LOADPLUS® for DB2 Reference Manual

Component of Database Administration for DB2

Version 7.1

October 24, 2003

Copyright 2003 BMC Software, Inc. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc. IBM and DB2 are registered trademarks of International Business Machines Corp. All other trademarks belong to their respective companies.

BMC Software considers information included in this documentation to be proprietary and confidential. Your use of this information is subject to the terms and conditions of the applicable End User License Agreement for the product and the proprietary and restricted rights notices included in this 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 UNPUBLISHED -- RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES.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., 2101 CityWest Blvd., Houston, TX 77042-2827, USA. Any contract notices should be sent to this address.

Contacting BMC Software

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

United States and Canada

Address BMC Software, Inc.
2101 CityWest Blvd.
Houston TX 77042-2827

Telephone 713 918 8800 or
800 841 2031

Fax 713 918 8000

Outside United States and Canada

Telephone (01) 713 918 8800

Fax (01) 713 918 8000

Customer Support

You can obtain technical support by using the Support page on the BMC Software Web site or by contacting Customer Support by telephone or e-mail. To expedite your inquiry, please see “Before Contacting BMC Software.”

Support Web Site

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

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software 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 800 537 1813. Outside the United States and Canada, please contact your local support center for assistance. To find telephone and e-mail contact information for the BMC Software support center that services your location, refer to the Contact Customer Support section of the Support page on the BMC Software Web site at **www.bmc.com/support_home**.

Before Contacting BMC Software

Before you contact BMC Software, have the following information available so that Customer Support can begin working on your problem 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, such as `file system full`
  - — messages from related software

Contents

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Figures

BMC Software, Inc., Confidential and Proprietary Information

Tables

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

About This Book

This book provides information that DB2 users need when using the LOADPLUS for DB2 product from BMC Software. This book is intended for DB2 system administrators, DB2 database administrators, and DB2 application programmers.

To use this book, you should be familiar with the following items:

- IBM DB2 Universal Database for OS/390 (DB2)
- Multiple Virtual Storage (MVS), OS/390, or z/OS operating systems
- job control language (JCL) and the Interactive System Productivity Facility (ISPF)

BMC Software, Inc., Confidential and Proprietary Information

How This Book Is Organized

This book is organized as follows. In addition, this book contains a summary of changes and an index.

Chapter/Appendix	Description
Chapter 1, “Introduction to LOADPLUS”	• discusses the need for loading DB2 tables • lists the functional differences offered by LOADPLUS over the functions of the IBM DB2 LOAD utility • describes and illustrates the LOADPLUS processing phases and the data sets that they use
Chapter 2, “Operational Considerations”	• lists the authorizations needed to execute the programs • discusses considerations for particular types of LOADPLUS operations such as dynamic work file allocation, SHRLEVEL access, CSV input, and identity columns • discusses serialization and concurrency, including the status of table and index spaces during the processing phases • discusses how to ensure recoverability of the loaded table space
Chapter 3, “Syntax of the LOAD Command”	• presents syntax rules • illustrates the command syntax • describes each of the command options • describes data type conversions
Chapter 4, “How to Build and Execute LOADPLUS Jobs”	• defines the utility execution parameters • describes how to specify JCL, the job statements, and DD statements • describes procedures for invoking, restarting, and terminating LOADPLUS, including specific instructions for recovery
Chapter 5, “Examples of LOADPLUS Jobs”	provides some example executions of the LOADPLUS utility, including typical job streams and SYSPRINT output
Chapter 6, “Performance Considerations”	explains how to get the most out of LOADPLUS, and provides suggestions for improving performance
Appendix A, “LOADPLUS Installation Options”	describes each installation option and lists the options modules that are shipped with the product
Appendix B, “LOADPLUS Database”	provides a definition of the tables that are shared by several BMC Software products
Appendix C, “LOADPLUS Messages and Codes”	contains the return codes and error messages that LOADPLUS might issue along with recommended responses
Appendix D, “Calculating LOADPLUS Work Data Set Sizes”	• provides formulas for calculating the size of the various work data sets used by LOADPLUS • illustrates how to use the formulas by providing examples
Appendix E, “RULES Installation Option Examples”	provides examples of how to use the RULES installation option
Appendix F, “LOADPLUS User Exits”	contains examples of user exits in various programming languages

BMC Software, Inc., Confidential and Proprietary Information

Related Documentation

BMC Software products are supported by several types of documentation:

- online and printed books
- online Help
- release notes and other notices

Note: LOADPLUS also provides online message information. For details about how to view the messages that LOADPLUS generates, see "Online Documentation for Messages" on page C-2.

In addition to this book, you can find useful information in the publications listed in the following table. As "Online and Printed Books" on page xxii explains, these publications are available on request from BMC Software.

Category	Document	Description
installation	*OS/390 and z/OS Installer Guide*	provides information about using the OS/390 and z/OS Installer to install LOADPLUS and other BMC Software mainframe products
	Utility Products for DB2 Customization Guide	provides information about customizing LOADPLUS and other BMC Software utilities for DB2
	online help panels for LOADPLUS options on Installer panels	provides help for fields that are required when installing the LOADPLUS product
product-specific documents	release notes, flashes, technical bulletins	explain the latest updates to LOADPLUS
related documents	*Utility Products for DB2 Reference Summary*	provides a quick reference for commands and syntax for the BMC Software CHECK PLUS for DB2, LOADPLUS, REORG PLUS for DB2, and UNLOAD PLUS® for DB2 products
	Apply Plus Reference Manual	provides detailed information about the Apply Plus component of the Log Master for DB2 product
	BMCDSN Command Processor Reference Manual	provides information about the BMC Software utility command processor, BMCDSN
	BMCSORT Reference Manual	provides information about the BMC Software sort engine, BMCSORT, including BMCSORT error messages
	UNLOAD PLUS for DB2 Reference Manual	provides information about UNLOAD PLUS

BMC Software, Inc., Confidential and Proprietary Information

Online and Printed Books

The books that accompany BMC Software products are available in online and printed formats. Online books are formatted as Portable Document Format (PDF) files. Some online books are also formatted as HTML files.

To Access Online Books

To view any online book that BMC Software offers, visit the Customer Support page of the BMC Software Web site at **http://www.bmc.com/support_home**. You can also access PDF books from the documentation compact disc (CD) that accompanies your product.

Use the free Acrobat Reader from Adobe Systems to view, print, or copy PDF files. In some cases, installing the Acrobat Reader and downloading the online books is an optional part of the product-installation process. For information about downloading the free reader from the Web, go to the Adobe Systems site at **http://www.adobe.com**.

To Request Additional Printed Books

BMC Software provides some printed books with your product order. To request additional books, go to **http://www.bmc.com/support_home**.

Release Notes and Other Notices

Printed release notes accompany each BMC Software product. Release notes provide current information such as

- updates to the installation instructions
- last-minute product information

In addition, BMC Software sometimes provides updated product information between releases (in the form of a flash or a technical bulletin, for example). The latest versions of the release notes and other notices are available on the Web at **http://www.bmc.com/support_home**.

BMC Software, Inc., Confidential and Proprietary Information

Conventions

This section provides examples of the conventions used in this book and explains how to read syntax diagrams.

Changes

In this book, change bars signify changes that clarify or correct existing information, or that provide new information corresponding to product changes. This book does not use change bars to denote editorial and formatting changes or typographical errors that have been fixed, unless these updates significantly affect your use of the information.

General Conventions

This book uses the following general conventions:

Item	Example
directories, file names, Web addresses	The BMC Software home page is at **www.bmc.com**.
nonspecific key names, option names	Use the HELP function key. KEEPDICTIONARY option
MVS calls, commands, control statements, keywords, parameters, reserved words	Use the SEARCH command to find a particular object. The product generates the SQL TABLE statement next.

code examples, syntax statements, system messages, screen text	`//STEPLIB DD`
	
	`The table table_name is not available.`

emphasized words, new terms, variables	The instructions that you give to the software are called *commands*. In this message, the variable *file_name* represents the file that caused the error.

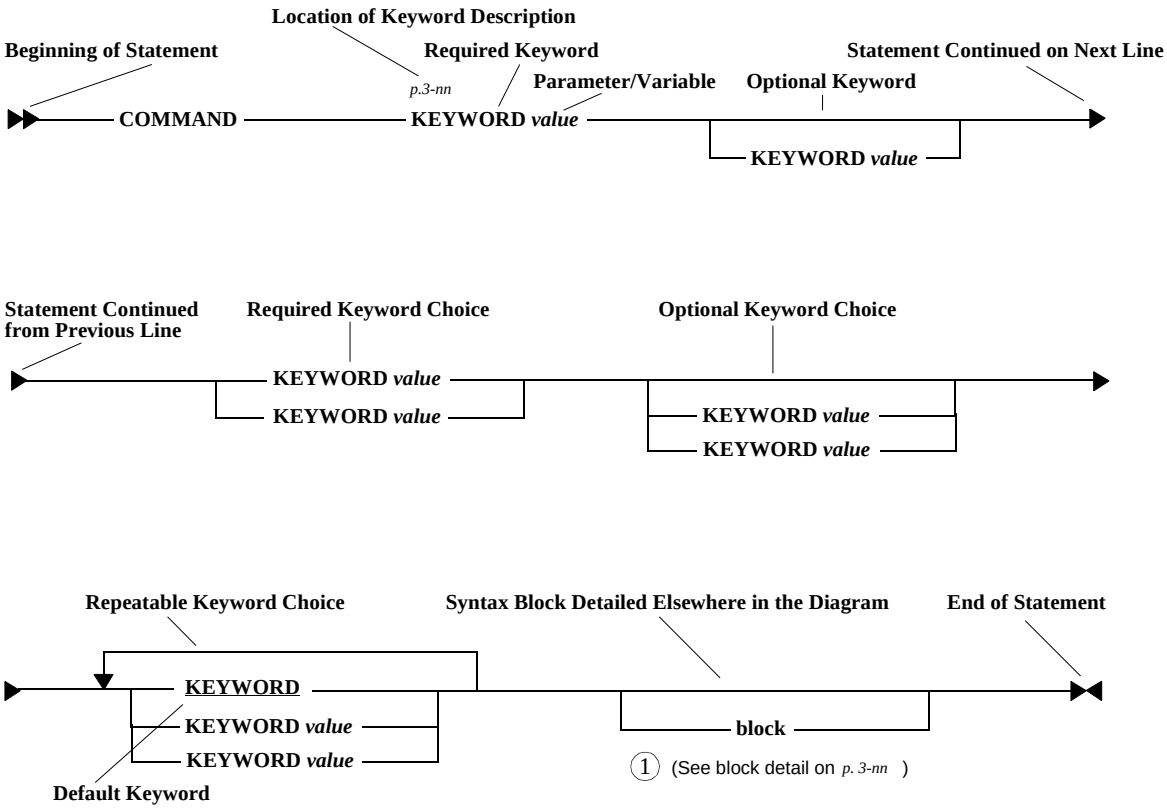
This book uses the following types of special text:

Note: Notes contain important information that you should consider.

Warning! Warnings alert you to situations that could cause problems, such as loss of data, if you do not follow instructions carefully.

Syntax Diagrams

Following are example syntax diagrams and information on the conventions used to create these diagrams.



The following guidelines provide information about syntax diagrams:

- Read diagrams from left to right and from top to bottom.
- Commands and keywords appear in all uppercase letters. In general, MVS commands, keywords, clauses, and data types appear in uppercase. However, if an item can be shortened, the minimum portion of the MVS command or keyword may appear in uppercase with the remainder of the word in lowercase (for example, CANcel).
- Required keywords are on the main line.
- Optional keywords are below the main line.

BMC Software, Inc., Confidential and Proprietary Information

- If you must select a keyword from two or more *required* keywords, the first keyword in the stack appears on the main line.

- If you can choose from two or more *optional* keywords, they are vertically stacked and the entire stack is below the main line.

- An underlined item is a default option.

- A recursive (left-pointing) arrow above a stack indicates that you may choose more than one item in the stack.

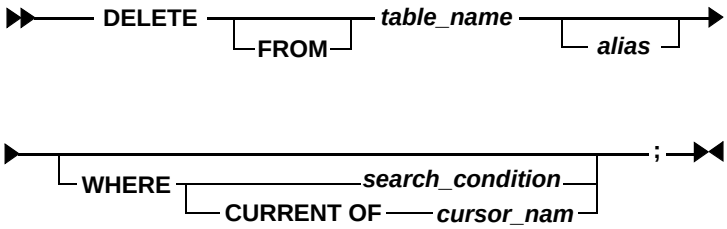
- The following conventions apply to variables in syntax diagrams:

  — Variables typically appear in lowercase.
  — If a variable is represented by two or more words, underscores connect the words (for example, *database_name* and *user_ID*).

- If punctuation marks, parentheses, or other such symbols are shown, you must enter them as part of the syntax.

- A part of the syntax diagram that is lowercase and followed by the word "block" indicates that the portion of syntax is too large or complex to include in the main syntax diagram. A numbered reference indicates its location in the "Syntax Diagram Details" section.

The following example illustrates the syntax for a hypothetical DELETE statement. Because the FROM keyword, *alias* variable, and WHERE clause are optional, they appear below the main command line. In contrast, the *table_name* variable appears on the command line because the table name is required. If the statement includes a WHERE clause, the clause must contain either a search condition or a CURRENT OF clause. (The *search_condition* variable appears on the main line for the WHERE clause, indicating that this choice is required.)



BMC Software, Inc., Confidential and Proprietary Information

Summary of Changes

This section provides an overview of new functions and changes in the LOADPLUS product.

Version 7.1.00 October 24, 2003

This release fixes known problems in the product. In addition, it includes the following enhancements:

- LOADPLUS now provides the ability to load data into tables whose table spaces or index spaces are created with DEFINE NO. You do not need to specify any additional syntax or other options to load these tables; however, the associated table spaces and index spaces must be in read/write (RW) status and cannot be in any restrictive status. LOADPLUS materializes the underlying data sets by performing a DB2 INSERT and ROLLBACK on the table.

- LOADPLUS now provides the option to not build or update secondary indexes that are associated with the table that you are loading. This option can provide additional flexibility in managing time and DASD resources when loading data. To use this feature, you specify one of the SKIPIX command options. See “SKIPIX” on page 3-30 for more information, including important considerations when using this option.

- LOADPLUS now includes a COPYPEND installation option, which allows you to set a default to have LOADPLUS set COPY pending when appropriate. Specifying the COPYPEND command option overrides this default value unless you specify ENFORCE on the installation option. For example, if you specify COPYPEND=(YES,ENFORCE) on the installation option, LOADPLUS always sets COPY pending when appropriate, regardless of the value of the COPYPEND command option. See page A-11 for more information.

BMC Software, Inc., Confidential and Proprietary Information

- LOADPLUS now includes a CHECKPEND command option that allows you to specify, on your LOAD command, whether to set dependent table spaces to CHECK pending (CHKP) status. This command option overrides the value of the CHEKPEND installation option, with the following exception. See page 3-67 for more information about this command option.

  To prevent the CHECKPEND command option from overriding the value of the CHEKPEND installation option, you can specify ENFORCE on the CHEKPEND installation option. For example, if you specify CHEKPEND=(YES,ENFORCE), LOADPLUS always sets CHKP when appropriate, regardless of the value of the CHECKPEND command option. See page A-10 for more information about the installation option.

- LOADPLUS now completes with a return code 0 instead of a return code 8 when the utility is not found during a TERM execution. For more information about running with the TERM parameter, see “TERM” on page 4-6.

  Warning! If you have jobs that interrogate the condition code of a LOADPLUS TERM step, BMC Software recommends that you evaluate those jobs to determine whether this change affects them.

- LOADPLUS now prints DSNHDECP values when printing MSGLEVEL(1) messages when the value of the LOADDECP installation option is YES. For an example of these messages, see “MAINT” on page 4-6.

- LOADPLUS now always deletes duplicates when you specify UNIQUECHECK NO or UNIQUECHECK CLUSTER. Duplicate processing under these options no longer results in the need to recover the objects in your load job. See “UNIQUECHECK” on page 3-52 for information about how LOADPLUS handles duplicates for each UNIQUECHECK specification.

- You can now use the DISCARDS command option to limit the number of discard records during a SQLAPPLY load job. See “DISCARDS” on page 3-40 for more information about this command option.

- You can now specify a value for the SYSDISC command option regardless of the value of the DELETEFILES option. This allows you to specify DELETEFILES NO SYSDISC YES to delete only the discard file. The DELFILES installation option has also been modified to allow you to make a comparable SYSDISC specification. For more information about these options, see “SYSDISC” on page 3-58 and “DELFILES” on page A-16.

BMC Software, Inc., Confidential and Proprietary Information

- The new STOPRETRY installation option allows you to set the number of times that LOADPLUS checks an object to determine whether DB2 has changed the status from stop pending (STOPP) to stopped (STOP). The new STOPDELAY installation option allows you to set the number of seconds that LOADPLUS waits between each check. See page A-9 for additional information about these two installation options.

- LOADPLUS no longer supports DB2 version 5. Although LOADPLUS version 7.1.00 does not run in a DB2 version 5 environment, earlier versions of LOADPLUS still support DB2 version 5.

- LOADPLUS no longer supports the ALTERINDEX command option and ALTIXTYP installation option because type 1 indexes are not supported by the versions of DB2 that this version of LOADPLUS supports.

- BMCSORT installation option and error message information has been moved from this book to the *BMCSORT Reference Manual*.

Version 6.2.00 September 20, 2002

This release fixes known problems in the product. In addition, it includes the following enhancements:

- LOADPLUS now provides the ability to load comma-separated-value (CSV) data. This data can originate from any product that generates CSV output, including distributed databases, spreadsheet applications, and the BMC Software UNLOAD PLUS for DB2 product. In support of this new feature, LOADPLUS provides the FORMAT CSV option.

- LOADPLUS now provides its single-phase architecture for LOAD RESUME YES jobs. Specifying PRELOAD LOAD with LOAD RESUME YES can improve the performance of your RESUME load job by combining the PRELOAD and LOAD phases into a single COMBINED phase.

  This feature includes the ability to specify LOAD RESUME YES PART REPLACE on a single-phase load regardless of whether you have nonclustering indexes and without having to replace all of the partitions that are involved in the load. However, see “PRESORTED” on page 3-44 for restrictions when specifying ORDER PRESORTED.

BMC Software, Inc., Confidential and Proprietary Information

- A new LOADPLUS command option, FORMAT BMCUNLOAD, allows you to load data that was unloaded by UNLOAD PLUS for DB2 with the FORMAT BMCLOAD option. The UNLOAD PLUS option unloads the data in an internal format that only LOADPLUS can read. The FORMAT BMCUNLOAD option significantly improves performance when transferring data between similarly defined tables by reducing the need for data verification and conversion. For information about unloading the data for this feature, see the *UNLOAD PLUS for DB2 Reference Manual*.

- LOADPLUS now provides support for running in a BatchPipes environment. This includes support for input from the Job Optimizer Pipes component of the BMC Software MAINVIEW® Batch Optimizer product.

- LOADPLUS now provides full identity column support.

  Included in this enhancement is the ability to control certain aspects of identity column processing. LOADPLUS provides this control through the following new command and installation options:

  — The IDCACHE command and installation options allow you to specify the size of the temporary cache that LOADPLUS uses for reserving identity column values.

  — The UPDATEMAXA command option and UPDMAXA installation option allow you to specify whether you want LOADPLUS to reset the MAXASSIGNEDVAL field in SYSIBM.SYSSEQUENCES when loading identity column values from an input file.

  — The IDERROR command and installation options allow you to tell LOADPLUS whether to terminate or discard when encountering a generated identity column value that is outside the range that is defined on the column.

- When you dynamically allocate copy data sets, you can now allocate them as generation data group (GDG) data sets. The following new command and installation options provide the ability to control certain aspects of this feature:

  — The new GDGLIMIT installation and command options allow you to specify the number of GDG data set generations to keep.

  — The new GDGEMPTY installation option allows you to specify whether the oldest generation or all generations are uncataloged when the GDGLIMIT is reached.

BMC Software, Inc., Confidential and Proprietary Information

— The new GDGSCRATCH installation option allows you to specify whether the uncataloged data set should be deleted.

- For dynamically allocated copy data sets, you can now use the new EXPDT command and installation options to specify an expiration date. You can also use the new RETPD command and installation options to specify a retention period.

- You can now have LOADPLUS dynamically allocate your SYSREC data sets. To do so, you specify one or more input data set names by using the new command option INDSN. This feature allows you to load data from different input data sets by updating just your syntax cards. In addition, if you use this feature, you do not need SYSREC DD statements in your JCL, which eliminates the need to override these DD statements in your cataloged JCL procedures (PROCs).

- For STOGROUP-defined staging data sets and multiple data set objects, LOADPLUS now creates additional data sets as needed and deletes any unused data sets even when the value of the REDEFINE command or installation option is NO.

- The SQLAPPLY load feature includes the following changes:

  — LOADPLUS has added new SQLAPPLY command and installation options that allow you to pass the collection ID and owner ID to Apply Plus for binding the Apply Plus plans. These options are APCOLLECTION and APOWNER, respectively.

  — To run SQLAPPLY, version 6.2.00 of LOADPLUS requires version 1.5.00 or later of the Apply Plus component of the Log Master for DB2 product. Version 1.5.00 of Apply Plus has new authorization requirements.

- LOADPLUS has added the following new installation options that are the equivalent of existing command options. These installation options allow you to set a default value for your jobs without having to specify command options on your JCL.

  — The REDEFINE option tells LOADPLUS whether to delete and redefine the VSAM data sets for the table space or index space as part of the load job.

  — The KEEPDICTIONARY option tells LOADPLUS whether to keep the existing compression dictionary or build a new one.

BMC Software, Inc., Confidential and Proprietary Information

To allow you to override the installation option, the KEEPDICTIONARY command option has been modified to accept a value of NO. To maintain compatibility with earlier releases, if you do not specify a value with the KEEPDICTIONARY command option, LOADPLUS assumes a value of YES.

— The PREFORMAT option tells LOADPLUS whether to preformat the unused pages of the data set.

To allow you to override the installation option, the PREFORMAT command option has been modified to accept a value of NO. To maintain compatibility with earlier releases, if you do not specify a value with the PREFORMAT command option, LOADPLUS assumes a value of YES.

- The BMC Software sort engine is now an individual component, separate from LOADPLUS. References in the documentation have been updated to reflect this change, including a change in the name of the component to BMCSORT.

- LOADPLUS now allows you to run a LOAD REPLACE job when the objects that are participating in the load are in logical page list (LPL) or write error page range (WEPR) status.

- LOADPLUS no longer supports DB2 version 4. Although LOADPLUS version 6.2.00 does not run in a DB2 version 4 environment, earlier versions of LOADPLUS still support DB2 version 4.

Version 6.1.00 July 15, 2001

This release fixes known problems in the product. In addition, it includes the following enhancements:

- LOADPLUS now provides online load capabilities:

— LOADPLUS provides the capability to perform a load while leaving the object that you are loading in read/write status (SHRLEVEL CHANGE).

If you specify **LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY** (also referred to as a SQLAPPLY load), LOADPLUS integrates with the BMC Software Apply Plus for DB2 for OS/390 product. This type of load job loads the data by using SQL INSERT statements. This feature does not require that you have a license for the Apply Plus product but does require that Apply Plus be installed and made available via STEPLIB, JOBLIB, or LINKLIST at execution time.

BMC Software, Inc., Confidential and Proprietary Information

If you specify **LOAD REPLACE SHRLEVEL CHANGE**, LOADPLUS loads the data to a staging data set. LOADPLUS then renames the staging data set, replacing the original object with the staging data set.

— LOADPLUS provides the capability to perform a load while leaving the object that you are loading in read-only status (LOAD REPLACE SHRLEVEL REFERENCE). LOADPLUS loads the data to a staging data set. LOADPLUS then renames the staging data set, replacing the original object with the staging data set.

- LOADPLUS provides initial support for the general availability (GA) version of DB2 for OS/390 version 7. Additionally, LOADPLUS recognizes either I or J instance qualifiers in your DB2 table space and index data set names.

- LOADPLUS provides initial support for identity columns.

- You can specify a new command option, NOSUBS, to tell LOADPLUS that you do not want to use a substitution character during translation between coded character set identifiers (CCSIDs).

- If you are running under DB2 version 4 and you specify INLINE YES on your LOAD command, LOADPLUS now converts that specification to INLINE NO. Likewise, if you provide a value of YES for your INLINECP installation option and you are running under DB2 version 4, LOADPLUS converts the value of that option to NO.

- LOADPLUS no longer supports the obsolete command option COPY IMAGE. If you attempt to use this command option, LOADPLUS terminates and issues message BMC50104E.

- LOADPLUS now uses the BMC Software sort engine instead of your system sort routine. This is the sort engine that became available in version 5.1.00.

  Because LOADPLUS uses its internal sort engine instead of your system sort routine, LOADPLUS no longer uses the SABOVE installation option. This release removes SABOVE from the installation options module. In addition, the SMCORE installation option now applies to the operation of the internal sort engine instead of your system sort routine.

- To maintain consistency with the other BMC Software DB2 utility products, two installation options have new names. The DB2NTRY installation option available in previous releases is now named SQLRETRY and the DB2WAIT installation option is now named SQLDELAY.

BMC Software, Inc., Confidential and Proprietary Information

- Because of architectural changes, the SYNCTO installation option has become obsolete. This release removes SYNCTO from the installation options module.

- The default for the CENTURY installation option has been removed, requiring that you provide the appropriate value for your site. If you do not supply a value, your installation options module will not assemble successfully. In previous releases, the default value supplied was (1900,1999).

- LOADPLUS no longer supports DB2 version 3. Although LOADPLUS version 6.1.00 does not run in a DB2 version 3 environment, earlier versions of LOADPLUS still support DB2 version 3.

Version 5.1.00 June 15, 2000

This release fixes known problems in the product. In addition, it includes the following enhancements:

- LOADPLUS supports DB2 version 6 functions, including

  — 8 KB and 16 KB data pages
  — user-defined data type (UDT)
  — support for new statuses
  — DSSIZE
  — ROWID
  — LOAD RESUME operations associated with base tables on large objects (LOBs)

  LOADPLUS currently has the following limitations on support for DB2 version 6 functions:

  — LOADPLUS does not load LOB data into auxiliary tables.

  — For LOADPLUS to load data into a table whose table space or index space was created with DEFINE NO, the table must first be materialized with INSERTs or with a previous load process.

  — LOADPLUS does not load data that is in IEEE Binary Floating Point (BFP) format.

  — LOADPLUS does not support tables that contain IDENTITY columns.

  — When you alter the length of a VARCHAR column that is part of an index, LOADPLUS requires that you specify LOAD REPLACE or specify or default to LOAD RESUME INDEX BUILD.

BMC Software, Inc., Confidential and Proprietary Information

- LOADPLUS now supports dynamic allocation of image copy data sets. This support includes the following new options:

  — new copy types for the DDTYPE command option.

  — a COPYLVL command option and installation option, which allows you to specify whether a dynamically allocated data set should be assigned to the full table space or to each partition being loaded.

- To improve performance, LOADPLUS provides the following enhancements:

  — LOADPLUS now multitasks the creation of image copy data sets and overlaps the copy process with the LOAD phase where possible.

  — LOADPLUS now performs multiple index tasks for unique indexes during the PRELOAD phase.

  — LOADPLUS now decides whether to sample or scan an index for cardinality, when determining cardinality is necessary. LOADPLUS also processes all data sets in a multi-data-set index and exploits multitasking throughout the ANALYZE phase. The ANALYZE command has been revised to reflect these changes in analyze processing. Changes to this command include new keywords, SAMPLE and SCAN, that allow you to control how LOADPLUS determines cardinality. This improves performance in the ANALYZE phase.

  — LOADPLUS now uses an internal sort engine instead of your system sort routine. This improves performance during sort processing.

- Using the new INLINE option of the LOAD command, or the new INLINECP installation option, you can tell LOADPLUS to create inline image copies *as* it loads your tables, rather than creating DSN1COPYs or image copies *after* it loads your tables. This can reduce the elapsed time of your load job.

- LOADPLUS now includes estimates for the copy data sets in the report it generates when you specify ANALYZE PAUSE or ONLY.

- LOADPLUS can now detect and clean up duplicates in the LOAD phase. LOADPLUS detects duplicates in either the LOAD phase or the PRELOAD phase, depending on the type of load you are running and the UNIQUECHECK option you specify.

- LOADPLUS now allows you to specify whether to delete your discard file (SYSDISC) if there have been no discards during load processing.

BMC Software, Inc., Confidential and Proprietary Information

- LOADPLUS now provides options that allow you to specify default ddnames or data set prefixes for your work files at installation. (Specifying these ddnames or data set prefixes with the corresponding LOAD command options allow you to override the installation defaults.)

- LOADPLUS now supports the SUBBYTE and ERRORBYTE columns of the SYSIBM.SYSSTRINGS table.

- The limit on SIZEPCT has been removed to allow larger dynamic work file allocations for multi-data-set DB2 objects.

- In most cases, if LOADPLUS attempts to stop a space but is unsuccessful due to long-running units of work, it no longer leaves a space in STOP pending (STOPP) status.

- The new installation and command options MAXEXTSZ allows you to specify the largest extent that LOADPLUS should allocate for each dynamically-allocated file.

- The new installation and command options SMSUNIT tells LOADPLUS to pass the UNIT parameter to SMS during dynamic allocation (along with the SMS class options and other values that LOADPLUS already passes).

BMC Software, Inc., Confidential and Proprietary Information

Chapter 1 Introduction to LOADPLUS

This chapter presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Overview of LOADPLUS

DB2 is a powerful relational database management system whose wide success in the database management community rests on its ability to provide fast application development and easy data access to the user. As users depend more and more on DB2 for critical business applications, the need for continuous operations becomes crucial.

The Need to Load Data

In a DB2 environment, there often is a need to load large amounts of data into DB2 tables. Data is loaded to initially populate a table, periodically replace the data in a table, periodically add data to a table, or reload data into a table.

This data can originate from sources such as

- other DB2 tables (possibly from a different DB2 system)
- other database management systems (DBMSs), including IMS or distributed-systems DBMSs
- applications that generate sequential files

In many cases, users load millions of rows. Because data in the table space is often unavailable to applications during the loading process, the loading process must be completed as quickly as possible. The DB2 LOAD utility supplied by IBM loads data from a sequential file into one or more tables or partitions in a table space.

The LOADPLUS Solution

The BMC Software product line for DB2 has been developed to respond to the real needs of database administrators and system administrators by providing high-performance database administration and utility products. The BMC Software LOADPLUS for DB2 product replaces most of the functions of the IBM DB2 LOAD utility and provides several additional functions. Advanced techniques and additional functions allow LOADPLUS to load more types of data faster than the IBM DB2 LOAD utility does.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS Benefits

LOADPLUS offers you the following significant benefits:

- **increases availability of DB2 data** because LOADPLUS reduces the time needed to load the data

  In addition, the online load options allow your tables to remain accessible in either read-only or read/write status while LOADPLUS loads your data.

- **reduces costs of loading DB2 data** because LOADPLUS uses fewer CPU cycles and EXCPs

- **provides a comprehensive set of data type conversions** that eliminates the need to write special application code

- **expands functionality** to eliminate the need to perform additional tasks before and after data is loaded

- **provides flexibility** by allowing you to load data from a variety of input formats

Solution Integration

LOADPLUS is also a component of the Database Administration for DB2 solution. Customers who acquire this solution benefit from all of the features of the individual component products and technologies as well as additional features that are available when one Database Administration component can rely on the presence of other components. For example, if you have a Database Administration solution password, you can take advantage of the ability to run multiple utility steps in parallel within a CHANGE MANAGER worklist.

BMC Software, Inc., Confidential and Proprietary Information

Solution Components

This solution integrates the features of the following products and technologies:

Product Components

- CATALOG MANAGER for DB2
- CHANGE MANAGER for DB2
- COPY PLUS for DB2
- LOADPLUS for DB2
- UNLOAD PLUS for DB2

Technology Components

- Apply Plus for DB2
  (a component of the Log Master for DB2 product that is required for the SQLAPPLY feature of LOADPLUS)
- BMCDSN Command Processor
  (used with COPY PLUS, LOADPLUS, and UNLOAD PLUS)
- BMCSORT
  (required for full sort support for LOADPLUS, UNLOAD PLUS, and RECOVER PLUS)
- Cross-System Image Manager (XIM) for DB2
  (required by CHANGE MANAGER to execute portions of a worklist concurrently)
- RECOVER PLUS for DB2
  (required by CHANGE MANAGER to migrate data from image copy data sets)
- SNAPSHOT UPGRADE FEATURE for DB2
  (licensed component of the EXTENDED BUFFER MANAGER for DB2 product that is used by COPY PLUS and UNLOAD PLUS)

Solution Features

You can use the BMC Software Database Administration for DB2 solution to manage your DB2 databases quickly, efficiently, and effectively. By using the tools in this solution, you can keep ahead of the workload growth curve because you can accomplish tasks with greater success and with less time required for error recovery.

BMC Software, Inc., Confidential and Proprietary Information

The Database Administration for DB2 solution provides the following features:

- analyzes the effects of changes to database structures
- automates creating, altering, and dropping DB2 objects
- provides easy navigation and management of the DB2 catalog
- aids in developing schema changes for application development and production maintenance
- increases application availability and optimizes resources by reducing the time that it takes to perform changes and to administer multiple DB2 environments
- maintains referential integrity and data integrity
- completes complex structure changes to databases quickly and accurately
- uses the fastest, most efficient utilities available to copy, unload, and reload data
- reduces the elapsed time required for executing a worklist that is generated by CHANGE MANAGER by executing portions of the worklist concurrently

Tasks That LOADPLUS Performs

LOADPLUS accomplishes standard load tasks and also offers numerous functional enhancements in the following areas:

Resources

- analyzes and allocates memory and CPU resources for maximum throughput
- optionally analyzes data set resources that are needed for the specific load job
- optionally allocates input, load data, index work, sort work, discard, error, and copy files dynamically

BMC Software, Inc., Confidential and Proprietary Information

Indexes

- optionally reorganizes the participating indexes
- optionally updates indexes, which can improve performance when loading a small number of rows into an existing table that has a large number of rows
- optionally skips building or updating secondary indexes

Performance

- optionally provides single-phase load processing, which combines the PRELOAD and LOAD phases into one phase (the COMBINED phase)

  In most cases, a single-phase load eliminates the requirement for work data sets (SORTOUT and SYSUT1) and provides additional performance gain.
- provides an option to order data by table and clustering key or by clustering key alone
- optionally skips checks for duplicate keys in unique indexes or checks for duplicate keys only in unique clustering indexes
- provides an option to improve performance by not sorting the data or clustering index if the data is already in correct clustering index sequence
- provides an option to improve performance when transferring data between similarly defined tables by loading data that was unloaded by UNLOAD PLUS in an internal format

Accessibility

- provides an option for the utility to pause in its processing before the data is loaded if any input records are discarded
- sets the status of all affected table spaces to CHECK pending (CHKP) if referential integrity constraints or table check constraints exist and were not checked
- runs concurrently with other BMC Software utilities on DB2 spaces
- provides restart capabilities

BMC Software, Inc., Confidential and Proprietary Information

- provides an option to leave tables in read-only status while loading your data
- provides an option to leave tables in read/write status while loading your data

Compression

- builds or keeps a compression dictionary when the object being loaded has the COMPRESS YES attribute

Loading Data

- loads data from one or more data sets into one or more tables or table space partitions in a single command execution, either replacing or adding to the existing data
- optionally sorts the input data by the clustering key, if one exists
- optionally orders rows by table
- allows you to load data that is in comma-separated-value (CSV) format
- allows you to load data that was archived by the REORG PLUS product
- allows you to load data that was unloaded by the UNLOAD PLUS product in an internal format
- allows you to load data from batch pipes
- provides powerful selection criteria for specifying which input records to load
- optionally deletes and redefines user-defined data sets and data sets that are defined in DB2 storage groups as part of the load
- avoids adding rows and keys to the table spaces and indexes if they cause duplicate keys in a unique index, so that the rows and keys do not need to be deleted later in the LOAD phase

  Following this process, your table spaces and indexes remain organized.

