.
To set the JCL options for worklist parallelism

1. Use the following menu selections to display the Execution Worklist Parallelism Options panel (Figure 79 on page 205):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>CHANGE MANAGER Main Options Menu</td>
<td>JCLGEN Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>ALTER and CHANGE MANAGER Options</td>
</tr>
<tr>
<td>ALTER and CHANGE MANAGER Options</td>
<td>Worklist Parallelism Options (CHANGE MANAGER only)</td>
</tr>
</tbody>
</table>

Figure 79: Execution Worklist Parallelism Options panel

```
AJXOWPL ------- EXECUTION WORKLIST PARALLELISM OPTIONS UPDATE -----------------
COMMAND ===> 
Type data and press Enter. 
Press PF3 or END to return to the main panel. 
Run Worklist in Parallel mode . . N (Y/N) 
Start XIM . . . . . . . . N (Y/N) 
Display Parallel Trace . . . . . N (Y/N) 
Minimum XIM Initiators . . . . 2 (1 - 8) 
Maximum XIM Initiators . . . . 3 (1 - 32) 
XIM Proc Name . . . . . . . . XIMACM 
XIM Group Name . . . . . . . . XIMACM 
NOTE: The number of maximum initiators will determine the number of data set allocations. 
```

JCL Generation inserts the keywords for the selected override options into the AEXPIN input stream in the JCL. These keywords override the default keywords in the POF. The Execution component uses the AEXPIN input stream to control parallel processing. Execution uses the AEXPIN input stream in conjunction with the AEXIN input stream.

Table 28 on page 205 lists the options on the panel, the AEXPIN keyword, and the AJXPOFIN keyword that is overridden by the AEXPIN keyword.

Table 28: Execution worklist parallelism options

<table>
<thead>
<tr>
<th>Option</th>
<th>AEXPIN keyword</th>
<th>AJXPOFIN keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run Worklist in Parallel mode</td>
<td>PARALLEL YES</td>
<td>ACM_PARALLEL_WORKLST</td>
</tr>
<tr>
<td>Start XIM</td>
<td>XIMSTART YES</td>
<td>ACM_PARALLEL_XIMSTRT</td>
</tr>
<tr>
<td>Display Parallel Trace</td>
<td>TRACE YES</td>
<td>ACM_PARALLEL_XIMTRCE</td>
</tr>
<tr>
<td>Minimum XIM Initiators</td>
<td>MAXINITS</td>
<td>ACM_PARALLEL_MAXINIT</td>
</tr>
<tr>
<td>Maximum XIM Initiators</td>
<td>MININITS</td>
<td>ACM_PARALLEL_MININIT</td>
</tr>
</tbody>
</table>
Setting options

<table>
<thead>
<tr>
<th>Option</th>
<th>AEXPIN keyword</th>
<th>AJXPOFIN keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>XIM Proc Name</td>
<td>XIMPROC</td>
<td>ACM_PARALLEL_XIMPROC</td>
</tr>
<tr>
<td>XIM Group Name</td>
<td>XIMGROUP</td>
<td>ACM_PARALLEL_XIMGRP</td>
</tr>
</tbody>
</table>

2. In the **Run Worklist in Parallel mode** field, type Y to enable worklist parallelism.

To run a worklist in parallel, the PARALLEL keyword must also be specified in the ALUIN input stream. For more information, see the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

3. In the **Start XIM** field, type Y to start XIM automatically when you execute a worklist.

XIM is started only on the OS/390 or z/OS image on which you submit a job. If you use a data sharing environment, XIM is not started on other images.

4. In the **Display Parallel Trace** field, type Y to display trace messages in an output log during the execution of the worklist.

5. In the **Minimum XIM Initiators** field, type the minimum number of XIM initiators.

6. In the **Maximum XIM Initiators** field, type the maximum number of XIM initiators.

   **Note**

   If the maximum number of XIM initiators that you specify is larger than the number of objects in the scope of the worklist, your DASD might not be used efficiently.

7. In the **XIM Proc Name** field, type the name of the started task procedure for XIM.

   **Note**

   BMC recommends that the name of the started task procedure for XIM be unique for each instance of XIM that is running on an OS/390 or z/OS image. In addition, you should copy the procedure into a system PROCLIB data set. BMC also recommends that you use the suggested default value unless you have a specific requirement for another value.

8. In the **XIM Group Name** field, type the group name for XIM.

   **Note**

   The group name for XIM must be unique for each instance of XIM that is running on an OS/390 or z/OS image. BMC recommends that you use the suggested default value unless you have a specific requirement for another value.
9 Press END to save your changes, and to return to the CHANGE MANAGER Main Menu.

## Setting the worklist options

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the ALTER or CHANGE MANAGER Worklist Options panel to specify the options for sorting a worklist.

### To set the JCL options for worklist sorting

1. Use the following menu selections to display the ALTER and CHANGE MANAGER Worklist Options panel (Figure 80 on page 207):

<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>CHANGE MANAGERMain Options Menu</td>
<td>JCLGEN Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>ALTER and CHANGE MANAGER Options</td>
</tr>
<tr>
<td>ALTER and CHANGE MANAGER Options</td>
<td>Worklist Options</td>
</tr>
</tbody>
</table>

   **Figure 80: ALTER and CHANGE MANAGER Worklist Options panel**

   `AJXOWKL ---------- ALTER and CHANGE MANAGER WORKLIST OPTIONS ------------------ COMMAND ===>`

   Type data and press Enter. Press PF3 or END to return to the main panel.

   - **Sort Worklist**: (Blank - No Sorting, A - Automatic sorting, C - Object Cardinality sorting, N - Table Name sorting)
   - **Print Sort Worklist Messages**: Y (Y - Print Messages, N - No messages)

2. In the **Sort Worklist** field, type A, C, or N (or leave the field blank) to specify the sorting method, as shown in Table 29 on page 207:

### Table 29: Worklist sorting options

<table>
<thead>
<tr>
<th>For this method</th>
<th>Type</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sorting the worklist by each table’s cardinality in a descending sequence</td>
<td>C</td>
<td>None</td>
</tr>
<tr>
<td>Sorting the worklist by table order in an ascending sequence, according to the table owner and table name</td>
<td>N</td>
<td>None</td>
</tr>
</tbody>
</table>
For this method | Type | Considerations
--- | --- | ---
Sorting the worklist by either table cardinality or table order, depending on whether the worklist is processed in parallel | A | If you use the Database Administration or BMC Object Administration for DB2 solution to process the worklist in parallel, Analysis sorts the worklist by table cardinality. Otherwise, it sorts the worklist by table order.

Generating the objects in the worklist in an unsorted, random order | None |

3 In the **Print Sort Worklist Messages** field, type **Y** to record in the SYSPRINT data set and in the worklist the amount of time to sort a worklist.

4 Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

**Setting the Analysis options**

Use the ALTER and CHANGE MANAGER Analysis Options panel to specify the options for dynamic sortwork data sets, the mapping table name, and the default output class.

**To set the Analysis options**

1 Use the following menu selections to display the ALTER and CHANGE MANAGER Analysis Options panel (Figure 81 on page 208):

2 In the **Dynamic Sortwork Number** field, type the number of dynamically allocated sortwork data sets that the REORG PLUS or IBM REORG utility uses.
3 In the **Dynamic Sortwork Unit** field, type the unit for dynamically allocated sortwork data sets.

The value of the unit can be a name from 1 to 8 characters long, blank, or NONE. To omit the UNIT parameter from the JCL, specify NONE.

4 In the **Require Mapping Table for IBM Reorg** field, type **Y** to include the name of the mapping table in the syntax for the IBM REORG utility.

The IBM REORG utility uses the mapping table to map the row IDs (RIDs) in the source table the RIDs in the target table. For more details, see “Setting the online reorg options” on page 202.

---

**Note**

The REORG PLUS utility invokes the IBM DSNUTILB utility control program to enable certain features. If you have specified to use the REORG PLUS utility, you still need to specify mapping table information. For information about the features for which REORG PLUS invokes DSNUTILB, see the *REORG PLUS for DB2 Reference Manual*.

---

5 In the **Sysout class** field, type the name of the default output class.

6 In the **Use NGT embedded automation** field, type **Y** to use your configured NGT automation control points instead of data set definitions in your product options file (POF).

7 Press END to save your changes and return to the ALTER or CHANGE MANAGER Main Menu.

---

**Setting user variables**

To define or modify the values in your ISPF profile and a user POF, you can use the Options panels of the JCL Generation component.

Use the User Defined Variables Update panel to specify character variables. Each variable has a corresponding symbolic variable that you can use in job cards or data set prefixes.

**To set the JCL options for user-defined variables**

1 Use the following menu selections to display the User Defined Variables Update panel (**Figure 82 on page 210**):
From this menu | Select this item and press Enter
--- | ---
CHANGE MANAGER Main Menu | Options
CHANGE MANAGER Main Options Menu | JCLGEN options
JCL Generation Main Menu | User Defined Variable Values

**Figure 82: User Defined Variables Update panel**

```plaintext
--------------- USER DEFINED VARIABLES UPDATE ---------------
COMMAND ===> Type data and press Enter.

Character Variables:
- User variable 1... (Symbolic &UVR1)
- User variable 2... (Symbolic &UVR2)
- User variable 3... (Symbolic &UVR3)
- User variable 4... (Symbolic &UVR4)
- User variable 5... (Symbolic &UVR5)

NOTE: Symbolic variables cannot be input to these values
```

2 Specify the values for the variables.

The maximum length of a variable name is eight characters.

3 Press **END** to save your changes, and to return to the CHANGE MANAGER Main Menu.

**Creating a user POF**

To define or modify the values in your ISPF profile and a user POF, use the Options panels of the JCL Generation component.

Use the Product Options File (POF) Functions panel to create a user POF or update the values in your ISPF profile. The panel displays the data set name of the initial POF. The panel also displays the value of the POFDATE parameter in the initial POF that was last used to update the ISPF profile.

**To create a user POF**

1 Use the following menu selections to display the JCL Generation Product Options File (POF) Functions panel (Figure 83 on page 211):

From this menu | Select this item and press Enter
--- | ---
ALTER or CHANGE MANAGER Main Menu | Options
Figure 83: JCL Generation Product Options File (POF) Functions panel

--- JCL GENERATION PRODUCT OPTIONS FILE (POF) FUNCTIONS ---

COMMAND ===> 

Type data and press Enter. Press PF3 or END to return to the main panel.

Initial POF name:
AUS.DOPSEC.CNTL(JX10QBDC)
Initial POF name Different from previous?  N
BROWSE Initial POF . . . . . . . . . . . . . . . . . . N (Y/N)
MSGCLASS for PDF Diagnostic Messages . X
POFDATE value used for initial POF Refresh Compare:  2011/02/09 08:15:23

Type User POF Name below:
AUS.DOPSEC.CNTL(JX10QBAC)
BROWSE, EDIT, VALIDATE User POF . . . . . . . . . . . . N (B/E/V/N)
RESET Profile Variables from User POF that are marked with Refresh ",(R)" . . . . . . N (Y/N)
RESET All Profile Variables from User POF . . . . . . N (Y/N)
WRITE User POF data set from Profile Variables . . . . . N (Y/N)
-- Edit/Validate will update the User POFDATE --

2 In the **Type User POF Name below** field, replace the displayed name (the initial POF) with the name of the data set for a user POF.

The name can be either an existing sequential, 80-column data set or a member of a PDS.

--- Note ---
Consider creating a user POF to set the options for processing a worklist in parallel.

3 In the **WRITE User POF data set from Profile Variables** field, type **Y** to write the ISPF variable values (located in the ISPF profile) to the user POF.

4 Press **Enter** to create the user POF.

### Updating a user POF

You can update a user POF by using the options panels or by directly editing the file.

**To update the user POF in the JCL Generation options panels**

1 In the various options panels, specify your changes to the JCL Generation options.

2 Use the following menu selections to display the JCL Generation Product Options File (POF) Functions panel:
<table>
<thead>
<tr>
<th>From this menu</th>
<th>Select this item and press Enter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTER or CHANGE MANAGER Main Menu</td>
<td>Options</td>
</tr>
<tr>
<td>JCL Generation Main Menu</td>
<td>Product Option File (POF) Functions</td>
</tr>
</tbody>
</table>

3 In the **Type User POF Name below** field, type the name of the data set for your user POF.

4 In the **WRITE User POF data set from Profile Variables** field, type **Y**.

5 Press **Enter**.

**To update the user POF directly**

1 From the JCL Generation Product Options File (POF) Functions panel, in the **Type User POF Name below** field, type the name of the data set for your user POF.