BMC Software, Inc., Confidential and Proprietary Information

Copies

- with a self-contained copy function, produces image copies or DSN1COPYs concurrently with the load
- optionally dynamically allocates the image copy data sets
- optionally produces inline image copies *as* it loads your tables, rather than DSN1COPYs or image copies *after* it loads your tables

Statistics and Reports

- produces a BMCSTATS report and optionally saves the statistics in the BMC Software DASD MANAGER PLUS tables, if that product is installed
- prevents the output of useless discard messages and records by allowing you to ignore specified types of discards
- optionally updates statistics in the DB2 catalog so that the DB2 optimizer can use them
- reports multiple load command errors, reducing the number of runs needed to correct your load specification

Operations on Data

- provides a comprehensive set of data type conversions and support for user-written exit routines that perform special data conversions
- provides an option to load ASCII or EBCDIC data
- provides options to perform data translation
- provides an additional phase (PRELOAD) for verifying that your input data is correct before your existing data is replaced (and lost), without incurring the overhead of actually loading the data
- provides full Boolean logic on the WHEN condition (allows AND, OR, NOT, and parentheses)
- provides control over expressions used in WHEN, NULLIF, and DEFAULTIF processing through the RULES installation option
- allows you to concatenate separate physical input records into one larger logical record

BMC Software, Inc., Confidential and Proprietary Information

- allows you to assign a constant value or CURRENT DATE, CURRENT TIME, or CURRENT TIMESTAMP to a column
- allows multiple NULLIF and DEFAULTIF conditions on a field specification
- provides powerful criteria for assigning NULL or default values to columns
  - — allows you to assign default values to nullable or nondefaultable columns
  - — allows you to specify the value for a DEFAULTIF value, overriding the standard default
  - — allows you to assign NULL or a default value to a column if a conversion error occurs while LOADPLUS is processing the column's input value

Functional and Operational Differences between LOADPLUS and the IBM DB2 LOAD Utility

This section describes those LOADPLUS functions and operations that differ from version 7 of IBM's DB2 LOAD utility:

Authorization

- LOADPLUS does not run as part of the DB2 subsystem. Therefore, you must have system authorization that is similar to that required by DB2 unless you are using the Resource Access Control Facility (RACF) and LOADPLUS is installed with OPNDB2ID=YES.

LOADPLUS Processing Phases

- LOADPLUS combines RELOAD, INDEXVAL, DISCARD, REPORT, SORT, BUILD, or SORTBLD into the PRELOAD and LOAD phases for a two-phase load, and into the COMBINED phase for a single-phase load. SORT and BUILD, however, still act together as a single phase for restart processing.

BMC Software, Inc., Confidential and Proprietary Information

Object Status

- For SHRLEVEL NONE (the default SHRLEVEL option), the table space that you are loading is stopped at the beginning of the PRELOAD or LOAD phase, or during the COMBINED phase. LOADPLUS starts the table space at the end of the LOAD or COMBINED phase. See “Object Status” on page 2-8 for more information.

- LOADPLUS allows you to run a LOAD RESUME YES PART REPLACE load of a table space when any of its partitions are in REORG pending (REORP) status as long as you are replacing all of the partitions that are in REORP status.

Compression

- If the object that you are loading has the COMPRESS YES attribute, LOADPLUS always reserves enough pages to hold the largest compression dictionary.

Indexes

- Unless you specify INDEX UPDATE or SKIPIX, LOADPLUS reorganizes all indexes that participate in a LOAD RESUME job by unloading existing index entries, merging in new index entries, and then sorting and reloading them into the index spaces. If you specify INTO TABLE PART REPLACE, then LOADPLUS rebuilds the clustering indexes for the partitions that are being replaced.

- You can tell LOADPLUS to skip the index building process for secondary indexes that are associated with the table that you are loading.

DEFINE NO Objects

- To enable loading data into tables whose table spaces or index spaces were created with DEFINE NO, LOADPLUS materializes the underlying data sets by performing a DB2 INSERT and ROLLBACK on the table.

  Note: For a SQLAPPLY load, DEFINE NO data sets are materialized by the DB2 INSERT processing that occurs during the COMBINED phase.

BMC Software, Inc., Confidential and Proprietary Information

Identity Columns

- LOADPLUS allows you to set the size of the cache that LOADPLUS uses when loading generated identity column values.

- LOADPLUS allows you to specify whether you want to reset the value of MAXASSIGNEDVAL in the SYSIBM.SYSSEQUENCES table when loading identity column values from an input file.

  Note: When generating identity column values, LOADPLUS always updates MAXASSIGNEDVAL.

- For most load types, you can tell LOADPLUS whether to discard or fail when a generated identity column value would fall outside the range that is defined on the column.

- In most cases, LOADPLUS allows you to load by partition regardless of whether you have an identity column as part of your clustering key.

- Unless you tell LOADPLUS not to enforce table check constraints, LOADPLUS fails when attempting to generate identity column values if the identity column is referenced in the check condition of a DB2 check constraint.

LOAD Command Options

- LOADPLUS allows only one LOAD command in the input data set (SYSIN).

- LOADPLUS allows you to specify SHRLEVEL REFERENCE with a LOAD REPLACE job. SHRLEVEL REFERENCE allows read-only access to the table space that you are loading.

- LOADPLUS allows you to specify SHRLEVEL CHANGE with a LOAD REPLACE job.

- LOADPLUS allows you to specify SHRLEVEL CHANGE with a LOAD RESUME job for all supported versions of DB2.

- LOADPLUS optimizes the processing of INTO statements by combining statements that are the same. For example, if you specify INTO PART 1 and INTO PART 1 on the same LOAD command, LOADPLUS combines these statements into a single INTO statement and loads one set of rows. In contrast, DB2 loads two sets of rows into partition 1 in this example.

- For a field specification, LOADPLUS requires that you specify the position before the data type in the command statement.

BMC Software, Inc., Confidential and Proprietary Information

WHEN, NULLIF, and DEFAULTIF Processing

- LOADPLUS provides the RULES installation option for specifying the set of comparison rules to be used in WHEN, NULLIF, and DEFAULTIF processing. Specifying RULES=STANDARD causes comparisons to be performed directly on the input record values rather than the values as they will be in the column. Specifying RULES=BMC causes comparisons to be performed on the values as they will be in the column.

Constraints and Referential Integrity

- LOADPLUS does not have an ENFORCE CONSTRAINTS option for referential integrity violations. Instead, LOADPLUS places the affected table spaces in CHECK pending status. After the LOADPLUS utility completes, you must run CHECK PLUS for DB2 (specifying the CHECK DATA SCOPE ALL option) or the IBM CHECK DATA utility (specifying SCOPE ALL) to ensure that referential integrity has not been violated.

  Note: If you are running a SQLAPPLY load, the Apply Plus architecture preserves referential integrity relationships.

- When table check constraint columns contain one or more of the following conditions, processing continues as though ENFORCE NO had been specified:

  — floating-point constants
  — VARCHAR columns with a length greater than 255

Data Types and Conversions

- LOADPLUS always truncates digits for floating-point-to-decimal conversions if digits are not needed in the precision. This action differs from IBM’s DB2 LOAD action of rounding in certain circumstances, but is consistent with the LOADPLUS definition of numeric-to-numeric conversions. If rounding is required, use the ROUND keyword on your field specification.

- LOADPLUS supports all floating-point formats except IEEE Binary Floating Point (BFP).

BMC Software, Inc., Confidential and Proprietary Information

Large Objects (LOBs)

- LOADPLUS loads data into base tables that are defined with LOB columns (BLOB, CLOB, and DBCLOB) with the exceptions noted below. LOADPLUS does not load data into the LOB auxiliary tables. LOADPLUS puts an empty LOB indicator, or a null LOB indicator if the column allows nulls, into the base table to indicate an empty or null LOB column. Base tables that are defined with LOB columns have the following limitations:

  — You cannot run a LOAD REPLACE job or specify INTO TABLE with the PART REPLACE option.

  — You cannot specify a LOB indicator column as part of your INTO field specifications.

Multiple Data Sets and Multitasking

- LOADPLUS automatically multitasks when you specify multiple data sets. You can specify multiple SYSREC input data sets and use multiple SORTOUT data sets to load partitioned tables. You can also specify multiple SORTWK data sets to multitask sort processing and multiple SYSUT1 data sets to multitask index processing.

Work Data Sets

- When allocating SORTOUT data sets, LOADPLUS requires that the total allocation be large enough to contain the DB2 row images that are created during the PRELOAD or COMBINED phase.

- For a single-phase LOAD REPLACE job or a LOAD RESUME YES SHRLEVEL CHANGE job, LOADPLUS does not use the SORTOUT and SYSUT1 work data sets and does not require that you define them in your JCL. For a single-phase LOAD RESUME YES SHRLEVEL NONE job, these data sets are required in certain cases only if you want to be able to restart the job.

Recoverability/Restartability

- LOADPLUS uses work files until it loads all the data. Therefore, the work files should not be temporary data sets if you want to be able to restart the job.

BMC Software, Inc., Confidential and Proprietary Information

- LOADPLUS allows an image copy to be made after a table is loaded even if LOADPLUS leaves the table in CHECK pending status. Because the CHECK DATA utility logs any changes made to the table, data recovery is possible using the RECOVER utility, which uses the full image copy and applies log records as necessary, ensuring the integrity of the data in the table. A RECOVER TOCOPY utility execution places the affected table in CHECK pending status again. A subsequent invocation of the CHECK DATA utility restores referential integrity.

- For most types of loads, LOADPLUS does not write to the DB2 log. Therefore, you must create a full image copy to ensure recoverability of the table space after loading. You can create a full image copy by using the LOADPLUS COPY YES option. You can also use the BMC Software COPY PLUS for DB2 product or the IBM DB2 COPY utility.

  Note: A LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY load (also referred to as a SQLAPPLY load) *does* write to the DB2 log. In addition, you cannot create an image copy with this type of load in LOADPLUS.

COPY Pending Status

- When you specify COPY YES, LOADPLUS resets the COPY pending status after loading the data and starting the table space.

- When you specify COPY NO, LOADPLUS places the table space in COPY pending status only if the value of COPYPEND is YES. If the value of COPYPEND is NO, LOADPLUS does not place the table space in COPY pending status.

SYSMAP Data Set

- LOADPLUS does not use a SYSMAP data set.

Multiple Volumes

- For multivolume STOGROUP-defined table spaces and indexes, LOADPLUS attempts to reallocate the data set on the volume on which it currently resides if the volume is still defined in the STOGROUP. After the current volume, the order of the volumes retrieved from the STOGROUP is unpredictable.

BMC Software, Inc., Confidential and Proprietary Information

Concurrent Utilities

- LOADPLUS uses the BMCSYNC table to control access to DB2 spaces by BMC Software utilities. This allows LOADPLUS and other BMC Software utilities to run concurrently on DB2 spaces.

How LOADPLUS Works

This section describes the LOADPLUS execution phases and the data sets that LOADPLUS uses. Figures 1-1 through 1-3, starting on page 1-17, illustrate the possible input and output data sets that are associated with each phase of execution.

Execution Phases of LOADPLUS

This section describes the execution phases of LOADPLUS and their primary functions.

UTILINIT This phase initializes the job; reads, parses, and verifies the LOAD command; performs DB2 catalog lookup; and, if any referenced object is VCAT-defined, verifies the commands in the SYSIDCIN data set.

ANALYZE This optional phase analyzes the objects that are being loaded and optionally produces statistics to help determine data set size for allocating those data sets. LOADPLUS executes this phase when dynamic work file allocation is active or when you specify the ANALYZE and ENUMROWS command options.

PRELOAD This phase reads data from the input data sets. In this phase, LOADPLUS converts data as needed and verifies that the data is correct, builds DB2 rows in the SORTOUT data set, builds index information in the SYSUT1 data set, and writes any input records that are in error to the SYSDISC discard data set. The PRELOAD phase also builds the compression dictionary and compresses the data rows.

If you are loading more than one table in a segmented table space, sorting the rows using the ORDER YES option, or you have unique indexes and are checking for duplicate keys, the PRELOAD phase invokes the BMCSORT component.

If you are loading rows into tables or partitions that already contain data, the PRELOAD phase reads the indexes and table space of the table or partition being loaded. The SYSREC and SYSDISC data sets are freed at the end of this phase and you should not specify FREE=CLOSE for these data sets.

BMC Software, Inc., Confidential and Proprietary Information

LOAD This phase redefines the VSAM data sets when the value of the REDEFINE option is YES. In this phase, LOADPLUS reads the rows from the SORTOUT and SYSUT1 data sets and loads the data into the table space and indexes. This phase invokes BMCSORT for indexes that are not sorted in the PRELOAD phase. The LOAD phase creates any requested copies and registers them with the DB2 catalog.

COMBINED This phase combines all functions of the PRELOAD and LOAD phases into a single processing phase. The COMBINED phase avoids the intermediate steps of writing to the SORTOUT and SYSUT1 data sets either by writing data directly to the table space and index space or by sending data directly to Apply Plus for inserting to the table space and index space. You do not have to specify the SORTOUT and SYSUT1 data sets in your JCL, eliminating the need to allocate disk and tape resources in large environments.

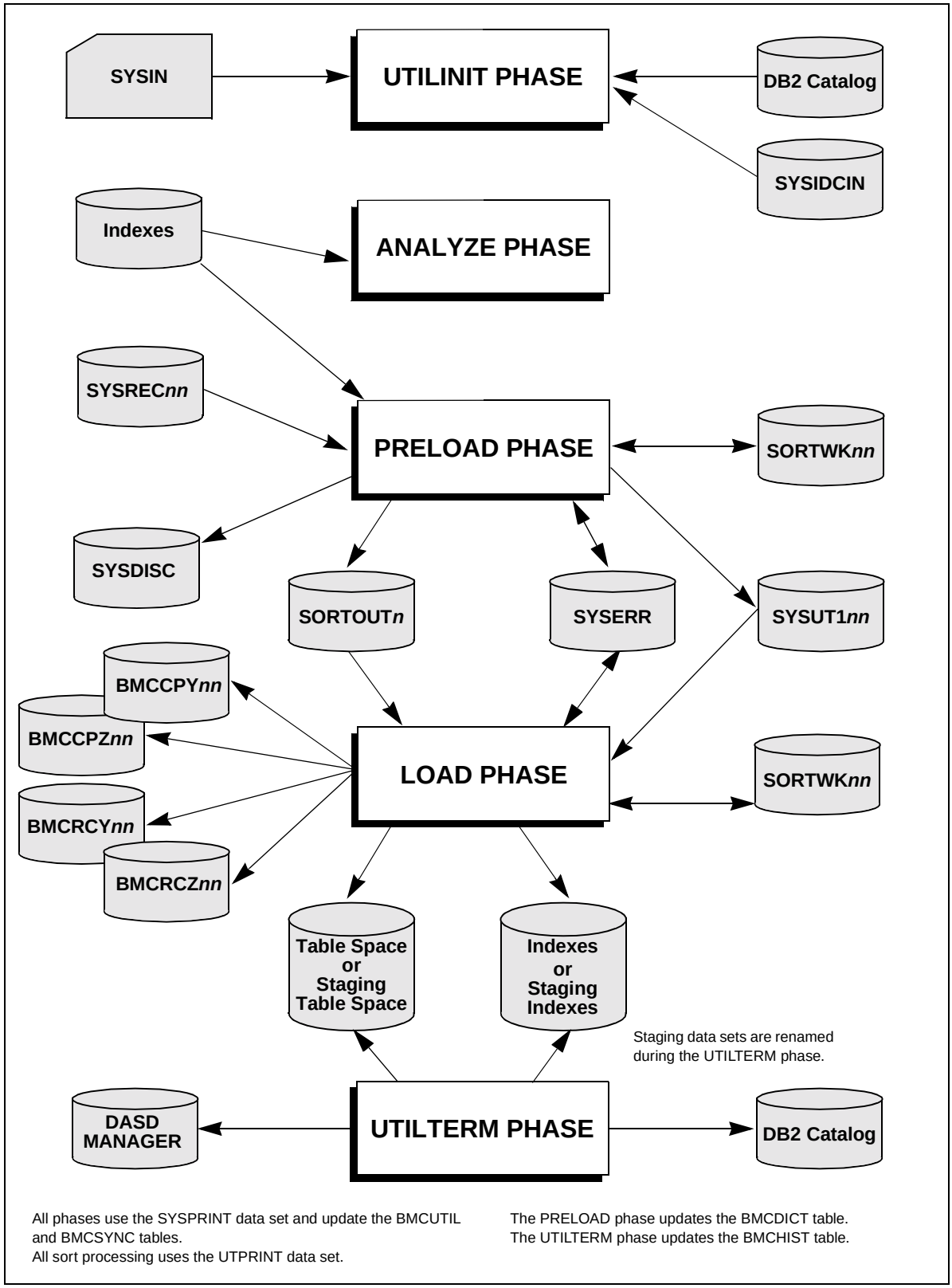
Note: For LOAD RESUME YES SHRLEVEL NONE, the SORTOUT and SYSUT1 data sets are optional. However, if you do not specify these data sets, your load job might not be restartable. If SORTOUT and SYSUT1 are available, LOADPLUS writes index entries and the new DB2 row images to these data sets.

UTILTERM This phase performs cleanup tasks. In this phase, LOADPLUS sets the pending status of all affected tables and table spaces and updates the BMC Software BMCHIST table, the BMCSTATS tables (if the BMC Software DASD MANAGER PLUS product is installed), and the DB2 catalog statistics. In addition, for LOAD REPLACE SHRLEVEL REFERENCE and for LOAD REPLACE SHRLEVEL CHANGE, this phase performs the rename and delete operations that are associated with the staging data sets.

All Phases All phases update the BMCUTIL and BMCSYNC DB2 tables.

BMC Software, Inc., Confidential and Proprietary Information

Figure 1-1 Two-Phase Load Processing Phases



BMC Software, Inc., Confidential and Proprietary Information

Figure 1-2 Single-Phase Load Processing Phases

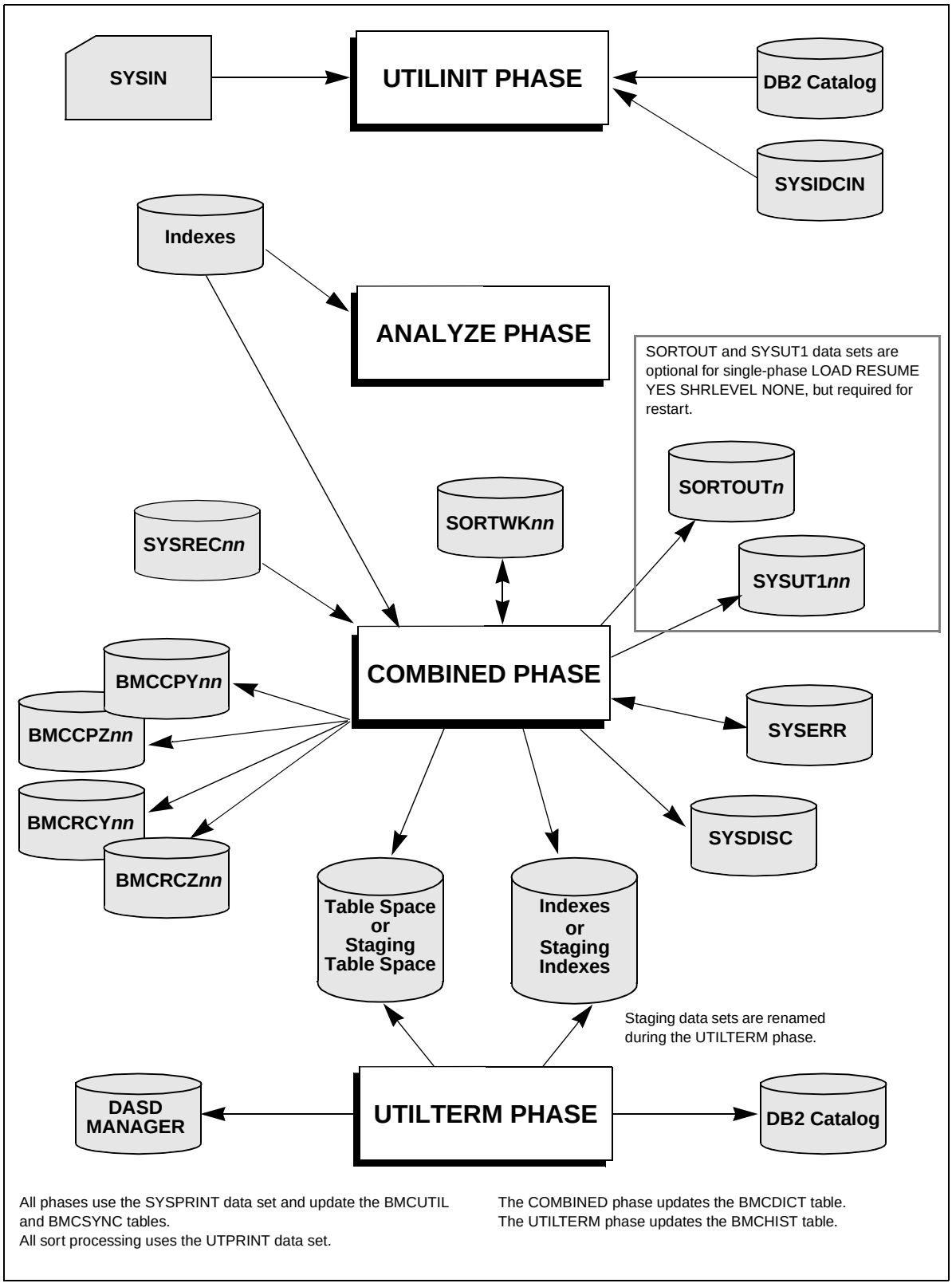
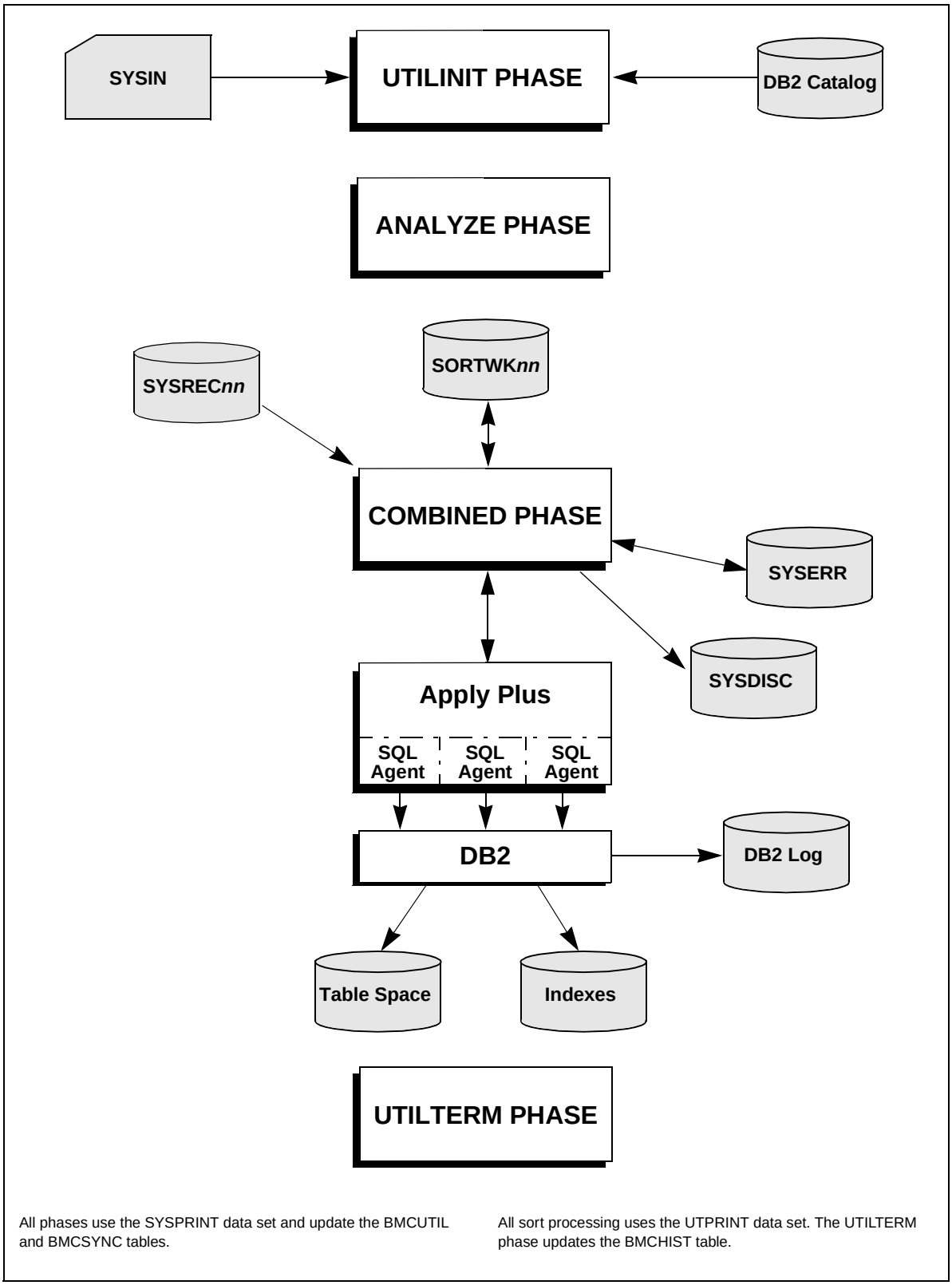


Figure 1-3 SQLAPPLY Processing Phases



BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS Data Sets

LOADPLUS uses the data sets specified by the ddnames that Table 1-1 describes. You can override most of the ddnames or ddname prefixes by using LOAD command options. Except when noted, the data sets can be allocated to a tape unit. See “LOADPLUS DD Statements” on page 4-8 for more information about how to specify the DD statement names and data set allocation sizes, and when the data sets are required.

Table 1-1 LOADPLUS Data Sets (Part 1 of 3)

Data Set	Description
SYSIN	SYSIN is the input data set that contains the LOAD command.
SYSREC*nn*	SYSREC*nn* is the input data set that contains the data that you are loading. The *nn* is required only if you specify multiple input data sets. LOADPLUS uses the data set in the PRELOAD phase or COMBINED phase. At the end of the PRELOAD or COMBINED phase, LOADPLUS deallocates the data set. **Note**: If you are using RECFM=U, or have concatenated an uninitialized data set in your SYSREC DD (which results in RECFM=U), only one record per block is processed and processing will probably not be as anticipated. As an alternative to specifying your SYSREC*nn* data set in a DD statement, you can have LOADPLUS dynamically allocate the data set.
SORTWK*nn*	SORTWK*nn* is the work data set that is used by BMCSORT. The data set is used in the PRELOAD and LOAD phases for a two-phase load, and in the COMBINED phase for a single-phase load. You cannot allocate the sort work files as VIO data sets or tape data sets. You must allocate each individual work file on a single DASD unit. As an alternative to specifying the SORTWK*nn* data set in a DD statement, you can have LOADPLUS or BMCSORT dynamically allocate the data set.
SYSUT1*nn*	SYSUT1*nn* is the data set that contains index key entries. This data set becomes the input data set for the index-building process. The *nn* is required only if you specify multiple data sets. To free the device, LOADPLUS deallocates the SYSUT1 data set after building the index. LOADPLUS uses this data set in the following ways: • For a two-phase load, LOADPLUS uses the data set in the PRELOAD and LOAD phases. • For a single-phase LOAD RESUME YES SHRLEVEL NONE job, this data set is not required unless you want to be able to restart the job. If a SYSUT1 data set is available for this type of load, LOADPLUS writes to the data set in the COMBINED phase and reads the data set only on restart. • For a single-phase LOAD REPLACE job or for a SQLAPPLY load, LOADPLUS does not require this data set and does not use it if specified. As an alternative to specifying the SYSUT1*nn* data set in a DD statement, you can have LOADPLUS dynamically allocate the data set.

BMC Software, Inc., Confidential and Proprietary Information

Table 1-1 LOADPLUS Data Sets (Part 2 of 3)

Data Set	Description
SORTOUT*n*	For a two-phase load, SORTOUT*n* is the data set that contains the output of the PRELOAD phase. This data set becomes the input data set to the LOAD phase. The *n* is required only if you specify multiple data sets. LOADPLUS uses the data set during the PRELOAD phase through the end of the LOAD phase. After the table space is loaded, LOADPLUS deallocates the data set to free the device. For a single-phase LOAD RESUME YES SHRLEVEL NONE job with ORDER YES specified, this data set is not required unless you want to be able to restart the job. If a SORTOUT data set is available for this type of load, LOADPLUS writes the clustering index and new row images to the data set in the COMBINED phase. LOADPLUS reads this data set only on restart. For the following types of load jobs, LOADPLUS does not require this data set and does not use it if specified: • SQLAPPLY • single phase LOAD REPLACE or LOAD RESUME NO • single phase LOAD RESUME YES SHRLEVEL NONE ORDER NO • single phase LOAD RESUME YES SHRLEVEL NONE with PART REPLACE As an alternative to specifying the SORTOUT*n* data set in a DD statement, you can have LOADPLUS dynamically allocate the data set.
SYSIDCIN	SYSIDCIN is the input data set that contains the IDCAMS command statements that LOADPLUS uses to redefine user-defined (VCAT-defined) data sets. These statements include statements to redefine the VCAT-defined staging data sets that LOADPLUS uses when running a load with LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE. LOADPLUS reads, parses, and partially verifies the SYSIDCIN data set in the UTILINIT phase. LOADPLUS issues the IDCAMS commands prior to loading the VCAT-defined data sets.
SYSDISC	SYSDISC is the data set that contains the discarded input records. LOADPLUS uses this data set in the PRELOAD phase for two-phase load and the COMBINED phase for single-phase processing. At the end of the PRELOAD or COMBINED phase, LOADPLUS deallocates the data set. As an alternative to specifying the SYSDISC data set in a DD statement, you can have LOADPLUS dynamically allocate the data set.
SYSERR	SYSERR is the data set that contains information about errors. LOADPLUS uses this data set in the PRELOAD phase through the end of the LOAD phase for two-phase load, and throughout the COMBINED phase for single-phase processing. As an alternative to specifying the SYSERR data set in a DD statement, you can have LOADPLUS dynamically allocate the data set.
Copy data sets	LOADPLUS creates one or more output copy data sets. These output data sets contain a DSN1COPY or image copy. Depending on the option that you choose, LOADPLUS creates either a standard image copy or DSN1COPY *after* it loads the data into the table, or an inline image copy *as* it loads the data into the table. After completing the copy, LOADPLUS deallocates the data set to free the device unless you specified VOL=(,RETAIN) in your JCL. Copy data sets should not be temporary data sets. As an alternative to specifying your a copy data set name in a DD statement, you can have LOADPLUS dynamically allocate the data set. LOADPLUS provides the following default names for each type of copy data set. This book uses the default name to refer to that type of copy data set. You can use a different name by changing it in the installation options module or overriding it by using options on the LOAD command. For more information, see “Copy Data Sets” on page 4-9. • BMCCPY*nn*—local primary • BMCCPZ*nn*—local backup • BMCRCY*nn*—remote primary • BMCRCZ*nn*—remote backup

BMC Software, Inc., Confidential and Proprietary Information

Table 1-1 LOADPLUS Data Sets (Part 3 of 3)

Data Set	Description
SYSPRINT	SYSPRINT is the output data set that contains LOADPLUS messages. All phases use this data set.
UTPRINT	UTPRINT is the data set that indicates that sort messages should be reported. However, the actual messages for each sort process appear in separate SYS*nnnnn* data sets, where *nnnnn* is a system-assigned sequential number. This data set is required by any phase that performs a sort. **Warning!** JES3 users should be aware of a limitation within JES3 that does not allow concurrent tasks to share SYSOUT data sets. (See IBM APAR OY23946 for a full description of this limitation.) This limitation means that you cannot use additional sort routine reporting DDs (other than UTPRINT) if they are defined as JES3 SYSOUT data sets and when LOADPLUS is multitasking its sort activity. If you attempt to use a nonsupported DD, you risk S1FB abends when concurrent sort tasks are running. JES3 version 4.2.1 users should also refer to IBM APARs OW00111 and OY63725.
DB2 data sets	LOADPLUS dynamically allocates the DB2 data sets (table spaces or index spaces) that you are loading. Therefore, you do not need to specify them in your JCL.

BMC Software, Inc., Confidential and Proprietary Information

Chapter 2 Operational Considerations

This chapter presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Required Authorization

Using the LOADPLUS product requires that you have the appropriate authorization within DB2 and through your system security package, such as RACF. You need sufficient authorization to access resources and perform the tasks that are accomplished during LOADPLUS processing.

Authorization Verification Mechanisms

Table 2-1 describes how LOADPLUS verifies authorization based on the authorization verification mechanism that is available for your system.

Table 2-1 Authorization Verification Mechanisms

Available Authorization Verification Mechanism	LOADPLUS Actions
DB2 security exit	LOADPLUS uses the DB2 security exit to verify the utility authorization. IBM provides a sample DB2 security exit that uses RACF.
CA-ACF2 for DB2 or CA-Top Secret for DB2 security product (Computer Associates)	LOADPLUS uses either Computer Associates product with any version of DB2. LOADPLUS detects the presence of the product in the subsystem where LOADPLUS is running. **Note**: You must perform *both* of the following steps: • Contact Computer Associates and request the appropriate APAR to enable SAF support in CA-ACF2 or CA-Top Secret. • Specify ACFORTSS=YES in your installation options module.
none	LOADPLUS uses the standard DB2 method to check security.

DB2 Authority

LOADPLUS requires certain authorizations for all types of load jobs. In addition, if you run LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY (also referred to as a SQLAPPLY load), LOADPLUS passes processing to Apply Plus, which requires additional authorizations. This section describes the authorizations required to run LOADPLUS.

BMC Software, Inc., Confidential and Proprietary Information

All LOADPLUS Jobs

To run LOADPLUS for any type of load job, you must have the following authorizations:

- sufficient DB2 authority to execute all LOADPLUS plans
- one of the following authorizations with your primary or secondary authorization ID:

  — SYSADM or SYSCTRL authority

  — DBADM or DBCTRL authority for the database that contains the named tables and their associated indexes

  — LOAD authority for the database that contains the named tables and their indexes

  — LOAD authority for the named tables, or LOAD authority for all the tables of the table space if you specify LOAD REPLACE

  LOAD authority means that the table's creator ID matches your authorization ID or an ID in your secondary authorization ID list.