2 In the **BROWSE, EDIT, VALIDATE User POF field**, type **E**.

   **Note**
   You can edit the user POF or the initial POF by using the ISPF edit macro AJXPODAT from the **HLQ.BMCCLIB** library. This library must be in your SYSPROC concatenation.

3 Edit and save the file.

4 In the **RESET All Profile Variables from User POF** field, type **Y** to update all of the ISPF variables in the ISPF profile with the variables in the user POF.

   **Note**
   You can update all of the ISPF variables in the ISPF profile with the variables in the POF by using one of the following **Command** line commands:

   - To use the variables in the initial POF, enter **TSO POFRESET**.
   - To use the variables in a specified initial or user POF, enter **TSO POFRESET POF (dataSetName(POFMember))**.

   To enable these commands, you must modify the POFRESET CLIST. The CLIST is located in the **HLQ.BMCCLIB** library. A current copy of this CLIST must be in the same SYSPROC concatenated library as your other CLISTs.

5 Press **Enter**.
Using multiple POFs

If you want to use different values for different applications, consider using more than one user POF. When you use multiple POFs, you can reset the values from a user POF that contains specifications for a particular application.

Some sample scenarios follow:

**Scenario 1**

The Payroll department needs backup copies of their data sets on tape, but other departments do not need backup copies.

You can create one user POF for the Payroll department and one for the other departments. Whenever you work with payroll objects, you can specify your payroll POF name to reset the profile variables from that POF.

**Scenario 2**

Your company is a service provider for several customers.

By using a separate POF for each customer, you can accommodate each customer’s naming standards.

**Scenario 3**

You have different requirements for test and production data.

For test data, you want to store the copies on DASD. For production data, you want to store the copies on tape. You can use separate POFs for test data and production data.

1. Follow the steps in “Creating a user POF” on page 210 to create an additional POF.

2. On the COMMAND line, enter TSO POFRESET POF(dataSetName(POFMember)).

---

**Note**

To specify the use of a different user POF, from the JCL Generation Product Options File (POF) Functions panel, in the Type User POF Name below field, type the name of the data set for your customized POF.
Refreshing the initial POF

Assume that your shop has revised its standards for naming data sets. To enforce the new standards, you need to refresh the initial POF so that users get the updated values.

The JCL Generation component uses the value of the POFDATE keyword and the refresh attribute when determining whether to reset the ISPF profile variables to the updated POF values. The refresh attribute of a POF keyword value indicates that the ISPF profile variable should be reinitialized from the POF value if one of the following conditions exists:

- The value of the POFDATE keyword is greater than that saved in the ISPF profile.
- The name of a new initial POF is different from the name of the POF that is saved in the ISPF profile.

To refresh the initial POF

1. Edit the initial POF outside of the product.
2. Change the value of the POFDATE keyword to the current date.
3. Append the refresh attribute,(R) to the values that you want to update.
4. Save the POF.

After you save the changes, users receive the updated ISPF variables the next time they invoke one of the products. However, users can still use their existing user POFs, which might not contain the updated values. To ensure that the user POFs use the updated values, users must reset all of their profile variables from the revised initial POF. If the users typically reset all of their variables from their user POFs, they must be sure to include the updated values.

Alternatively, specify the name of a new initial POF in the POFDS installation option.

Generating POF reports

Periodically, you might need to review POF keyword values, determine whether values are missing, or diagnose a problem. The following reports can assist you:

- The POF Validation Report lists a POF keyword, the action taken on the keyword, and the value of the keyword in the POF. For example, you can generate this report when you create a new user POF and change the values of
several keywords. The report shows the changes, the number of values that were refreshed, and any errors that resulted.

- The **Variables Initialized with Default** report lists the keywords that are missing from the initial POF and the default ISPF variables that are used to populate the keywords. You can generate this report when you want to view the new keywords and their values for a release.

**To generate the reports**

1. From the JCL Generation Product Options File (POF) Functions panel, in the **MSGCLASS for POF Diagnostic Messages** field, type the MSGCLASS for the SYSOUT field that is used to display messages.

   Consider using a SYSOUT class that is designated to go to the held queue so that you can view the output. Two SYSOUT files are allocated: AJXPOFER and AJXPOFVL.

   **Note**
   
   The default value for the **MSGCLASS for POF Diagnostic Messages** field is blank, which indicates that JCL Generation does not generate a report when you invoke the product.

   When you invoke the product or reset the POF, the reports are listed on the output for your TSO session.

**Reusing a POF in a subsequent installation**

Assume that you customized the values in your POF, and now you are installing a new release of a product. To avoid having to customize the values again, you can specify that the Installation System use your existing POF to populate the values in the new initial POF.

The new POF contains your current values plus any new keywords (and their values) for the new release.

**To reuse a POF**

1. Run the Installation System.

2. From the Install System JCL Generation File Information panel, in the **Use Existing POF to Populate the New Product Options File** field, type **Y**.

   **Figure 84: Reusing an existing POF**
3 Enter the names of the data set and member for the existing POF (Figure 84 on page 215).

For the member name, use the name of the POF that is used as the initial POF when you invoke the product.

**Figure 85: Specifying the name of the existing POF**

Enter the data set and member name of an existing Product Options File (POF) that is to be used to seed the new POF. An existence check will be performed, including verifying the specified member has a format consistent with the POF format.

<table>
<thead>
<tr>
<th>Command</th>
<th>Install System JCL Generation File Information</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>POF Data Set</th>
<th>____________________________</th>
</tr>
</thead>
<tbody>
<tr>
<td>POF Member</td>
<td>____________________________</td>
</tr>
</tbody>
</table>

### Overriding POF values in SLIBs

One of the primary advantages of using POFs is that you can customize your JCL without having to modify your SLIBs. Nonetheless, you might need to modify your SLIBs from time to time.

SLIB variables (or ISPF variables) are used in the SLIBs. Some of these SLIB variables correspond to the parameters in the POF. Note, however, that the names of the SLIB variables differ from the names of the POF keywords.

For example, if you specify the data set prefix for local primary copies (Figure 86 on page 217), the name of the primary copy data set resolves to the following name without modifications to the SLIB:

```
<SSIDname>.IC.T.ICPY.<databaseName>.<tableSpaceName>.<ddname>
```
For data sets that are not dynamically allocated, JCL Generation appends the *ddname* to the prefix to create the name of the data set.

**Figure 86: Specifying the prefix for a copy data set**

<table>
<thead>
<tr>
<th>JCL GENERATION DATA SET OPTIONS FOR LOCAL PRIMARY COPY UPDATE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>COMMAND ====&gt;</strong></td>
</tr>
<tr>
<td>Enter Data Set Prefix below:</td>
</tr>
<tr>
<td>. . &amp;SSID..IC.T.ICPY..&amp;DB..&amp;TS</td>
</tr>
<tr>
<td>Unit Name . . . . . . . . . SYSDA</td>
</tr>
<tr>
<td>Primary Space . . . . . . . . 10 (Cylinders)</td>
</tr>
<tr>
<td>Secondary Space . . . . . . . . 2 (Cylinders)</td>
</tr>
<tr>
<td>Tape EXPDT. . . . . . . . . (Blank or YYDDD or YYYY/DDD)</td>
</tr>
<tr>
<td>SMS Data Class. . . . . . . . (Blank or Data Class)</td>
</tr>
<tr>
<td>SMS Storage Class . . . . . (Blank or Storage Class)</td>
</tr>
<tr>
<td>SMS Management Class . . . (Blank or Management Class)</td>
</tr>
<tr>
<td>Threshold Value . . . . . . . . 0 (Cylinders, 0 means no Threshold)</td>
</tr>
<tr>
<td>Alternate Unit Name . . . . . . (SYSDA, TAPE, etc)</td>
</tr>
<tr>
<td>Alternate SMS Data Class . . . . . (Blank or Data Class Name)</td>
</tr>
<tr>
<td>Alternate SMS Storage Class . . . . (Blank or Storage Class Name)</td>
</tr>
<tr>
<td>Alternate SMS Management Class . . . (Blank or Management Class Name)</td>
</tr>
</tbody>
</table>

Now, assume that your site’s DBA decides that users should not have the ability to change the data set prefix from the options panels. The DBA can override the value in the SLIB. By specifying the value for the copy data set in the AJX#DSNS SLIB (Figure 87 on page 217), the DBA can uphold your site’s naming standards.

**Figure 87: Changing the SLIB variable for the copy data set in AJX#DSNS**

```cm
)SEL &AJXC1PRF ^= &Z
)SET SYSC1PR  = &AJXC1PRF => Resolved values from ISPF profile
)ENDSEL
)SEL &AJXC1PRF = &Z
)SEL &AJXSYCOP ^= &Z
)SET SYSC1PR  = &AJXSYCOP
)ENDSEL
)SEL &AJXSYCOP = &Z
)SET SYSC1PR  = &AJXHLQ..&AJXDB..&AJXTS
)ENDSEL
)ENDSEL
)SEL &AJXC2PRF ^= &Z
)SET SYSC2PR  = &AJXC2PRF
)ENDSEL
)SEL &AJXC2PRF = &Z
)SEL &AJXSYCOP ^= &Z
)SET SYSC2PR  = &AJXSYCOP
)ENDSEL
)SEL &AJXSYCOP = &Z
)SET SYSC2PR  = &AJXHLQ..&AJXDB..&AJXTS
)ENDSEL
)ENDSEL
)SET SYSC1PR  = user.shop.standards.&AJXDB..&AJXTS <= Override with SLIB variables and fixed values

After changing an SLIB variable, the DBA should use JCL Generation to test the changes. If the SLIB is coded correctly, the DBA must then recompile the SLIB. The
DBA can use the SLIB compiler tool that is supplied with the Administrative products to compile the SLIB.

For more information about testing the changes or using the SLIB compiler, see the *ALTER and CHANGE MANAGER for DB2 User Guide, Volume 2*.

### Creating OUTPUT or TEMPLATE descriptors for copies

In ALTER and CHANGE MANAGER, you can use BMC and IBM utilities to dynamically allocate data sets that are used to create valid image copies.

You can use an option in the POF to create an OUTPUT or a TEMPLATE descriptor for the Analysis component of ALTER and CHANGE MANAGER. When generating a worklist, Analysis uses that descriptor for local and recovery primary and backup copies (Figure 88 on page 218).

**Figure 88: OUTPUT descriptor in worklist**

```
BMCC_D01250
OPTIONS IXDSNUM ALL
OUTPUT DCPYLOCP
UNIT SYSDA
DSNAME PAYROLL.IC.LP.&DB.&TS
MODELDCB SYS1.MODEL
COPY INDEX ACMX01.Z_X06SSB1
COPYDDN (DCPYLOCP)
```

To create OUTPUT or TEMPLATE descriptors for copies

1. From the ALTER and CHANGE MANAGER Main Menu, select **Options** and press Enter.
2. Select **JCLGEN Options** and press Enter.
3. Select **Individual Data Set Options** and press Enter.
4. Select **Copy** and press Enter.
5. Select **Local Primary Copy** and press Enter.
6. Type the name of the prefix for the data set that will be dynamically allocated (Figure 89 on page 218).

**Figure 89: Specifying the prefix for a dynamically allocated data set**

```
--------------------------- JCL GENERATION DATA SET OPTIONS FOR LOCAL PRIMARY COPY UPDATE ----
COMMAND ===> 
Type data and press Enter. Press PF3 or END to return to the main panel.
```

Enter Data Set Prefix below:

```
... PAYROLL.IC.LP..&DB..&TS
```
If you are using foreground Analysis to generate the worklist, JCL Generation uses the prefix in your ISPF profile variables to create the OUTPUT descriptor. However, if you are using batch Analysis, JCL Generation generates a partial POF from your ISPF profile variables and includes this partial POF in the batch Analysis JCL (Figure 90 on page 219).

Figure 90: Excerpt of batch Analysis JCL with POF keywords

```
//AJXPOFIN DD   *
PCPY1_DATACLASS_ALT=
PCPY1_DATACLASS=
PCPY1_EXPDT=
PCPY1_MGMTCLASS_ALT=
PCPY1_MGMTCLASS=
PCPY1_PREFIX=PAYROLL.IC.LP..&DB..&TS
PCPY1_PRIQTY=10
PCPY1_RETPD=
PCPY1_SECQTY=2
PCPY1_STACK=N
PCPY1_STORCLASS_ALT=
```
Glossary

A

ACM

The product code that BMC uses to identify the CHANGE MANAGER product.

ACT

The product code that BMC uses to identify the CATALOG MANAGER product.

action

In DASD MANAGER PLUS, an action is the fundamental work unit for generating utilities. An action names a set of services. After you create an action, you can copy, edit, or delete it. See also corrective action.

action code

A one-character or two-character command that you can enter on one of the lines of a list panel. Although you can only enter one command per line, you can enter multiple commands on a single panel.

action name

An action name is a string of up to eight alphanumeric characters excluding percent (%), asterisk (*), underscore (_), and space. When specifying an action name, you can use a wildcard pattern to display a group of similar names.

action owner

Authorization ID of the creator of the action.
**action POF**

A POF that can be written from the ISPF variables that are set in the product or edited. An action POF can be used to reset all of the options that will be used in the current session to create JCL.

**action status**

The Execution status of the action. DASD MANAGER PLUS sets the action status to **N** (not started) when the worklist is created. The Execution program updates the action status to **S** (started) if the worklist has started but not completed, or to **R** (rerunnable) if the worklist has completed. If you submit a worklist for UNDO, the status becomes **U**. You can start over or restart a worklist that has an action status of **S**. You must rebuild a worklist having status **U** before the worklist can be resubmitted.

**action type**

The type of action: **U** for utility.

**Administrative Products for DB2**

A collection of products from BMC that includes ALTER, CATALOG MANAGER, CHANGE MANAGER, and DASD MANAGER PLUS. These integrated products are designed to help database administrators, system programmers, and application developers automate the tasks associated with the implementation and administration of a DB2 Universal Database for z/OS system.

When all Administrative products are installed properly, they can access some of each other’s functionality. In addition, these products can access the IBM and BMC utilities.

**AEXIN**

The ddname of the input stream that the Execution component uses.

**AEXPIN**

The ddname of the input stream that the Execution component uses to control parallel processing.

**AEXPRINT**
The ddname of diagnostic output of the Execution component. This diagnostic output data set contains all output from the Execution process, including DB2 for z/OS utility messages, BMC utility messages, dynamic SQL messages, IDCAMS messages, and any other messages that are generated by the actions of the worklist. AEXPRINT is frequently referred to as the worklist execution log.

AEXPRnnn
The ddname of one of the initiator outputs of the Execution component. When you use the worklist parallelism feature of the Database Administration or BMC Object Administration for DB2 solution, this output data set contains AEXPRINT output from each BMC Cross-System Image Manager (XIM) initiator.

AEXPTRAC
The ddname of the trace output of the Execution component. When you use the worklist parallelism feature of the Database Administration or BMC Object Administration for DB2 solution, this output data set contains tracing records.

AEXSYyynn
The ddname of one of the initiator outputs of the Execution component. When you use the worklist parallelism feature of the Database Administration or BMC Object Administration for DB2 solution, this output data set contains system messages and job information for each BMC Cross-System Image Manager (XIM) initiator.

AJX variables
A group of variables (JOB, STEP, and DD) that are used during Job Control Language (JCL) generation. JOB global variables are set once per JCL creation session. STEP global variables are set at the beginning of a JCL creation session and are updated when a new JOBSTEP is detected. Data Definitions (DD) variables are local variables that are set for each creation of a JCL DD entry. Descriptions of these variables are provided in HLQ.BMCSLIB($AJXDOCV).

AJXIN
The ddname of the input stream that the Batch Execution JCL Generation component uses.

AJXPOFIN
The ddname of the override input stream that the Batch Execution JCL Generation component uses.
AJXPOFVL

The ddname of the product options file (POF) validation report that the Batch Execution JCL Generation component uses.

AJXPRINT

The ddname of the diagnostic output of the Batch Execution JCL Generation component.

alloc unit

The allocation unit that is used for space estimation calculations, volume placement, and primary and secondary quantities. Possible values are K (kilobytes), T (tracks), or C (cylinders). The default value comes from the installation options modules.

ALTDD

A BMCSTATS parameter that specifies whether to use an alternate data set, such as the name of a DSN1COPY data set, against which to collect statistics.

ALTE for DB2

A BMC product that provides advanced database administration and manipulation within a single DB2 subsystem. ALTER streamlines the process of changing and migrating database objects, handles analysis for both changes and migrations, and automatically generates SQL, DB2 commands, and utilities.

alter-type work ID

A work ID that ALTER and CHANGE MANAGER use to perform modifications to the local DB2 subsystem.

ALU

The product code that BMC uses to identify the ALTER product.

ALUIN

The ddname of the input stream that the Import, Baseline, Baseline Report, Compare, Analysis, and CM/PILOT components of CHANGE MANAGER use.
ALUPRINT

The ddname of the diagnostic output of the Import, Baseline, Baseline Report, Compare, and Analysis components of CHANGE MANAGER.

Analysis

A component of ALTER and CHANGE MANAGER that reads the change and migration requests for a particular work ID and generates a worklist to implement the requests. Analysis checks for consistency with the DB2 catalog before it builds the worklist.

Application

In the CM/PILOT component, the association of a group of CHANGE MANAGER profiles that are used repeatedly for the change management tasks of a specific DB2 application.

ARU

The product code that BMC uses to identify the REORG PLUS for DB2 product.

ASU

The product code that BMC uses to identify the DASD MANAGER PLUS product.

attribute

A value that defines certain properties of an object. Each attribute of an object can occur only once and has a single value from a finite list of possible values. For example, some of the attributes of the TABLE object are database, table space, and EDITPROC. Attributes differ from sub-elements because sub-elements can occur multiple times and can have attributes of their own. For example, columns are sub-elements of tables.

AUTHID

See “authorization ID” on page 225.

authorization ID
An identifier that is allowed a set of privileges. An example of the authorization ID is the owner of a table space, database, storage group, or synonym. An authorization ID is the implicit qualifier of a table, view, alias, or index name.

**automation modes**

Levels of automation that the Database Performance for DB2 solution provides to suit different user requirements. You choose the automation mode at installation, and you can change the mode to match your site’s changing needs. See also full automation mode and standard JCL mode.

**automation spawner**

The processing component of the Database Performance for DB2 solution that manages registered candidates, object processing, JCL generation, and job management. The automation spawner consists of the automation spawner address space and one or more automation spawner subtasks. The automation spawner address space functions as a control point for automation spawners that correspond to discrete instances of DB2. The automation spawner is the component responsible for all of the Database Performance for DB2 solution utility automation processing.

**AutoSub**

A BMCTRIG parameter that automatically submits utility JCL on an exception. The data set named in the **Util DSN** field will be submitted without modification.

**auxiliary index**

An index on an auxiliary table. Each index refers to a large object (LOB) column.

**auxiliary list**

A list of several auxiliary objects (such as auxiliary table spaces, tables, or indexes) of only one type.

**auxiliary table**

A table that contains a single large object (LOB) column. An auxiliary table resides in an auxiliary (or LOB) table space.
auxiliary table space

A nonpartitioned table space that contains the data for a large object (LOB) column in an auxiliary table.

B

base table

A table that contains a ROWID column and the definition for a large object (LOB) column. A base table is incomplete if it does not contain any auxiliary objects, such as auxiliary table spaces, tables, and indexes.

base table space

A table space that contains base tables.

baseline

A component of CHANGE MANAGER that captures a set of DB2 structure definitions from either the DB2 catalog or a DDL file at a specific point in time.

baseline name template

A template that a baseline profile can contain. This template is used to create the names of the baselines that are established with the profile. The template can include arbitrary text and the special character sequences #### and @@@@@@ or @@@@@@@@, or a combination of both types of sequences. When the baseline is created, ascending numbers replace the #### sequence and the current date replaces the @@@@@@ or @@@@@@@@ sequence. For @@@@@@, the date is in the form YYMMDD. For @@@@@@@@, the date is in the form YYYYMMDD.

baseline profile

A BMC object that contains the information that is necessary to establish a baseline. The two types of baseline profiles are catalog and DDL.

batch component

A component that can run in batch mode. Batch components of CHANGE MANAGER include Analysis, Baseline, Baseline Report, Compare, Execution, and Import.
BMC object

A logical entity that contains data that is necessary for performing tasks. BMC objects have a two-part name \((a.b)\). The BMC objects for CHANGE MANAGER include work IDs, worklists, unload data sets, baselines, profiles, CDL files, DDL files, internal tables, task IDs, CM/PILOT worklists, applications, and script tables.

BMCCOPY

The NGT Copy utility that is used to create an image copy. The short form of the command is BMCI.

BMCCPRS

A DASD MANAGER PLUS utility command that copies statistics from the DB2 catalog to the DASD MANAGER PLUS historical database. The short form of the command is BMCC.

BMCLOAD

The BMC LOADPLUS utility that is used to load DB2 tables. The short form of the command is BMCL.

BMCREORG

The BMC REORG PLUS utility that is used to perform a reorganization. The short form of the command is BMCR.

BMCSTATS

A command that invokes the statistics collecting function of the DASD MANAGER PLUS product. The short form of the command is BMCS. BMCSTATS is similar to the IBM RUNSTATS utility. It provides the same statistics plus many additional statistics.

BMCTRIG

A DASD MANAGER PLUS program that evaluates exception thresholds and optionally generates corrective actions. Thresholds are user-defined limits that enable automatic utility generation. Thresholds are based on information from the statistics tables in the DASD MANAGER PLUS database, IDF catalog information, DB2 status, RTS data, and user-defined exceptions through REXX programs. Percentage increase thresholds are based on a comparison between the most current statistics run and the previous statistics run. Thresholds are based on a statistical limit. The short form of the command is BMCT.
BMCUNLOAD

The BMC UNLOAD PLUS utility that is used to unload data from a full image copy of one or more tables in a table space. The short form of the command is BMCD.

BMCUPRS

A DASD MANAGER PLUS utility command that updates the DB2 catalog statistics with BMCSTATS statistics (using SQL UPDATE). The short form of the command is BMCU.

box

A DASD MANAGER PLUS graphic display option that specifies whether to place a box around the legend of statistical graphs: KBOX (box) or NKBO (no box).

browse function

A DASD MANAGER PLUS function that enables you to display statistics from the DB2 catalog. This function is available on the DB2 Object List panel. Compare with display function.

buffer pool

Main storage reserved to satisfy the buffering requirements for one or more table spaces or indexes.

Buffers

A BMCSTATS parameter that specifies the number of four-kilobyte I/O buffers each task in a multitasking job uses for reading data (2 through 999).

build JCL

A DASD MANAGER PLUS job generation option that instructs JCL Generation to build the JCL from the worklist.

build worklist

A DASD MANAGER PLUS job generation option that builds (or rebuilds) the worklist using the information specified for a particular action.
CANCEL (CAN)

An ISPF command that returns you to the previous panel without saving any changes made to the current panel.

catalog indirection

An optional method of implementing the Administrative products that allows them to access the DB2 catalog indirectly when making information queries. Catalog indirection is accomplished by using aliases that point to a copy of the DB2 catalog. The major benefit is to reduce catalog contention.

catalog baseline

A baseline that is established on part of the DB2 catalog. A catalog baseline must include a scope, which might be defined directly in a baseline profile or by reference to a scope in a migrate profile.

CATALOG MANAGER for DB2

A tool that is designed to automate the day-to-day tasks associated with administering DB2. This product provides an interactive, intuitive, and easy-to-use interface for submitting DB2 commands and retrieving catalog information using qualified lists, wildcard searches, and dependency lists. CATALOG MANAGER provides the ability to create and drop DB2 objects, re-create dropped structures and data, and browse and edit table data.