SQLAPPLY Load Jobs

If you run a SQLAPPLY load, LOADPLUS passes processing to Apply Plus during the COMBINED phase. Apply Plus requires the following DB2 authorizations. If you need additional information, see the *Apply Plus Reference Manual*. The GRANT member of the *HLQ*.CNTL installation data set (where *HLQ* is the high-level qualifier that is set during installation) contains sample authorization statements.

Note: You can use secondary authorization IDs to limit access as necessary for your site.

The following DB2 authorizations are normally granted during Apply Plus installation:

- execute privileges (EXECUTE) for the Apply Plus restart package
- execute privileges (EXECUTE) for the plan that Apply Plus uses to access its own restart table and the catalog
- appropriate table privileges (SELECT, INSERT, UPDATE, or DELETE) for the Apply Plus restart table

BMC Software, Inc., Confidential and Proprietary Information

The following DB2 authorizations are normally granted after Apply Plus installation:

- bind privileges with the add option (BINDADD) for the plans and packages that Apply Plus creates during apply processing
- create privileges (CREATE IN) for the collections that Apply Plus creates
- if any of the tables that you are loading have referential integrity (RI) relationships, DISPLAY system privilege or DISPLAYDB authority on the databases that contain the tables in the RI relationship, including those tables that are not participating in the load
- INSERT privileges on the tables that you are loading

If you are running Apply Plus version 1.6, Apply Plus provides several ways to grant these privileges. Some techniques avoid granting bind privileges to the user ID that runs Apply Plus. For more details, see the *Apply Plus Reference Manual*.

Data Set Authorization

Because LOADPLUS does not run as part of the DB2 subsystem, to use LOADPLUS, you must have system authorization that is equivalent to the authorization that is required by DB2. If RACF or a similar system security package protects both the underlying data sets and the ICF (Integrated Catalog Facility) catalog of a table or index space, you must have the minimum levels of authority as shown in Table 2-2.

Table 2-2 Minimum Levels of Authority That LOADPLUS Requires

Table or Index Space Definition	To Access, Update, and Define DB2 Data Sets	To Access and Update the ICF Catalog
VCAT-defined	CONTROL	UPDATE
STOGROUP-defined	ALTER or CONTROL	UPDATE or CONTROL

BMC Software, Inc., Confidential and Proprietary Information

If you establish authority at a node lower than the highest node, you must have this same authority (as shown in Table 2-2 on page 2-5) for the following data sets. LOADPLUS uses these data sets during the renaming process for LOAD REPLACE SHRLEVEL CHANGE and LOAD REPLACE SHRLEVEL REFERENCE.

- *vcat*.BMCDBD
- *vcat*.BMCDBC
- *vcat*.OLDDBD
- *vcat*.OLDDBC

Using RACF

If you are using RACF and LOADPLUS is installed with option OPNDB2ID=YES, the user who is running LOADPLUS is not required to have these authorizations.

Note: OPNDB2ID=YES tells LOADPLUS to use the DB2 RACF ID instead of the RACF ID of the user.

Using Security Packages Other Than RACF

The following procedure illustrates one method for granting these data set authorizations when your site uses a system security package other than RACF:

Step 1 Associate users with a security group.

Step 2 Grant EXECUTE authority on the LOADPLUS product program (AMUUMAIN) to the security group.

Step 3 Grant the data set authorizations that are described in Table 2-2 on page 2-5 to the LOADPLUS product program.

BMC Software, Inc., Confidential and Proprietary Information

Number of DB2 Threads Used by LOADPLUS

For most load jobs, the maximum number of batch threads that LOADPLUS uses concurrently is six per job. LOADPLUS uses up to six threads during the UTILINIT phase and two threads for the duration of the job.

For a SQLAPPLY job, however, processing passes to Apply Plus during the COMBINED phase. Apply Plus uses a minimum of six threads concurrently, but can use up to two plus the value that you specified for the LOAD command option APMAXAGENTS (or the LOADPLUS installation option APMXAGNT).

Serialization and Concurrency

This section discusses concurrency issues and object status requirements, which can vary with the command statement specifications.

LOADPLUS and Other BMC Software Utilities for DB2

All BMC Software utility products use the BMCUTIL table to control the use of utility IDs, identifiers of BMC Software utility runs. The BMCUTIL table requires that each BMC Software utility product have a unique ID for restart purposes. For more information, see “BMCUTIL Table” on page B-3.

BMC Software utility products use the BMCSYNC table to coordinate access to DB2 objects. DB2 objects that participate in a BMC Software utility job are registered in the BMCSYNC table. When each object is registered, the registering utility assigns a sharing level to control access to that object from other BMC Software utilities. For partitioned DB2 spaces, registration is performed at the partition level. For the values that LOADPLUS assigns to the objects that it registers, see “BMCSYNC Table” on page B-4.

This use of the BMCSYNC table allows multiple BMC Software utilities or multiple instances of a single utility to operate concurrently on different partitions of the same DB2 space if no nonpartitioning indexes are involved. In addition, some BMC Software utilities can operate concurrently on the same object or partition. See “Serialization and Concurrency” and the BMCSYNC table description in the reference manual for each BMC Software utility to determine its serialization and concurrency issues.

Warning! Do not run an IBM utility that attempts to manipulate data within the same objects on which a BMC Software utility is currently processing.

BMC Software, Inc., Confidential and Proprietary Information

Depending on the setting of the LOCKROW installation option, LOADPLUS uses MVS enqueues or SQL LOCK TABLE statements to serialize the BMCSYNC and BMCUTIL tables. For information about the LOCKROW option, see Appendix A, “LOADPLUS Installation Options.”

Object Status

This section describes the initial status requirements for each type of load. The section also describes how LOADPLUS changes the status of the objects during load processing and after load processing completes.

Initial Status

The database associated with the objects that are participating in the load must have an initial status of read/write (RW). The initial status that LOADPLUS requires of the table space and index spaces depends on the type of load that you are running.

LOAD REPLACE or LOAD RESUME YES PART REPLACE

When you specify LOAD REPLACE or LOAD RESUME YES PART REPLACE, LOADPLUS requires that the table space and index spaces that are participating in the load have an initial status of RW, read-only (RO), or utility-only (UT). In addition, table and index spaces can be in any restrictive status with the exception of the following restriction and the restrictions that Table 2-3 describes.

Note: Objects that are associated with a table whose table space or index spaces are created with DEFINE NO and are not already materialized must be in RW status and cannot be in any restrictive status.

Table 2-3 LOAD REPLACE Restrictive Status Restrictions (Part 1 of 2)

Status	Restrictions
RESTP	Table and index spaces cannot be in restart pending (RESTP) status.
REORP	If you are running a LOAD RESUME YES PART REPLACE job, nonparticipating partitions cannot be in REORG pending (REORP) status. If any of the partitions are in REORP status, you must replace all partitions that are in REORP status.

BMC Software, Inc., Confidential and Proprietary Information

Table 2-3 LOAD REPLACE Restrictive Status Restrictions (Part 2 of 2)

Status	Restrictions
LPL or WEPR	If the objects are in LPL or WEPR status, they must be in RW,LPL or RW,WEPR status.
	For a table space that is in LPL or WEPR status, the LOAD command must include one of the following options: • COPY YES with REGISTER for one or more copies • COPY NO COPYPEND NO **Note**: If the value of the COPYPEND installation option is (YES,ENFORCE), you cannot specify COPYPEND NO on your LOAD command.
	If a table space that is in LPL or WEPR status contains a table that is defined with referential constraints, the value of CHECKPEND must be NO.
	An index that is in LPL or WEPR status must not be in informational COPY pending (ICOPY) status.
	If an index that is in LPL or WEPR status was created with COPY YES, you must specify COPY NO COPYPEND NO on the LOAD command. **Note**: If the value of the COPYPEND installation option is (YES,ENFORCE), you cannot specify COPYPEND NO on your LOAD command.
	You can only specify LOAD RESUME YES PART REPLACE if no associated nonpartitioning indexes are in LPL or WEPR status.

LOAD RESUME YES (except SQLAPPLY)

For LOAD RESUME YES load jobs other than SQLAPPLY loads, LOADPLUS requires that the table space and index spaces that are participating in the load have an initial status of RW, RO, or UT. In addition, the table spaces and index spaces cannot be in any of the following restrictive statuses. (For status requirements for partitions when you specify LOAD RESUME YES PART REPLACE, see the requirements for LOAD REPLACE in the preceding section.)

Note: Objects that are associated with a table whose table space or index spaces are created with DEFINE NO and are not already materialized must be in RW status and cannot be in any restrictive status.

- auxiliary CHECK pending (ACHKP)
- advisory restart pending (AREST)
- group RECOVER pending (GRECP)
- logical page list (LPL)
- page set REBUILD pending (PSRBD)
- REBUILD pending (RBDP)
- logical part REBUILD pending (RBDP*)
- RECOVER pending (RECP)
- refresh pending (REFP)
- restart pending (RESTP)
- REORG pending (REORP)
- write error page range (WEPR)

BMC Software, Inc., Confidential and Proprietary Information

If objects participating in the load are in any of these statuses, LOADPLUS issues message BMC50263E and terminates.

LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY

For LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY jobs, LOADPLUS requires that the table space and index spaces that are participating in the load have an initial status of RW. In addition, the table spaces and index spaces cannot be in any restrictive status. If any objects participating in the load do not meet these requirements, LOADPLUS issues message BMC50263E and terminates.

Status Changes

Table 2-4 describes how the object status changes during load processing for each type of load. The statuses that this table describes apply to all objects that are participating in the load.

Table 2-4 Summary of Status Changes during Load Processing (Part 1 of 2)

Load Type	Initial Status	PRELOAD Phase	LOAD Phase	COMBINED Phase	UTILTERM Phase
LOAD REPLACE SHRLEVEL NONE two phase	RW, RO, or UT[a,b]	unchanged	STOP[c]	not applicable	reset to initial status[b]
LOAD RESUME YES SHRLEVEL NONE two phase	RW, RO, or UT[b]	STOP[c,d]	STOP[c,d]	not applicable	reset to initial status[b]
SHRLEVEL NONE single phase	RW, RO, or UT[b]	not applicable	not applicable	STOP[c,d]	reset to initial status[b]
LOAD REPLACE SHRLEVEL REFERENCE two phase	RW, RO, or UT[a,b]	unchanged	RO	not applicable	• STOP when the renaming process starts • reset to initial status after renaming process complete[b]
LOAD REPLACE SHRLEVEL REFERENCE single phase	RW, RO, or UT[a,b]	not applicable	not applicable	RO	• STOP when the renaming process starts • reset to initial status after the renaming process completes[b]

BMC Software, Inc., Confidential and Proprietary Information

Table 2-4 Summary of Status Changes during Load Processing (Part 2 of 2)

Load Type	Initial Status	PRELOAD Phase	LOAD Phase	COMBINED Phase	UTILTERM Phase
LOAD REPLACE SHRLEVEL CHANGE two phase	RW, RO, or UT[a,b]	unchanged	unchanged	not applicable	• STOP when the renaming process starts • reset to initial status after the renaming process completes[b]
LOAD REPLACE SHRLEVEL CHANGE single phase	RW, RO, or UT[a,b]	not applicable	not applicable	unchanged	• STOP when the renaming process starts • reset to initial status after the renaming process completes[b]
LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY (always uses single-phase architecture)	RW	not applicable	not applicable	unchanged	unchanged

[a] If the objects are in LPL or WEPR pending status, they must be in RW status.

[b] If you specify SKIPIX SIX or SKIPIX NUSIX, the nonparticipating indexes can be in any initial status. They are placed in a REBUILD pending status (RBDP, RBDP*, or PSRBD) after the load job completes.

[c] STOP status ensures that LOADPLUS has exclusive use of the space while it is being loaded.

[d] When operating on partitioned objects with RESUME YES and the PART option specified, LOADPLUS stops and starts only those partitions that are specified in the LOAD command.

Status after Load

After the loading of the data is complete, LOADPLUS starts all table spaces and participating index spaces with the same status that they had at the start of the load job. This status change occurs at different times, depending on the SHRLEVEL option:

- For SHRLEVEL NONE, LOADPLUS changes the status at the beginning of the UTILTERM phase.
- For LOAD REPLACE SHRLEVEL REFERENCE and LOAD REPLACE SHRLEVEL CHANGE, LOADPLUS changes the status during the UTILTERM phase after the renaming process is complete.

In addition, LOADPLUS takes the actions shown in Table 2-5 on page 2-12, depending on the load options.

BMC Software, Inc., Confidential and Proprietary Information

Note: Note the following exceptions:

- If the table space is in both COPY and CHKP statuses, the table space status remains unchanged and LOADPLUS issues message BMC50380I.
- If you specify LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY, LOADPLUS does not set pending statuses.

Table 2-5 Statuses Set or Reset by LOADPLUS

LOAD Command Specified	LOADPLUS Actions
any load	sets ICOPY for all participating indexes that are defined with COPY YES
SKIPIX SIX or SKIPIX NUSIX	with the exception described below, sets all secondary (SIX) or all nonunique secondary indexes (NUSIX) to one of the following REBUILD pending statuses: • RBDP • RBDP* • PSRBD **Exception**: If you specify RESUME YES (without PART REPLACE) and 0 rows are loaded during the load job, LOADPLUS leaves the nonparticipating indexes in the same status that they were in at the start of the job.
COPY YES	resets the COPY status
COPY NO COPYPEND YES[a]	places the table space in COPY status
COPY NO COPYPEND NO[b] or LOG NO NOCOPYPEND[b]	does not place the table space in COPY status If the table space was in COPY status before the load, LOADPLUS does not reset the status.
ENFORCE NO CHECKPEND YES[c]	places the table space in CHKP status if table check constraints exist
CHECKPEND YES[c]	places the table space in CHKP status if referential constraints exist
LOAD REPLACE	resets the following statuses of the table space and participating index spaces if the objects were in that status prior to the load: • RECP • RECP* • REFP • RBDP • RBDP* • GRECP • PSRCP • PSRBD • REORP

[a] If the value of the COPYPEND installation option is (NO,ENFORCE), you cannot specify COPYPEND YES on the LOAD command.

[b] If the value of the COPYPEND installation option is (YES,ENFORCE), you cannot specify COPYPEND NO or NOCOPYPEND on the LOAD command.

[c] If the value of the CHEKPEND installation option is (NO,ENFORCE), you cannot specify CHECKPEND YES on the LOAD command.

BMC Software, Inc., Confidential and Proprietary Information

Multiple Data Set DB2 Objects

When loading multiple data set table spaces and index spaces that are VCAT-defined, you must perform one of the following tasks:

- Predefine the required data sets.
- Specify REDEFINE YES and include a SYSIDCIN data set that defines the required data sets.

If you do not perform one of these tasks, the LOADPLUS job fails.

For STOGROUP-defined multiple data sets, LOADPLUS defines additional data sets if needed for expansion, even if the value of the REDEFINE option is NO.

The PRELOAD phase of a two-phase load uses the data set allocations, the number of extents allowed, and DB2 restrictions to determine whether the DB2 object can hold the data that you are loading. If the DB2 object cannot hold the data, the PRELOAD phase issues message BMC51487E and terminates. However, the COMBINED phase of a single-phase load cannot make this prediction due to real-time loading of data. As a result, if the DB2 object is unable to hold the data that you are loading, LOADPLUS fails with an extent failure or similar condition instead of issuing message BMC51487E.

Dynamic Work File Allocation

When you activate dynamic work file allocation, either through the command syntax or installation defaults, LOADPLUS calculates the optimal size and number of load data files, index work files, sort work files, discard files, error files, and copy files. LOADPLUS then allocates these files for you.

Note: You can also have LOADPLUS dynamically allocate your input data sets by using the INDSN command option. However, LOADPLUS does not analyze requirements for these data sets and does not use the standard LOADPLUS dynamic allocation options that this section describes. The considerations in this section do not apply to dynamic allocation of input data sets. See “INDSN” on page 3-34 for more information about dynamically allocating input data sets.

BMC Software, Inc., Confidential and Proprietary Information

In a two-phase load, dynamic allocation takes place at the beginning of the PRELOAD phase. For a single-phase load, this allocation occurs at the beginning of the COMBINED phase.

Note: Sort work files can also be dynamically allocated by BMCSORT. See "Allocating Sort Work Data Sets" for additional information.

Allocating Sort Work Data Sets

Sort work data sets are allocated in one or more of the following ways:

- with SORTWK DD statements in your JCL
- dynamically by LOADPLUS
- dynamically by BMCSORT

This section describes how you can control this allocation.

Using SORTWK DD Statements

Under the following circumstances, LOADPLUS uses any SORTWK DD statements that you specify in your JCL to allocate your sort work data sets:

- You specify ACTIVE NO for dynamic allocation of SORTWK data sets.
- You specify ACTIVE YES IFALLOC USE for dynamic allocation of SORTWK data sets.

If BMCSORT determines that it needs more sort work space than you have allocated, additional sort work data sets are dynamically allocated. See "Dynamically Allocating Sort Work Data Sets."

For more information about specifying SORTWK DD statements in your JCL, see "SORTWKnn Data Sets" on page 4-14.

Dynamically Allocating Sort Work Data Sets

Dynamic allocation takes place when one of the following actions occurs:

- You specifically request dynamic allocation through command or installation options.

- BMCSORT determines that it needs more sort work space than other allocation methods provide (allocated in your JCL, dynamically allocated by LOADPLUS, or both).

BMC Software, Inc., Confidential and Proprietary Information

Several factors affect this dynamic allocation. Table 2-6 describes the results of combinations of these factors.

Table 2-6 Factors That Affect Dynamic Allocation of Sort Work Data Sets

LOADPLUS Dynamic Allocation Options	SORTNUM or SORTDEVT	BMCSORT DYNALOC Option	Results
ENUMROWS[a] ACTIVE YES IFALLOC FREE[b]	not applicable[c]	not applicable[c]	LOADPLUS allocates 12 sort work data sets. The information that you provide with ENUMROWS affects the size of the sort work data sets.
ACTIVE NO	SORTNUM *n* or SORTDEVT	value of third parameter is ON or OFF[d]	If you specify a SORTNUM value that is greater than 32, BMCSORT allocates the number of data sets that it determines it needs, up to the specified number of data sets. Otherwise, BMCSORT allocates up to 32 data sets.
	not specified	value of third parameter is ON	BMCSORT allocates the number of data sets that it determines it needs up to 32.
	not specified	value of third parameter is OFF	BMCSORT does not allocate any sort work data sets and attempts to perform sort processing in memory.

[a] If you do not specify ENUMROWS, dynamic allocation of sort work data sets is done by BMCSORT as if you specified ACTIVE NO.

[b] If you specify IFALLOC USE, LOADPLUS allocates sort work data sets in addition to the ones specified in your JCL for a total of 12.

[c] The values for SORTNUM, SORTDEVT, and the DYNALOC option apply for this scenario only if BMCSORT determines that it needs more than 12 sort work data sets. In this case, these options affect BMCSORT processing *only for the additional data sets* as if you specified ACTIVE NO. Refer to the other rows in this table to determine how BMCSORT allocates the additional data sets.

[d] If the value of the third parameter of the BMCSORT DYNALOC option is OFF, specifying SORTNUM or SORTDEVT changes this value to ON.

Allocating Copy Data Sets

For the copy data sets, LOADPLUS determines the size of the dynamically allocated files based on the following information:

- for LOAD REPLACE, the value that you specify for ENUMROWS
- for LOAD RESUME YES, the high-used relative byte address (RBA) of the table space that you are loading and the value that you specify for ENUMROWS

BMC Software, Inc., Confidential and Proprietary Information

You can decide when to delete the dynamically allocated copy data sets by specifying an expiration date (with the EXPDT command or installation option) or by specifying a retention period (with the RETPD command or installation option). For more information about these commands, see “EXPDT” on page 3-151 and “RETPD” on page 3-151.

Running in a Worklist Environment

When LOADPLUS runs in a worklist environment, it ignores the ACTIVE option in your installation module. LOADPLUS dynamically allocates your work files only if the invoking product (DASD MANAGER PLUS for DB2, CATALOG MANAGER for DB2, or CHANGE MANAGER for DB2) supplies the ACTIVE YES syntax.

Generating Data Set Names

You can use the DSNPAT option to specify a pattern to generate a unique data set name. For copy files, you can also specify a generation data group (GDG) name as the data set name.

Names Created with DSNPAT

The data set name pattern (DSNPAT) installation or command option allows you to specify text and variable data for building data set names. If you cannot construct a data set name that meets your organization’s standards by using text and the supplied variables, LOADPLUS provides an exit point that allows you to create your own variables for use with DSNPAT. Sample exits written in assembler, COBOL, C, and LE C are provided in Appendix F, “LOADPLUS User Exits” as well as in the *HLQ*.CNTL library. You can find additional information about the DSNPAT option on page 3-145.

The pattern that you specify in your DSNPAT option must allow LOADPLUS to generate unique data set names. For multiple SORTOUT and SYSUT1 files, you must include the &DDNAME variable to generate unique names. For copy data sets, you might need to include additional variables, such as &VCAT, &DATEJ, or &TIME4, to generate unique names across multiple loads. If LOADPLUS encounters nonunique data set names, the job terminates.

BMC Software, Inc., Confidential and Proprietary Information

Generation Data Group Names

You can use GDG names for your dynamically allocated copy data sets. Each DDTYPE must have a different GDG base.

GDG Name Format

The GDG format that you use to construct data set names is the same as the format that you use in your JCL when you allocate copy data sets through DD statements. Simply append the generation number in parentheses. The open parenthesis tells LOADPLUS that the pattern is a GDG name. The generation number must be an integer from 1 through 255.

An example of a GDG name is &UTILPFX.&DDNAME.(+1). If you are using a substitution variable as the last variable before the open parenthesis, you must include a period before the open parenthesis.

GDG Base

LOADPLUS has the following requirements for the number of GDG bases that you specify:

- Each DDTYPE must have a different GDG base.
- If you specify COPYLVL PART on the LOAD command, each partition must have a different GDG base.

If the base does not exist, LOADPLUS creates it for you, using everything in the pattern up to the open parenthesis as the base name.

When defining the base, LOADPLUS uses the values of the following options:

- The GDGLIMIT installation or command option allows you to specify the number of generations to keep.

- If the GDGLIMIT value is exceeded, the GDGEMPTY installation option tells the system to uncatalog either all preexisting generations of this data set or only the oldest generation.

- The GDGSCRATCH installation option tells the system whether to delete the entry that was just uncataloged from the volume's table of contents (VTOC).

For more information, see the installation option descriptions in Appendix A, starting on page A-29, and the command option description for "GDGLIMIT" on page 3-152.

BMC Software, Inc., Confidential and Proprietary Information

Specifying ddname Prefixes

If you specify more than one ddname prefix for dynamic allocation, the prefix must be different enough for LOADPLUS to differentiate one prefix from another. *Different enough* means that these prefixes must be different, *and*, if they are different only because one prefix has additional trailing bytes, then these trailing bytes must contain at least one nonnumeric byte. For example, the first set of prefixes that follow is different enough, but the second set is not:

- acceptable:

```
BMCRD
BMCRDWK
```

- not acceptable:

```
BMCRD
BMCRD11
```

Reaching the MAXTAPE Limit

When UNIT and THRESHLD specifications require that LOADPLUS dynamically allocate tape units, allocation occurs in the following priority order:

1. LOADPLUS attempts to allocate the greatest number of tape units required that will optimize multitasking.

2. If this number of tape units exceeds the MAXTAPE value, LOADPLUS decreases the multitasking level until the number of tape units required is less than or equal to the MAXTAPE value.

3. If the minimum number of tape units required exceeds the MAXTAPE value, LOADPLUS issues a message and terminates.

The value that you specify for the MAXTAPE option includes the units that are required for image copy data sets.

BMC Software, Inc., Confidential and Proprietary Information

Changing Options on Restart

LOADPLUS has certain restrictions governing changes that you make to dynamic work file allocation options on restart. You might find that the values provided for some options need to be changed for dynamic work file allocation to complete properly. For example, specifying an invalid UNIT or overly restrictive MAXTAPE value can cause LOADPLUS to terminate. The following restrictions apply to dynamic allocations on restart:

- You cannot change the value for the ACTIVE option or the ENUMROWS option on any restart.
- Changing any option on restart such that it results in different ddnames or a different number of DDs than the original option can produce an error. If you need to change the number of SORTOUT or SYSUT1 work files, you must resubmit the job with a parameter of NEW.
- If you change the value of other dynamic work file allocation options, specify RESTART(PHASE).

Frequently Asked Questions about Dynamic Work File Allocation

Question: What is dynamic work file allocation?

Answer: With dynamic work file allocation, LOADPLUS analyzes the requirements for the load (SORTOUT), work (SYSUT1), sort work (SORTWK), discard (SYSDISC), error (SYSERR), and copy files for the current execution. LOADPLUS uses the optimal arrangement of these files, dynamically allocating them to produce the highest multitasking level possible. Using the dynamic work file allocation options, you do not need to specify any DD statements for these files in the JCL for your LOADPLUS job.

Question: How do I enable and use dynamic work file allocation?

Answer: To quickly and simply enable and use dynamic work file allocation, make the following specifications:

- Specify ENUMROWS on your LOAD command.
- For any of the DDTYPEs that you want to dynamically allocate, specify ACTIVE YES in your LOAD command or in your installation options.
- To test this feature on existing LOADPLUS jobs without changing your JCL, use the IFALLOC FREE option for each of the DDTYPEs. You can specify this option either in your LOAD command or in your installation options.

BMC Software, Inc., Confidential and Proprietary Information

For additional options that you can use with dynamic work file allocation, see “Dynamic Work File Allocation Options” on page 3-133.

Question: How do I make my smaller work files go to DASD and my larger work files go to TAPE?

Answer: Perform the following steps.

Note: Sort work files cannot go to tape.

Step 1 For each DDTYPE, specify the UNIT option with a DASD device type for the first parameter and a tape device type for the second. The following example illustrates a typical UNIT specification:

```
UNIT(WORK,TAPE)
```

Step 2 For each DDTYPE, specify the THRESHLD option with the largest work file size (in kilobytes) that you want on DASD. This forces LOADPLUS to send the larger work files to tape. The following example supplies LOADPLUS with a number approximately equal to one 3380 volume:

```
THRESHLD 1890000
```

Step 3 Specify the MAXTAPE option to control the maximum number of tape devices that LOADPLUS dynamically allocates. The following example tells LOADPLUS to dynamically allocate no more than three tape devices:

```
MAXTAPE 3
```

Note: If work files go to tape, the MAXTAPE value might restrict the multitasking level of LOADPLUS.

You can find additional information about these options in “Dynamic Work File Allocation Options” on page 3-133.

Question: How do I create SMS-managed data sets with dynamic work file allocation?

Answer: For any DDTYPE, specify SMS YES in your LOADPLUS command or in your installation options. You must also specify class names using the DATACLAS, MGMTCLAS, and STORCLAS options. SMS YES instructs LOADPLUS to use these SMS classes for dynamic work file allocation.

Question: What if my SMS allocation routines use the UNIT parameter to influence data set allocation?

BMC Software, Inc., Confidential and Proprietary Information

Answer: When you specify SMSUNIT YES, LOADPLUS passes the UNIT option to SMS allocation in addition to passing the SMS class options and other normally passed options. When you specify SMSUNIT NO, LOADPLUS does not pass the UNIT option.

Question: How do I delete the work files that LOADPLUS dynamically allocates?

Answer: Specify DELETEFILES YES on your LOAD command. After the load job completes successfully, LOADPLUS automatically deletes the work files that it dynamically allocated and those allocated in your JCL, except SYSDISC. If you do not specify DELETEFILES YES, you must manually delete the dynamically allocated work files when your load completes successfully. DELETEFILES YES does not apply to image copy data sets that LOADPLUS dynamically allocates.

You can also have LOADPLUS delete SYSDISC, if no records were written to this data set during the load process, by specifying SYSDISC YES. If LOADPLUS wrote records to SYSDISC during the load process, LOADPLUS does not automatically delete SYSDISC, allowing you to correct the discarded records that it contains and resubmit the job. You must manually delete SYSDISC in this case.

The SYSPRINT from your LOADPLUS job contains a report of the dynamically allocated work files. When you need to manually delete work files, you can use this report to determine which files to delete.

These preferences can also be specified at installation with the DELFILES installation option. For more information about these options, see “DELETEFILES” on page 3-57 or the DELFILES installation option on page A-16.

Question: What is the performance cost for using dynamic work file allocation?

Answer: The time to perform analysis (which is comparable to ANALYZE ONLY or ANALYZE PAUSE) and the time to perform dynamic allocation of the work files increases the elapsed time of the run.

Question: What is the benefit of using dynamic work file allocation?

Answer: Because LOADPLUS automatically determines the optimal size and number of work files, dynamically allocates the work files, and (optionally) automatically deletes the work files, you spend less time performing analysis to set up optimized JCL for LOADPLUS jobs. Dynamic work file allocation also prevents you from having to modify the JCL for the LOADPLUS job as DB2 objects change size or structure over time.

BMC Software, Inc., Confidential and Proprietary Information

Finding More Information

For more information about dynamic allocation, see the following sections:

Additional Dynamic Allocation Information	Reference
syntax associated with dynamic allocation	"Dynamic Work File Allocation Options" on page 3-133
default dynamic allocation options	Appendix A, "LOADPLUS Installation Options."
user-defined variables for use in data set name patterns that LOADPLUS uses to determine data set names for dynamic allocation	Appendix F, "LOADPLUS User Exits."

LOADPLUS Work File Validity and Integrity Checks

LOADPLUS ensures that the work files it uses for load processing are valid by performing the following types of file verification:

- The first check determines whether a work file has been allocated as a temporary data set or with DISP=MOD.
- The second check ensures that the work file that LOADPLUS reads during the LOAD phase is the same one that was created during the PRELOAD phase.

Check for Data Set Attributes

This section describes how LOADPLUS responds when determining that your work files and copy data sets are temporary data sets.

LOADPLUS defines a temporary data set as one whose normal or abnormal DISP is defined with one of the following values:

- DELETE
- NEW,PASS
- OLD,PASS *and* the original status is not OLD

BMC Software, Inc., Confidential and Proprietary Information

SORTOUT, SYSERR, and SYSUT1 Data Sets

If LOADPLUS determines that your SORTOUT, SYSERR, or SYSUT1 data sets are temporary, LOADPLUS responds based on the value for the FILECHK installation option.

FILECHK=FAIL

If your installation options specify FILECHK=FAIL and if LOADPLUS finds that any of the SORTOUT, SYSERR, or SYSUT1 data sets are temporary, LOADPLUS terminates. On output, if the disposition is MOD, LOADPLUS resets it to empty.

If LOADPLUS is restarting in the LOAD phase and the data set is temporary due to its normal disposition, LOADPLUS overrides the normal disposition to KEEP to ensure against data loss after termination.

FILECHK=WARN

You can force LOADPLUS to process the SORTOUT, SYSERR, and SYSUT1 files in WARN mode by specifying FILECHK=WARN in the LOADPLUS installation options. In WARN mode, LOADPLUS issues a warning message for each work file that is allocated as a temporary data set, but continues load processing.

SYSDISC and Copy Data Sets

If your installation options specify FILECHK=FAIL and LOADPLUS finds that the SYSDISC work file or any of the copy data sets are temporary data sets, LOADPLUS does not fail, but continues processing and issues a warning message.

Note: The warning message for the copy files occurs at open time during the LOAD phase. If you run LOADPLUS in two steps by using the PRELOAD PAUSE option, you can code the DDs with DUMMY in the PRELOAD phase to avoid receiving message BMC50391E. This message states that LOADPLUS is unable to locate the copy data set.

BMC Software, Inc., Confidential and Proprietary Information

Check for Data Integrity

LOADPLUS performs a second check of the SORTOUT, SYSERR, and SYSUT1 work files to ensure data integrity.

Two-Phase Load

During the PRELOAD phase of two-phase load processing, LOADPLUS creates the SORTOUT, SYSERR, and SYSUT1 work files with header information about the current utility execution. During the LOAD phase, LOADPLUS checks this header information to ensure that it is the file it is expecting to load. If the header information does not match, LOADPLUS terminates processing and issues an error message.

Single-Phase Load

During the COMBINED phase of single-phase load processing, LOADPLUS writes and verifies the header information of the SYSERR file.

For SORTOUT and SYSUT1 files in a LOAD RESUME YES SHRLEVEL NONE job, LOADPLUS writes the header information during the COMBINED phase of the original load job. If you restart this job, LOADPLUS verifies the header information during the COMBINED phase of the restarted job. If the header information does not match, LOADPLUS terminates processing and issues an error message.

SHRLEVEL Considerations

The SHRLEVEL option specifies the level of access that DB2 has to the objects that you are loading during LOADPLUS processing.

- With SHRLEVEL NONE (the default), the objects that you are loading are stopped and unavailable during load processing.
- With SHRLEVEL REFERENCE, the objects that you are loading are available in read-only status during load processing.
- With SHRLEVEL CHANGE, the objects that you are loading are available in read/write status during load processing.

Use Table 2-7 on page 2-25 to determine which SHRLEVEL option to use with your LOAD command.

BMC Software, Inc., Confidential and Proprietary Information

Table 2-7 Determining Which SHRLEVEL Option to Use

Type of Access to Your Data During the Load Process	Additional Site or Application Requirements	SHRLEVEL Option
do not need access	not applicable	SHRLEVEL NONE
read-only access	• want very limited outage (which occurs during the renaming process) • want to replace existing data	LOAD REPLACE SHRLEVEL REFERENCE
read/write access	• want very limited outage (which occurs during the renaming process) • want a full replacement of the data in the object (or partition) Full replacement includes replacing inserts, updates, and deletes to the base object that occur during the load process.	LOAD REPLACE SHRLEVEL CHANGE
	• want *no* outage to the object • want to retain all insert, update, and delete operations that affect the table during the load process • do not want to replace existing data	LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY

The rest of this section discusses considerations for the SHRLEVEL REFERENCE and SHRLEVEL CHANGE options. For additional information about SHRLEVEL topics, see the following references:

- For information about DB2 and system authority, see “DB2 Authority” on page 2-3 and “Data Set Authorization” on page 2-5.
- For information about object statuses, see “Object Status” on page 2-8.
- For syntax information, see “SHRLEVEL” on page 3-19.
- For information about recovering from a failure or termination, see “Recovering the DB2 Object after Terminating or Canceling the Job” on page 4-34.

SHRLEVEL REFERENCE

When you specify SHRLEVEL REFERENCE, LOADPLUS places the object that you are loading in read-only status while it loads the data to a staging data set. After the load completes, LOADPLUS renames this staging data set, replacing the original object with the staging data set.

BMC Software, Inc., Confidential and Proprietary Information

During the load process, the original object is available in read-only status. During the brief rename process, which occurs in the UTILTERM phase, the object is stopped. This nondestructive process leaves the original data sets intact throughout the load process and allows you to easily restart from a failure or make the objects available without having to recover the objects.

To run a load with SHRLEVEL REFERENCE, perform the following tasks:

Step 1 Depending on the value for your REDEFINE command or installation option, perform any staging data set tasks that are required as described in “Staging Data Sets” on page 2-31.

Step 2 With your LOAD command, specify SHRLEVEL REFERENCE and, if necessary, ORIGINALDISP. For information about these commands, see “SHRLEVEL” on page 3-19 and “ORIGINALDISP” on page 3-59.

Note: SHRLEVEL REFERENCE is currently available only with LOAD REPLACE.