CDL

See “Change Definition Language (CDL)” on page 230.

Change Definition Language (CDL)

A BMC proprietary language that is used to specify changes to DB2 data structures.

CHANGE MANAGER for DB2

A BMC product that enables database administrators, system administrators, and developers to manage user applications and individual database objects globally. As a robust extension of
the ALTER product, CHANGE MANAGER automates data structure changes across multiple DB2 subsystems and between DB2 and data modeling tools by providing a way to implement, migrate, synchronize, and back out data structure changes while preserving structure modifications that might be unique to a specific subsystem.

change rule

An element of a migrate-type work ID or a migrate profile. Change rules define the automatic object attribute changes that are made during a migration or change migration process. These rules can include changes, or sub-element inclusion or exclusion. Change rules can also force VCAT definitions for table spaces and indexes.

char H/W

A DASD MANAGER PLUS graphic display option for statistical graphs that specifies the character height relative to its width. For example, if you specify 200, the height will be multiplied by twice the width.

char size

A DASD MANAGER PLUS graphic display option that specifies the character size multiplier for statistical graphs. This multiplier is divided by 100 and the spacing sizes are multiplied by this factor.

checkpoint

A point at which information about the status of a job and the system can be recorded so that the job step can later be restarted.

See also “sync point” on page 258.

child

A DB2 object that contains the foreign keys which reference the primary key in a parent table.

See also “parent” on page 249.

CLIST

Command list.
CM/PILOT for DB2

A component of CHANGE MANAGER that automates the DB2 change management processes that you perform using CHANGE MANAGER.

CM/PILOT worklist

A data set that contains the ordered commands, keywords, and parameters that CHANGE MANAGER needs to process a task ID.

color

A DASD MANAGER PLUS graphic display option that specifies the color of headers, titles, labels, legends and boxes. Specify the color by number according to the Color Key provided on the panel.

columns

A DASD MANAGER PLUS printing option that specifies the number of columns per row.

This is also a parameter used by BMCSTATS, BMCCPRS, and BMCUPRS to name the columns for which statistics are to be collected, copied, or updated. You can type column names in the format COL1, COL2, and so on; type ALL for all columns; or leave blank for none.

command

A token that you can enter at the command prompt on a panel.

See also “action code” on page 221.

commit

An operation that terminates a unit of work. A commit releases all locks. Data that was changed is now consistent.

Compare

A component of CHANGE MANAGER that identifies the differences between two sets of data structures and then generates a CDL file. You can review the differences and decide how to apply the differences to implement the necessary changes. You can compare data structures that are stored in a DDL file, baseline, worklist, or DB2 catalog.
Compare1

The primary input to the compare process.

See also “source” on page 255.

Compare2

The secondary input to the compare process.

See also “target” on page 260.

component

A major functional unit of ALTER or CHANGE MANAGER, such as Analysis, Execution, Specification, or Import.

constraint

See “referential constraint” on page 252.

copies

A DASD MANAGER PLUS printing option that specifies the number of image copies to make.

CopyPend

An exception you can set using BMCTRIG to determine whether the copy pending flag is on.

corrective action

An action specifying a corrective action (such as REORG, COPY, and so on) to perform based on the exceptions that are initiated by a BMCTRIG job. A corrective action is predefined, specifies no objects, and can be initiated when specific objects meet certain criteria. A skeleton specifies only utilities and commands and can be designed to suit multiple objects and situations. See also action.

Cross-System Image Manager (XIM)

A BMC technology that provides sysplex performance improvements by enabling the distribution and management of discrete units of work (UOW) across one or more OS/390 and
z/OS systems. The BMC products that use XIM can divide single, long-running tasks into multiple parallel tasks that can be run across multiple computers in the sysplex, thus decreasing the overall elapsed time. The products can also be used with XIM in a data sharing environment on a single OS/390 or z/OS image.

D

DASD MANAGER PLUS for DB2

A BMC product that automates utility generation, gathers comprehensive statistics, monitors changes in the database, and enables you to perform maintenance based on the condition of the data instead of a rigid schedule.

DASD MANAGER PLUS database

The database provided and maintained by DASD MANAGER PLUS. The database is created during installation and named BMCASU. The DASD MANAGER PLUS database contains statistics tables (RS_%), utility job tables (UT_%), worklist tables (WL_%), and an action table (DO_WORKIDS).

Data Control Language (DCL)

A category of SQL statements that control data security.

data definition language (DDL)

A category of SQL statements that create, modify, or delete database objects.

data definition name (ddname)

The name of a data definition (DD) statement in job control language (JCL) that corresponds to a data control block that contains the same name.

Data Manipulation Language (DML)

In the CM/PILOT component of CHANGE MANAGER, SQL-like statements that can be used to update, delete, and migrate data structures.

data set sizing
The process of determining data set allocations, especially as used by the JCL Generation component. Data set sizing is distinct from space estimation.

**data structure**

An object that is defined in the DB2 catalog. Objects include storage groups, databases, table spaces, tables, indexes, foreign keys, views, synonyms, aliases, and triggers.

**database administrator (DBA)**

An individual who is responsible for the design, development, operation, security, maintenance, and use of databases.

**database request module (DBRM)**

A module that contains SQL statements which the DB2 precompiler has extracted from a source program.

**DB2 catalog**

System tables, maintained and used by DB2, that contain descriptions of DB2 objects such as tables, views, and indexes.

**DB2 command**

An instruction to the DB2 subsystem. Some example command processes enable you to start or stop DB2, display information on current users, start or stop databases, and display information about databases. DB2 commands always begin with a hyphen (-).

**DBA**

See “database administrator (DBA)” on page 235.

**DBCS**

See “double-byte character set (DBCS)” on page 237.

**DD statement**

Data Definition statement.
DDL

See “data definition language (DDL)” on page 234.

DDL baseline

A baseline that is established on a file that contains DDL.

ddname

See “data definition name (ddname)” on page 234.

default options module (DOPTS)

See “installation options module” on page 243.

default value

A predetermined value, attribute, or option that is assumed when no other is explicitly specified.

DeleteAge

A parameter used by BMCSTATS to delete statistics that have reached a certain age or by BMCTRIG to delete exceptions that have reached a certain age. Both deletions are made in the DASD MANAGER PLUS database.

delimited identifier

An SQL identifier that is enclosed within escape characters.

dependencies

The name or values of objects which another object uses as part of its definition or as a hierarchical subordinate.

dependent

A child object (row or table) that has at least one parent.
dependent object

An object whose definition relies on the name or the values of another object. The dependent object references the other object.

destination

The intended receiving location for CDL or a worklist.

device name

A DASD MANAGER PLUS printing option that specifies the LUNAME of the printer.

device type

The type of disk device used for DB2 data set allocation, such as 3380 and 3390, or generic types TAPE and CART.

DOPTS

See “installation options module” on page 243.

double-byte character set (DBCS)

A delimited set of characters in which each character is represented by two bytes. Katakana and other lowercase characters are nonstandard characters and must be contained within double quotes.

duplicate

An action that you can make on a database object. You can duplicate single or multiple objects within the same database system. If you want to create an object that is like an existing object, you can use the Create Like command to duplicate the existing object. You can then change the object name and make any other necessary changes.

E

edit procedure
EDITPROC

An edit procedure that defines an editing routine to be invoked just after a record that corresponds to a table row is retrieved and just before that record is stored. Editing routines allow for data compression, decompression, and encrypting.

END

An ISPF command, similar to Enter, that validates and processes the information on a panel and returns you to the previous panel, but does not execute commands. This command is typically programmed on your keyboard as a function key, such as PF3.

Enter key

The key that executes any commands that have been specified. For a sequence of panels, the Enter key displays the next panel.

entry field delimiter

A user option that enables you to specify the highlighting to be used for the user input fields of DASD MANAGER PLUS panels (underscore, reverse video, blink, or no highlighting).

estimate source

A DASD MANAGER PLUS user option that enables you to specify the source to be used for estimating the size of the utility work data sets. Valid entries are N (no estimation), B (current BMCSTATS statistics), C (DB2 catalog statistics), or O (object sampling). See also space estimation function.

events log

A collection of records that describe the events occurring during utility execution and their sequence. If you enable the Record Events option on the Action Generation panel, these records will be stored in the DASD MANAGER PLUS EVENTS table. Event records are useful for recovery in case of failure or for general analysis purposes.

EVENTS table
A DASD MANAGER PLUS statistics table that sequentially stores the events recorded during utility execution. One row is recorded for each utility execution event. This table is named BMCASU nn.EVENTS.

**exception**

A statistical value in the DASD MANAGER PLUS historical database that meets or exceeds a user-specified threshold value that has been set with BMCTRIG. Such exceptions are stored in the exceptions table, and they can be examined through ISPF dialogs (Exceptions report). A predefined corrective action can be generated automatically based on the identification of an exception.

**exception status**

In DB2, an abnormal table space or partition status (for example, check pending, copy pending, or recover pending).

**exceptions report**

A list of exceptions from the DASD MANAGER PLUS historical database. Each exception line includes the object name, type of exception, and a timestamp.

**EXCEPTIONS2 table**

A DASD MANAGER PLUS statistics table that stores exceptions identified by BMCTRIG. This table is named BMCATS nn. RS_EXCEPTIONS2.

**Execution**

A component of ALTER and CHANGE MANAGER that carries out the commands in a worklist.

**exit routine**

A program (BMC, IBM, or user-written) that receives control from DB2 to perform specific functions. Exit routines run as extensions of DB2 (for example, authorization checking).
field procedure

See “FIELDPROC” on page 240.

FIELDPROC

A user-written exit routine that is designed to receive a single value and transform (encode or decode) it in any way that the user specifies.

foreground component

A component that can run in foreground mode. Foreground components in CHANGE MANAGER include Front End, Specification, Analysis, Compare, JCL Generation, and Import.

Front End

A component of ALTER and CHANGE MANAGER that acts as the interface between the user and the other components. Front End is an interactive ISPF dialog that is responsible for creating and maintaining BMC objects and facilitating the generation of JCL.

full-recovery baseline

A baseline that captures data and the data structure definitions at a specific point in time.

generation data group (GDG)

See “generation data group (GDG)” on page 240.

generation data group (GDG)

A finite number of data sets that are kept in chronological order. Each data set is a generation data set.
Group ID

In an outbound migrate profile, a four-character identifier that links locations (or application instances) together. If a migrate profile defines one or more Group IDs, Analysis and Compare generate a single output file (worklist or CDL) for each group. If groups are not defined, Analysis and Compare generate one output file for each location. Locations within a group must reside on the same physical DB2 subsystem.

H

hash value

A number that appears at the end of commands in worklists or CDL files. The hash value is generated based on the contents of the command line and allows the products to determine whether the line has been manually modified since the file was generated.

historical database

A statistics database created by DASD MANAGER PLUS that contains the first, last, and current statistics collected by BMCSTATS for an object. You can display information from the historical database. BMCTRIG reads the historical database for statistics used during exception threshold evaluation.

HLQ

High-level qualifier of a data set.

I

image copy

An exact reproduction of all or part of the data in a table space. IBM provides utility programs to make full image copies (copy the entire table space) or incremental image copies (copy only the pages that have been modified since the last image copy). The NGT Copy utility can perform the same function. You can make an image copy of an index.

Import