SHRLEVEL CHANGE

You can specify SHRLEVEL CHANGE with either LOAD REPLACE or LOAD RESUME YES. The load process is different for these two options.

LOAD REPLACE

When you specify LOAD REPLACE SHRLEVEL CHANGE, the object that you are loading remains in read/write status while LOADPLUS loads the data to a staging data set. After the load completes, LOADPLUS renames this staging data set, replacing the original object with the staging data set.

During the load process, the original object is available in read/write status. During the brief rename process, which occurs in the UTILTERM phase, the object is stopped. This nondestructive process leaves the original data sets intact throughout the load process and allows you to easily restart from a failure or make the objects available without having to recover the objects.

Warning! This type of load fully replaces the data in the object (or partition), including any inserts, updates, or deletes to the base object that occur during the load process.

BMC Software, Inc., Confidential and Proprietary Information

To run a LOAD REPLACE SHRLEVEL CHANGE job, perform the following tasks:

Step 1 Depending on the value for your REDEFINE command or installation option, perform any staging data set tasks that are required as described in “Staging Data Sets” on page 2-31.

Step 2 With your LOAD command, specify SHRLEVEL CHANGE and, if necessary, ORIGINALDISP. For information about these commands, see “SHRLEVEL” on page 3-19 and “ORIGINALDISP” on page 3-59.

LOAD RESUME YES

When you specify LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY (hereafter referred to as SQLAPPLY), the object that you are loading remains in read/write status while LOADPLUS loads the data by using SQL insert processing.

For this type of load, LOADPLUS integrates with the Apply Plus component of the BMC Software Log Master for DB2 product. This feature does not require that you have a license for the Log Master product but does require that the Apply Plus component be installed and made available through the STEPLIB, JOBLIB, or LINKLIST at execution time. The Apply Plus component is installed separately from the Log Master product.

To run a SQLAPPLY load, perform the following tasks:

Step 1 Ensure that Apply Plus version 1.5.00 or later is available to LOADPLUS.

Step 2 Ensure that you have the appropriate authorizations. See “DB2 Authority” on page 2-3 for information about these authorizations.

Step 3 With your LOAD command, specify SHRLEVEL CHANGE and, if necessary, one or more of the options that are available with SHRLEVEL CHANGE.

See “SHRLEVEL” on page 3-19 for information about these options.

The rest of this section describes considerations for running SQLAPPLY.

DB2 Logs

Because this type of load uses insert processing, it writes to your DB2 logs. With this additional activity, you might need to increase the size of your active logs. Refer to your IBM DB2 installation guide for guidelines to use when sizing your active logs.

BMC Software, Inc., Confidential and Proprietary Information

Processing Phases

For this type of load, LOADPLUS uses its single-phase load architecture to pass your data to Apply Plus. For additional information about the processing phases involved in this type of load, see Figure 1-3 on page 1-19.

Work Data Sets

For this type of load, LOADPLUS does not require the SORTOUT and SYSUT1 data sets, and does not use them if they are specified.

When Apply Plus needs to take a snap dump, Apply Plus always directs the snap dump to the SYSERR data set, resulting in a data set name conflict if you use SYSERR for the standard LOADPLUS error data set. Therefore, BMC Software recommends that you use the ERRDDN installation or command option to specify a default ddname other than SYSERR for the standard error data set that LOADPLUS uses. See “ERRDDN” on page 3-39 for more information about this option.

Table Space Organization

Apply Plus loads your data by using SQL INSERT statements. Therefore, following a SQLAPPLY load, your table functions as a table whose data was inserted, not loaded. Because insert processing does not preserve the table’s free space and free pages, this type of load can affect the organization of your table. Therefore, you might need to reorganize your table space following this type of load.

If you are adding new rows to an empty table space or to a table space with little or no free space, and you want the new records inserted in clustering key order, specify ORDER YES. Doing so can help maintain the clustering order of your table space and reduce the need to reorganize the table space.

Multiple Table Loads

For performance reasons with this type of load, LOADPLUS sorts multiple table data only if you specify ORDER YES.

Referential Integrity

If you are loading multiple tables that have foreign key relationships between them, and you want your parent tables to load before their child tables, specify ORDER YES. This causes LOADPLUS to assign a table number based on the parent-child relationships and to sort your data by this assigned table number.

BMC Software, Inc., Confidential and Proprietary Information

If you are loading multiple tables that are self-referential or that have circular relationships, LOADPLUS cannot determine the correct order. To maintain referential integrity in this case, order the data prior to your load job, and specify ORDER NO on your LOAD command.

Duplicate Keys

When you run a SQLAPPLY load, LOADPLUS handles duplicates differently from other load types. For this type of load, if the input file contains duplicate rows, LOADPLUS loads the first one that it encounters and discards any subsequent ones.

Note: The record that LOADPLUS loads and the duplicate records that LOADPLUS discards might not be the ones that you expect. For example, if you specify ORDER YES, the sequence of the records that LOADPLUS loads might be different from the sequence of the records in your input file.

This type of load uses DB2 insert processing and DB2 rejects a duplicate row in the first index in which it detects that the row is a duplicate. Therefore, LOADPLUS might not detect all indexes in which a row is a duplicate.

For this type of load, LOADPLUS detects and discards duplicates during insert processing in the COMBINED phase.

Copies

For a SQLAPPLY load, your table space is always available. Therefore, for this type of load, LOADPLUS does not create an image copy or DSN1COPY. If you include COPY YES in your LOAD command, LOADPLUS terminates with return code 8.

Restart

If you restart a SQLAPPLY job, you must use the same input data that you used originally and it must be in the same order. For example, if you are loading concatenated data sets, these data sets must be in the same order as they were originally, and you must not include any additional data sets. Similarly, if LOADPLUS encountered duplicates during the original job, you should not remove them before you restart.

Warning! Data that is missing or that is in a different order in the restarted job than in the original job produces unpredictable results.

BMC Software, Inc., Confidential and Proprietary Information

Statistics

For a SQLAPPLY load, LOADPLUS does not update DB2 or BMC statistics tables.

For this type of load, LOADPLUS generates a message that indicates the number of loaded rows and the number of discarded rows. LOADPLUS provides this information by table space, partition, and table.

Incompatible LOADPLUS Command Options

Some LOADPLUS command options are not available when you perform a SQLAPPLY load. Table 2-8 describes the options that are not valid with this type of load and how LOADPLUS responds if your job contains these options.

Table 2-8 Command Options That Are Incompatible with SQLAPPLY (Part 1 of 2)

Command Option	Response From LOADPLUS If You Include the Option
BMCSTATS	ignores this option and issues message BMC50109I
CHECKPEND YES	ignores this option
CLUSTERRATIO	ignores this option
COPY NO COPYPEND YES	ignores this option
COPY YES	issues message BMC50115E and terminates with return code 8
ENFORCE	ignores this option
IDCACHE	ignores this option
IDERROR	ignores this option
INDEX BUILD INDEX UPDATE	ignores these options
LOG NO	issues message BMC50115E and terminates with return code 8
MAXSORTS	ignores this option
NLPCTFREE	ignores this option
ORDER PRESORTED	issues message BMC50115E and terminates with return code 8
ORDER YES ASSOCIATE BYCLUSTERKEY	changes this option to ASSOCIATE BYTABLE if referential integrity (RI) exists between the tables that you are loading If LOADPLUS changes this option, LOADPLUS issues message BMC50138I.
PART REPLACE	issues message BMC50115E and terminates with return code 8

BMC Software, Inc., Confidential and Proprietary Information

Table 2-8 Command Options That Are Incompatible with SQLAPPLY (Part 2 of 2)

Command Option	Response From LOADPLUS If You Include the Option
PART PREFORMAT PREFORMAT	ignores these options and issues message BMC50109I
PRELOAD LOAD PRELOAD CONTINUE	ignores these options LOADPLUS always uses single-phase processing for a SQLAPPLY load.
PRELOAD PAUSE	issues message BMC50115E and terminates with return code 8
REDEFINE	ignores this option
SKIPIX	issues message BMC50115E and terminates with return code 8
SYNC	ignores this option
UNIQUECHECK	ignores this option
UPDATEDB2STATS	ignores this option and issues message BMC50109I

Staging Data Sets

This section describes the actions that you need to take with regard to staging data sets when using LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE. The section also describes the rename process that LOADPLUS performs during the UTILTERM phase of your load job.

Tasks to Perform Before Running the Load Job

Use Table 2-9 on page 2-32 to determine what action to take, regarding staging data sets, for LOADPLUS to perform your LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE job correctly.

BMC Software, Inc., Confidential and Proprietary Information

Table 2-9 Staging Data Set Actions

REDEFINE Command or Installation Option	Object You Are Loading	Action
REDEFINE NO	VCAT-defined	Preallocate the staging data sets and any data sets that you might need for expansion of a multiple data set object. If you do not preallocate staging data sets and you specify REDEFINE NO, LOADPLUS fails because it cannot find the data sets.
	STOGROUP-defined	Preallocate the staging data sets and any data sets that you might need for expansion of a multiple data set object. If you do not preallocate the staging or expansion data sets, LOADPLUS creates the data sets for you using the same rules as if you had specified REDEFINE YES.
REDEFINE YES	VCAT-defined	Provide the IDCAMS statements in the SYSIDCIN data set to delete and define the staging data sets. If you do not provide any statements in SYSIDCIN, LOADPLUS treats the job as if you specified REDEFINE NO.
	STOGROUP-defined	Ensure that you have enough space available for LOADPLUS to automatically allocate the staging data sets.

If you do need to preallocate or define the staging data sets, you must use the original DB2 data set names as a starting point, but replace the **DSN**DBC and **DSN**DBD nodes with **BMC**DBC and **BMC**DBD. Table 2-10 illustrates this naming convention.

Table 2-10 Naming Convention for Staging Data Sets

If the Name of the Existing Data Set You Are Loading Is This	Then Define the Staging Data Set Name as This
vcat.DSNDBC.*database*.*tablespace*.I0001.A*nnn*	*vcat*.BMCDBC.*database*.*tablespace*.I0001.A*nnn*
vcat.DSNDBD.*database*.*tablespace*.I0001.A*nnn*	*vcat*.BMCDBD.*database*.*tablespace*.I0001.A*nnn*
vcat.DSNDBC.*database*.*tablespace*.J0001.A*nnn*	*vcat*.BMCDBC.*database*.*tablespace*.J0001.A*nnn*
vcat.DSNDBD.*database*.*tablespace*.J0001.A*nnn*	*vcat*.BMCDBD.*database*.*tablespace*.J0001.A*nnn*

BMC Software, Inc., Confidential and Proprietary Information

Data Set Rename Process

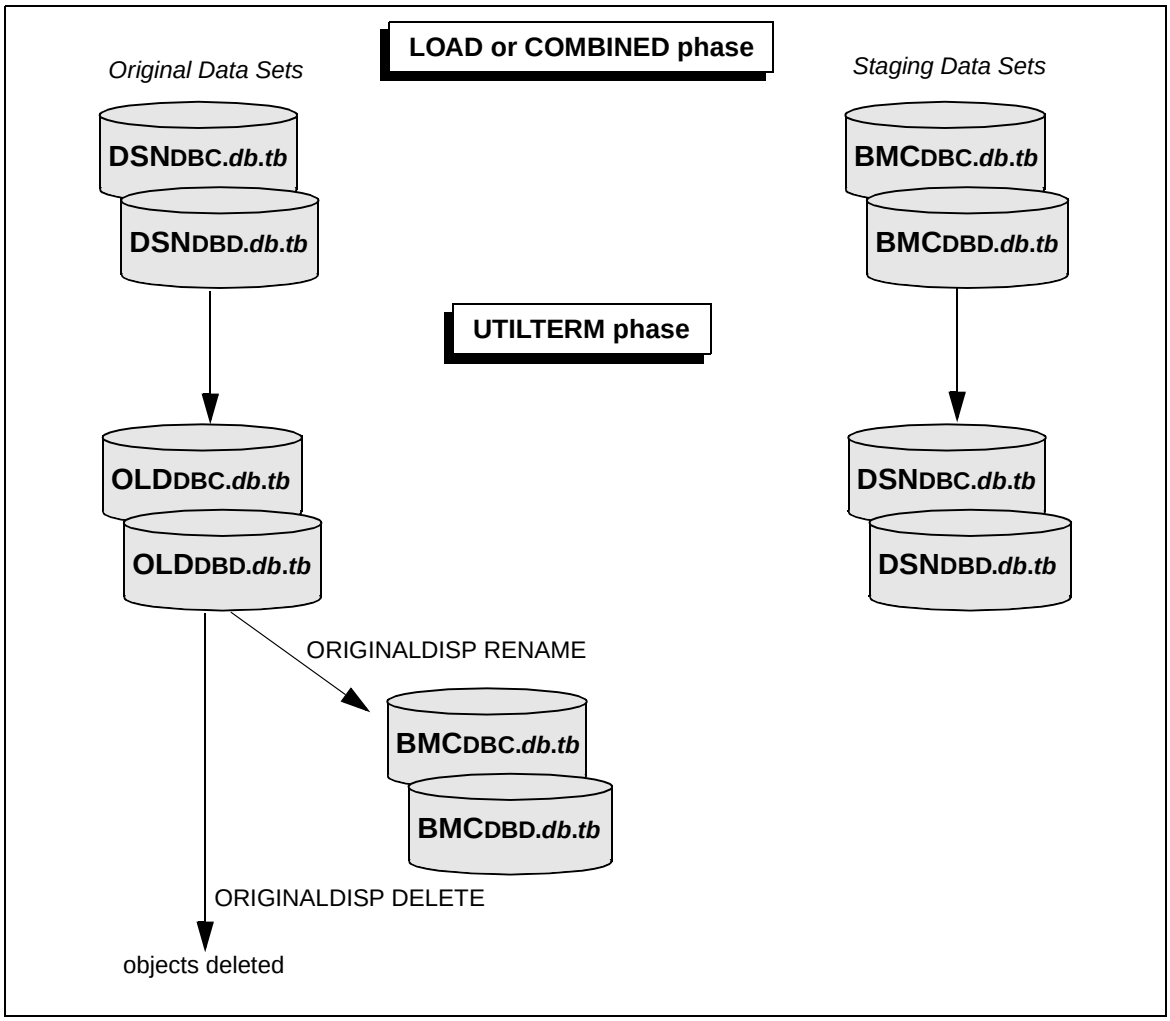
Figure 2-1 on page 2-34 illustrates the data set rename process.

After LOADPLUS writes the loaded objects to the staging data sets, LOADPLUS stops the objects and renames the original DB2 VSAM data sets. During the renaming process, LOADPLUS replaces **DSN** in the DSNDBC node of the cluster and in the DSNDBD node of the data component with **OLD**. This part of the process makes the original data sets obsolete. LOADPLUS then renames the staging data sets by replacing **BMC** in the BMCDBC node of the cluster and in the BMCDBD node of the data component with **DSN**. Doing so gives the staging data sets the same name as the original data sets, even though they are in a new location.

After successfully renaming the data sets, LOADPLUS starts the objects with their original statuses. If the value for the ORIGDISP option in the installation options module is DELETE, or you set the ORIGINALDISP command option to DELETE, LOADPLUS deletes the original data sets, which are now obsolete. If the value for this option is RENAME, LOADPLUS renames the original data sets to the staging data set names (that is, replaces **OLD** with **BMC** in the names). Doing so preserves the space that was initially allocated for the original data sets. As a result, the staging data sets are ready to be used with the next LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE load.

BMC Software, Inc., Confidential and Proprietary Information

Figure 2-1 Data Set Renaming Process



Tasks to Perform If Your Load Job Terminates

If you terminate the load job or the job abnormally terminates and you do not intend to restart it, you can delete the staging data sets. However, if the job terminates during the rename process, BMC Software recommends that you restart your job rather than attempt to rename the staging data sets manually.

BMC Software, Inc., Confidential and Proprietary Information

Identity Columns

When loading identity columns, LOADPLUS generates values as necessary and updates the DB2 catalog to reflect the maximum assigned value.

How LOADPLUS Generates Identity Column Values

For all types of load jobs except SQLAPPLY, LOADPLUS reserves one or more caches of identity column values by updating the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES during the PRELOAD or COMBINED phase. LOADPLUS reserves caches independently for each READ task that the load job starts.

You can use the IDCACHE command or installation option to control the size of the cache that LOADPLUS reserves. If you specify a large cache size, there is a greater possibility of gaps in the set of identity column values and a higher risk of running out of identity column values. In addition, if your identity column is defined with CYCLE YES, a large cache size introduces a greater likelihood of generating duplicate values. However, specifying a smaller cache size can impact performance because LOADPLUS must access the DB2 catalog more frequently.

Note: For a SQLAPPLY load job, LOADPLUS does not reserve an identity column cache on the DB2 catalog. For this type of load job, LOADPLUS uses DB2 to generate identity column values.

How LOADPLUS Updates MAXASSIGNEDVAL

LOADPLUS updates the value of the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES based on the values that you are loading.

Generated Values

When generating your identity column values, LOADPLUS always updates the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES with the last value in the cache that it is reserving. LOADPLUS updates this field at the time that it reserves each cache.

BMC Software, Inc., Confidential and Proprietary Information

Input File Values

When loading your identity column values from an input file (when the column is defined as GENERATED BY DEFAULT), LOADPLUS updates MAXASSIGNEDVAL only if you specify a value of YES for the UPDATEMAXA command option or the UPDMAXA installation option. As Table 2-11 describes, the value with which LOADPLUS updates this field depends on the identity column values loaded.

Table 2-11 How LOADPLUS Updates MAXASSIGNEDVAL When Loading Values from an Input File

Loaded Identity Column Values	LOADPLUS Action on MAXASSIGNEDVAL
The maximum loaded identity column value is • greater than the value of MAXASSIGNEDVAL (for positive INCREMENT values; less than the value of MAXASSIGNEDVAL for negative INCREMENT values) *and* • within the range that is defined on the identity column	LOADPLUS updates MAXASSIGNEDVAL with the maximum loaded identity column value.
The maximum loaded identity column value is less than the value of MAXASSIGNEDVAL (for positive INCREMENT values; greater than the value of MAXASSIGNEDVAL for negative INCREMENT values).	LOADPLUS does not update MAXASSIGNEDVAL.
The *maximum* loaded identity column value is not within the range that is defined on the identity column, but the *minimum* loaded identity column value *is* within this range.	LOADPLUS updates MAXASSIGNEDVAL with the highest value in the range defined on the identity column (for positive INCREMENT values). **Note**: If INCREMENT is a negative value, LOADPLUS updates MAXASSIGNEDVAL with the lowest value in the range defined on the identity column.
The maximum and minimum loaded identity column values are not within the range that is defined on the identity column.	LOADPLUS does not update MAXASSIGNEDVAL.

How LOADPLUS Handles Invalid Value Errors

LOADPLUS validates that identity column values are within the range that is defined on the column. LOADPLUS handles any invalid identity column values based on the type of load job and the value for the IDERROR command or installation option.

- For SQLAPPLY load jobs, LOADPLUS discards the rows with the invalid values and ignores the value for the IDERROR option.

BMC Software, Inc., Confidential and Proprietary Information

- For all other load jobs:

  — If you specify a value of DISCARD for IDERROR, LOADPLUS discards the rows with the invalid values.

  — If you specify a value of FAIL for IDERROR, LOADPLUS terminates the load job with a return code 8 as soon as an identity column value exceeds the range that is defined on the column.

Additional Identity Column Considerations

This section describes additional considerations for loading identity columns.

Partitioned Table Spaces

With the following exception, LOADPLUS loads into partitions regardless of the presence of an identity column on the table. For a SQLAPPLY load, if the identity column is part of the limit key of a partition and identity column values are being generated, LOADPLUS does not load the partitioned table space and terminates with a return code 12.

FORMAT BMCUNLOAD

If you specify FORMAT BMCUNLOAD with your LOAD command, the identity column in the table that you are loading must be defined as GENERATED BY DEFAULT. For additional considerations when using these two features together, see “Identity Columns” on page 2-47.

Order of Identity Column Values

When multitasking your load job, you cannot ensure that LOADPLUS will assign generated identity column values in input order. To force LOADPLUS to assign identity column values in input order, build your load job in one of the following ways. If you do not use one of the following specifications, the order in which the generated values are assigned is unpredictable.

- For a SQLAPPLY load, specify APMAXAGENTS 1 and ORDER NO.

  Note: For a SQLAPPLY load, you can also have values assigned in clustering order. To do so, specify APMAXAGENTS 1 and ORDER YES.

- For all other load types, specify only one SYSREC data set per load job.

BMC Software, Inc., Confidential and Proprietary Information

DB2 Table Check Constraints

When all of the following circumstances apply, LOADPLUS terminates:

- You are loading an identity column that is referenced in the check condition of a DB2 table check constraint.
- The identity column definition and LOADPLUS field specification for the identity column *could* result in LOADPLUS generating values.
- You do not specify ENFORCE NO on the LOAD command.

Field Specifications

For certain field specifications, LOADPLUS terminates or takes other actions that you might not expect when you are loading identity columns. Table 2-12 describes these actions.

Table 2-12 Considerations for Identity Columns and Related Field Specifications

Identity Column Definition	Identity Column Field Specification	Other Field Specification	LOADPLUS Action
GENERATED BY DEFAULT	field specification includes DEFAULTIF	not applicable	LOADPLUS defaults to the generated value.
	field specification could cause LOADPLUS to generate a value	identity column is referenced in a WHEN, DEFAULTIF, or NULLIF statement on the field specification of another column	LOADPLUS terminates.
GENERATED ALWAYS	field specification is included on the LOAD command	not applicable	LOADPLUS terminates.
	not applicable	identity column is referenced in a WHEN, DEFAULTIF, or NULLIF statement on the field specification of another column	LOADPLUS terminates.

BMC Software, Inc., Confidential and Proprietary Information

CSV Considerations

LOADPLUS provides the capability of loading data from comma-separated-value (CSV) files.

Note: CSV files are also referred to as comma-delimited or delimited files.

How to Load CSV Data

To load CSV data, perform the following tasks:

Step 1 Ensure that your data is formatted within the requirements that are specified in "CSV Data Format."

Step 2 Specify FORMAT CSV with your LOAD command. Depending on the format of your CSV file, you might also need to specify one or more CSV options. For more information about this option, see "CSV" on page 3-26.

Step 3 Include appropriate field specifications with your LOAD command. For considerations regarding field specifications, see "CSV Field Specifications" on page 2-41. For syntax information, see "field_specification" on page 3-102.

CSV Data Format

Before you load a CSV file, you must ensure that your data meets the following requirements:

- Your CSV file must contain delimiters.

  Note: *Enclosure characters* (characters that enclose each field) in your CSV file are optional.

- Input strings should not contain the same characters as your field delimiters or enclosure characters. If they do, LOADPLUS might load the record incorrectly or reject it. To prevent this problem, perform one of the following tasks:

  — Remove these characters from your input strings.
  — Change the delimiter or enclosure character in your input file so that you do not have a conflict with embedded characters. Then ensure that the TERMINATEDBY and ENCLOSEDBY options on your FORMAT CSV command are compatible with the delimiter and enclosure characters in your input file.

BMC Software, Inc., Confidential and Proprietary Information

- Your CSV data must be in an OS/390 or z/OS file that is in EBCDIC format.

- Your CSV input file should use a variable-length record format (RECFM=VB). If your input file uses a fixed-length record format, and the last field on a record does not extend to the end of that record, the field must end with a delimiter.

- A one-to-one correspondence should exist between the input fields of your CSV data and the field specifications on your LOAD command. Otherwise, the following results occur. (See “CSV Field Specifications” on page 2-41 for more information.)

  — If you have more input fields than field specifications, LOADPLUS ignores the extra input file data. LOADPLUS maps field specifications to input fields in order, starting with the first specification and the first input field.

  — If you have fewer input fields than field specifications, LOADPLUS ignores the additional field specifications.

Installation Options

When you specify FORMAT CSV, LOADPLUS changes the value of the RULES installation option to STANDARD, regardless of the value that was specified at installation. This option affects how LOADPLUS handles NULLIF and DEFAULTIF specifications. For more information about the RULES option, see Appendix E, “RULES Installation Option Examples.”

LOAD Command Limitations with FORMAT CSV

The CONTINUEIF option is not valid when you specify FORMAT CSV. If you specify CONTINUEIF with FORMAT CSV, LOADPLUS issues message BMC51412E and terminates.

BMC Software, Inc., Confidential and Proprietary Information

CSV Field Specifications

You must include a field specification on your LOAD command for every column that you are loading. In addition, each field specification must map to an input field in your CSV file. LOADPLUS maps field specifications to input fields in order, starting with the first specification and the first input field.

Note: Alternatively, you can elect not to include any field specifications. If you do not include any field specifications, LOADPLUS loads the data as if you had included a field specification that maps to every column in the table that you are loading.

Excluding Particular Input Fields

If you do not want to load data from a particular input field, perform one of the following tasks:

- If the input field that you do not want to load is the last field (or fields) in the CSV file, you do not need to include a field specification for that field. LOADPLUS ignores any extra fields at the end of the input file.
- If the input field that you do not want to load *is not* the last field (or fields) in the CSV file, include a placeholder field specification on your LOAD command. In this field specification, use a field name that is not the name of a column in the DB2 table. You must also specify SKIPFIELDS YES on your LOAD command.

Figure 2-2 on page 2-43 illustrates how LOADPLUS maps your field specifications for CSV files to your DB2 table columns. Example 2a does not supply a field specification for the employee department field. Therefore, LOADPLUS loads employee department data into the employee number column, which is not the desired result. Example 2b corrects this problem by adding a placeholder field specification.

Field Specification Requirements

Your field specifications must meet the following requirements:

- For the fields that you want to load, the field name must match the corresponding column name.
- DATE, TIME, or TIMESTAMP field specifications must be in one of the supported external formats, as listed in the tables in “DATE, TIME, and TIMESTAMP Formats” on page 3-119.

BMC Software, Inc., Confidential and Proprietary Information

- To have LOADPLUS interpret two consecutive delimiter characters as a null field, include the following NULLIF option:

  ```
  NULLIF field_name = ''
  ```

  If you do not include this NULLIF option, LOADPLUS treats these input fields in the following ways:

  — For DATE, TIME, and TIMESTAMP fields where the column is nullable, LOADPLUS loads a null value.

  — For DATE, TIME, and TIMESTAMP fields where the column is defined as NOT NULL, LOADPLUS loads the default date, time, or timestamp.

  — LOADPLUS treats all other fields as VARCHAR fields with length 0.

Additional Field Specification Considerations

In addition, consider the following information about field specification options:

- You can use NULLIF and DEFAULTIF field specifications. Note, however, that LOADPLUS changes the value of the RULES installation option to STANDARD, which affects how LOADPLUS handles NULLIF and DEFAULTIF specifications.

- LOADPLUS ignores the following field specifications:

  — POSITION (*start*:*end*)

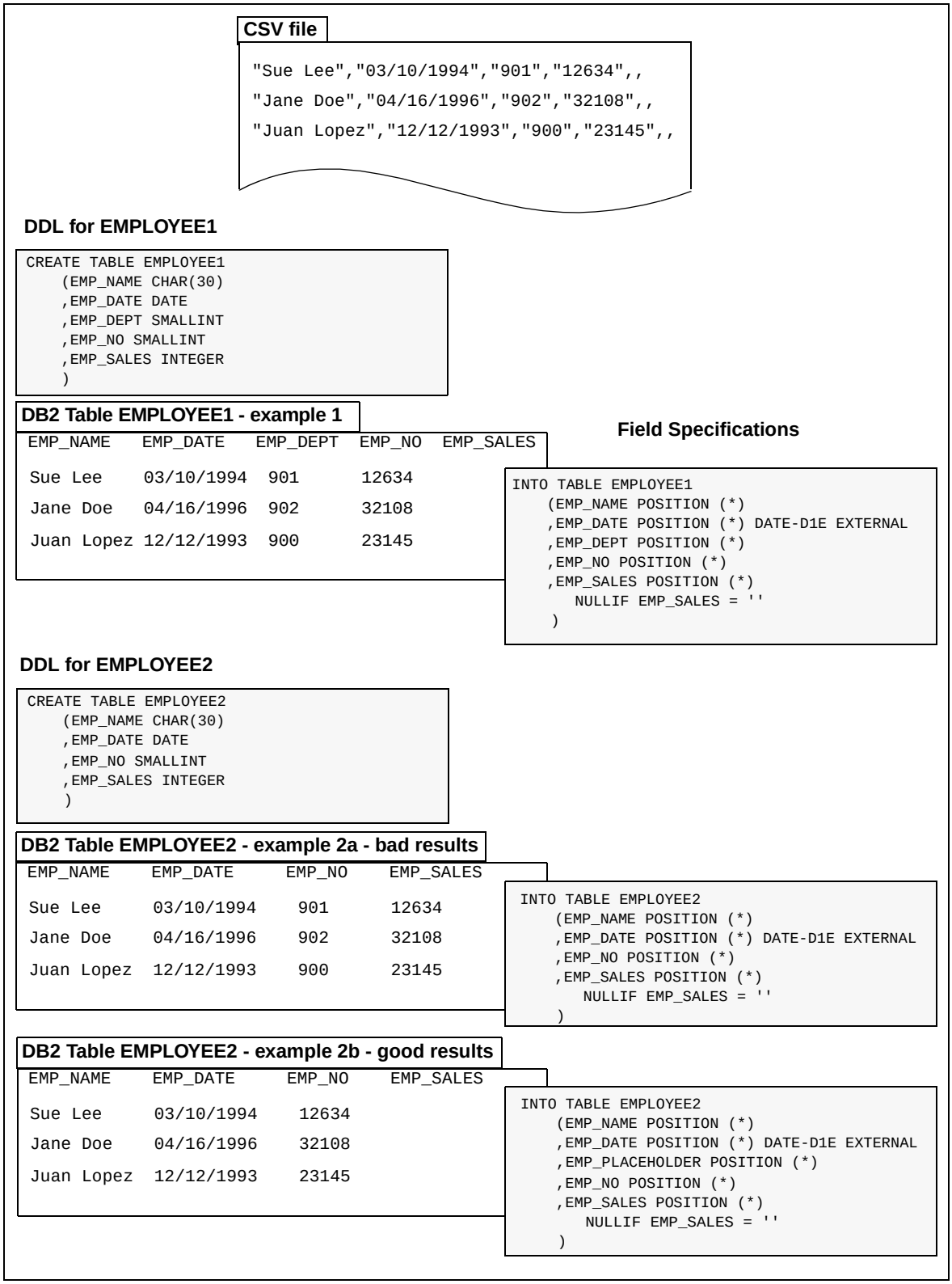
  LOADPLUS currently starts from the first position of each field. However, BMC Software recommends that you specify POSITION(*).

  — data type, except on specifications for DATE, TIME, and TIMESTAMP fields

  — length

BMC Software, Inc., Confidential and Proprietary Information

Figure 2-2 CSV Field Specification Mapping



BMC Software, Inc., Confidential and Proprietary Information

Data from UNLOAD PLUS in Internal Format

When used in conjunction with UNLOAD PLUS, LOADPLUS provides a high-speed option to move data from one table to another table that has a similar table structure. This feature is useful, for example, for migrating data from test to development databases. The FORMAT BMCLOAD option of UNLOAD PLUS unloads the data in an internal format that only LOADPLUS can read. Using the FORMAT BMCUNLOAD option of LOADPLUS to read this data significantly improves performance for this type of load job by reducing the need for data verification and conversion. See the *UNLOAD PLUS for DB2 Reference Manual* for information about unloading the data for this type of data movement.

How to Load the Data

To load data that was unloaded by using the FORMAT BMCLOAD option of UNLOAD PLUS, perform the following tasks. For an example that shows you how to use this feature, see "Example 14: LOAD REPLACE from UNLOAD PLUS Data in Internal Format" on page 5-106.

Step 1 Ensure that the table that you are loading is structured within the requirements specified in "Table Structure" on page 2-45.

Step 2 Specify FORMAT BMCUNLOAD with your LOAD command. For more information about this option, see "BMCUNLOAD" on page 3-25.

Step 3 Include a WHEN TABLE=*obid* clause with your INTO option. For more information about this option, see "TABLE=obid" on page 3-94.

Warning! BMC Software recommends that you specify input files from a single UNLOAD PLUS step in a single LOADPLUS step. If you specify input files from multiple UNLOAD PLUS steps in a single LOADPLUS step, you can encounter data corruption or other unpredictable results.

Step 4 If you do not use the LOAD control statements that UNLOAD PLUS generated, ensure that your LOAD command does not contain field specifications or any of the incompatible command options that are described in "Incompatible Command Options" on page 2-46.

Note: When you use this feature, BMC Software strongly recommends that you use the LOAD control statements that UNLOAD PLUS generates. Using these control statements ensures that your command options comply with the requirements for using this feature.

BMC Software, Inc., Confidential and Proprietary Information

Table Structure

In general, when you use the FORMAT BMCUNLOAD option, the structure of the table that you are loading must be identical to the structure of the table that you unloaded with UNLOAD PLUS. This section describes considerations regarding the structure of the two tables. For information about how LOADPLUS handles discrepancies in table structure, see “Validation Errors” on page 2-46

General Structure Considerations

The type of table space (for example segmented) does not have to be the same on the target as it is on the source, and the indexes do not have to be the same.

Column Definitions

The following considerations apply to column definitions in the two tables:

- The number, order, and data type of the columns in the two tables must be the same, but column names can be different.
- For VARCHAR and VARGRAPHIC columns, your target table columns can have a length that is greater than the source table columns.

  Note: For ROWID columns, LOADPLUS does not verify the column length.

- Target identity columns and ROWID columns must be defined as GENERATED BY DEFAULT. If your identity column or ROWID column is defined as GENERATED ALWAYS, LOADPLUS discards all rows for that table and issues message BMC51590E.
- If identity column definitions are not the same (for example, the values for START and INCREMENT), LOADPLUS loads the data. If the value of the UPDATEMAXA or UPDMAXA option is YES, LOADPLUS also issues a warning message.

DB2 User Exits

The following considerations apply to user exits that are defined for the two tables:

- If the source has an EDITPROC, the target must also have an EDITPROC with the same name.

BMC Software, Inc., Confidential and Proprietary Information

- If the source has a VALIDPROC or FIELDPROC, the target must also have one, but it can have a different name. LOADPLUS displays a warning but loads the rows.

Validation Errors

LOADPLUS validates that your table structure is the same by comparing the verification records of the input file that UNLOAD PLUS creates with the structure of the table that you are loading. If conflicts exist between the verification records and the table that you are loading, LOADPLUS issues the appropriate verification messages as described in this section. Depending on the type of conflict, LOADPLUS takes one of the following actions:

- The load job fails.
- LOADPLUS loads the rows with a warning.
- LOADPLUS discards the rows for the table until it encounters an accurate set of verification records for the same table.

LOADPLUS issues the following messages when validating the table structure:

- BMC51590E or W displays the type of error.
- BMC51591E or W displays the discrepancy between the two tables.
- BMC51592I displays the number of verification records that were read.

If the input file does not contain any verification records, LOADPLUS issues message BMC51590E and ends with a return code 16.

Incompatible Command Options

The following command options are not valid when you specify FORMAT BMCUNLOAD:

- You cannot include a field specification when you specify FORMAT BMCUNLOAD. If your LOAD command includes a field specification, the load job fails and LOADPLUS issues message BMC51412E.

- If you specify CONTINUEIF, the load job fails and LOADPLUS issues message BMC51412E.

- If you specify ENFORCE CHECK CONSTRAINTS, LOADPLUS ignores the option and issues message BMC50109I. If there are check constraints on the table and the value of the CHECKPEND option is YES, LOADPLUS sets CHKP following the load.

BMC Software, Inc., Confidential and Proprietary Information

- If you specify any of the following LOAD command options with FORMAT BMCUNLOAD, LOADPLUS ignores the option:

  — ASCII/EBCDIC
  — CCSID
  — NOSUBS

Additional FORMAT BMCUNLOAD Considerations

This section describes additional requirements to consider when using the FORMAT BMCUNLOAD option.

RULES Installation Option

LOADPLUS changes the value of the RULES installation option to BMC, regardless of the value that was specified at installation.

Data Translation

Data translation between the two tables is not supported. If you specify the CCSID option on your LOAD command, LOADPLUS ignores the option but loads the data.

DB2 Table Check Constraints

When you specify FORMAT BMCUNLOAD, LOADPLUS does not validate table check constraints, regardless of the value that you specify for the ENFORCE option. If there are check constraints on the table and the value of the CHECKPEND option is YES, LOADPLUS sets CHKP following the load.

Identity Columns

If any of the tables that are specified on your INTO statements contain an identity column and the tables are valid based on their verification records, LOADPLUS updates the value for MAXASSIGNEDVAL in SYSIBM.SYSSEQUENCES when the value of the UPDATEMAXA or UPDMAXA option is YES. The value that LOADPLUS uses for MAXASSIGNEDVAL is the value for the source table that was in the DB2 catalog at the beginning of the unload process in UNLOAD PLUS. LOADPLUS updates MAXASSIGNEDVAL by using the same criteria as with any other load job. See “How LOADPLUS Updates MAXASSIGNEDVAL” on page 2-35 for additional information.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS validates the identity column definition in the verification records of the input file against the identity column definition in the table that you are loading. If these definitions are not the same, LOADPLUS issues a warning message only if the value of the UPDATEMAXA or UPDMAXA option is YES. LOADPLUS loads the data regardless of whether the definitions are the same.

When you specify FORMAT BMCUNLOAD, LOADPLUS does not validate the identity column definition against the input data values. Therefore, you might have values in your identity column that are outside the range that you specified.

BatchPipe Input

LOADPLUS allows you to run in a BatchPipes environment, including the ability to use input from the Job Optimizer Pipes component of the BMC Software MAINVIEW Batch Optimizer product. The following considerations apply when using data pipes:

- If your input data set is a pipe, LOADPLUS does not write discards to a SYSDISC data set.
- You can specify a mixture of DASD files and pipes as input. However, you cannot concatenate these files within one SYSREC DD.
- If you use multiple pipe data sets, you must specify a separate SYSREC DD statement for each data set and the DCB parameters for all of the pipe data sets must be the same.
- If you are loading data with multiple pipes, each pipe must be generated by an independent job and each of these jobs must be able to run concurrently. For example, you cannot load data by using multiple pipes that are generated by multiple steps within a single job.
- You cannot specify a pipe data set with the INDSN command option.

BMC Software, Inc., Confidential and Proprietary Information

Hardware Compression

Hardware compression is in effect when both of the following conditions apply to your load job:

- The table space or partition that you are loading has the COMPRESS YES attribute.
- You specify LOAD REPLACE or, for the specified partitions, you specify LOAD RESUME YES PART REPLACE.

You can use the following compression dictionaries interchangeably:

- any BMC Software compression dictionary that the BMC Software utilities build
- any DB2 compression dictionary that the IBM REORG or LOAD utility builds

The value of the KEEPDICTIONARY installation option determines how LOADPLUS handles compression at a global level. However, you can override this value by specifying the KEEPDICTIONARY command option at either the table space or partition level. For information about the command options, see page 3-17 and page 3-92. For information about the installation option, see page A-20.

Note: If you specify the KEEPDICTIONARY command option without a value, LOADPLUS assumes KEEPDICTIONARY YES, regardless of the value that was specified for the installation option.

LOADPLUS reports the percentage of compressed rows in messages BMC50512I and BMC50522I. This percentage does not include rows that are compressed by an EDITPROC.

Building a Dictionary

LOADPLUS builds a new dictionary under either of the following circumstances:

- The value of KEEPDICTIONARY is NO.
- The value of KEEPDICTIONARY is YES, but a dictionary does not exist.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS builds the compression dictionary during the PRELOAD phase for a two-phase load and during the COMBINED phase for a single-phase load. After completely building the dictionary, LOADPLUS compresses the data. For a two-phase load, LOADPLUS stores the dictionary in the BMCDICT table between phases.

LOADPLUS reserves enough pages for the largest dictionary size.

Note: If LOADPLUS does not load enough rows to build a complete dictionary, LOADPLUS does not compress any rows.

Keeping a Dictionary

If the value of KEEPDICTIONARY is YES and a dictionary already exists, LOADPLUS uses it for compression.

Image Copies and Recoverability of the Loaded Table Space

If you do not take an image copy as part of the LOADPLUS job, you must take some action to ensure that DB2 can recover the table space. If you want an image copy, run COPY PLUS or the IBM DB2 COPY utility. You can also run DSN1COPY or any suitable substitute that your system provides.

If you intend to create the copy outside of LOADPLUS, specify LOADCPY=YES in your installation options so that LOADPLUS inserts a LOAD LOG(NO) row into SYSIBM.SYSCOPY. This row tells DB2 that a load has been performed on the table space without logging. This row is necessary for recovery to work during fallback processing and to prevent you from making an incremental image copy before you make a full image copy.

If you specify COPY NO and your installation options specify LOADCPY=NO, LOADPLUS does not insert a LOAD LOG(NO) row into SYSIBM.SYSCOPY. This action can render the table space unrecoverable or only partially recoverable if the full image copy that you make after the load is not usable or if you make incremental image copies after the load before you make a full image copy.

Note: This information does not apply to a SQLAPPLY load job. This type of load logs all inserts, so there is no need to insert a LOAD LOG(NO) row into SYSIBM.SYSCOPY. When performing a SQLAPPLY load, the table space is always recoverable, assuming that it was recoverable before the load job.

BMC Software, Inc., Confidential and Proprietary Information

Referential Integrity and Check Constraints

LOADPLUS handles referential integrity and DB2 table check constraints differently depending on the type of load job that you are running.

SQLAPPLY

LOADPLUS has no ENFORCE CONSTRAINTS option for referential integrity violations. However, for a load job that uses Apply Plus (a SQLAPPLY load), the Apply Plus architecture preserves referential integrity relationships. Therefore, when referential integrity is involved and you specify ORDER YES, LOADPLUS sorts your data by table so that LOADPLUS loads parent tables before their child tables.

If you attempt to load tables that are self-referencing or that have circular relationships and you specify ORDER YES, LOADPLUS cannot determine the correct order. In this case, LOADPLUS issues warning message BMC51582W, but continues processing. When processing passes to DB2, you might receive SQL -530 errors. To maintain referential integrity for this type of load, order the data before your load job and specify ORDER NO with your LOAD command.

If DB2 table check constraints exist for a table being loaded, the Apply Plus architecture automatically checks table check constraints and discards any violations. If you specify ENFORCE NO for this type of load, LOADPLUS ignores it.

Other Load Types

LOADPLUS has no ENFORCE CONSTRAINTS command option for referential integrity violations. Therefore, for load types other than SQLAPPLY, LOADPLUS does not preserve referential integrity relationships. For load types other than SQLAPPLY, if the value of the CHECKPEND option is YES and a table has referential constraints defined, LOADPLUS places the table spaces in CHKP status. After the LOADPLUS job completes, you must run CHECK PLUS (specifying the CHECK DATA SCOPE ALL option) or the IBM CHECK DATA utility (specifying SCOPE ALL) to ensure that referential integrity has not been violated.

BMC Software, Inc., Confidential and Proprietary Information

If DB2 table check constraints exist for a table that you are loading and you specify or default to ENFORCE CHECK CONSTRAINTS, LOADPLUS verifies all constraints and discards any violations.

Note: If you specify FORMAT BMCUNLOAD, LOADPLUS does not verify check constraints, regardless of the value of the ENFORCE option.

If table check constraints exist, you specify ENFORCE NO, and the value of the CHECKPEND option is YES, LOADPLUS places the table space in CHKP status. After the LOADPLUS utility completes, you must run CHECK PLUS (specifying the CHECK DATA SCOPE ALL option) or the IBM CHECK DATA utility (specifying SCOPE ALL) to ensure that table check integrity has not been violated.

LOADPLUS allows an image copy to be made after a table is loaded, even when the table is in CHKP status. Because both check utilities log any changes that are made to the table, a full RECOVER utility execution that uses the copy applies the log records, thus ensuring the integrity of the data in the table. A RECOVER TOCOPY utility execution places the affected table in CHKP status again. If the table is in CHKP status due to referential integrity constraints, a subsequent invocation of either check utility restores referential integrity.

The CHEKPEND installation option and CHECKPEND command option tell LOADPLUS whether to set CHKP status when table check constraints are not enforced or RI constraints exist. For more information about these options, see page A-10 and page 3-67.

Additional Information About Check Constraints

Under certain circumstances, when a table that you are loading has a check constraint, LOADPLUS terminates. This section describes those circumstances.

DEFINE NO Objects

If you are loading a table whose table space or index spaces are created with DEFINE NO and a check constraint exists on the table, LOADPLUS attempts to build a row that satisfies the check constraint (attempts to *resolve* the constraint) while materializing the table. However, if LOADPLUS cannot resolve the check constraint, LOADPLUS terminates and issues message BMC51491E. Before terminating, LOADPLUS issues message BMC51492W, which describes the check constraint that it cannot resolve.

BMC Software, Inc., Confidential and Proprietary Information

To load this table, you must manually materialize the associated data sets and restart the load job.

Identity Columns

If you are loading an identity column that is referenced in the check condition of a DB2 table check constraint and the following circumstances also apply, LOADPLUS terminates:

- You do not specify ENFORCE NO on the LOAD command.
- The identity column definition and LOADPLUS field specification for the identity column *could* result in LOADPLUS generating values.

How LOADPLUS Invokes DB2 User Exits

LOADPLUS can invoke the following types of DB2 user exits during processing:

- EDITPROCs
- VALIDPROCs
- FIELDPROCs
- date exits:
  - DSNXVDTX (EBCDIC date exit routine)
  - DSNXVDTA (ASCII date exit routine)
- time exits:
  - DSNXVTMX (EBCDIC time exit routine)
  - DSNXVTMA (ASCII time exit routine)
- an authorization exit

By default, LOADPLUS invokes these exits in supervisor state (and PSW key=7). LOADPLUS calls the MODESET SVC to perform this switch before invoking an exit, and again to reset the mode after returning from the exit. For information about changing the LOADPLUS installation option to achieve better performance, see “Invoking MODESET for DB2 User Exits” on page 6-4.

BMC Software, Inc., Confidential and Proprietary Information

Image Copies After Recovery

The LOADPLUS copy function, when used with the LOAD RESUME YES option, provides performance benefits by not resetting the modified-page indicators in the table space data pages and space maps on existing pages, thus substantially reducing I/O operations. Other vendor products or product options (like the RESETMOD NO option in COPY PLUS) also use this technique and do not reset the modified-page indicators in the table space.

Because of the way DB2 handles these modified-page indicators, you need to routinely make a *full* image copy of any table space that you recover by using this type of image copy before continuing to make incremental image copies. You should make the full image copy by using COPY PLUS with the RESETMOD YES option specified or DB2 COPY.

DB2 Features That LOADPLUS Does Not Currently Support

The current release of LOADPLUS has the following limitations on support for the features of DB2:

- LOADPLUS does not support the following features:
  - — renaming data sets by using FASTSWITCH processing

    However, LOADPLUS *does* recognize the I and J instance qualifiers in your DB2 table space and index data set names.
  - — translating data by using Unicode encoding schemes
  - — updating statistics for catalog tables that are new for DB2 version 7
  - — using LISTDEF and TEMPLATE control statements
  - — loading data in IEEE Binary Floating Point (BFP) format
- LOADPLUS does not enforce referential integrity constraints unless you are running a SQLAPPLY load job.
- LOADPLUS does not load LOB data into auxiliary tables.
- When you alter the length of a VARCHAR column that is part of an index, LOADPLUS requires that you specify LOAD REPLACE or specify or default to LOAD RESUME YES INDEX BUILD.

BMC Software, Inc., Confidential and Proprietary Information

Chapter 3 Syntax of the LOAD Command

This chapter presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Syntax Rules for LOADPLUS

The following general rules apply to the LOAD command syntax:

- The LOADPLUS product does not permit you to split a token (such as a keyword, identifier, or constant) across a line.
- In a LOAD command, LOADPLUS considers any line beginning with an asterisk (*) in column 1 to be a comment and ignores it.
- When encountering two consecutive hyphens in a line, LOADPLUS considers everything on that line after those hyphens to be a comment and ignores the line.
- If you specify the same command option more than once, LOADPLUS uses only the last option that you specify. For example, if you specify

```
LOAD DATA
     ORDER NO ORDER YES
```

  LOADPLUS accepts ORDER YES as the processing option.

- In the syntax diagrams in the following pages, underlined options indicate default options.

For more information about how to read syntax diagrams, see “Syntax Diagrams” on page xxiv.

BMC Software, Inc., Confidential and Proprietary Information

Alphabetical Listing of LOADPLUS Options

Table 3-1 lists the LOADPLUS command options in alphabetical order. The table provides the corresponding page numbers where you can find description and usage information about each option.

Table 3-1 LOADPLUS Command Options (Part 1 of 3)

Command Options	See Page
ACTIVE	3-139
ANALYZE	3-61
APCOLLECTION	3-22
APCOMMIT	3-21
APMAXAGENTS	3-21
APOWNER	3-23
APRETRYLIM	3-21
APRETRYVAL	3-22
ASCII	3-28
BMCSTATS	3-74
CCSID	3-28
CENTURY	3-70
CHECKPEND	3-67
CLUSTERRATIO	3-76
CONTINUEIF	3-51
COPY	3-80
COPYDDN	3-85
COPYLVL	3-84
COPYPEND	3-80
DATA	3-14
DATACLAS	3-153
DDTYPE	3-137
DEFAULTIF	3-108
DELETEFILES	3-57
DISCARDDN	3-39
DISCARDS	3-40
DSNPAT	3-145
DSNUEXIT	3-135
EBCDIC	3-28

BMC Software, Inc., Confidential and Proprietary Information

Table 3-1 LOADPLUS Command Options (Part 2 of 3)

Command Options	See Page
ENFORCE	3-66
ENUMROWS	3-46
ERRDDN	3-39
EXPDT	3-151
FORMAT	3-24
GDGLIMIT	3-152
IDCACHE	3-71
IDCDDN	3-57
IDERROR	3-73
IFALLOC	3-140
IGNORE	3-40
INDDN	3-36
INDEX	3-15
INDSN	3-34
INLINE	3-82
INTO TABLE	3-89
KEEPDICTIONARY	3-17, 3-92
LOAD	3-14
LOADDN	3-37
LOG	3-68
MAXEXTSZ	3-142
MAXSORTS	3-50
MAXTAPE	3-134
MGMTCLAS	3-154
NLPCTFREE	3-60
NOSUBS	3-29
NULLIF	3-108
ORDER	3-43
ORIGINALDISP	3-59
PART	3-90
POSITION	3-104
PREFORMAT	3-69, 3-90
PRELOAD	3-32
RECOVERYDDN	3-86
REDEFINE	3-54

BMC Software, Inc., Confidential and Proprietary Information

Table 3-1 LOADPLUS Command Options (Part 3 of 3)

Command Options	See Page
REGISTER	3-83
REPLACE	3-17, 3-91
REPORT	3-41
RESUME	3-15
RETPD	3-151
REUSE	3-56
SHRLEVEL	3-19
SIZEPCT	3-144
SKIPFIELDS	3-87
SKIPIX	3-30
SMS	3-141
SMSUNIT	3-141
SORTDEVT	3-49
SORTKEYS	3-31
SORTNUM	3-49
STORCLAS	3-154
SYNC	3-65
SYSDISC	3-58
THRESHLD	3-144
UNIQUECHECK	3-52
UNIQUEINTO	3-88
UNIT	3-143
UPDATEDB2STATS	3-76
UPDATEMAXA	3-72
WHEN	3-93
WORKDDN	3-38

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS Command Syntax Diagram

Figure 3-1 illustrates the syntax of the LOAD command.

Figure 3-1 LOADPLUS Command Syntax Diagram (Part 1 of 5)

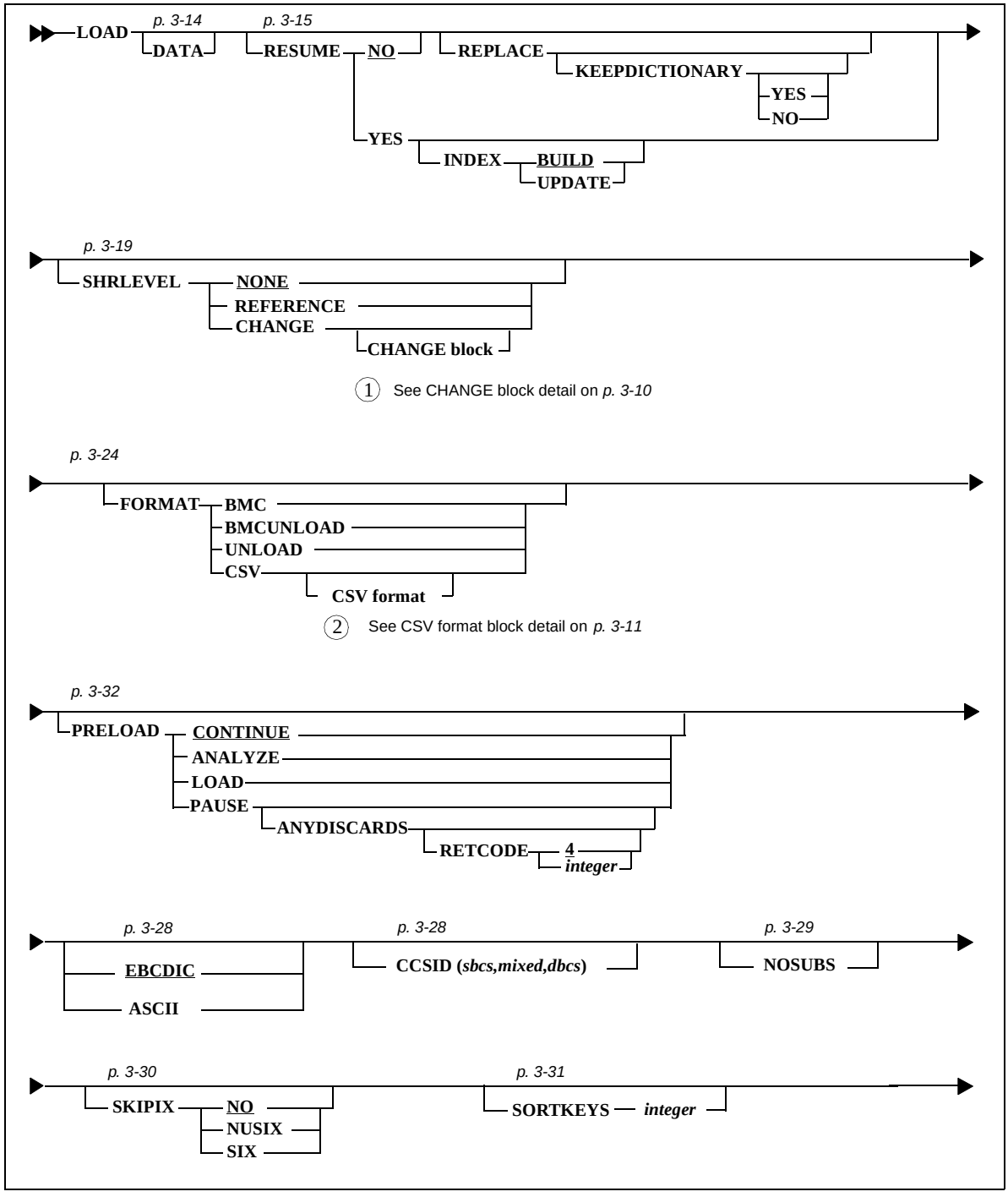


Figure 3-1 LOADPLUS Command Syntax Diagram (Part 2 of 5)

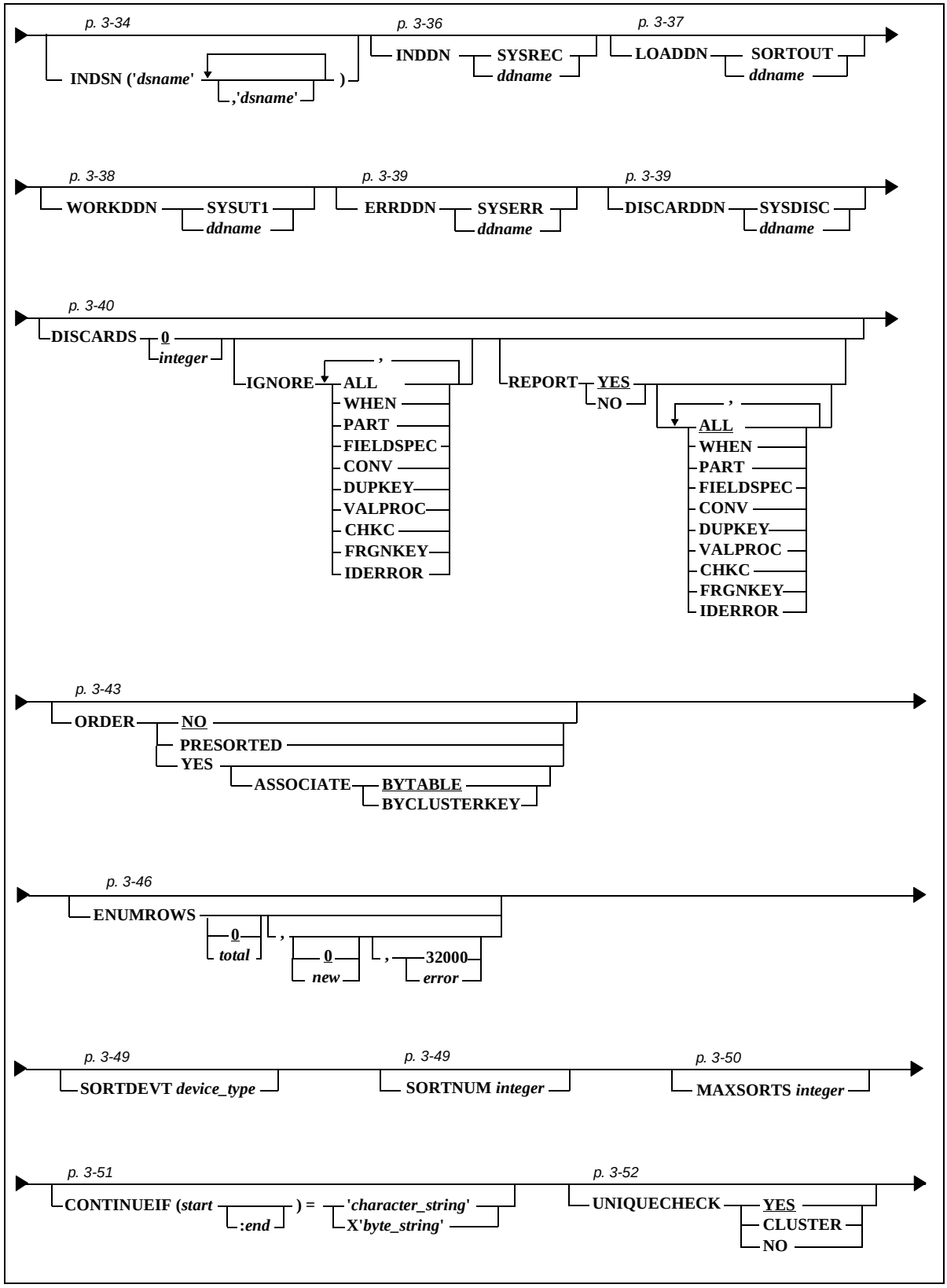


Figure 3-1 LOADPLUS Command Syntax Diagram (Part 3 of 5)

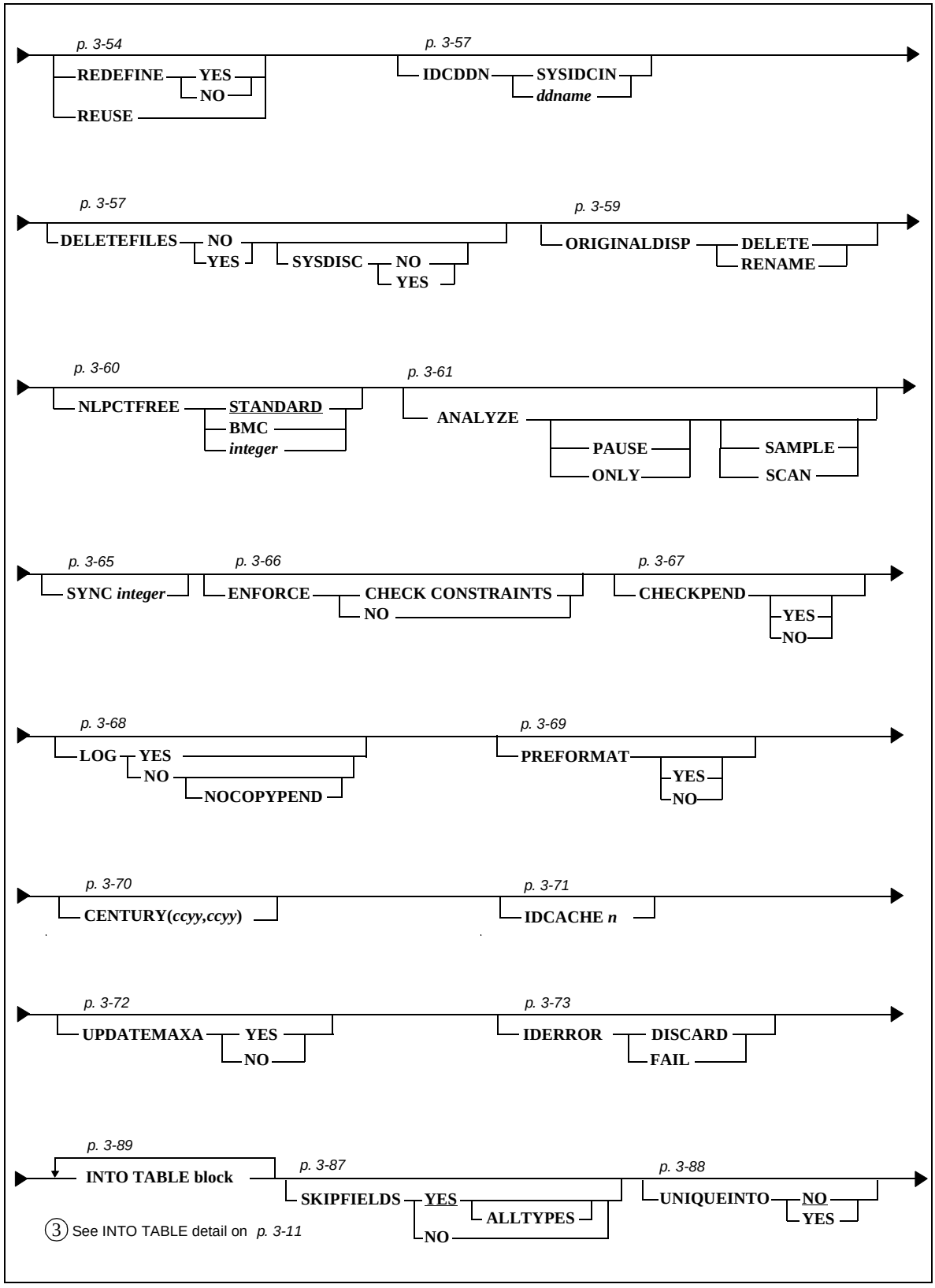


Figure 3-1 LOADPLUS Command Syntax Diagram (Part 4 of 5)

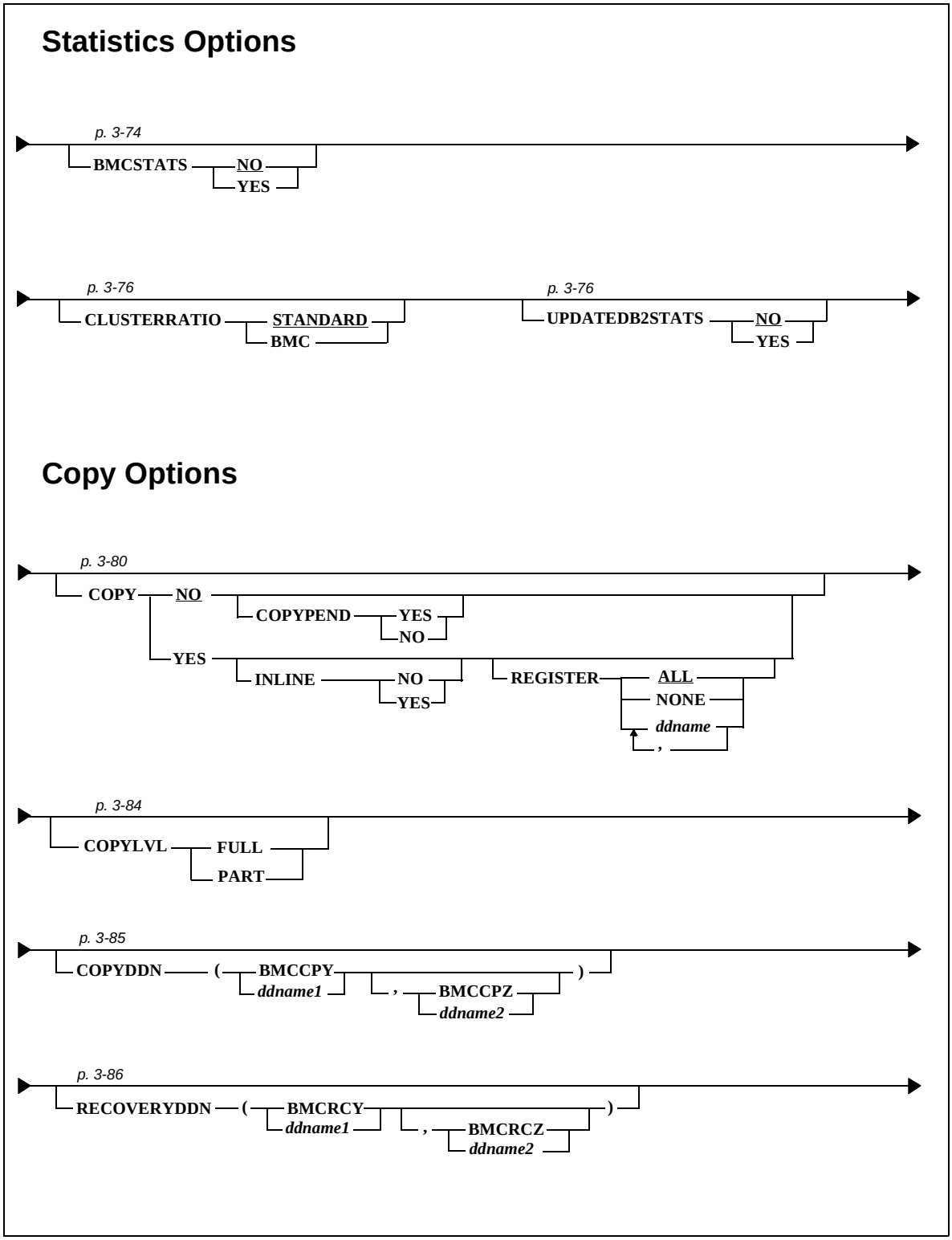
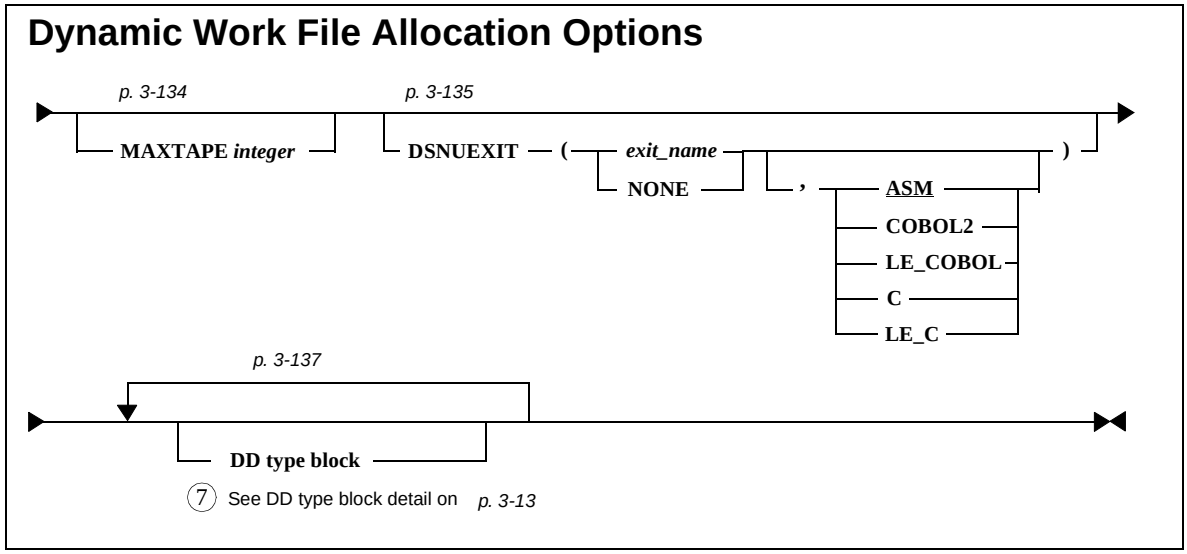


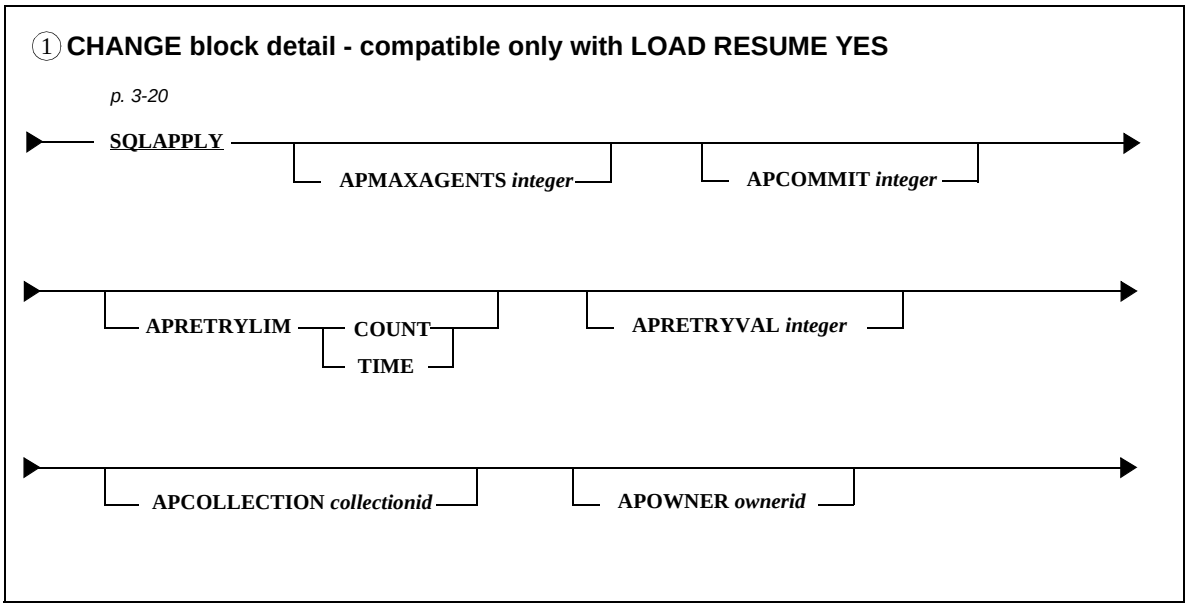
Figure 3-1 LOADPLUS Command Syntax Diagram (Part 5 of 5)



Syntax Diagram Details

This section shows syntax diagram details. Each detail diagram is numbered or lettered to correspond to a preceding section of the LOADPLUS command syntax or to another section of the detail.