A component of CHANGE MANAGER that converts statements that are stored in a CDL, DDL, or DML file into change requests in a work ID. In ALTER, Import converts statements that are stored in a DDL file.
import or importation

The process of obtaining an object or an object-set definition from a file or an external database and applying it to an alter-type work ID.

inbound migrate profile

A profile that can be used with the Import component to automate changes to object attributes. An inbound migrate profile can contain only change rules (no locations or scope).

incremental DDL

The DDL that changes the data structures that exist in the DB2 catalog, by using either an alter strategy or a drop-then-rebuild strategy.

Index Cardinality

A DASD MANAGER PLUS statistical graph that plots the FULLKEY and FIRSTKEY cardinality of an index over time.

Index Pages

ADASD MANAGER PLUS statistical graph that plots the relationship over time between active pages, allocated pages, and pages required if the index were reorganized.

Index Partition Cardinality

A DASD MANAGER PLUS statistical graph that plots the number of rows of an index partition over time. The graph plots the number of NEAROFFPOS and FAROFFPOS references as well as the number of keys in relation to the number of rows.

Index Partition Extents

A DASD MANAGER PLUS statistical graph that plots the number of extents in an index partition over time. The maximum extents per per data set allowed by VSAM is 7257.

Index Partition Leaf Distribution

A DASD MANAGER PLUS statistical graph that plots the leaf distribution in an index partition over time.
Index Partition Page Group

A DASD MANAGER PLUS statistical graph that plots page group information, specifically the
distribution of data in an index partition. (To display this graph, BMCSTATS must have been
run with a PAGEGROUP specification greater than zero.) With dual vertical axes, this graph
shows how rows and keys are distributed in the data set, and how leaf and full pages are
distributed based on the last BMCSTATS values.

Index Partition Pages

A DASD MANAGER PLUS statistical graph that plots the relationship between the number of
allocated pages and the number of active pages in an index partition over time.

installation options module

An assembler module that contains keywords whose global values determine the operating
environment for a BMC product.

internal table

A table that ALTER or CHANGE MANAGER use to store information.

ISPF skeletons

Data definition statement templates that the JCL Generation (JCLGEN) component uses. The
skeletons are described in HLQ.BMCSLIB($AJXDOC).

J

JCL

Job Control Language that is used to execute processes in batch mode.

JCL DSN

The name of the data set that contains job control language (JCL). The data set must exist and
can be partitioned or sequential. You must specify a member name for partitioned data sets.
You can use symbolic variables.
JCL Generation (JCLGEN)

A component of ALTER and CHANGE MANAGER that constructs a job control language (JCL) file for running the components in batch mode. When you choose to build JCL, JCLGEN is passed to the worklist that contains the control statements. ALTER and CHANGE MANAGER resolve all data set names that are entered with symbolic variables on the interface panels. JCLGEN resolves all data sets that are passed from the option panels and the unload data sets that are used by the Execution facility. The generated JCL includes data definition statements (ddnames) for all data sets that are needed by Execution, as well as the EXEC statement for the program and any necessary control parameters.

JCL variable display

The resulting output of a user option that includes debugging comments within any generated JCL. All AJX-prefixed variables are displayed as //* comments in the JCL to assist in diagnosing JCL Generation problems.

JCLGEN

See “JCL Generation (JCLGEN)” on page 244.

job

A batch unit of work that is defined by JCL, a work ID, and a worklist to perform tasks. ALTER and CHANGE MANAGER use the information that you supply for a work ID to generate the worklist, which provides the BMC JCL Generation component (JCLGEN) with the information that is necessary to build the JCL to run a job.

JOB table

A DASD MANAGER PLUS utility definition table that stores utility jobs generated and submitted through DASD MANAGER PLUS. This table is named BMCASUnn.UT_JOB.

justify

A graphic display option that specifies the alignment of the legend on statistical graphs: C (center), R (right), L (left), T (top), or B (bottom).
large object (LOB) column

A type of column that is used to store large objects (LOBs), such as images, audio, video, text, or graphics, as strings. The data type of the column is defined as LOB (such as a binary large object, or BLOB; character large object, or CLOB; or double-byte character large object, or DBCLOB) or as a distinct or user-defined type (UDT) that is based on a LOB data type. A LOB column resides in an auxiliary table.

large object (LOB) table space

See “auxiliary table space” on page 227.

Leaf distribution

An exception you can set using the DASD MANAGER PLUS BMCTRIG utility to monitor the leaf distribution in an index. An increase over time implies several leaf page splits and might indicate that the index should be reorganized.

Level

An exception you can set using BMCTRIG to indicate that the number of index levels has increased.

LEVELINC

The level increase; an exception you can set using BMCTRIG to monitor increases in the number of index levels.

LEVELMIN

The level minimum; an exception you can set using BMCTRIG to monitor the minimum number of levels required if the index were reorganized.

lines

ADASD MANAGER PLUS graphic display option that specifies whether to draw lines on statistical graphs: LINE (lines) or NOLI (no lines).
LINK library

A partitioned, cataloged data set that is used to store and retrieve all or part of a program in a form that is suitable for loading into main storage for execution. The LINK library contains executable modules that perform a product’s processes.

location

An arbitrary identifier that groups change rules for a particular destination. An outbound migrate profile can define one or more locations. A location is not a DB2 location identifier.

M

margin

ADASD MANAGER PLUS graphic display option that specifies the placement of the legend on statistical graphs: B (bottom) or T (top) for horizontal orientation; R (right) or L (left) for vertical orientation.

markers

ADASD MANAGER PLUS graphic display option that specifies whether to use place markers on statistical graphs: MARK (markers) or NOMA (no markers).

menu

A list of action options. You select an action by typing its corresponding number in the option input field and pressing Enter. A menu panel might contain other fields that you can use to qualify the action.

migrate

The process of moving DB2 data structures, data structures and data, or data structure changes, from an origin to a destination subsystem.

migrate profile

A profile that can be reused to select a set of objects, to customize changes to objects which are migrated to different locations or used in a comparison, or both.
migrate-type work ID

A work ID that is used to migrate data structures, data, or both. Worklists that are generated from a migrate-type work ID do not contain SQL DROP statements. These work IDs can have migrate options and change rules.

mixed list

A panel in the Specification component that can display multiple DB2 object types.

Most Frequent Value display

A BMCSTATS column statistics display that shows the ten most frequent values found in the column and their sequence.

N

name template

See “baseline name template” on page 227 and “work ID name template” on page 263.

name propagation

The process of extending to dependent objects the changes that you make in a referenced object. For example, if you change the name of a table that is referenced in a view, name propagation replicates the new name in the view definition.

NonUniform

An exception you can set using BMCTRIG to monitor the nonuniformity of an index based on the values that might appear in the SYSFIELDS catalog table. There are up to 10 distinct values.

null

A special value that indicates the absence of information.

NumIncremt

An exception you can set using BMCTRIG to monitor the number of incremental copies since the last full copy.
object

A DB2 entity that can be created or dropped. Objects are storage groups, databases, table spaces, tables, indexes, foreign keys, views, synonyms, aliases, or triggers. Unlike DB2, which treats foreign keys and check constraints as table attributes, ALTER and CHANGE MANAGER treat foreign keys and check constraints as independent objects.

object list

A list of one or more objects of different types.

OPNDB2ID

The DB2 authorization ID to update BMCSTATS tables. With the installation option of OPNDB2ID = Y, users with STATS authority can collect statistics even if their logon ID does not have RACF authority to read the data set.

option

A named value that is used to control one or more components. Global options are defined in the installation options module. The user can override the installation options by specifying the user options or by specifying keywords in the component’s ALUIN, AEXIN, or AJXIN input stream.

order

A DASD MANAGER PLUS graphic display option that specifies the order of the legend: KNOR (normal; left to right for horizontal and top to bottom for vertical) or KREV (reversed).

orient

A DASD MANAGER PLUS graphic display option that specifies the orientation of the legend on statistical graphs: H (horizontal) or V (vertical).

origin

The sending location or source of a migration or a change migration.
orphaned auxiliary index

An auxiliary index that is not associated with an existing auxiliary table.

orphaned auxiliary table space

An auxiliary table space that does not contain an auxiliary table.

outbound migrate profile

A profile that is used in performing a change migration. An outbound migrate profile might define one or more locations that contain change rules that are defined for any or all of those locations. Optionally, an outbound migrate profile can have a scope to select the DB2 objects on which to operate.

P

page

A unit of storage within a table space (4K or 32K) or index space (4K). A page in a table space contains one or more rows of a table. 8K and 16K pages can be used.

PageGroup

A feature of BMCSTATS that allows you to review the statistics on a specified grouping of pages to uncover additional information on hot spots in the data. When you generate control statements for the BMCSTATS utility, you determine the number of pages to group (0 through 99999). This facility and the graphic displays can help you locate areas of concentrated activity within a table space.

parent

A DB2 object that contains the primary key which might be referenced by one or more foreign keys in the child table.

See also “child” on page 231.

partitioned data set (PDS)

A data set in direct access storage that is divided into partitions, called members, each of which can contain a program, part of a program, or data. Synonymous with program library.
partitioned table space

A table space that is subdivided into parts (based upon index key range), each of which can be processed independently by utilities.

PartLvl

A BMCTRIG parameter that specifies whether to generate a utility job for each partition in exception or to generate a job at the table space or index level.

pattern

A rule that is applied to the naming of objects of a specified type. You can use wildcard characters (\% and *) when you define a pattern.

PctActivHi

An exception you can set using BMCTRIG to monitor the percentage of active pages. Use this parameter to specify the maximum percentage of active pages allowed.

PctActivLo

An exception you can set using BMCTRIG to monitor the percentage of active pages. Use this parameter to specify the minimum percentage of active pages allowed.

PDS

See “partitioned data set (PDS)” on page 249.

privilege

The capability of performing a specific function (authorization) on an object. Privileges might be explicitly or implicitly granted.

profile

A collection of scope rules, change rules, and locations that enables you to define and control a change, migrate, or baseline process.
See also “baseline profile” on page 227, “migrate profile” on page 246, “inbound migrate profile” on page 242, and “outbound migrate profile” on page 249.

**protected baseline**

A baseline that is designated as protected from deletion. A protected baseline cannot be deleted until the protected designation is removed.

See also “unprotected baseline” on page 261.

**Q**

**QMF**

Query Management Facility. DASD MANAGER PLUS provides QMF procedures for accessing the statistics tables to obtain useful information. DASD MANAGER PLUS provides sample QMF queries (in the CNTL library member ASURVIEW) that are used for the sample QMF reports.

**R**

**Recall**

A BMCSTATS parameter that enables you to recall archived data sets when collecting statistics.

**receive-type work ID**

A work ID that is used to create new data structures and load migrated data on a different subsystem.

**recovery**

The process of restoring a set of data structure definitions to their state at a particular point in time. Recovery involves comparing the DB2 catalog to a baseline, importing the CDL, generating a worklist with Analysis, and executing the worklist. If the baseline is a full-recovery baseline, you can recover data and the data structure definitions.

**reference location**
A location in a profile whose rules are used by another location. For example, a location called Houston might have that rules that are explicitly defined, while locations called Austin and Dallas might use Houston as a reference location.

**reference profile**

A profile whose scope is used by another profile. A baseline profile can reference a migrate profile, and a migrate profile can reference a baseline profile. Using a reference profile enables you to define the scope only once and thus eliminates errors that might arise from redesigning the scope.

**referenced object**

An object that a dependent object references. If you change the definition of a referenced object, dependent objects might not continue to function properly.

**referential constraint**

The requirement that nonnull values of a designated foreign key are valid only if they equal values of the primary key of a designated table. The relationship between the primary key in the parent table and a foreign key in a dependent table is used to establish referential integrity in a database. A referential constraint is always assigned a name to distinguish it from other constraints.

**referential integrity**

The condition that exists when all intended references from data in one column of a table to data in another column of the same or a different table are valid. Maintaining referential integrity requires enforcing referential constraints on all LOAD, RECOVER, INSERT, UPDATE, and DELETE operations.

**Report**

A parameter used by BMCSTATS to print a report on the statistics collected, by BMCTRIG to print a report on the initiated exceptions, by BMCCPRS to print a report on the statistics copied, and by BMCUPRS to print a report on the statistics updated.

**Restart option**

A DASD MANAGER PLUS option that causes JCL Generation to generate a RESTART keyword in the AEXIN(SYSIN) parameters of the JCL job stream. The restart option lets you
restart the job from the point at which the job failed. This option is available on the DASD MANAGER PLUS job generation panel.

**restart parameter**

An option that enables you to pass parameters to utilities being restarted and thereby ensure proper restart based on the utility and objects being processed. Restart parameters can be generated in RESTART JCL by entering the parameters on the DASD MANAGER PLUS job generation panel or by editing the worklist. The format is `RESTARTPARM RESTARTPARMstring` where `RESTARTPARMstring` is a list of `lineoneparms`.

**rows**

A DASD MANAGER PLUS printing option that specifies the number of rows per page.

**rows/key**

The number of rows per index key (1 through 999999). Might be fractional. For a unique index, this value should be 1. This is one of the parameters you can adjust when estimating space requirements for an object. With BMCTRIG, you can set the `Rows/Key` exception to specify the maximum number of rows per key to allow.

**S**

**Sample**

A BMCSTATS and RUNSTATS feature that you can use for random sampling of objects when collecting statistics. You can use sampling on all objects, on table spaces only, or on indexes only. Sampling is much faster on large objects. If the object has fewer than 1000 pages, sampling is not performed. Regular statistics will be collected even if sampling is requested. Sampling on table columns estimates values using probability. If you want more detail, do not use sampling.

**Save**

A BMCTRIG parameter that specifies whether to save exceptions in the DASD MANAGER PLUS exceptions table.

**save last used**
A DASD MANAGER PLUS user option that specifies whether to use the last values specified for the options as the default values.

**SAVESTATS**

A historical attribute indicating whether the statistics collected by the BMCSTATS utility were saved in the DASD MANAGER PLUS statistics database.

**SaveStats**

A BMCSTATS parameter that you can use to save the statistics collected in the DASD MANAGER PLUS statistics database.

**SBCS**

See “single-byte character set (SBCS)” on page 255.

**scope**

The final set of DB2 objects that are selected from the catalog on which the product operates. A scope includes objects that are explicitly selected and any of their object dependencies. For the Baseline component, the scope selects the objects that are captured in the baseline. For the Compare and Analysis components of CHANGE MANAGER, the scope selects the objects that are included in the comparison or analysis process. A scope consists of one or more scope rules.

**scope rule**

A specification for selecting DB2 objects from the catalog by object type and name.

**Script table**

In the CM/PILOT component of CHANGE MANAGER, ordered steps that prompt you for the information that is required to perform a change management task.

**security exit**

A user exit that enables the establishment of some form of system security. DASD MANAGER PLUS provides the following user exits to allow different forms of security:

- Front End Security Exit (limits access to actions)
Execution Manager Security Exit (provides installation security checking and option enforcement)

Execution Manager Unload Exit (provides testing and modification of each row of unloaded data)

**selection list**

A list of related items from which you can select one for further action. The actions (line commands) that you can specify in the Act field are typically displayed across the top of the panel.

**sequence number**

A six-digit, zero-filled number that identifies a statement in a worklist. The sequence number appears in columns 7 through 12 of the first line of each worklist command.

**service**

A user-specified command or utility that makes up an action. You can add, delete duplicate, and modify services to customize a particular job.

**Simple Space Estimation (SSE)**

A COMMAND line tool that allows you to estimate simple space for table spaces or index objects, giving you "what if" capability. Unlike DASD MANAGER PLUS statistics, you do not need to run BMCSTATS before using SSE. See also space estimation function.

**single-byte character set (SBCS)**

A character set in which each character is represented by a one-byte code.

**source**

In CHANGE MANAGER, the original object of a migration or the original object of a Compare process. When migrating objects or databases, the source database is the database from which you are migrating. The Compare process compares a source data structure with a target data structure. This process synchronizes two data structures, and the source is the data structure that needs to change.

*See also “Compare1” on page 233.*
Space Estimation

A feature of ALTER and CHANGE MANAGER that enables you to determine the amount of space that a table space or index will require, based on the object definitions and their estimated usages.

space estimation function

A DASD MANAGER PLUS function that you can use to estimate space requirements for existing table spaces and indexes. Estimates are based on statistics from the DASD MANAGER PLUS database. See also Simple Space Estimation (SSE).

SpaceOnly

A BMCSTATS parameter that you can use to collect only space information (from the VSAM catalog) when collecting statistics. This option is very fast if you need only size and extents information.

Specification

A component of ALTER and CHANGE MANAGER that enables you to create or edit data structure change or migration requests. Specification stores its change or migration requests in a work ID.

SQL

See “Structured Query Language (SQL)” on page 257.

SQLID

The authorization ID that is used as the implicit qualifier of table, view, synonym, and index names in dynamic SQL statements. The SQL ID, along with the other authorization IDs of a process, is used for authorization checking of dynamic SQL statements.

SSID

A DB2 subsystem identifier.

stack tapes

A DASD MANAGER PLUS user option that specifies whether to stack image copy tapes.
Start Over option

A RESTART option provided on the DASD MANAGER PLUS Action Generation panel that reprocesses a worklist that did not complete from the beginning.

STATAUTH

The statistics authorization indicator. If the indicator is set to Y (the default), DASD MANAGER PLUS checks users’ authorization to run BMCSTATS and requires the same authorization as for RUNSTATS.

STEP table

A DASD MANAGER PLUS utility definition table that stores services generated through DASD MANAGER PLUS. This table is named BMCASUnn.UT_STEP.

structure-only baseline

A baseline that contains only data structure definitions. No data from those data structures are included.

Structured Query Language (SQL)

An ANSI-standard language for database definition, manipulation, and query.

sub-element

A component of an object. For example, a column is a sub-element of a table, and a volume is a sub-element of a storage group.

symbolic variable

A user interface variable that has its value set interactively for the current user and session at the time of job control language (JCL) generation. The BMC JCL Generation component (JCLGEN) uses symbolic variables to perform ISPF file tailoring services. A symbolic variable should be preceded with an ampersand (&). In the installation options modules, a symbolic variable should be preceded with two ampersands (&&). Symbolic variables should not be confused with global job variables (AJX-type), which have their values set for all users and all sessions.

See also “JCL Generation (JCLGEN)” on page 244.
SYNC

A worklist command that invokes a checkpoint to use for restart processing and commits data to DB2.

sync point

A completion flag that is set during the execution of a worklist. The Execution program writes sync points to the SYNC table whenever it encounters -SYNC or -STOP commands in the worklist input stream. All SQL statements between sync points are executed as a single DB2 transaction. If a worklist is halted before completion for any reason, sync points enable you to begin processing the worklist from the last sync point.

SYNC table

A DASD MANAGER PLUS utility definition table containing sync point information from the execution of a worklist. This table is named BMCASUnn.UT_SYNC.

synchronization

The process of identifying structural differences between two copies of the same data structure and then making the data structures identical. For example, separate groups of developers might be independently modifying several copies of a set of data structures. At various times, the copies need to be synchronized to ensure that all of the developers are using the same structure definitions.

T

Table Pages

A DASD MANAGER PLUS statistical graph that plots the number of pages in a table over time.

Table Average Row Length

A DASD MANAGER PLUS statistical graph that plots the average row length in a table over time.

Table Cardinality
A DASD MANAGER PLUS statistical graph that plots the number of rows in a table over time. The graph also plots the number of indirect references in relation to the number of rows.

**Table Percent Pages**

A DASD MANAGER PLUS statistical graph that plots the percentage of pages in a table over time.

**Table Space Pages**

A DASD MANAGER PLUS statistical graph that plots the relationship over time of page statistics. The statistics plotted include allocated pages, active pages, and pages required if the table space were reorganized.

**Table Space Partition Cardinality**

A DASD MANAGER PLUS statistical graph that plots the number of rows in a table space over time. The graph also plots the number of NEARINDREF and FARINDREF in relation to the number of rows in the partition over time.

**Table Space Partition Extents**

A DASD MANAGER PLUS statistical graph that plots the number of extents in a table space over time. The maximum number of extents per data set allowed by VSAM is 7257.

**Table Space Partition Page Group**

A DASD MANAGER PLUS statistical graph that plots the distribution of data in the partition. The graph shows the number of rows, number of dirty pages, and number of full pages in relation to the number of pages in the data set.

**Table Space Partition Pages**

A DASD MANAGER PLUS statistical graph that plots the relationship over time of page statistics. The statistics plotted include allocated pages, active pages, pages required if the table space were reorganized, dirty pages, and full pages.

**Table Space Partition Percent Active/Drop**

A DASD MANAGER PLUS statistical graph that plots the percentage of active and dropped pages in the table space partition over time.
Tables

A BMCSTATS parameter that displays a panel from which to select tables and columns for collecting statistics.

target

The object of a migration or the object of a Compare process. When migrating objects or databases, the target database is the database to which you are migrating. The Compare process compares a source data structure with a target data structure. This process synchronizes two data structures, and the target is the data structure that contains the wanted changes.

See also “Compare2” on page 233.

task ID

A unit of work in the CM/PILOT component.

Tasks

A BMCSTATS parameter that specifies the level of multitasking to use for processing partitioned objects (1 through 16). If you use this option, specify a Buffers value that is at least this large.

template

A method of obtaining specifications for the definition of an auxiliary table space, table, or index. This definition can be replicated for each of the partitions in the base table space.

See also “baseline name template” on page 227 and “work ID name template” on page 263.

TOTALIND

The sum of FARIND and NEARIND; an exception you can set using BMCTRIG to monitor the percentage of rows of a table space that are not in their original page.

TOTALOFF

The sum of FAROFF and NEAROFF; an exception you can set using BMCTRIG to monitor the percentage of rows of an index that are not in optimal position.
TSO submit exit

A DASD MANAGER PLUS user option that specifies whether to use a TSO submit exit to generate job statements (default = N).

Type

A DASD MANAGER PLUS input or information field identifying the type of object: TS (table space), TT (table space set), IX (index), SG (storage group), or VL (volume).

U

unit

A specific device, device type, or group of devices that are used in data set allocation.

unload data set

An object that is used to store data while DB2 objects are dropped and rebuilt. The unload data set is also referred to as a SYSREC data set.

unprotected baseline

A baseline that is not designated as protected from deletion.

See also “protected baseline” on page 251.

UOW

Unit of Work. A unit of work consists of the worklist commands that are bounded by the -BEGU and -ENDU commands and that are run in a BMC Cross-System Image Manager (XIM) initiator.

UOWTRnnn