Figure 3-2 Detail Syntax Diagrams (Part 1 of 4)



BMC Software, Inc., Confidential and Proprietary Information

Figure 3-2 Detail Syntax Diagrams (Part 2 of 4)

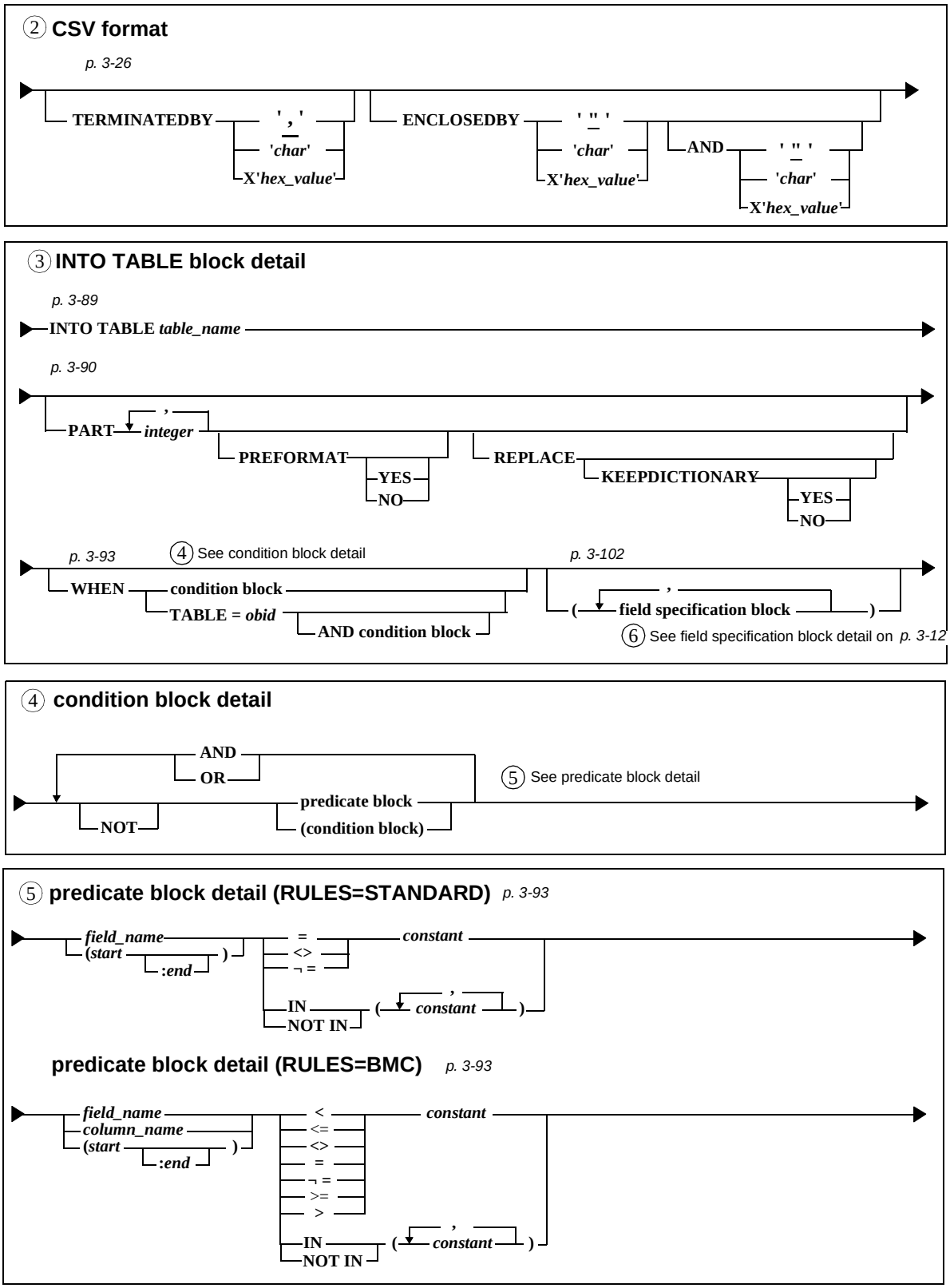
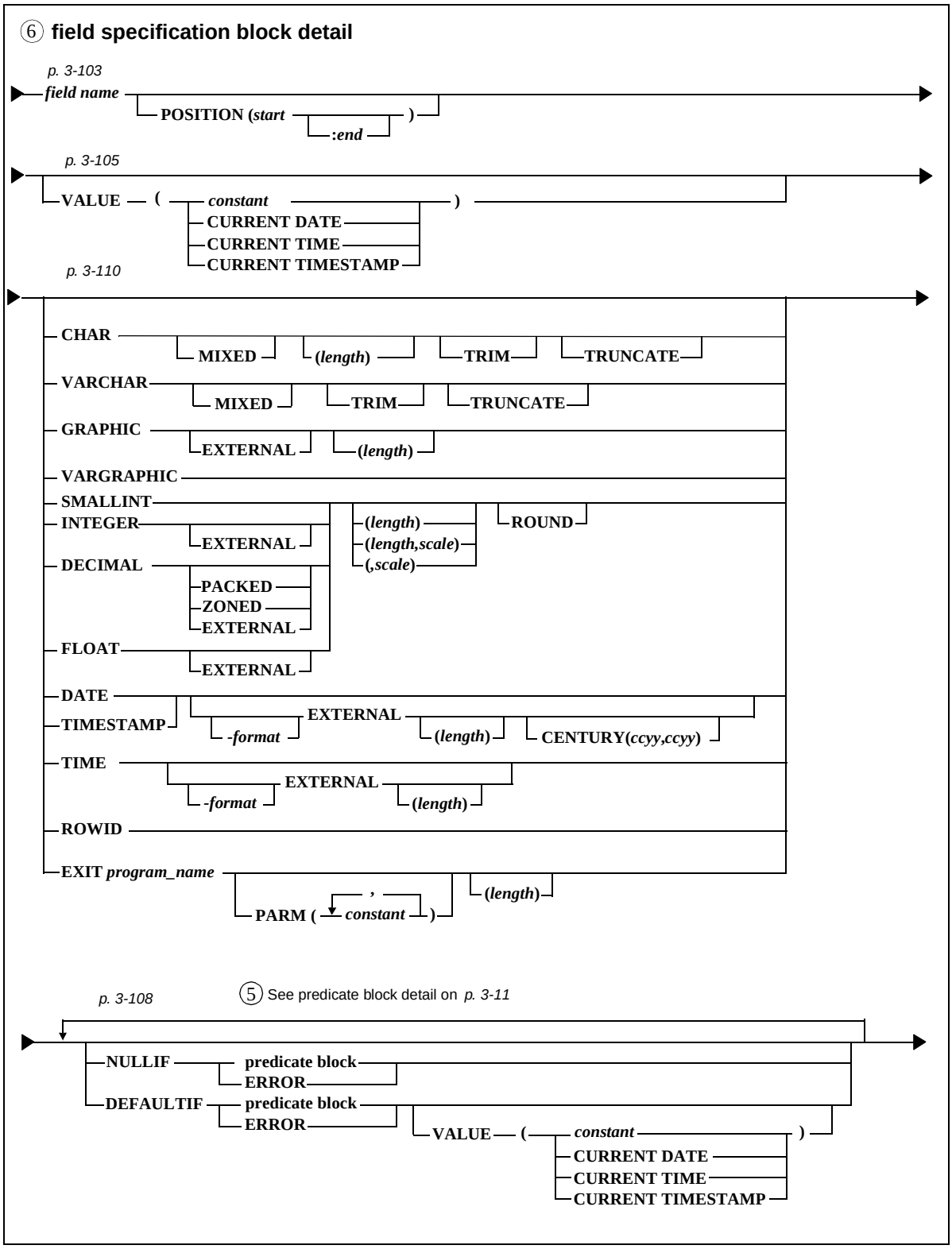
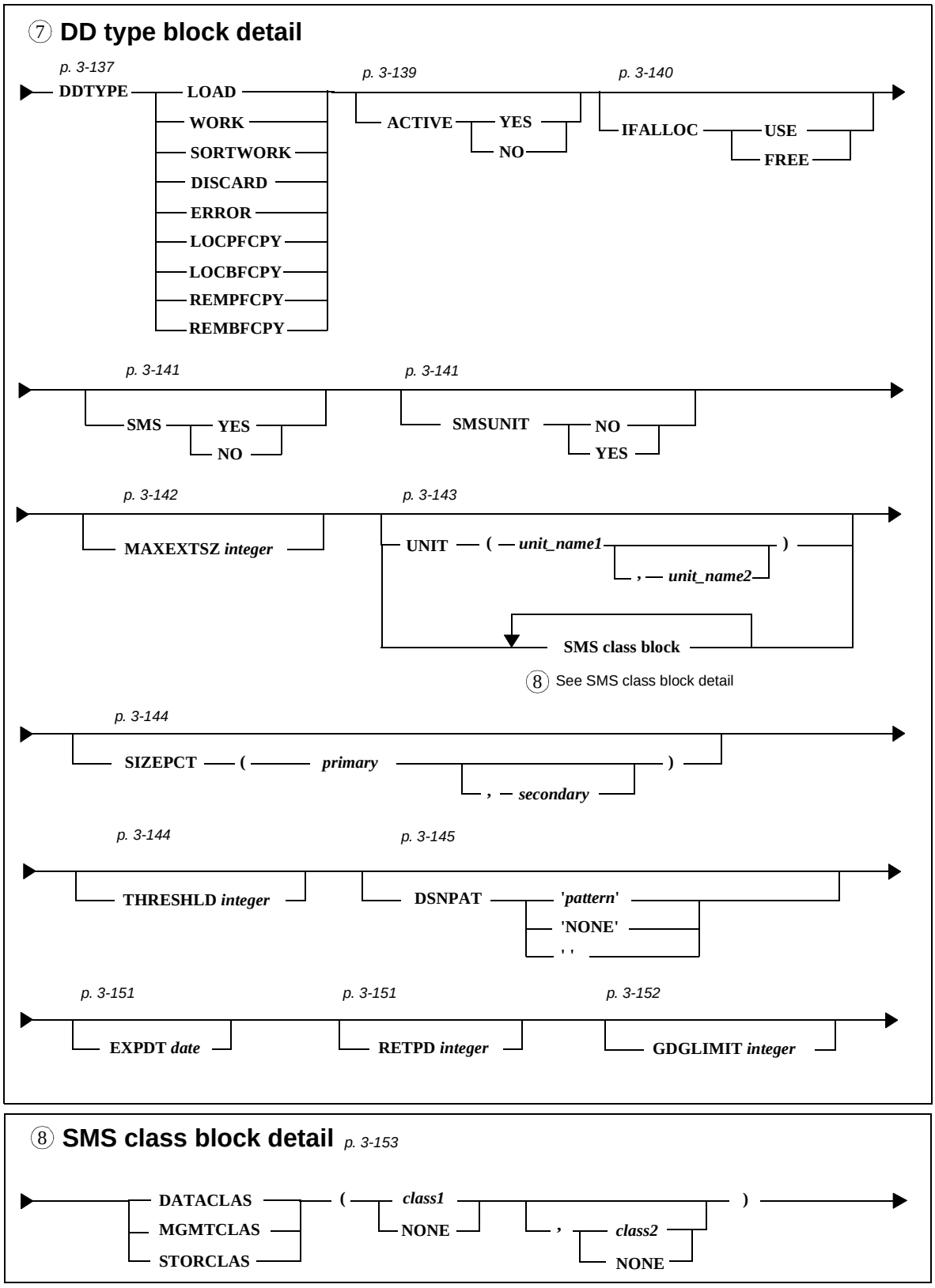


Figure 3-2 Detail Syntax Diagrams (Part 3 of 4)



BMC Software, Inc., Confidential and Proprietary Information

Figure 3-2 Detail Syntax Diagrams (Part 4 of 4)

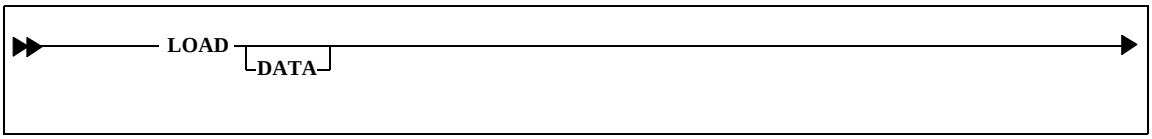


LOADPLUS Options

The remainder of this chapter describes each LOADPLUS option. The descriptions are in the order in which they appear in the preceding main syntax diagram.

Basic Processing Options

The basic load processing options control most aspects of LOADPLUS execution, except for copy options, statistics options, and the options that identify the tables and the data to be loaded. There are also additional options that control dynamic work file allocation.



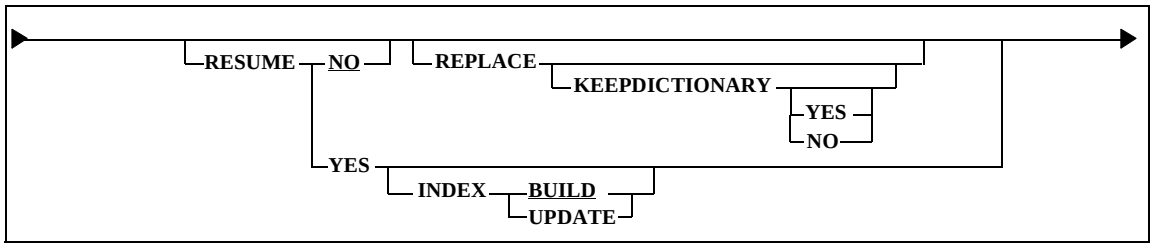
LOAD

LOAD is the keyword for the LOADPLUS utility command.

DATA

DATA is used only for compatibility with the IBM DB2 LOAD utility command, and LOADPLUS treats this option as a comment. With LOADPLUS, you identify the data to load by specifying the table name in the INTO TABLE option.

BMC Software, Inc., Confidential and Proprietary Information



RESUME

RESUME tells LOADPLUS whether to add data to an empty table space or to a table space that already contains data. RESUME support is available at the global or partition level.

NO

RESUME NO, the default, tells LOADPLUS that the table space is empty. If the table space is not empty and you do not specify the REPLACE option, LOADPLUS terminates.

YES

RESUME YES indicates that the table space might or might not be empty. LOADPLUS adds the new data to any existing data. Unless you specify INDEX UPDATE or SKIPIX, LOADPLUS unloads the existing indexes and merges them with the new index keys from the input data. LOADPLUS then sorts the merged index data and uses that data to build organized indexes during the LOAD phase or the COMBINED phase.

You cannot use this option with LOAD REPLACE. If you want to replace only specific partitions of a table space, specify RESUME YES INTO TABLE with the PART REPLACE option.

INDEX

This option tells LOADPLUS whether to build or update an index when specifying RESUME YES. BUILD is the default.

LOADPLUS ignores the INDEX option in the following cases:

- If you specify LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY (also referred to as a SQLAPPLY load), LOADPLUS does not build or update indexes.
- If you specify SKIPIX SIX or SKIPIX NUSIX, LOADPLUS does not build or update associated secondary indexes (SIX) or associated nonunique secondary indexes (NUSIX).

BMC Software, Inc., Confidential and Proprietary Information

BUILD

Under the BUILD option, which is the default, LOADPLUS unloads the existing indexes and merges them with the new index keys from the input data. LOADPLUS then sorts the merged index data and uses that data to build organized indexes during the LOAD phase or the COMBINED phase.

UPDATE

This option tells LOADPLUS to add index entries to the existing indexes. With INDEX UPDATE, LOADPLUS does not reorganize indexes as it does when you specify or default to INDEX BUILD.

Note: If you specify PART REPLACE, LOADPLUS always rebuilds the index for that partition.

BMC Software recommends that you limit the use of UPDATE to those cases where you are adding a small percentage of the total amount of existing data. If you are adding a large percentage, using UPDATE can impact optimal performance of the SQL that uses the index in processing.

Warning! When you specify INDEX UPDATE and a failure occurs during index update processing, you will need to recover your table space and indexes. If you attempt to restart, LOADPLUS terminates and issues message BMC51435S.

Additional Considerations—The following considerations apply to the INDEX UPDATE option:

— When you specify INDEX UPDATE for a two-phase load, you must also specify UNIQUECHECK NO. If you specify UNIQUECHECK YES or CLUSTER, LOADPLUS fails and displays message BMC51419E. See “UNIQUECHECK” on page 3-52 for more details.

— When you specify INDEX UPDATE, LOADPLUS does not unload the existing index records. Therefore, LOADPLUS cannot check for uniqueness in the PRELOAD phase. If duplicates exist, LOADPLUS detects them in the LOAD or COMBINED phase, loads the table space, deletes the duplicate records, and issues messages BMC50258E and BMC51477I. For a two-phase load, LOADPLUS does not write these duplicate records to the discard data set.

BMC Software, Inc., Confidential and Proprietary Information

— When you specify INDEX UPDATE and index update processing completes, but the job fails in the LOAD phase, you must restart the job by specifying RESTART without PHASE. If you specify RESTART(PHASE), LOADPLUS terminates and issues message BMC51436E.

REPLACE

REPLACE tells LOADPLUS to replace any existing data in the table space (not just the data in the tables or partitions that you specify in the INTO TABLE option). In other words, REPLACE tells LOADPLUS to delete the existing data before loading. If you specify the REPLACE option, you cannot also specify the PART REPLACE option.

Additional Considerations—The following additional considerations apply to the REPLACE option:

- LOADPLUS terminates if you specify REPLACE when loading a base table that is defined with LOB columns (BLOB, CLOB, and DBCLOB).

KEEPDICTIONARY

This option tells LOADPLUS whether to keep the existing compression dictionary. For compatibility with earlier releases, if you specify KEEPDICTIONARY without a value, LOADPLUS assumes KEEPDICTIONARY YES.

LOADPLUS also provides KEEPDICTIONARY support at the partition level. See “PART” on page 3-90 for information on how to use the PART REPLACE options.

You can specify the default for the KEEPDICTIONARY option in your installation options module (using the KEEPDICTIONARY installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

YES

If you specify KEEPDICTIONARY YES, LOADPLUS keeps the existing compression dictionary.

This option is valid only if the table space that you are loading has the COMPRESS YES attribute. If a dictionary already exists, LOADPLUS uses it for compression. If a dictionary does not exist, LOADPLUS builds one in the PRELOAD phase or the COMBINED phase. After completely building the dictionary, LOADPLUS compresses the data.

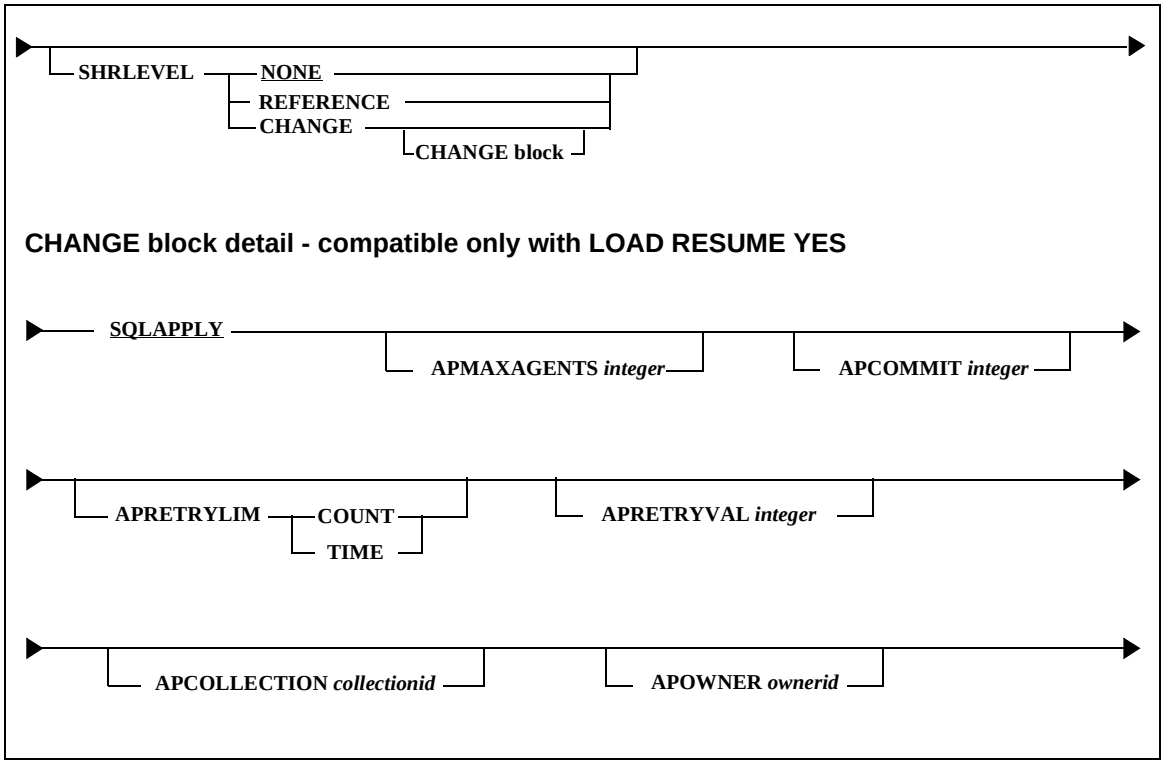
BMC Software, Inc., Confidential and Proprietary Information

NO

If you specify KEEPDICTIONARY NO, LOADPLUS builds a new compression dictionary.

This option is valid only if the table space that you are loading has the COMPRESS YES attribute. LOADPLUS builds the dictionary in the PRELOAD phase or the COMBINED phase. After completely building the dictionary, LOADPLUS compresses the data.

BMC Software, Inc., Confidential and Proprietary Information



SHRLEVEL

SHRLEVEL specifies the level of access that DB2 has to the objects that you are loading during LOADPLUS processing. See “SHRLEVEL Considerations” on page 2-24 for important information about using this option.

NONE

SHRLEVEL NONE, the default, tells LOADPLUS to stop the objects that you are loading, making them unavailable during the entire load process.

REFERENCE

(available only for LOAD REPLACE) SHRLEVEL REFERENCE tells LOADPLUS to make the objects that you are loading available in read-only status. LOADPLUS writes the loaded data to staging data sets. The original VSAM data sets remain intact throughout the load process. This nondestructive process allows you to easily restart from a failure or make the objects available without having to recover the objects.

This option is not available when you specify PART REPLACE.

BMC Software, Inc., Confidential and Proprietary Information

CHANGE

SHRLEVEL CHANGE allows the objects that you are loading to remain in read/write status.

For SHRLEVEL CHANGE with LOAD REPLACE, LOADPLUS writes the loaded data to staging data sets. The original VSAM data sets remain intact throughout the load process. This nondestructive process allows you to easily restart from a failure or make the objects available without having to recover.

Warning! LOAD REPLACE SHRLEVEL CHANGE fully replaces the data in the object, including any inserts, updates, or deletes to the base object that occur during the load process.

For SHRLEVEL CHANGE with LOAD RESUME YES, the object that you are loading remains in read/write status while LOADPLUS loads the data by using SQL insert processing. For this type of load, LOADPLUS integrates with the Apply Plus component of the BMC Software Log Master for DB2 product. This feature does not require that you have a license for the Log Master product but does require that the Apply Plus component be installed and made available through the STEPLIB, JOBLIB, or LINKLIST at runtime.

Note: See “SHRLEVEL CHANGE” on page 2-26 for important information about using this option, including compatibility with other LOAD command options.

If you specify SHRLEVEL CHANGE with LOAD RESUME YES, the following additional command options allow you to control your load job.

SQLAPPLY

SQLAPPLY specifies that you want to use the type of online load that integrates with Apply Plus. Currently, this option is the only one that is available for LOAD RESUME YES SHRLEVEL CHANGE, and specifying the keyword is optional.

Note: For ease of reference, this book refers to LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY as SQLAPPLY.

When you specify more than one of the following SQLAPPLY options, your syntax must follow the sequence that is shown in the syntax diagram. For example, if you specify APCOMMIT, it must always appear before APOWNER.

BMC Software, Inc., Confidential and Proprietary Information

APMAXAGENTS

APMAXAGENTS defines the maximum number of agents that Apply Plus can start for this load. Valid values for this option are 1 through the number of batch threads that are available in your DB2 subsystem. APMAXAGENTS affects only partitioned or multiple-table table spaces. LOADPLUS uses at most one agent per partition and table.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the MaxAgents parameter in the *Apply Plus Reference Manual*.

You can specify the default for the APMAXAGENTS option in your installation options module (using the APMXAGNT installation option). The command option overrides the default that was established at installation. For details, see Appendix A, "LOADPLUS Installation Options."

APCOMMIT

APCOMMIT specifies the maximum number of records that each Apply Plus agent is to load before issuing a COMMIT statement. Valid values for this option are 1 through 32767.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the StatementCount parameter of the CommitTriggers section in the *Apply Plus Reference Manual*.

You can specify the default for the APCOMMIT command option in your installation options module (using the APCOMMIT installation option). The command option overrides the default that was established at installation. For details, see Appendix A, "LOADPLUS Installation Options."

APRETRYLIM

APRETRYLIM tells Apply Plus what measure to use to determine the limit for retry attempts following SQL -911, -913, or -904 errors. The value for this option is used in conjunction with the APRETRYVAL option.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the RetryLimit parameter in the *Apply Plus Reference Manual*.

BMC Software, Inc., Confidential and Proprietary Information

You can specify the default for the APRETRYLIM command option in your installation options module (using the APRETLIM installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

COUNT

COUNT tells Apply Plus to base the limit for these retry attempts on the number of attempts that the APRETRYVAL option specifies.

TIME

TIME tells Apply Plus to retry until it reaches the number of seconds that the APRETRYVAL option specifies.

APRETRYVAL

APRETRYVAL provides Apply Plus with the retry limit for each unit of recovery in the load job. The value for this option is used in conjunction with the APRETRYLIM option. Valid values are 0 through 32767. A value of 0 tells Apply Plus not to retry.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the RetryValue parameter in the *Apply Plus Reference Manual*.

You can specify the default for the APRETRYVAL command option in your installation options module (using the APRETVAL installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

APCOLLECTION

APCOLLECTION provides Apply Plus with the ID for the collection to which Apply Plus dynamically binds packages during execution. You can specify a collection ID with a length of up to 18 bytes. You cannot specify a null value for this option.

You can specify the default for the APCOLLECTION command option in your installation options module (using the APCOLLECTION installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS passes this value to Apply Plus. If you do not specify a value for this command option, one of the following actions occurs:

— If the value of the APCOLLECTION installation option *is not* null, LOADPLUS passes the value of this installation option to Apply Plus.

— If the value of the APCOLLECTION installation option *is* null, LOADPLUS does not pass any value to Apply Plus and Apply Plus uses the value that is provided in its CollectionID parameter.

For additional information, see the *Apply Plus Reference Manual*.

APOWNER

APOWNER provides Apply Plus with the authorization ID that Apply Plus uses to bind the DB2 plan and packages for the apply request. You can specify an owner ID with a length of up to 8 bytes. You cannot specify a null value for this option.

You can specify the default for the APOWNER command option in your installation options module (using the APOWNER installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

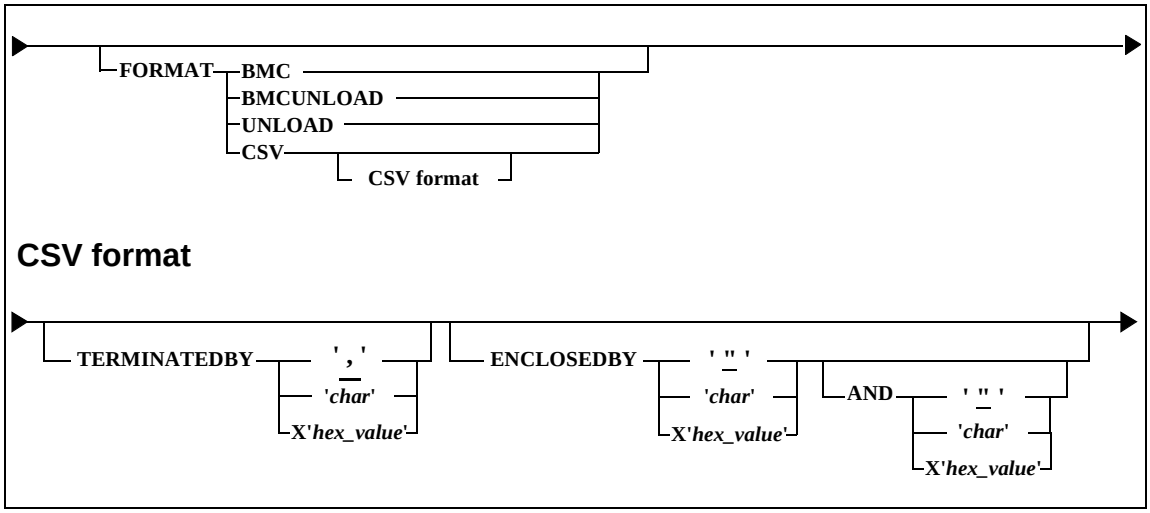
LOADPLUS passes this value to Apply Plus. If you do not specify a value for this command option, one of the following actions occurs:

— If the value of the APOWNER installation option *is not* null, LOADPLUS passes the value of this installation option to Apply Plus.

— If the value of the APOWNER installation option *is* null, LOADPLUS does not pass any value to Apply Plus and Apply Plus uses the value that is provided in its BindOwner parameter.

For additional information, see the *Apply Plus Reference Manual*.

BMC Software, Inc., Confidential and Proprietary Information



FORMAT

FORMAT specifies the format of the input data in the SYSREC data set when your input data is in one of the following formats:

- format generated by the BMC Software REORG PLUS for DB2 utility by using the SYSARC option
- format generated by the BMC Software UNLOAD PLUS for DB2 utility by using the FORMAT BMCLOAD option
- format generated by the DB2 REORG utility by using the UNLOAD ONLY option
- comma-separated-value (CSV) format

BMC

BMC specifies that the input data in the SYSREC data set is in the format that was generated by REORG PLUS by using the SYSARC data set (an archive data set that contains rows that are discarded during the reorganization as a result of SELECT and DELETE operations). If you specify FORMAT BMC, you cannot include field specifications on your LOAD command.

With the exception of EDITPROCs, the table that you are loading must have the same definition as the table from which the archive rows originated. LOADPLUS loads the records from the input data set that match the table OBID of the table that you specify in the INTO TABLE option or the OBID that you specify in the WHEN TABLE=*obid* option of the LOAD command.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS allows you to specify a WHEN condition with the FORMAT BMC option. However, the condition must reference columns of the table only; it cannot reference a (*start*:*end*) field. See the INTO TABLE and WHEN options beginning on page 3-89 for more information about specifying these options.

BMCUNLOAD

BMCUNLOAD specifies that the input data in the SYSREC data set is in an internal format that was generated by UNLOAD PLUS. This input data is data that was unloaded by using the UNLOAD PLUS option FORMAT BMCLOAD. For information about how to unload data in this format, see the *UNLOAD PLUS for DB2 Reference Manual*.

This option is useful for migrating data to duplicate tables or from development to production databases. This option provides an improved performance benefit because the data is never converted to an external format and because LOADPLUS does not need to perform data verification.

If you specify FORMAT BMCUNLOAD, you cannot include field specifications on your LOAD command. With a few exceptions, the table that you are loading must have the same definition as the table that was unloaded with UNLOAD PLUS. LOADPLUS loads the records from the input data set that match the table OBID that you specify in the WHEN TABLE=*obid* option of the LOAD command. See the WHEN TABLE option on page 3-93 for more information about specifying this option with FORMAT BMCUNLOAD.

For considerations when using the FORMAT BMCUNLOAD option, see “Data from UNLOAD PLUS in Internal Format” on page 2-44. For an example that uses this option, see “Example 14: LOAD REPLACE from UNLOAD PLUS Data in Internal Format” on page 5-106.

UNLOAD

This option specifies that the input data in the SYSREC data set is in the format that was generated by the DB2 REORG utility by using the UNLOAD ONLY option. If you specify FORMAT UNLOAD, you cannot include any field specifications on your LOAD command.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS loads the records from the input data set that match the object identifier (OBID) of the table that you specify in the INTO TABLE option or the OBID that you specify in the WHEN TABLE=*obid* option with FORMAT UNLOAD. Unlike the IBM DB2 LOAD utility, LOADPLUS allows you to specify a WHEN condition with the FORMAT UNLOAD option. However, the condition must reference columns of the table only; it cannot reference a (*start*:*end*) field. See the INTO TABLE and WHEN options beginning on page 3-89 for more information about specifying these options.

CSV

The CSV option allows you to load data that has been unloaded in comma-separated-value (CSV) format. This data can originate from any product that generates CSV output, including distributed databases, spreadsheet applications, and UNLOAD PLUS. In CSV-formatted data, a particular character separates each field from other fields. Also, in most cases, a particular pair of characters encloses each non-numeric field.

Note: CSV files are also referred to as comma-delimited or delimited files.

To use this option, your data and LOAD command must meet the requirements that are specified in “CSV Considerations” on page 2-39.

When you use more than one of the following CSV options, your syntax must follow the sequence that is shown in the syntax diagram. For example, if you specify TERMINATEDBY, it must always appear before ENCLOSEDBY.

TERMINATEDBY

This option specifies the character that your input data uses to delimit fields of data. The default is a comma (,). You cannot specify a null value for TERMINATEDBY.

ENCLOSEDBY

The ENCLOSEDBY option specifies the character that your input data uses on the left side to enclose fields of data. The default is a double quotation mark ("). You cannot specify a null value for ENCLOSEDBY.

For each field, if LOADPLUS does not find the ENCLOSEDBY character that you specify or default to, LOADPLUS assumes that the field is not enclosed by any character.

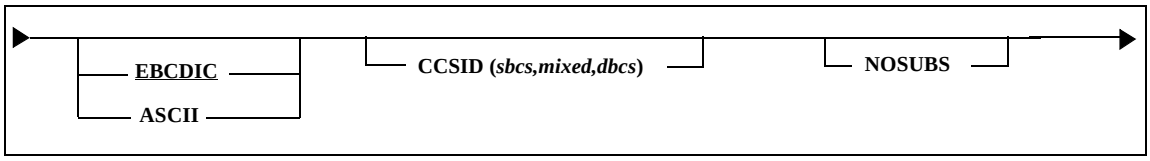
BMC Software, Inc., Confidential and Proprietary Information

AND

This option specifies the character that your input data uses on the right side to enclose fields of data. You cannot specify a null value for AND.