The ddname of the unit of work (UOW) output of the Execution component. When you use the worklist parallelism feature of the Database Administration or BMC Object Administration for DB2 solution, this output data set contains tracing records for each BMC Cross-System Image Manager (XIM) initiator.
user options

A set of options that are stored in the user’s profile and that are used by Front End and JCL Generation for running ALTER or CHANGE MANAGER components. The user options are initially set from the installation options module the first time that the user runs ALTER or CHANGE MANAGER.

V

variable

See “symbolic variable” on page 257.

VCAT allocation

A volume placement parameter that specifies the data set high-level qualifier that is appropriate for the DB2 subsystem. Use this field for nonstorage group allocation only.

versioning

The process of comparing baselines that represent the same set of structures at different points in time in order to change a version of a data structure.

W

wildcard

A symbol that you can use to represent a value in SQL statements, filters, and name patterns. Valid wildcards for SQL statements and filters include the following symbols:

- The % and * represent any character string.
- The _ and ? represent a single character.

work ID

A unit of work with a two-part name (owner.name) that contains change or migration requests in change definition (CD) tables. The change or migration requests can be either imported or created manually through the Specification component.
work ID name

A work ID name is a string of up to eighteen alphanumeric characters excluding percent (%), asterisk (*), underscore (_), and space. When you specify a work ID name, you can use a wildcard pattern to display a group of similar names.

work ID name template

A template from which the name of a work ID is created. You can use this template to create or replace a work ID in batch mode. The name template contains the special character sequences #### and @@@@@@ or @@@@@@@@@, in addition to text characters. An ascending sequence of numbers replaces the #### sequence, and the current date replaces the @@@@@@ or @@@@@@@@@ sequence when the work ID is generated. You can create and replace receive-type work IDs (using name templates) with the Import and Execution components.

work ID owner

The Authorization ID of the creator of the work ID.

WORKIDS table

A DASD MANAGER PLUS object definition table that stores actions and task IDs. This table is named BMCASUnn.DO_WORKIDS.

worklist

A data set that contains commands for implementing a data structure change or migration.

worklist parallelism

A feature in the Database Administration and BMC Object Administration for DB2 solutions that reduces the elapsed time for executing a worklist that the CHANGE MANAGER product generates.

worklist execution log

See “AEXPRINT” on page 222.
X

XIM

See “Cross-System Image Manager (XIM)” on page 233.

XIM initiator

A program that executes one or more units of work (UOW).
Index

_ (underscore)
  as a literal 137
  as a wildcard 137
? (question mark) as a wildcard 137
* (asterisk) as a wildcard 137
% (percent sign) as a wildcard 137
$AJXDOC data set 150

A
above-the-bar storage 149
access
  controlling 141
  to database 157
  to parallelism feature, restricting 148
ACM
  definition 221
ACM_PARALLEL_MAXINIT AJXPOFIN keyword 204
ACM_PARALLEL_MININIT AJXPOFIN keyword 204
ACM_PARALLEL_WORKLST AJXPOFIN keyword 204
ACM_PARALLEL_XIMGRP AJXPOFIN keyword 204
ACM_PARALLEL_XIMPROC AJXPOFIN keyword 204
ACM_PARALLEL_XIMSTRT AJXPOFIN keyword 204
ACM_PARALLEL_XIMTRCE AJXPOFIN keyword 204
ACMvryA Analysis plan 141
ACMvryB Baseline plan 142
ACMvryC Compare plan 142
ACMvryE Environment plan 141
ACMvryF Front End plan 141
ACMvryI Import plan 141
ACMvryR Baseline Report plan 142
ACMvryS Specification plan 141
ACT
  definition 221
action
  definition 221
  action code
    definition 221
  action name
    definition 221
  action owner
    definition 221
  action POF
    definition 222
  action status
    definition 222
  action type
    definition 222
Administrative Assistant for DB2 46
Administrative Products for DB2
  definition 222
AEXAUNLD package 143
AEXEFAUH bit 147
AEXESTDL package 143
AEXIN
  definition 222
AEXIN input stream
  user options 131
AEXPIN
  definition 222
AEXPIN input stream
  using in a parallel worklist 204
AEXPRnnn
  definition 223
AEXPRINT
  definition 222
AEXPRnnn
  definition 223
AEXPTRAC
  definition 223
AEXSQLIO package 143
AEXSynnn
  definition 223
AEXUSERX
  DSECT 144
AEXvrHA Execution Monitor Entry plan 143
AEXvrHM Execution Monitor plan 143
AJX variables
  definition 223
AJXIN
  definition 223
AJXIN input stream 131
AJXPODAT edit macro 211
AJXPOFER message file 214
AJXPOFIN
  definition 223
AJXPOFVL
  definition 224
AJXPOFVL message file 214
AJXPRINT
  definition 224
aliases, INSERT statement 157
alloc unit
  definition 224
allocating data sets 136
ALTDD
  definition 224
ALTER
  architecture 25
  components 27
  Execution plans 143
  objects 30
  overview 17
  plans, list of 141
ALTER for DB2
  definition 224
ALTER objects
  DDL file 32
  internal table 32
  unload data set 32
  work ID 30
  worklist 31
alter-type work ID 30
  definition 224
ALU
  definition 224
ALUDSYNC member 156
ALUEUSX1 member 144
ALUIN
  definition 224
ALUPRINT
  definition 225
Analysis
definition 225
Analysis component
  default data set name 163
  of ALTER 28
  of CHANGE MANAGER 35
  setting default processing options 163
Application
  definition 225
Application ID 132
applications, user options 170
ARU
  definition 225
asterisk (*) as a wildcard 137
ASU
  definition 225
ASXEFLAG 147
attribute
  definition 225
AUTHID
  definition 225
authorization
  setting 129
authorization ID
  definition 225
authorization switching
  controlling 144
automation modes
  definition 226
automation spawner
  definition 226
AutoSub
  definition 226
auxiliary index
  definition 226
auxiliary list
  definition 226
auxiliary table
  definition 226
auxiliary table space
  definition 227

B

base table
  definition 227
base table space
  definition 227
baseline
baseline component
definition 227
deleting 156
default data set name 168
setting default processing options 168
Baseline component
doing CHANGE MANAGER 35
definition 227
baseline name template
definition 227
baseline profile
definition 227
Baseline Report component
default data set name 169
setting default processing options 169
baselines
default run type 168
object 39
profile 40
table maintenance 156
table sizing and growth 156
worklist 39
batch component
definition 227
BL_ALIAS product table 152
BL_AUXTABLE product table 152
BL_CHECK product table 152
BL_CHECK2 product table 152
BL_COLAUTH product table 152
BL_COLUMN product table 152
BL_CITY product table 152
BL_DATABASE product table 152
BL_DBAUTH product table 152
BL_INDEX product table 152
BL_KEY product table 152
BL_KEYTARGETS product table 152
BL_PACKAGE product table 152
BL_PARM product table 152
BL_REL product table 152
BL_RELKEY product table 152
BL_RESAUTH product table 152
BL_ROUTINES product table 152
BL_ROUTINETEXT product table 152
BL_RTIAUTH product table 152
BL_SEQUENCES product table 152
BL_SQAUTH product table 152
BL_STOGROUP product table 152
BL_SYNONYM product table 152
BL_TABAUTH product table 152
BL_TABLE product table 152
BL_TABLEPART product table 152
BL_TABLESPACE product table 152
BL_TRIG_COL product table 152
BL_TRIG_TEXT product table 152
BL_TRIG_TEXT_AUX product table 152
BL_TRIGGER product table 152
BL_UNQCNST product table 152
BL_UNQCNST_COL product table 152
BL_SERVICE product table 152
BL_VIEW product table 152
BL_VIEW_TEXT product table 152
BL_VIEW_TEXT_AUX product table 152
BL_VOLS product table 152
BLRP data set 186
BMC object
definition 228
BMC Object Administration for DB2 47
BMC Software solutions
  Administrative Assistant for DB2 46
  BMC Object Administration for DB2 47
  Database Administration for DB2 46
BMCCAMD2 CLIST 52
BMCALTER command 52
BMCCAT command 52
BMCCCHG command 52
BMCCCOPY
definition 228
BMCCPRS
definition 228
BMCDASD command 52
BMCDB2 CLIST 132
BMCCLOAD
definition 228
BMCREORG
definition 228
BMCCSTATS
definition 228
BMCTRIG
definition 228
BMCU UNLOAD
definition 229
BMCM UPSRS
definition 229
box
definition 229
browse function
definition 229
buffer pool
definition 229
Buffers
definition 229
build JCL
definition 229
build worklist
definition 229

C
CAN
definition 230
CANCEL
definition 230
CANCEL ISPF command 53
carriage return, hexadecimal format 140
CASE tools 24
catalog baseline
definition 230
catalog baseline profiles, user option for CM/PILOT 170
catalog indirection
definition 230
CATALOG MANAGER for DB2
definition 230
CD_ALIAS product table 152
CD_AUXTABLE product table 152
CD_CHECK product table 152
CD_CHECK2 product table 152
CD_COLUMN product table 152
CD_DATABASE product table 152
CD_INDEX product table 152
CD_KEY product table 152
CD_KEYTARGETS product table 152
CD_PACKAGE product table 152
CD_PARMS product table 152
CD_REL product table 152
CD_RELKEY product table 152
CD_ROUTINES product table 152
CD_ROUTINES2 product table 152
CD_ROUTINES2NEWTEXT product table 152
CD_ROUTINES2TEXT product table 152
CD_SEQUENCES product table 152
CD_STOGROUP product table 152
CD_SYNONYM product table 152
CD_TABLE product table 152
CD_TABLEPART product table 152
CD_TABLESPACE product table 157
CD_TRIG_COL product table 152
CD_TRIG_TEXT product table 152
CD_TRIG_TEXT_AUX product table 152
CD_TRIGGER product table 152
CD_UNQCNST product table 152
CD_UNQCNST_COL product table 152
CD_VIEW product table 152
CD_VIEW_TEXT product table 152
CD_VIEW_TEXT_AUX product table 152
CD_VOLS product table 152
CDL
definition 230
CDL (Change Definition Language)
data set name, specifying 167
data set, allocating 136
Change Definition Language CDL
definition 230
CHANGE MANAGER
architecture 32
components 35
database access 157
Execution plans 143
objects 38
overview 38
plans, list of 141
CHANGE MANAGER for DB2
definition 230
CHANGE MANAGER objects
application 45
baseline 39
CDL file 44
CM/PILOT worklist 45
DDL file 44
DML 45
internal table 44
profile 40
script 45
task ID 45
unload data set 32
work ID 30
worklist 31
change migration process 22
change rule
definition 231
change rules, used in outbound migrate profiles 43
changes
managing with ALTER 17
changes to the product 11
char H/W
definition 231
char size
definition 231
characters
delimited identifiers 139
nonprintable 140
check constraints, INSERT statement 157
checkpoint  
  definition 231
child  
  definition 231
cleanup, data set 177
CLIST  
  BMCADMF2 52
  BMCDB2 132
  definition 231
CM_APPLICATION product table 152
CM_BASELINE product table 152
CM_BLPROFILE product table 152
CM_HISTORY product table 152
CM_LOCATION product table 152
CM_MIGPROFILE product table 152
CM_RENAME product table 152
CM_RULES product table 152
CM_SCOPE product table 152
CM_SECURITY product table 152
CM_SYNC product table 152
CM_TASKID product table 152
CM_UNLOADDSN product table 152
CM_WORKID product table 152
CM_XML product table 152
CM_XML_AUX product table 152
CM/PILOT component  
  of CHANGE MANAGER 37
CM/PILOT user options  
  catalog baseline profile 170
  CDL 170
  changing 170
  data sets 170
  DDL baseline profile 170
  processing a worklist 170
  worklist 170
CM/PILOT worklist 45
CMPILOT for DB2  
  definition 232
CMPILOT worklist  
  definition 232
color  
  definition 232
columns  
  definition 232
  columns, INSERT statement 157
command  
  definition 232
commands  
  DOPTS 133, 134
  ENVI 51
  ENVIRONMENT 51
  Fast Path Navigation 52
  ISPF 53
  MAINT 51
commit  
  definition 232
Compare  
  definition 232
Compare component  
  default data set name 167
  of CHANGE MANAGER 36
Compare1  
  definition 233
Compare2  
  definition 233
comparisons  
  primary input 36
  secondary input 36
  setting default processing options 167
component  
  definition 233
components  
  Analysis 28
  Baseline 35
  CM/PILOT 37
  Compare 36
  Execution 29
  Front End 27
  Import 28, 37
  JCL Generation 30
  Specification 27
constraint  
  definition 233
control characters, using in object names 140
control point 35
conventions, documentation 10
copies  
  definition 233
CopyPend  
  definition 233
corrective action  
  definition 233
CP_DML_STMT product table 152
CP_SCRIPT product table 152
CP_STEP product table 152
creating  
  additional installation options module 134
  objects 55
Cross-System Image Manager. See XIM
Cross-System Image Manager XIM
definition 233

D
DASD MANAGER PLUS
Execution plans 143
DASD MANAGER PLUS database
definition 234
DASD MANAGER PLUS for DB2
definition 234
Data Control Language DCL
definition 234
data conversions, using in change process 20
data definition language DDL
definition 234
data definition name ddname
definition 234
Data Manipulation Language DML
definition 234
data modeling tools, using 24
data set
map 177
data set sizing
definition 234
data sets
$AJXDOC 150
CDL 136
cleanup 177
DATAWORK 184
deleting automatically 177
diagnostic output 136
discard 177
error 177
JCL 136
LOGSORT 184
permanent work 186
prefixes 186
punch 177
report 136
resolving names 186
sample VSAM object 177
setting default processing options 186
setting installation options 162
sizing 151, 177
SORTOnnn 177
SORTOUT 177, 186
SORTPnnn 177
SORTWORK 177, 184
SUThnnn 177
SYSCOPY 186
SYSDISC 186
SYSERR 186
SYSMAP 186
SYSPUNCH 186
SYSREC 186
SYSUT 186
SYSUTnmm 177
unload 177
user options, task ID 170
work 177
worklist 136
WRKnnn 177
data structure
definition 235
data structures, importing 20
Database Administration for DB2 46
database administrator DBA
definition 235
database request module DBRM
definition 235
databases, INSERT statement 157
DATASETSIZING B keyword 178
DATASETSIZING C keyword 178
DATASETSIZING O keyword 179
DATAWORK data set 184
DB2 catalog
definition 235
DB2 command
definition 235
DB2 objects, security 157
DB2 SSID, specifying 162
DB2 utilities
  used by ALTER 17
  used by CHANGE MANAGER 17

DBA
definition 235

DCS
definition 235

DD statement
definition 235

DDL
  baseline 39
  baseline profile 40
definition 236
DDL baseline
definition 236
ddname
definition 236
debugging, setting default processing options 198
default options. See installation options
default options module
definition 236
default output class 208
default value
definition 236
defaults, setting product options
data sets 186
defaulting 198
generation data groups 196
JCL jobcard 173