If you do not specify a value for this option, LOADPLUS assumes that your input data uses the value that you specified for the ENCLOSEDBY option. If you did not specify a value for the ENCLOSEDBY option, LOADPLUS assumes that your input fields are enclosed by double quotation marks or are not enclosed by any character.

BMC Software, Inc., Confidential and Proprietary Information



Note: LOADPLUS does not currently support Unicode encoding schemes. If you attempt to load data that is encoded with a Unicode encoding scheme or if your subsystem's default encoding scheme is Unicode, LOADPLUS terminates.

EBCDIC

EBCDIC specifies that the input data is encoded in EBCDIC using the DB2 installation default coded character set identifiers (CCSIDs). EBCDIC is the default.

LOADPLUS ignores this option if you specify it with FORMAT BMCUNLOAD.

ASCII

The ASCII option specifies that the input data is encoded in ASCII using the DB2 installation default CCSIDs.

LOADPLUS ignores this option if you specify it with FORMAT BMCUNLOAD.

CCSID

This option allows you to specify up to three CCSIDs used to encode input data. The three CCSIDs are for single-byte character set (SBCS), MIXED character set, and double-byte character set (DBCS) data. If a CCSID is specified as 0 or omitted, LOADPLUS uses the corresponding DB2 system default CCSID. For more details, see "Data Translation" on page 3-129.

LOADPLUS supports translation from one CCSID to another with the following restrictions:

- MIXED and DBCS translations are not supported.
- Translations that require the use of a conversion procedure are not supported.
- LOADPLUS ignores this option if you specify it with FORMAT BMCUNLOAD.

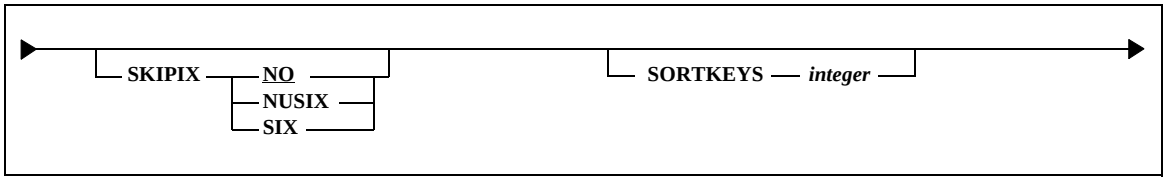
BMC Software, Inc., Confidential and Proprietary Information

NOSUBS

NOSUBS tells LOADPLUS not to accept substitution characters during translation between CCSIDs. If you specify NOSUBS and LOADPLUS encounters a record that requires substitution, LOADPLUS does not load the record, but writes it to your discard file.

LOADPLUS ignores this option if you specify it with FORMAT BMCUNLOAD.

BMC Software, Inc., Confidential and Proprietary Information



SKIPIX

The SKIPIX option tells LOADPLUS whether to skip building or updating the secondary indexes that are associated with the partitioned table space that you are loading. Skipped indexes no longer participate in any portion of the load job, including checking for duplicate keys.

Warning! BMC Software strongly recommends that you do not specify SKIPIX SIX to ignore unique secondary indexes if there is any possibility of loading duplicate values. If running with this option results in loading duplicate values, follow your standard procedures for eliminating duplicates when creating a unique index.

If you specify NUSIX or SIX, consider the following information:

- If you specify NUSIX or SIX under one of the following circumstances, LOADPLUS terminates with a return code 8:

  — You are loading a nonpartitioned table space. LOADPLUS issues message BMC50119E.

  — You are running a SQLAPPLY load job. LOADPLUS issues message BMC50115E.

- With the following exception, the nonparticipating indexes can be in any restrictive status at the beginning of the load job. Exception: If the nonparticipating indexes were created as DEFINE NO, they must not be in any restrictive status.

- With the following exception, LOADPLUS leaves the nonparticipating indexes in one of the rebuild pending statuses (RBDP, RBDP*, or PSRBD) and completes with a return code 4. Exception: If you specify RESUME YES (without PART REPLACE) and no rows are loaded, LOADPLUS leaves the nonparticipating indexes in the same status that they were in at the start of the load job.

  In those cases where LOADPLUS leaves the nonparticipating indexes in a rebuild pending status, you must take appropriate action to remove the pending status (for example, rebuilding the index) following the load job.

BMC Software, Inc., Confidential and Proprietary Information

NO

SKIPIX NO, the default, tells LOADPLUS to build or update all secondary indexes that are associated with the table that you are loading.

NUSIX

SKIPIX NUSIX tells LOADPLUS to not build or update any nonunique secondary indexes that are associated with the table that you are loading. See page 3-30 for considerations when specifying this option.

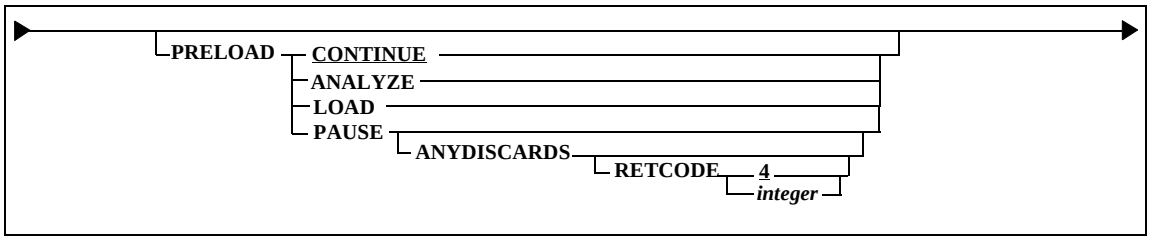
SIX

SKIPIX SIX tells LOADPLUS to not build or update any secondary indexes, unique or nonunique, that are associated with the table that you are loading. See the warning and information to consider on page 3-30 when specifying this option.

SORTKEYS

This option is used only for compatibility with the IBM DB2 LOAD utility command syntax. LOADPLUS treats this option as a comment.

BMC Software, Inc., Confidential and Proprietary Information



PRELOAD

The PRELOAD option controls the execution of LOADPLUS.

CONTINUE

PRELOAD CONTINUE, the default, directs LOADPLUS to continue with the LOAD phase after the PRELOAD phase is completed.

ANALYZE

The PRELOAD ANALYZE option directs LOADPLUS to stop after the optimization phase of PRELOAD. No data is read and no tables are affected. LOADPLUS displays message BMC51496I indicating the optimal number of SORT tasks and READER tasks, regardless of the number of SORTOUT and SYSREC data sets that you specified in your JCL. You can use this information to adjust the number of SORTOUT data sets that you specify in your JCL.

LOAD

If you specify PRELOAD LOAD, LOADPLUS uses single-phase load processing, which combines the functions of the PRELOAD and LOAD phases into the COMBINED phase. LOADPLUS reads the input data and copies it directly to the table space. This phase can improve performance in most cases. See “Single-Phase Load Performance Considerations” on page 6-35 for details.

PAUSE

The PRELOAD PAUSE option directs LOADPLUS to pause after the PRELOAD phase. You can restart the utility at the beginning of the LOAD phase.

If you also specify LOAD REPLACE, the table space being loaded is not stopped during the PRELOAD phase. By specifying PRELOAD PAUSE, you can schedule running the LOAD phase so that it has the least impact on data availability.

BMC Software, Inc., Confidential and Proprietary Information

PRELOAD PAUSE is not valid with a SQLAPPLY load. If you specify this option with a SQLAPPLY load, LOADPLUS issues message BMC50115E and terminates with return code 8.

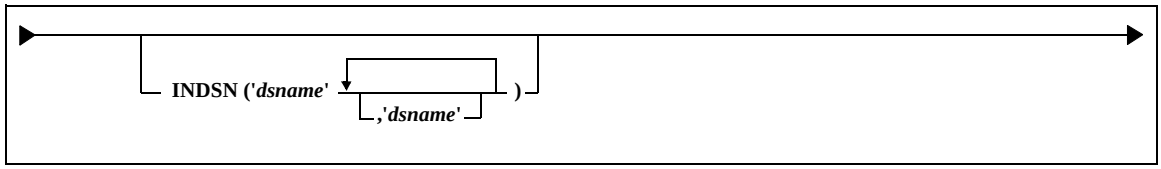
ANYDISCARDS

The PRELOAD PAUSE ANYDISCARDS option directs LOADPLUS to pause after the PRELOAD phase only if input records are discarded. You can restart the utility in the LOAD phase.

RETCODE

RETCODE 4 is the default. This option of PRELOAD PAUSE ANYDISCARDS allows you to designate the return code when the utility pauses because of discards. If the integer is greater than 31, LOADPLUS issues a user abend that is equal to the integer specified. If the integer specified is greater than 4095, LOADPLUS issues system abend 001.

BMC Software, Inc., Confidential and Proprietary Information



INDSN

The INDSN option enables dynamic allocation for your input (SYSREC) data sets. You can specify from 1 through 256 data set names, enclosed in single quotation marks, and separated by commas. LOADPLUS allocates the input data sets for use during the load job.

Use this option for one or more of the following reasons:

- when loading data from UNLOAD PLUS by using FORMAT BMCUNLOAD
- when you want to load data from different input data sets by updating just your syntax cards
- to eliminate the need to override SYSREC DD statements in your cataloged JCL procedures (PROCs) with SYSREC DD statements in your JCL

The input data sets must already exist and must be cataloged tape or DASD data sets. You cannot specify any parameters for these data sets (such as VOLSER, DCB, or UNIT) other than the data set name.

Considerations—The following considerations apply to the INDSN option. In addition, because each data set name that you specify with INDSN is assigned to a separate SYSREC, all considerations that apply to multiple SYSREC data sets apply to specifying multiple data set names on this option. See “SYSRECnn Data Sets” on page 4-18 for additional information.

- You can specify a GDG name as your data set name. LOADPLUS interprets a digit, plus sign, or hyphen following an open parenthesis in *dsname* as an indication that the data set name is a GDG name.
- You can specify a partitioned data set (PDS) name as your data set name as long as you include a member name. LOADPLUS interprets an alphabetic character following an open parenthesis in *dsname* as the beginning of a member name of a PDS.
- You can specify a dummy data set by specifying INDSN('NULLFILE').
- You cannot use this option when loading data from a batch pipe.

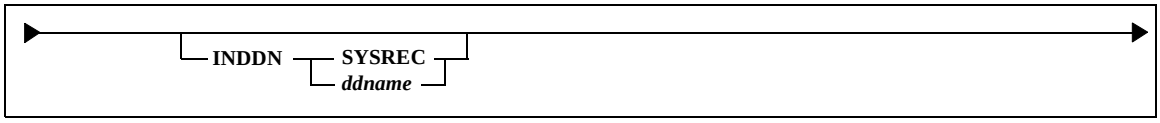
BMC Software, Inc., Confidential and Proprietary Information

- You cannot include SYSREC DD statements in your JCL when you specify this option. If there are SYSREC DD statements in your JCL, LOADPLUS fails and issues message BMC50460E.

- You can use the INDDN option to override the ddname prefix for your input data sets during dynamic allocation. For information about the INDDN option, see page 3-36.

- You cannot specify multiple data sets that are on a stacked tape.

BMC Software, Inc., Confidential and Proprietary Information



INDDN

The INDDN option allows you to override the default ddname (SYSREC) or ddname prefix of the input data set.

Note: You can specify the default for the INDDN command option in your installation options module (using the INDDN installation option). The command option overrides the default that was established at installation. For details, see Appendix A, "LOADPLUS Installation Options."

The value for INDDN tells LOADPLUS which DD statements in your JCL are input data sets. The following considerations apply when you use this option:

- If you also specify INDSN on your LOAD command, LOADPLUS uses the value for INDDN as the ddname or ddname prefix when constructing your input data set DD statements. If any DD statements in your JCL match the INDDN value, LOADPLUS fails and issues message BMC50460E.

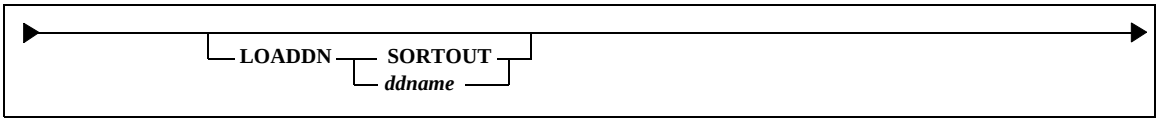
- The following considerations apply when you *do not* also specify INDSN on your LOAD command:

  — If you use the default ddname SYSREC in your JCL, you do not need to use this option. However, if you want to use a ddname other than SYSREC, you must specify it both in this option and in your JCL.

  — If you use multiple data sets, thus specifying only the ddname prefix in this option, you must append *nn* to the DD statements in your JCL.

If you use multiple input data sets, specify only the ddname prefix (no *nn*) in this option. This prefix can have a maximum of seven characters. For detailed information about using the SYSREC data set in LOADPLUS, see "SYSRECnn Data Sets" on page 4-18.

BMC Software, Inc., Confidential and Proprietary Information



LOADDN

The LOADDN option allows you to override the default ddname (SORTOUT) or ddname prefix for the output data set from the PRELOAD phase.

Note: You can specify the default for the LOADDN command option in your installation options module (using the LOADDN installation option). The command option overrides the default that was established at installation. For details, see Appendix A, "LOADPLUS Installation Options."

For multiple SORTOUT data sets, specify only the ddname prefix (no *n*) in this option. However, in your JCL DD statements, you must append *n* to the prefix that you specify in this option. This prefix can have a maximum of seven characters. For detailed information about using the SORTOUT data set in LOADPLUS, see "SORTOUTn Data Sets" on page 4-12.

If you use the default ddname SORTOUT in your JCL, you do not need to use this option. However, if you want to use a ddname other than SORTOUT, you must specify it both in this option and in your JCL.

Dynamic work file allocation—If dynamic work file allocation is active and you specify more than one ddname prefix for dynamic allocation, the prefix for each ddname must be different enough for LOADPLUS to differentiate one prefix from another. *Different enough* means that these prefixes must be different *and*, if they are different only because one prefix has additional trailing bytes, then these trailing bytes must contain at least one nonnumeric byte. For example, the first set of prefixes that follow is different enough, but the second set is not:

- acceptable:

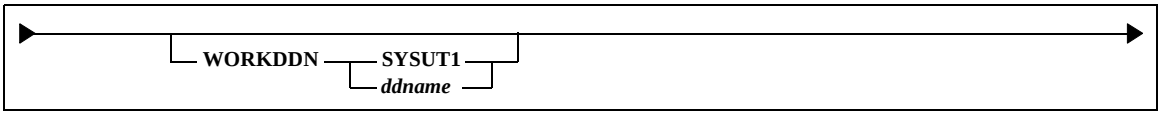
```
BMCRD
BMCRDWK
```

- not acceptable:

```
BMCRD
BMCRD11
```

To dynamically allocate more than nine SORTOUT data sets, do not use the default ddname. Instead, use this option to specify a ddname prefix that has six or fewer characters.

BMC Software, Inc., Confidential and Proprietary Information



WORKDDN

The WORKDDN option allows you to override the default ddname (SYSUT1) or ddname prefix of the work data set.

Note: You can specify the default for the WORKDDN command option in your installation options module (using the WORKDDN installation option). The command option overrides the default that was established at installation. For details, see Appendix A, "LOADPLUS Installation Options."

If you use multiple work data sets, specify only the ddname prefix (no *nn*) in this option. However, in your JCL DD statements, you must still append *nn* to the prefix that you specified in this option. For detailed information about using the SYSUT1 data set in LOADPLUS, see "SYSUT1nn Data Sets" on page 4-20.

If you use the default ddname SYSUT1 in your JCL, you do not need to use this option. However, if you want to use a ddname other than SYSUT1, you must specify it both in this option and in your JCL.

Dynamic work file allocation—If dynamic work file allocation is active and you specify more than one ddname prefix for dynamic allocation, the prefix for each ddname must be different enough for LOADPLUS to differentiate one prefix from another. *Different enough* means that these prefixes must be different *and*, if they are different only because one prefix has additional trailing bytes, then these trailing bytes must contain at least one nonnumeric byte. For example, the first set of prefixes that follow is different enough, but the second set is not:

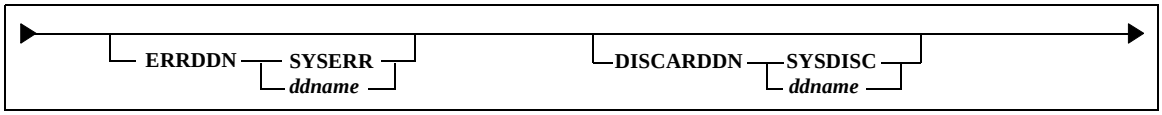
- acceptable:

```
BMCRD
BMCRDWK
```

- not acceptable:

```
BMCRD
BMCRD11
```

BMC Software, Inc., Confidential and Proprietary Information



ERRDDN

The ERRDDN option allows you to override the default data set ddname (SYSERR). This data set contains information about records that are discarded because of errors. For detailed information about using the SYSERR data set in LOADPLUS, see “SYSERR Data Set” on page 4-17.

If you use the default ddname SYSERR in your JCL, you do not need to use this option. However, if you want to use a ddname other than SYSERR, you must specify it both in this option and in your JCL.

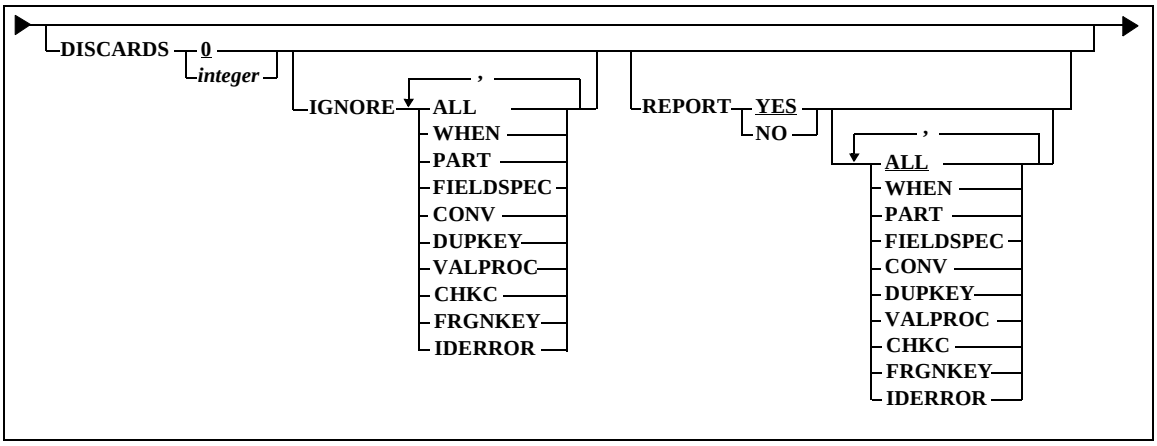
You can specify the default for the ERRDDN command option in your installation options module (using the ERRDDN installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

DISCARDDN

The DISCARDDN option allows you to override the default ddname (SYSDISC) of the discard data set. If you use the default ddname SYSDISC in your JCL, you do not need to use this option. However, if you want to use a ddname other than SYSDISC, you must specify it both in this option and in your JCL. For detailed information about using the SYSDISC data set in LOADPLUS, see “SYSDISC Data Set” on page 4-15.

You can specify the default for the DISCARDDN command option in your installation options module (using the DISCDDN installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

BMC Software, Inc., Confidential and Proprietary Information



DISCARDS

The DISCARDS option defines the limit on the number of discard records. If LOADPLUS reaches the limit, the job terminates. You must specify a SYSDISC data set when you use this option (see “SYSDISC Data Set” on page 4-15).

The default is 0 (no limit) when you do not specify the DISCARDS keyword. When you specify the DISCARDS keyword, you must supply a value and this value must be an integer (which can be 0).

IGNORE

The IGNORE option tells LOADPLUS to ignore the specified discard types during the PRELOAD phase or the COMBINED phase. If the specified type of discard occurs, LOADPLUS does not write an error record to the SYSERR data set (unless it is a duplicate key error), does not report the error in the error summary report, and does not write the corresponding input record to the discard data set if all discards were ignored for the record.

Note: If you use MSGLEVEL(1), LOADPLUS issues messages BMC51501E, BMC51502E, BMC51503E, BMC51505E, and BMC51506E to help diagnose problems regardless of the IGNORE option.

The IGNORE option is especially useful for WHEN and PART discards when all records in your input file do not participate in the load.

Table 3-2 on page 3-41 lists the keywords and discard types, along with their descriptions, that you can specify with the IGNORE option.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-2 Valid Discard Types Used with the IGNORE Option

Discard Type	Description
ALL	Ignore all discard types.
WHEN	Ignore records that are not selected by any INTO statement or that are discarded as a result of a table or column definition mismatch when specifying FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD.
PART	Ignore records that are not selected by any parts being loaded.
FIELDSPEC	Ignore records that are discarded because of an error on the field specification.
CONV	Ignore records that are discarded because of an error in conversion.
DUPKEY	Ignore records that are discarded because of a duplicate key.
VALPROC	Ignore records that are discarded because of a validation procedure.
CHKC	Ignore records that are discarded because of table check constraints.
FRGNKEY	Ignore records that are discarded because of an invalid value in a foreign key. This option is applicable only for SQLAPPLY load.
IDERROR	Ignore records that are discarded because they are outside the range that is defined on the identity column.

For additional information about how to enhance LOADPLUS performance when using this option, see “DISCARDS IGNORE Command Option” on page 6-25.

REPORT

The REPORT option tells LOADPLUS which discard types to report or not report in the error summary report. Although this option has no effect on actual discard processing, it allows you to limit the number of SYSPRINT records when you expect many discards.

YES

When you specify DISCARDS REPORT YES, the discard types that you specify are reported in the LOADPLUS error summary report. DISCARDS REPORT YES ALL is the default.

LOADPLUS does not report any discard types that you specify with the IGNORE option, even if you specify them in the REPORT YES option.

BMC Software, Inc., Confidential and Proprietary Information

NO

The discard types that you specify in this option are not reported in the LOADPLUS error summary report. This option is beneficial when you expect many discards.

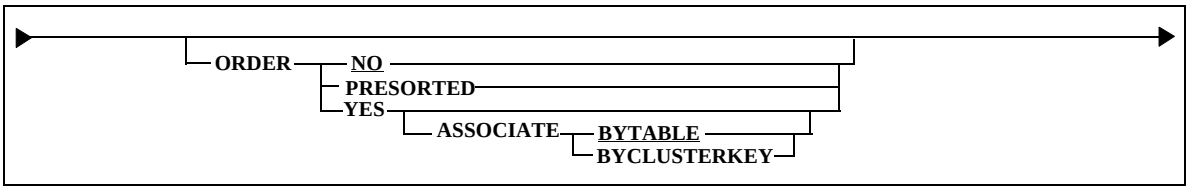
REPORT NO has no effect on actual discard processing. Depending on the use of the IGNORE option, LOADPLUS still writes error records to the SYSERR data set and still writes corresponding input records to the discard data set.

Table 3-3 lists the keywords and discard types, along with their descriptions, that you can specify with the REPORT option.

Table 3-3 Valid Discard Types Used with the REPORT Option

Discard Type	Description
ALL	Affects all discard types.
WHEN	Affects records that are not selected by any INTO statement or that are discarded as a result of a table or column definition mismatch when specifying FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD.
PART	Affects records that are not selected by any parts being loaded.
FIELDSPEC	Affects records that are discarded because of an error on the field specification.
CONV	Affects records that are discarded because of an error in conversion.
DUPKEY	Affects records that are discarded because of a duplicate key.
VALPROC	Affects records that are discarded because of a validation procedure.
CHKC	Affects records that are discarded because of table check constraints.
FRGNKEY	Affects records that are discarded because of an invalid value in a foreign key. This option is applicable only for SQLAPPLY load.
IDERROR	Affects records that are discarded because they are outside the range that is defined on the identity column.

BMC Software, Inc., Confidential and Proprietary Information



ORDER

The ORDER option tells LOADPLUS how and whether to order the table rows. ORDER affects only new rows that you are loading. For a LOAD RESUME YES job, LOADPLUS does not mix new rows with old rows.

You can always omit the ORDER option from your LOAD command without error. If you do not specify ORDER, the default is ORDER NO except when you are loading a multiple-table, segmented table space. In this case, ORDER YES is the default.

Although you can omit the ORDER option without error, if you explicitly specify ORDER, the restrictions in Table 3-4 apply.

Table 3-4 Restrictions for Specifying the ORDER Option

ORDER YES			
ASSOCIATE BYTABLE	**ASSOCIATE BYCLUSTERKEY**	**ORDER NO**	**ORDER PRESORTED**
This option is meaningless for a single-table table space with no clustering index. In this case, LOADPLUS processes this option as if you specified ORDER NO.	Do not specify this option if there is no clustering index or for a multiple-table, segmented table space. LOADPLUS changes this option to ASSOCIATE BYTABLE for a SQLAPPLY load when referential integrity exists between the objects that you are loading.	Do not specify this option if you are loading multiple tables in a segmented table space unless you are performing a SQLAPPLY load. You *can* specify this option if you are loading one table of a multiple-table, segmented table space.	This option is meaningful for a LOAD REPLACE or LOAD RESUME NO of a single table with a single SYSREC data set. This option is also meaningful for a single-phase LOAD RESUME YES PART REPLACE if you are replacing all partitions that are involved in the load, have no participating nonclustering indexes, and are using only one SYSREC data set.

For ORDER PRESORTED and ORDER YES, the encoding scheme of the table determines the collating sequence for character data. For example, if the table is defined as EBCDIC, the collating sequence will be EBCDIC. If the table is defined as ASCII, the collating sequence will be ASCII.

BMC Software, Inc., Confidential and Proprietary Information

NO

ORDER NO is the default except when you are loading a multiple-table, segmented table space. (In this case, ORDER YES is the default.) If you specify ORDER NO, LOADPLUS performs no ordering at all, and the rows retain the order of the input data set from which they were read. The table space might require reorganization for adequate DB2 performance. LOADPLUS sorts clustering indexes together with participating nonclustering indexes.

PRESORTED

If you specify ORDER PRESORTED, LOADPLUS verifies that the rows are in the correct clustering index order. LOADPLUS performs no sorts on either the data or the clustering index and terminates the job if the data is not in the correct order.

You must specify only one SYSREC data set when using ORDER PRESORTED. Otherwise, LOADPLUS terminates the job.

ORDER PRESORTED is valid for the following types of load jobs. If you use this option with other types of load jobs, LOADPLUS issues message BMC50115E or BMC51430E and terminates processing.

- LOAD REPLACE or LOAD RESUME NO of a single table
- single-phase LOAD RESUME YES PART REPLACE of a partitioned table space if you are replacing all partitions that are participating in the load and no nonclustering indexes are participating in the load

YES

If you specify ORDER YES, LOADPLUS orders the rows in the tables of the table space by their clustering key.

ASSOCIATE

Use the ASSOCIATE option to specify how you want LOADPLUS to sort the rows for multiple-table table spaces.

BYTABLE

BYTABLE, which is the default if you specify ORDER YES, tells LOADPLUS to sort the rows by table as well as by each table’s clustering key. If no clustering key exists, LOADPLUS uses X'00's. LOADPLUS always sorts segmented table spaces by table.

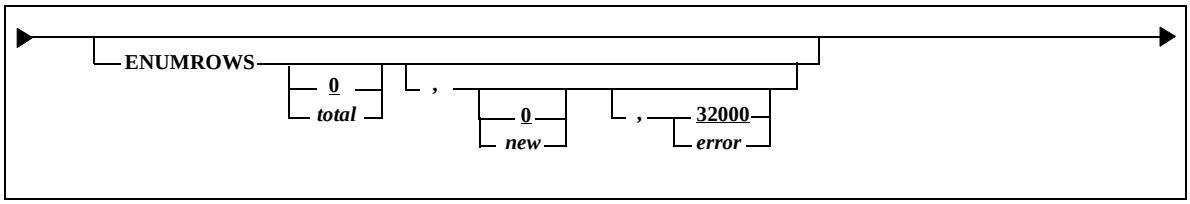
BMC Software, Inc., Confidential and Proprietary Information

BYCLUSTERKEY

BYCLUSTERKEY sorts the rows by each table's clustering key only. This option is useful for multiple-table, nonsegmented table spaces to group rows of different tables together by a common clustering key.

Note: For a SQLAPPLY load when referential integrity exists between the objects you are loading, LOADPLUS changes this option to ASSOCIATE BYTABLE and continues processing.

BMC Software, Inc., Confidential and Proprietary Information



ENUMROWS

The ENUMROWS option provides estimated information that LOADPLUS and BMCSORT use to dynamically allocate your work files and to determine the most efficient method for performing sort processing. If you specify ENUMROWS, you must specify a value for at least one of the parameters (*total*, *new*, or *error*). These parameters are positional.

Use ENUMROWS if any of the following conditions apply:

- You specify the ANALYZE option.

  You must specify the appropriate *total* or *new* parameter so that LOADPLUS can accurately estimate the size of the work files.

- Dynamic work file allocation is active.

  You must specify the appropriate *total* or *new* parameter so that LOADPLUS can accurately estimate the size of the work files.

- BMCSORT is dynamically allocating the sort work data sets.

  BMC Software recommends that you specify ENUMROWS to allow BMCSORT to most accurately and efficiently allocate your sort work data sets. When you specify ENUMROWS in this case, you must supply the value for *total*. For information about when BMCSORT dynamically allocates sort work data sets, see “Allocating Sort Work Data Sets” on page 2-14.

  Note: When BMCSORT is dynamically allocating your sort work data sets, BMC Software recommends that you also specify the SORTNUM option. Specifying SORTNUM lets LOADPLUS run multiple concurrent sort tasks when needed. If you do not specify SORTNUM and only BMCSORT is allocating sort work data sets, LOADPLUS uses a single sort process. For more information, see “SORTNUM” on page 3-49.

BMC Software, Inc., Confidential and Proprietary Information

- You are loading a very large number of rows.

  In this case, BMC Software strongly recommends that you specify a value for ENUMROWS so that BMCSORT can determine the most efficient method for performing sort processing.

For any of the ENUMROWS values, BMC Software recommends that you overestimate rather than underestimate. Overestimating results in LOADPLUS allocating files that are larger than needed, but underestimating can cause LOADPLUS to terminate the job.

Do not include commas in the numbers that you specify.

Use Table 3-5 to determine whether to specify *total* or *new* for the type of load job that you are running.

Table 3-5 Which Parameter to Specify with ENUMROWS

LOAD RESUME or REPLACE	*total* or *new*[a]
LOAD REPLACE	either *total* or *new*
LOAD RESUME (including LOAD RESUME YES PART REPLACE)	*new*

[a] If you specify *new* and BMCSORT is dynamically allocating the SORTWK data set, you must also supply the value for *total*.

total

This value is an integer that specifies the total number of rows that you expect to have in all tables of the table space following the load. The default is 0. A value of 0 tells LOADPLUS to dynamically allocate data sets of a minimal size for those DDTYPEs for which dynamic allocation is active.

If BMCSORT is dynamically allocating SORTWK data sets, LOADPLUS passes this integer to BMCSORT through the SIZE sort parameter. BMCSORT uses the SIZE parameter to calculate how much SORTWK space to dynamically allocate.

BMC Software, Inc., Confidential and Proprietary Information

new

This value is an integer that specifies the number of records in your SYSREC data set. The default is 0. A value of 0 tells LOADPLUS to dynamically allocate data sets of a minimal size for those DDTYPEs for which dynamic allocation is active.

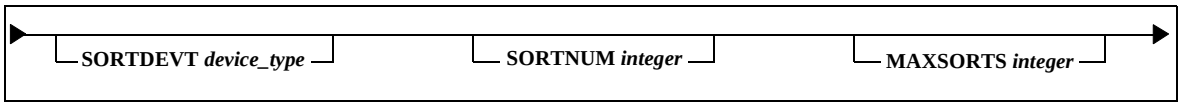
Note: For LOAD RESUME YES PART REPLACE with INDEX UPDATE, *new* is the total number of rows that you expect to have in the parts that you are replacing.

error

This value is an integer that specifies the number of rows that you expect to be discarded. The default is 32000 or the number of rows being loaded if that number is less than 32000. Specify a value for *error* only if you expect that the number of discarded rows will exceed the default value. This option applies only if LOADPLUS is dynamically allocating the SYSERR or SYSDISC data sets.

Note: If you specify the DISCARDS option with a value greater than zero and you are dynamically allocating SYSDISC, LOADPLUS uses the DISCARDS value instead of the value for *error* to determine the size of the SYSDISC data set.

BMC Software, Inc., Confidential and Proprietary Information



SORTDEVT

The SORTDEVT option specifies the device type for the sort work files that BMCSORT will allocate dynamically. This option overrides the value that was specified in the BMCSORT DYNALOC installation option.

Note: If the value of the third parameter in the BMCSORT DYNALOC installation option is OFF, specifying the SORTDEVT option turns BMCSORT dynamic allocation on.

SORTNUM

The SORTNUM option affects the allocation of sort work files when BMCSORT is allocating your sort work files dynamically. You can specify an integer value of 1 through 99.

When you specify this option, BMCSORT dynamically allocates the number of sort work files that it needs for each sort task up to the maximum that is illustrated in the following formula:

```
maximum dynamically allocated sort work files =
n – preallocated sort work files
```

- If you specify *integer* from 1 through 32, *n* equals 32.
- If you specify *integer* greater than 32, *n* equals *integer.*

Note: *Preallocated sort work files* include sort work files that are allocated in your JCL and any sort work files that LOADPLUS dynamically allocates.

Specifying the SORTNUM option also produces the following results:

- If the value of the third parameter in the BMCSORT DYNALOC installation option is OFF, specifying the SORTNUM option turns BMCSORT dynamic allocation on and BMCSORT allocates sort work files as needed.

- This option provides the information that lets LOADPLUS run multiple concurrent sort tasks when applicable. If you *do not* specify this option and only BMCSORT is allocating your sort work files, LOADPLUS uses a single sort task.

BMC Software, Inc., Confidential and Proprietary Information

The sort work files that BMCSORT allocates dynamically might be used in one of the following circumstances:

- if you do not specify any SORTWK DD statements in your JCL
- in addition to any SORTWK data sets that you specify in your JCL if BMCSORT determines that additional data sets are needed

For information about when BMCSORT allocates your sort work files dynamically, see “Allocating Sort Work Data Sets” on page 2-14.

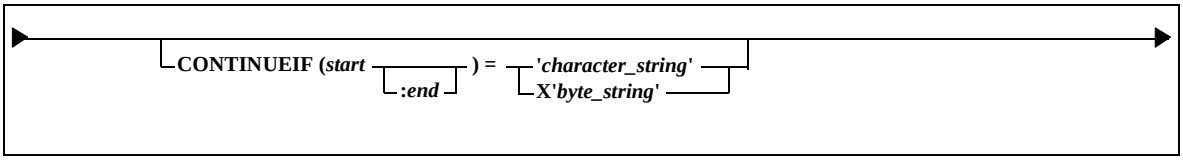
Note: BMC Software recommends that, if you depend on BMCSORT to dynamically allocate your sort work data sets, you also specify the ENUMROWS command option. For more information, see “ENUMROWS” on page 3-46.