JCL static data sets 177
JCL STEPLIBs 175
product options file 210
SORTWORK data sets 184
tapes 182
utility module names 201
defaults, setting user options
Analysis options 208
Analysis processing 163
baseline processing 168
Baseline Report processing 169
Compare processing 167
data set allocation 162
data sets 186
debugging 196
Execution processing 166
general 129
generation data groups 196
Import processing 165
online reorgs 202
screen display 160
SHRLEVEL CHANGE 202
user variables 209
worklist parallelism 204
worklist sorting 207
DeleteAge
definition 236
deleting
  baselines 156
data sets automatically 177
task IDs 156
work IDs 156
delimited identifier
definition 236
delimited identifiers
  valid and invalid wildcard use 137
  wildcard support 139
dependencies
definition 236
dependent
definition 236
dependent object
definition 237
destination
definition 237
device name
definition 237
device type
definition 237
diagnostic output
data sets 136
default data set name for Baseline 168
default data set name, for Analysis 163
default data set name, for Baseline Report 169
default data set name, for Compare 167
default data set name, for Execution 166
default data set name, for Import 165
Front End component
    description 27
genrating input streams 133
full-recovery baseline
    definition 240

G
GDG
    definition 240
GDG (generation data group)
    setting default processing options 196
GDG symbolic variable 186
GDG(generation data group)
    specifying 186
    symbolic variable 186
    using for data sets 186
general resource profile 149
generation data group. See GDG
generation data group GDG
    definition 240
GLID(global authorization ID), default 163
Group ID
    definition 241

H
hash value
    definition 241
HELP ISPF command 53
historical database
    definition 241
HLQ
    definition 241

I
IEFUSI exit 149
image copy
    definition 241
import
    definition 242
Import
    definition 241
Import component
    default data set name 165
    of CHANGE MANAGER 28, 37
importing
    default run type 165
    setting default processing options 165
inbound migrate profile
    definition 242
incremental DDL
    definition 242
Index Cardinality
    definition 242
Index Pages
    definition 242
Index Partition Cardinality
    definition 242
Index Partition Extents
    definition 242
Index Partition Leaf Distribution
    definition 242
Index Partition Page Group
    definition 243
Index Partition Pages
    definition 243
indexes, INSERT statement 157
initial POF, refreshing 214
input streams 131
installation options
    creating 134
    user options 133
installation options module
    definition 243
internal table
    definition 243
ISPF
    AJXPODAT macro 211
    commands 53
    file tailoring, used by JCL Generation 150
    help 53
ISPF interface
    Main Menu 50
    using 49
ISPF profile
    user options 133
ISPF skeletons
    definition 243

J
JCL
    data set, allocating 136
    definition 243
JCL data set name
  Analysis 163
  Baseline Report 169
  Compare 167
  Execution 166
  Import 165
JCL DSN
definition 243
JCL Generation
data set sizing 151
  product options 132
  RUNSTATS utility 151
  VSAM object sampling 151
JCL Generation component 30
JCL Generation JCLGEN
definition 244
JCL job card, setting default processing options 173
JCL job statement, defaults 173
JCL static data sets, setting default processing options 177
JCL STEPLIBs, setting default processing options 175
JCL variable display
definition 244
JCLGEN
definition 244
job
definition 244
JOB table
definition 244
justify
definition 244

L
large object LOB column
definition 245
large object LOB table space
definition 245
Leaf distribution
definition 245
LEFT ISPF command 53
Level
definition 245
LEVELINC
definition 245
LEVELMIN
definition 245
lines
definition 245
LINK library
definition 246
literal strings 140
LOB SYSREC data set 186
location
definition 246
locations, using 43
locking options 52
LOGSORT data set 184

M
macros
  AJXPODAT 211
MAINT command 51
maintaining, sync tables 156
maintenance
  MAINT command 51
managing change
  using ALTER 17
  using CHANGE MANAGER 21
map data set 177
mapping database 202
mapping table 202
margin
definition 246
markers
definition 246
MAXINITS AEXPIN keyword 204
MEMLIMIT system parameter 149
menu
definition 246
message files 214
message identifier 54
migrate
definition 246
migrate profile
definition 246
migrate-type work ID
definition 247
migration
  inbound migrate profile 170
  objects 90
MININITS AEXPIN keyword 204
mixed list
definition 247
modifying, user options 170
Most Frequent Value display
definition 247
msg_ID 54
multitasking, using tape stacking 182

N
name propagation
definition 247
name template
definition 247
naming objects 136
native SQL stored procedure, INSERT statement 157
negative acknowledgegment, hexadecimal format 140
NEXT ISPF command 53
NGT utility options 204
NGT_UTILDB POF keyword 204
nonprintable characters 140
NonUniform
definition 247
nonviewable characters 140
null
definition 247
null character, hexadecimal format 140
null characters 140
NumIncremt
definition 247

O
object
definition 248
object list
definition 248
objects
ALTER 30
creating 55
migrating 90
naming 136
security 136
online reorgs 202
OPNDB21D
definition 248
option
definition 248
options

Analysis 163
baseline 168
Baseline Report 169
changing 129
CM/PILOT 170
Compare 167
data set allocation 162
data sets 186
debugging and display 198
Execution 166
generation data group 196
Import 165
installation 130
JCL job cards 173
JCL static data sets 177
JCL STEPLIBs 175
online reorg 202
panel display 160
product options file 210
refreshing 131
setting 129, 160
SHRLEVEL CHANGE 202
SORTWORK data sets 184
tapes 182
used by products 135
user 131
user variables 209
utility module names 201
worklist parallelism 204

order
definition 248
orient
definition 248
origin
definition 248
orphaned auxiliary index
definition 249
orphaned auxiliary table space
definition 249
outbound migrate profile
definition 249
outbound migrate profiles, used by CHANGE MANAGER 41
override options, using keywords 133

P
packages, Execution component 143
PageGroup
definition 249
panel display 160
PARALLEL AEXPIN keyword 204
parameters
  MEMLIMIT 149
  system 149
parent
definition 249
partitioned data set PDS
definition 249
partitioned table space
definition 250
PartLvl
definition 250
pattern
definition 250
PCPY1_SUPPRESS_SUFF POF keyword 186
PCPY2_SUPPRESS_SUFF POF keyword 186
PctActivHi
definition 250
PctActivLo
definition 250
PDS
definition 250
percent character, hexadecimal format 140
percent sign (%) as a wildcard 137
permanent work data sets 186
PFSHOW ISPF command 53
plans
  ALTER 141
  authorizations 141
  CHANGE MANAGER 141, 142
    Execution component 143
    names 141
    security 141
    synonyms 51
POF
  AJXPODAT macro 211
  creating a user POF 210
  setting default processing options 210
  updating a user POF 211
POF (product options file)
  generating reports 214
  initial 132
  initializing 132
  overriding values in SLIBs 216
  POF Validation Report 214
  populating 132
  renewing the initial POF 214
  reusing in a subsequent installation 215
  user 132
  using multiple POFS 213
  Variables Initialized with Default report 214
POFDATE POF keyword 132
POFDS installation option 132
prefixes 186
PREV ISPF command 53
privilege
definition 250
product changes 11
product database 157
product options file. See POF
product tables
  list of 152
  maintaining 152
  sizing 156
profile
definition 250
profiles
  baseline 40
  catalog baseline 40
  DDL baseline 40
  inbound migrate 41
  migrate 41
  outbound migrate 41
  user options, application 170
  propagating, changes 16
protected baseline
definition 251
publications, related 9
punch data set 177
Q
QMF
definition 251
question mark (?) as a wildcard 137
R
RACF (IBM Resource Access Control Facility) 149
RCPY1_SUPPRESS_SUFF POF keyword 186
RCPY2_SUPPRESS_SUFF POF keyword 186
Recall  definition 251
receive-type work ID  definition 251
recovery  definition 251
reference location  definition 251
reference profile  definition 252
referenced object  definition 252
referential constraint  definition 252
referential integrity  definition 252
refreshing user options 131
related publications 9
rename table
  security 157
  updating 157
Report  definition 252
report data set 136
reports
  setting installation options 169
reports, generating POF 214
Restart option  definition 252
restart parameter  definition 253
restricting access
  specifying a resource profile 149
  to worklist parallelism 148
RIGHT ISPF command 54
ROWID SYSREC data set 186
rows  definition 253
rows/key  definition 253
RUNSTATS utility 151

S
Sample  definition 253
Save  definition 253
save last used  definition 253
SaveStats  definition 254
SAVESTATS  definition 254
SBCS  definition 254
scope  definition 254
scope rule  definition 254
scope rules, CHANGE MANAGER object 42
scope, used by Compare 42
Script table  definition 254
script, CM/PILOT 37
security
  BMC object 136
  component 141
  DB2 objects 157
  plan 141
  rename table 157
security authorization 144
security exit
  definition 254
  Execution exit 144
  testing the execution 147
  used by Execution 144
security, setting authorizations 129
selection list  definition 255
sequence number  definition 255
sequences, INSERT statement 157
service  definition 255
SHRLEVEL CHANGE 202
Simple Space Estimation (SSE)  definition 255
Simple Space Estimation (SSE) feature 54
single-byte character set SBCS  definition 255
skeleton library, overriding POF values 216
SMFPRM member 149
SMS (Storage Management Subsystem) 139
SMS(Storage Management Subsystem) 186
SORTOnnn data set 177
SORTOUT data set 177, 186
SORTPnnn data set 177
SORTWORK data set 184

source
definition 255

Space Estimation
ALTER 21
CHANGE MANAGER 27
definition 256

space estimation function
definition 256

SpaceOnly
definition 256

Specification
definition 256

Specification, component of ALTER and CHANGE MANAGER 27

SPLIT ISPF command 54

SQL
definition 256

SQLID
definition 256

SSE ISPF command 54

SSID
definition 256

stack tapes
definition 256

Start Over option
definition 257

STATAUTH
definition 257

STEP table
definition 257

STEPLIB libraries, setting default values 173

storage group, INSERT statement 157

Storage Management Subsystem. See SMS storage, virtual 149

structure-only baseline
definition 257

Structured Query Language SQL
definition 257

sub-element
definition 257

summary of changes 11

SUTnnndata set 177

SWAP ISPF command 54

Switching authorization 144

symbolic variable
definition 257

symbolic variables
GDG 186

SYNC
definition 258

csync point
definition 258

csync table 156

SYNC table
definition 258

synchronization
definition 258

synonyms

ENVI command 51

INSERT statement 157

syntax statement conventions 10

SYSCOPY data set

setting JCL options 186

SYSDISC data set 186

SYSERR data set 186

SYSMAP data set 186

SYSPUNCH data set 186

SYSREC data sets

LOB 186

ROWID 186

setting JCL options 186

SYSUT data set 186

SYSUTnnn data set 177

T

Table Average Row Length
definition 258

Table Cardinality
definition 258

Table Pages
definition 258

Table Percent Pages
definition 259

Table Space Pages
definition 259

Table Space Partition Cardinality
definition 259

Table Space Partition Extents
definition 259

Table Space Partition Page Group
definition 259

Table Space Partition Pages
definition 259

Table Space Partition Percent Active/Drop
definition 259
table spaces, INSERT statement 157
tables
   baseline 156
   column, INSERT statement 157
   INSERT statement 157
   maintenance 152
   security 157
   sizing 156
Tables
   definition 260
tapes
   setting default processing options 182
   stacking, disabling 182
   using 186
target
   definition 260
task ID
   definition 260
task IDs
   CHANGE MANAGER object 45
   deleting 156
   user options 170
Task List 55, 90
Tasks
   definition 260
template
   definition 260
temporary work data sets 184
threshold, for alternate unit 186
TOTALIND
   definition 260
TOTALOFF
   definition 260
TRACE AEXPIN keyword 204
triggers, INSERT statement 157
TSO region 49
TSO region size 129
TSO submit exit
   definition 261
Type
   definition 261

U
underscore (_)
   as a literal 137
   as a wildcard 137
unique constraints, INSERT statement 157
unit
   definition 261
unit name, ROWID SYSREC data set 186
UNLD1_SUPPRESS_SUFF POF keyword 186
UNLD2_SUPPRESS_SUFF POF keyword 186
unload data set
   definition 261
unload data set, used by utilities 177
unprotected baseline
   definition 261
UOW
   definition 261
UOWTRnnn
   definition 261
UP ISPF command 54
user exits
   Execution security 144
user options
   CDL 170
   changing CM/PILOT 170
   definition 262
   general 129
   JCL Generation 132
   override options 133
user POF
   creating 210
   updating directly 211
   updating in options panels 211
   using multiple 213
user variables 209
utility module names, setting default processing options 201

V
values
   hexadecimal 168
   nonprintable characters, using 140
variable
   definition 262
VCAT allocation
   definition 262
versioning
   definition 262
view column, INSERT statement 157
viewing, INSERT statement 157
virtual storage 149
VSAM object
data set sizing 177
JCL generation 151
sampling 151

W
wildcard
   definition 262
wildcard characters
   delimited identifier support 139
   used with change rules 137
   used with scope rules 137
   used with Specification filters 137
   valid and invalid wildcards 137
work data sets
   permanent 186
   temporary 184
   used by utilities 177
work ID
   definition 262
work ID name
   definition 263
work ID name template
   definition 263
work ID owner
   definition 263
work IDs
   deleting 156
   name template 170
WORKIDS table
   definition 263
worklist
   definition 263
worklist execution log
   definition 263
worklist parallelism
   definition 263
   overriding defaults 204
   setting default processing options 204
worklists
   data set 136
   DDL baseline 39
   overview 31
   setting default data set name 163
   used by Baseline 39
   WRKnnn data set 177

X
XIM
   definition 264
   XIM (Cross System Image Manager) 148
   XIMGROUP AEXPIN keyword 204
   XIMPRC AEXPIN keyword 204
   XIMSTART AEXPIN keyword 204

Z
zaps
   applying to restrict access 148
   using the MAINT command 51
ZOOM command 160
ZOOM ISPF command 54