MAXSORTS

The MAXSORTS option specifies the maximum number of index sort tasks that LOADPLUS can run concurrently during the LOAD phase. If you do not specify MAXSORTS, LOADPLUS uses the SMAX value, which is specified in the installation options member. See “Controlling Sort Processing” on page 6-12 and Appendix A, “LOADPLUS Installation Options” for additional information.

Note: If you specify LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY, LOADPLUS ignores the MAXSORTS option.

BMC Software, Inc., Confidential and Proprietary Information



CONTINUEIF

The CONTINUEIF option tells LOADPLUS whether the current logical input record continues with the next physical input record.

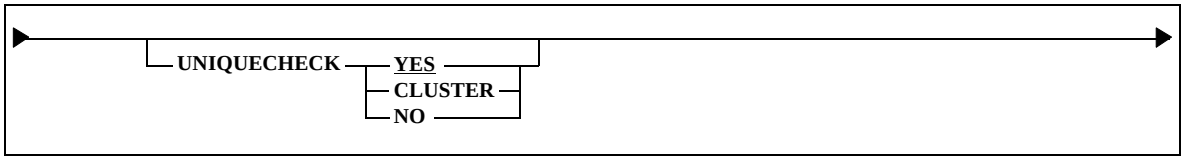
Note: This option is not valid when you specify any of the FORMAT options. If you specify CONTINUEIF with one of the FORMAT options, LOADPLUS fails and issues message BMC51412E.

The two numbers, *start*:*end*, indicate the starting and ending columns of the continuation indicator in the physical input record. If the comparison of the continuation indicator with the string is equal, LOADPLUS concatenates the next physical input record. You can concatenate any number of physical records into a logical record as long as the length of the logical record does not exceed 32 KB.

You can specify either a character string or a hexadecimal string. The length of the continuation indicator cannot be greater than 70 or less than the length of the string. If you do not specify *:end*, LOADPLUS uses the length of the string. When determining the value to specify, note that in the physical input record, the first column of the record is column 1.

The continuation indicator is not included in the logical record. Thus, if you specify CONTINUEIF(73:80)=*string*, the first byte of the first physical record is column 1 of the logical record. The first byte of the second physical record is column 73 of the logical record.

BMC Software, Inc., Confidential and Proprietary Information



UNIQUECHECK

The UNIQUECHECK option tells LOADPLUS whether to check unique index keys in the input records for duplicate values.

LOADPLUS ignores this option in the following cases:

- SQLAPPLY load—DB2 checks for duplicates during the apply process.
- Single-phase load—LOADPLUS always checks all participating unique indexes for duplicates.

YES

The default is UNIQUECHECK YES, which tells LOADPLUS to check for duplicates in key values for all participating unique indexes. Table 3-6 on page 3-53 describes the actions that LOADPLUS takes when you specify or default to UNIQUECHECK YES and LOADPLUS detects a unique key violation.

CLUSTER

UNIQUECHECK CLUSTER indicates that you want LOADPLUS to check for duplicates in key values for only the clustering indexes. Specifying this option eliminates the need to sort and check participating nonclustering indexes in the PRELOAD phase, which can reduce the elapsed time of your load job. Table 3-6 on page 3-53 describes the actions that LOADPLUS takes when you specify UNIQUECHECK CLUSTER and LOADPLUS detects a unique key violation.

Note: If you specify UNIQUECHECK CLUSTER and specify or default to ORDER NO, LOADPLUS changes the value of this option to UNIQUECHECK YES.

NO

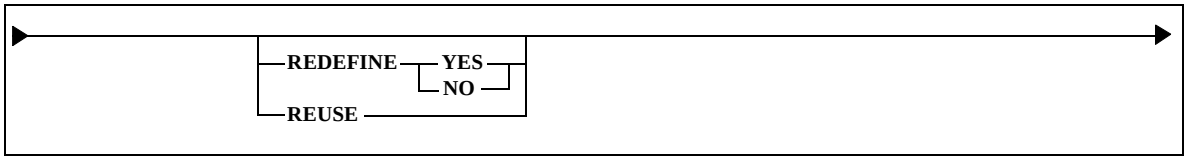
Specifying this option indicates that you do not want LOADPLUS to check key values. Using this option eliminates the need to sort and check the indexes in the PRELOAD phase, which can reduce the elapsed time of your load job. Table 3-6 on page 3-53 describes the actions that LOADPLUS takes when you specify UNIQUECHECK NO and LOADPLUS detects a unique key violation.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-6 UNIQUECHECK Options

UNIQUECHECK Option	Unique Index Type	LOADPLUS Action on Unique Key Violation
YES	any	• detects duplicates in the PRELOAD phase • discards incoming records that violate unique key requirement • places records in the discard data set
CLUSTER	clustering	see the UNIQUECHECK YES option
	nonclustering	see the UNIQUECHECK NO option
NO	any	• detects duplicates in the LOAD phase • deletes duplicates from loaded table space and participating indexes, but does not place them in the discard data set

BMC Software, Inc., Confidential and Proprietary Information



REDEFINE

This option controls whether LOADPLUS deletes and redefines the VSAM data sets for the table space and index spaces that are participating in the load. LOADPLUS can redefine both user-defined (VCAT-defined) data sets and data sets that are defined in DB2 storage groups (STOGROUP-defined).

For a SQLAPPLY load, LOADPLUS ignores this option.

You can specify the default for the REDEFINE command option in your installation options module (using the REDEFINE installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

YES

The REDEFINE YES option tells LOADPLUS to delete and redefine the VSAM data set for each DB2 object prior to loading it. LOADPLUS also defines any additional VSAM data sets that might be required.

Table 3-7 describes which DB2 objects that LOADPLUS redefines when you specify REDEFINE YES.

Table 3-7 DB2 Objects Redefined with REDEFINE YES (Part 1 of 2)

Command Issued	Objects Redefined
LOAD REPLACE	• table space • clustering index • participating nonclustering indexes
LOAD RESUME YES SHRLEVEL NONE	if you specify or default to INDEX BUILD, clustering index and participating nonclustering indexes
LOAD RESUME YES PART *n*	if you specify or default to INDEX BUILD, clustering index part and participating nonclustering indexes

BMC Software, Inc., Confidential and Proprietary Information

Table 3-7 DB2 Objects Redefined with REDEFINE YES (Part 2 of 2)

Command Issued	Objects Redefined
LOAD RESUME YES PART *n* REPLACE	• clustering index partition for the PART that you are replacing • table partition for the PART that you are replacing • if you specify or default to INDEX BUILD, participating nonclustering indexes
LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE	staging data sets **Note**: If you are loading VCAT-defined objects, you must provide the IDCAMS statements in the SYSIDCIN data set to delete and define the staging data sets. See “Staging Data Sets” on page 2-31 for information about naming the staging data sets.

For VCAT-defined objects, consider the following information:

- When you specify PRELOAD PAUSE, you still have the option of deleting and redefining VCAT-defined data sets.

- When you specify or default to PRELOAD CONTINUE, the only way to delete and redefine VCAT-defined data sets as part of the load is to use this option and provide the SYSIDCIN data set that contains the necessary IDCAMS control statements. Otherwise, LOADPLUS issues message BMC50391E and resets the high-used RBA.

For STOGROUP-defined objects, consider the following information:

- When you specify this option for STOGROUP-defined data sets, do not provide a SYSIDCIN data set.

  For detailed information about using the SYSIDCIN data set in LOADPLUS, see “SYSIDCIN Data Set” on page 4-22.

- For a STOGROUP-defined table space or index space with multiple volumes in the STOGROUP, LOADPLUS attempts to reallocate the data set on the volume on which it currently resides if that volume is still defined in the STOGROUP. The order in which LOADPLUS retrieves subsequent volumes from the STOGROUP for the purpose of allocating VSAM data sets is not predictable.

BMC Software, Inc., Confidential and Proprietary Information

NO

The REDEFINE NO option tells LOADPLUS not to delete and redefine the existing VSAM data sets for the table space and index spaces that are participating in the load. Instead, LOADPLUS issues message BMC50391I, reuses the existing data sets, and resets the high-used RBA.

For VCAT-defined objects, LOADPLUS

- extends to another data set, if needed, as long as that data set is already defined
- does not define any additional data sets

If you specify REDEFINE NO with LOAD REPLACE SHRLEVEL REFERENCE or with LOAD REPLACE SHRLEVEL CHANGE, you must preallocate the staging data sets for VCAT-defined objects. For information about naming the staging data sets, see “Staging Data Sets” on page 2-31.

For STOGROUP-defined objects, LOADPLUS

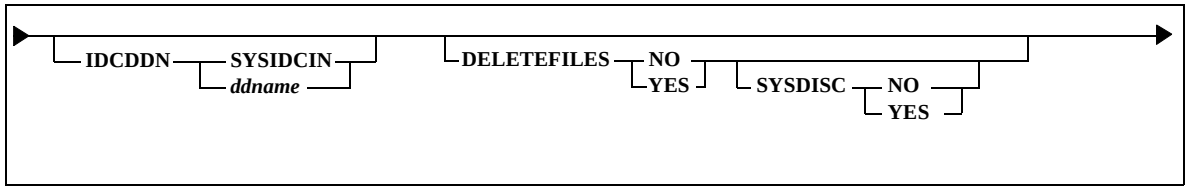
- (for LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE) creates any staging data sets that you do not preallocate, but reuses any staging data sets that you do preallocate
- (for multiple data set objects) extends to another data set if needed, and creates the data set if it does not exist
- (for multiple data set objects) deletes any data set that it did not use

REUSE

This option is used for compatibility with the IBM DB2 LOAD utility command syntax. If you specify REUSE, LOADPLUS functions as if you specified REDEFINE NO.

If you specify REUSE and REDEFINE YES, LOADPLUS honors the last keyword that it finds in the command string.

BMC Software, Inc., Confidential and Proprietary Information



IDCDDN

The IDCDDN option allows you to override the default ddname (SYSIDCIN) for the input data set that contains the IDCAMS command statements that LOADPLUS uses to redefine VSAM data sets. For detailed information about using the SYSIDCIN data set in LOADPLUS, see “SYSIDCIN Data Set” on page 4-22.

You can specify the default for the IDCDDN command option in your installation options module (using the IDCDDN installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

DELETEFILES

The DELETEFILES option tells LOADPLUS whether to delete the SORTOUT, SORTWK, SYSUT1, and SYSERR files after the load completes successfully.

You can specify the default for the DELETEFILES command option in your installation options module (using the DELFILES installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

If you are running in a worklist environment, LOADPLUS ignores the value that was specified in the DELFILES installation option. If you want to delete your work files, you must specify DELETEFILES YES on the LOAD command.

NO

DELETEFILES NO tells LOADPLUS not to delete any SORTOUT, SORTWK, SYSUT1, or SYSERR files.

Note: If your work files are defined with DISP=(*any*,DELETE,DELETE), LOADPLUS deletes these work files after the load completes even if you specify DELETEFILES NO.

BMC Software, Inc., Confidential and Proprietary Information

YES

DELETEFILES YES tells LOADPLUS to delete all corresponding SORTOUT, SORTWK, SYSUT1, and SYSERR files after the load completes successfully, regardless of the JCL disposition of these files and whether the files were used.

LOADPLUS deletes all physical sequential data sets whose ddnames match the SORTOUT, SORTWK, SYSUT1, and SYSERR names or ddname prefixes. This occurs after either the LOAD phase or the COMBINED phase completes successfully.

If the work files are on tape, you can use the TAPEDISP installation option to specify the final disposition of the work files. See Appendix A, “LOADPLUS Installation Options,” for details.

SYSDISC

This option tells LOADPLUS whether to delete the SYSDISC file after the load completes successfully.

You can specify the default for the SYSDISC command option in your installation options module (using the second parameter of the DELFILES installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

If you are running in a worklist environment, LOADPLUS ignores the value that was specified in the DELFILES installation option. If you want to delete your SYSDISC file, you must specify SYSDISC YES on the LOAD command.

NO

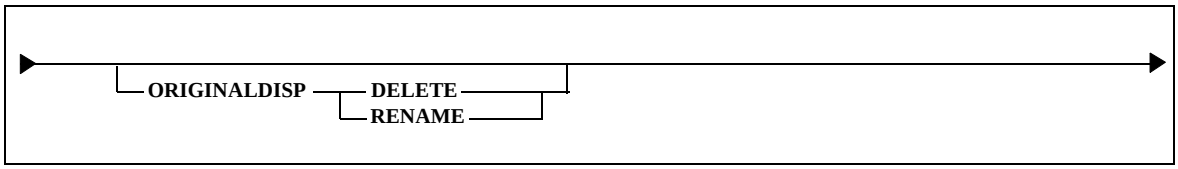
SYSDISC NO tells LOADPLUS not to delete the SYSDISC file automatically, even if it does not contain any discarded records.

Note: If SYSDISC is defined with DISP=(*any*,DELETE,DELETE), LOADPLUS deletes these work files after the load completes even if you specify SYSDISC NO.

YES

SYSDISC YES tells LOADPLUS to delete the SYSDISC file if it does not contain any discarded records.

BMC Software, Inc., Confidential and Proprietary Information



ORIGINALDISP

(applies to LOAD REPLACE SHRLEVEL REFERENCE and LOAD REPLACE SHRLEVEL CHANGE only) ORIGINALDISP allows you to specify whether you want LOADPLUS to delete or rename the original data sets after LOADPLUS renames the staging data sets and completes a successful load.

You can specify the default for the ORIGINALDISP command option in your installation options module (by using the ORIGDISP installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

For information about the staging data sets for LOAD REPLACE SHRLEVEL REFERENCE and LOAD REPLACE SHRLEVEL CHANGE, see “Staging Data Sets” on page 2-31.

DELETE

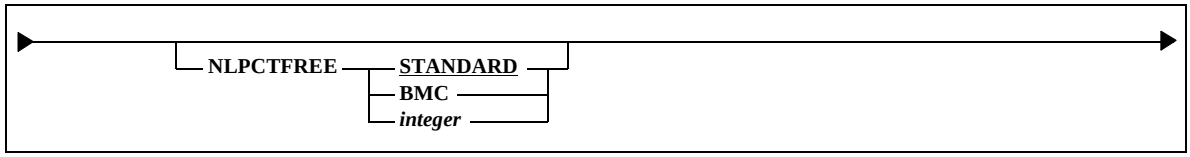
This option tells LOADPLUS to delete the original data sets.

RENAME

This option tells LOADPLUS to rename the original data sets to the staging data set names. LOADPLUS changes the **OLD** in the OLDDBC node of the cluster and in the OLDDBD node of the data component to **BMC**.

This option allows you to preserve the space that was initially allocated for the original data sets by renaming them to the staging data set names. The renamed data sets are then ready to use as the staging data sets in a subsequent load process.

BMC Software, Inc., Confidential and Proprietary Information



NLPCTFREE

NLPCTFREE specifies the percentage of each nonleaf index page to reserve as free space when LOADPLUS builds the indexes. This option is effective only if you specify INDEX BUILD.

STANDARD

STANDARD, which is the default, tells LOADPLUS to use the value that was provided in the DB2 PCTFREE option when the index was created. Like DB2, LOADPLUS leaves up to 10 percent of a nonleaf page free. If you specify a value greater than 10, only 10 percent is left free.

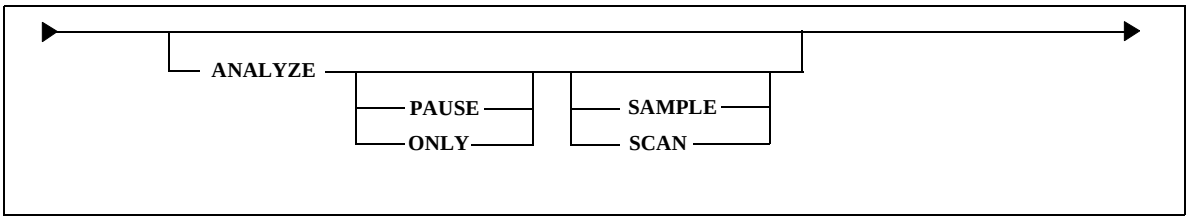
BMC

This option directs LOADPLUS to honor the DB2 PCTFREE value in the DB2 catalog, even if the value is greater than 10.

integer

This value identifies the percentage of each nonleaf index page to reserve as free space. You can specify an integer value of 0 through 99.

BMC Software, Inc., Confidential and Proprietary Information



ANALYZE

ANALYZE gathers information about the table space and provides estimated data set sizes for the following data sets:

- load (SORTOUT)
- work (SYSUT1)
- sort work (SORTWK)
- discard (SYSDISC)
- error (SYSERR)
- image copy (BMCCPY, BMCCPZ, BMCRCY, and BMCRCZ)

LOADPLUS provides all estimates in both kilobytes and cylinders by device type. LOADPLUS calculates the data set size based on two values: cardinality (the number of rows) and average row length. To obtain the cardinality, LOADPLUS either samples or scans (depending on what you specify) one index per participating table. LOADPLUS determines the average row length from information in the DB2 catalog.

LOADPLUS scans or samples all data sets of a multiple data set index. LOADPLUS multitasks its analysis, scanning or sampling one data set per task.

During the ANALYZE phase, LOADPLUS does not stop the index space and associated table space that are participating in the load. See “Tuning the ANALYZE Phase” on page 6-16 for performance considerations when using this option.

PAUSE

If you specify ANALYZE PAUSE, LOADPLUS generates a report and ends the processing after the ANALYZE phase completes. You can use the output of the ANALYZE phase to specify the number and allocations of the work data sets. LOADPLUS leaves the utility ID active so that you can restart the load at the next phase. For an example that uses this option, see “Example 8: LOAD REPLACE of a Partitioned Table Space Using ANALYZE PAUSE” on page 5-55.

BMC Software, Inc., Confidential and Proprietary Information

ONLY

If you specify ANALYZE ONLY, LOADPLUS generates a report and terminates after the ANALYZE phase. You cannot restart the load job. However, you can use the output of the ANALYZE phase to specify the number and allocations of the work data sets for subsequent load jobs.

SAMPLE

SAMPLE tells LOADPLUS to determine the estimated cardinality by reading a subset of pages from the most appropriate index. LOADPLUS determines the most appropriate index based on key length and the number of data sets in the index.

Note: If LOADPLUS attempts to sample a data set that contains less than 100 pages, LOADPLUS actually scans the data set instead of sampling it.

SCAN

SCAN tells LOADPLUS to determine the exact cardinality by reading every leaf page in the most appropriate index. LOADPLUS determines the most appropriate index based on key length and the number of data sets in the index.

ANALYZE Requirements

You must specify ENUMROWS to allow ANALYZE processing to estimate the required space. If you specify ANALYZE PAUSE or ANALYZE ONLY without specifying ENUMROWS, LOADPLUS ends with a return code of 8. For information on how to specify the ENUMROWS option, see “ENUMROWS” on page 3-46.

LOADPLUS determines cardinality only when running LOAD RESUME YES INDEX BUILD (because the indexes will be built by using existing keys). LOADPLUS does not determine cardinality in the following cases:

- if you specify LOAD RESUME NO or LOAD REPLACE
- if you specify LOAD RESUME YES INDEX UPDATE
- if you specify LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY

BMC Software, Inc., Confidential and Proprietary Information

- if you specify LOAD RESUME YES PART REPLACE and no secondary indexes are participating in the load

- for partitions or tables that are not part of the load, unless they have keys in a secondary index that the load will build

ANALYZE Defaults

Under either of the following circumstances, LOADPLUS determines whether to use sampling or scanning. LOADPLUS performs a full analysis and continues processing, using the information from the ANALYZE phase to dynamically allocate work files.

- You specify ANALYZE with no other options.
- You do not specify ANALYZE, but you do specify ENUMROWS and enable dynamic work file allocation.

You can enable dynamic work file allocation by using the DDTYPE installation or command option.

How LOADPLUS Determines Cardinality and Row Length

Table 3-8 shows the various combinations of ANALYZE options and the effect that they have on how LOADPLUS determines cardinality and average row length.

Table 3-8 How ANALYZE Options Affect Cardinality and Average Row Length (Part 1 of 2)

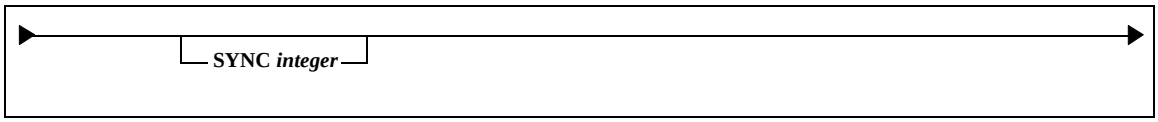
ANALYZE Keywords	How LOADPLUS Determines Cardinality	How LOADPLUS Determines Average Row Length
(ANALYZE not specified)	If you specify ENUMROWS and enable dynamic allocation, LOADPLUS defaults to ANALYZE and issues message BMC51424I. Otherwise, LOADPLUS does not perform an analysis.	
ANALYZE ANALYZE PAUSE ANALYZE ONLY	If running LOAD RESUME YES INDEX BUILD, LOADPLUS decides whether to sample or scan the index leaf pages. Otherwise, LOADPLUS does not determine cardinality, but uses the number of new rows to determine work data set size.	LOADPLUS obtains this information from the DB2 catalog, basing it on half the length of any VARCHAR columns in the table. LOADPLUS does not consider compression in the calculations.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-8 How ANALYZE Options Affect Cardinality and Average Row Length (Part 2 of 2)

ANALYZE Keywords	How LOADPLUS Determines Cardinality	How LOADPLUS Determines Average Row Length
ANALYZE SAMPLE ANALYZE PAUSE SAMPLE ANALYZE ONLY SAMPLE	If running LOAD RESUME YES INDEX BUILD, LOADPLUS samples the index space. Otherwise, LOADPLUS does not determine cardinality, but uses the number of new rows to determine work data set size.	LOADPLUS obtains this information from the DB2 catalog, basing it on half the length of any VARCHAR columns in the table. LOADPLUS does not consider compression in the calculations.
ANALYZE SCAN ANALYZE PAUSE SCAN ANALYZE ONLY SCAN	If running LOAD RESUME YES INDEX BUILD, LOADPLUS scans the most appropriate index. Otherwise, LOADPLUS does not determine cardinality, but uses the number of new rows to determine work data set size.	LOADPLUS obtains this information from the DB2 catalog, basing it on half the length of any VARCHAR columns in the table. LOADPLUS does not consider compression in the calculations.

BMC Software, Inc., Confidential and Proprietary Information



SYNC

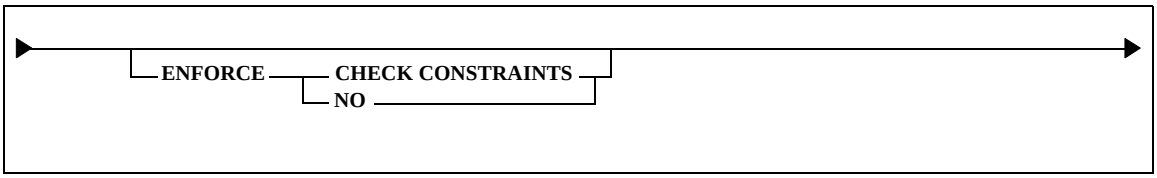
LOADPLUS writes records to the BMCSYNC table, that indicate the number of 1K rows that LOADPLUS processes during the LOAD phase or the COMBINED phase. You can use this information to determine how far the load job has progressed. By default, LOADPLUS writes BMCSYNC records only after loading the last row or key in a table space, index, or partition. The SYNC option identifies the number of 1K rows that LOADPLUS processes between writing to the BMCSYNC table. If you want LOADPLUS to write records more often, specify an integer value with this option.

LOADPLUS records monitoring and restart sync points in the BMCSYNC table as the job progresses. You can issue an SQL statement or use BMCDSN (the BMC Software utility command processor) to query this table to determine how far the load has progressed and the status of the objects that you are loading.

The SYNC option does not control the sync points that are used to restart LOADPLUS. The restart sync points are established only after LOADPLUS loads the last row or key in a table space, index, or partition.

For a SQLAPPLY load, LOADPLUS ignores this option.

BMC Software, Inc., Confidential and Proprietary Information



ENFORCE

ENFORCE tells LOADPLUS whether to check for violations of DB2 table check constraints.

LOADPLUS does not support ENFORCE CONSTRAINTS and, except for a SQLAPPLY load, LOADPLUS does not check for referential integrity violations. Instead, if a table has referential integrity constraints defined, LOADPLUS places the affected table spaces in CHECK pending status if the value of the CHECKPEND option is YES. For information about how to ensure that referential integrity has not been violated, see “Referential Integrity and Check Constraints” on page 2-51. For more information about the CHECKPEND option, see “CHECKPEND” on page 3-67.

For a SQLAPPLY load, LOADPLUS ignores this option because DB2 checks all constraints during apply processing.

If you specify FORMAT BMCUNLOAD, LOADPLUS ignores this option and issues message BMC50109I. LOADPLUS sets CHECK pending if there are check constraints on the table that you are loading and the value of the CHECKPEND option is YES.

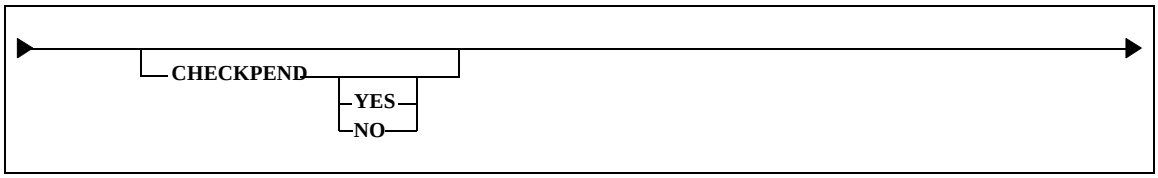
CHECK CONSTRAINTS

When you specify CHECK CONSTRAINTS, which is the default, LOADPLUS checks the rows to be loaded for violations of DB2 table check constraints, if any constraints exist. LOADPLUS handles any violating rows according to your DISCARDS option specification.

NO

Specify NO when you do not want to check for table check constraints during LOADPLUS processing. When you specify NO, LOADPLUS loads all rows regardless of whether they violate table check constraints or referential integrity. If the value of the CHECKPEND option is YES, LOADPLUS automatically sets the object status to CHECK pending.

BMC Software, Inc., Confidential and Proprietary Information



CHECKPEND

CHECKPEND tells LOADPLUS whether to set dependent table spaces to CHECK pending (CHKP) status.

You can specify the default for the CHECKPEND command option in your installation options module by using the CHEKPEND installation option. With the following exception, the command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

If the value of the CHEKPEND installation option includes the ENFORCE keyword, you cannot override the installation option with the CHECKPEND command option.

YES

After loading, LOADPLUS sets CHKP status if appropriate and completes with a return code 4. Run either CHECK PLUS (specifying the CHECK DATA SCOPE ALL option) or the IBM CHECK DATA utility (specifying SCOPE ALL) to ensure that referential integrity and table check integrity have not been violated.

If the value for the CHEKPEND installation option is (NO,ENFORCE), specifying CHECKPEND YES causes LOADPLUS to issue message BMC50115E and terminate processing.

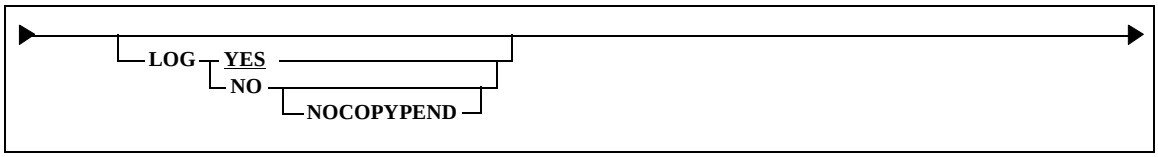
For a SQLAPPLY load, LOADPLUS ignores this option.

NO

After loading, LOADPLUS does not set CHKP status and the job completes with a return code 0.

If the value for the CHEKPEND installation option is (YES,ENFORCE), specifying CHECKPEND NO causes LOADPLUS to issue message BMC50115E and terminate processing.

BMC Software, Inc., Confidential and Proprietary Information



LOG

LOADPLUS provides this option for compatibility with the IBM DB2 LOAD utility command syntax.

YES

(applies to a SQLAPPLY load only) LOADPLUS provides this option for compatibility with the IBM DB2 LOAD utility command syntax and treats it as a comment.

If you specify LOG YES with any load type other than SQLAPPLY, LOADPLUS issues message BMC50110E and terminates with return code 8.

NO

LOADPLUS provides this option for compatibility with the IBM DB2 LOAD utility command syntax.

If you specify this option with a SQLAPPLY load, LOADPLUS issues message BMC50115E and terminates with return code 8.

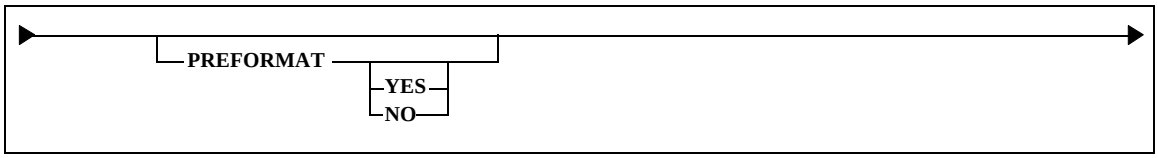
For all other load types, if you specify this option without the NOCOPYPEND keyword, LOADPLUS treats it as a comment.

NOCOPYPEND

LOADPLUS provides this option for compatibility with the IBM DB2 LOAD utility command syntax. When you specify this option, LOADPLUS functions as if you specified COPY NO COPYPEND NO.

Note: If the value for the COPYPEND installation option is (YES,ENFORCE), specifying NOCOPYPEND causes LOADPLUS to issue message BMC50115E and terminate processing.

BMC Software, Inc., Confidential and Proprietary Information



PREFORMAT

The PREFORMAT option tells LOADPLUS whether to preformat the unused pages of the data set. For compatibility with earlier releases, if you specify PREFORMAT without a value, LOADPLUS assumes PREFORMAT YES.

LOADPLUS also provides PREFORMAT support at the partition level. See “PART” on page 3-90 for information.

You can specify the default for the PREFORMAT option in your installation options module (using the PREFORMAT installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

YES

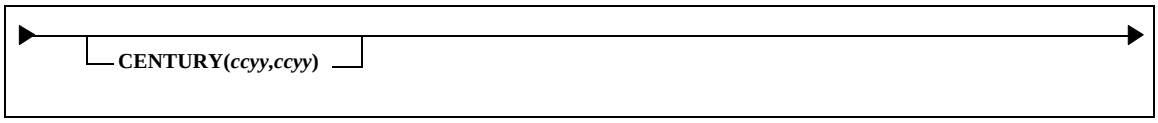
If you specify PREFORMAT YES, LOADPLUS preformats the unused portion of the data set. Preformatting writes pages that have been initialized with zeros up to the high-allocated relative byte address (RBA) of the table space and index spaces. Preformatting occurs after LOADPLUS loads the data and builds the indexes.

For a SQLAPPLY load, LOADPLUS ignores this option and issues message BMC50109I.

NO

If you specify PREFORMAT NO, LOADPLUS does not preformat the unused pages.

BMC Software, Inc., Confidential and Proprietary Information



CENTURY

The CENTURY option specifies the 100-year range that determines the century for DATE and TIMESTAMP external formats that contain two-digit year values. The first four-digit year value must be less than the second four-digit year. You must specify both values and these values must span 100 years.

You specify the default for the CENTURY command option in your installation options module (using the CENTURY installation option). The command option overrides the default that was established at installation. For details, see Appendix A, "LOADPLUS Installation Options."

You can also specify CENTURY with the field specification option if you want to apply it to a particular DATE or TIMESTAMP column. See "DATE-format EXTERNAL(length)" on page 3-115 or "TIMESTAMP-format EXTERNAL(length)" on page 3-116 for more information.

(*ccyy,ccyy*)

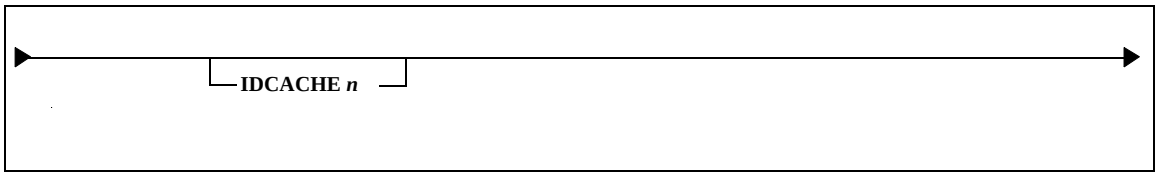
Any two-digit year in the input data that lies between the first *yy* specification and 99 is prefixed with the first *cc* value to create a four-digit year. Any two-digit year in the input data that lies between 00 and the second *yy* specification is prefixed with the second *cc* value to create a four-digit year.

The following examples illustrate this option:

- If you specify CENTURY(1950,2049), LOADPLUS places 19 in front of each two-digit year with a value 50 through 99, and places 20 in front of each two-digit year with a value 00 through 49. The date 99/12/31 becomes 1999/12/31 and 00/12/31 becomes 2000/12/31.

- If you specify CENTURY(1925, 2024), LOADPLUS places 19 in front of each two-digit year with a value 25 through 99, and places 20 in front of each two-digit year with a value 00 through 24. The date 45/12/31 becomes 1945/12/31 and 15/01/31 becomes 2015/01/31.

BMC Software, Inc., Confidential and Proprietary Information



IDCACHE

The IDCACHE option allows you to specify the size of the cache that LOADPLUS uses when generating values for an identity column. LOADPLUS uses one cache for each READ task. The valid values for this option are 1 through 100000.

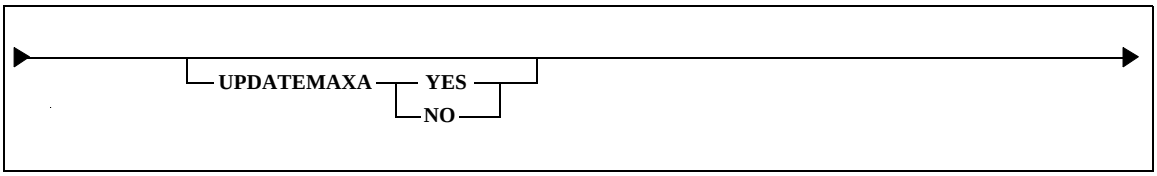
Note: This cache is a temporary cache that LOADPLUS uses. This option does not affect the cache that you specified when defining your identity column.

LOADPLUS reserves this cache of numbers in SYSIBM.SYSSEQUENCES. If you specify a smaller size for this cache, LOADPLUS accesses and updates SYSIBM.SYSSEQUENCES more frequently, which can reduce the performance of your load job. However, if you specify a larger size for this cache, there is a greater possibility of gaps in your identity column values and a higher risk of running out of numbers. In addition, if your identity column is defined with CYCLE YES, a larger cache size introduces a greater likelihood of generating duplicate values.

For a SQLAPPLY load, LOADPLUS ignores this option.

You can specify the default for the IDCACHE command option in your installation options module (using the IDCACHE installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

BMC Software, Inc., Confidential and Proprietary Information



UPDATEMAXA

The UPDATEMAXA option tells LOADPLUS whether to update the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES when loading identity column values from an input file. (When generating identity column values, LOADPLUS always updates MAXASSIGNEDVAL.)

You can specify the default for the UPDATEMAXA command option in your installation options module (using the UPDMAXA installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

YES

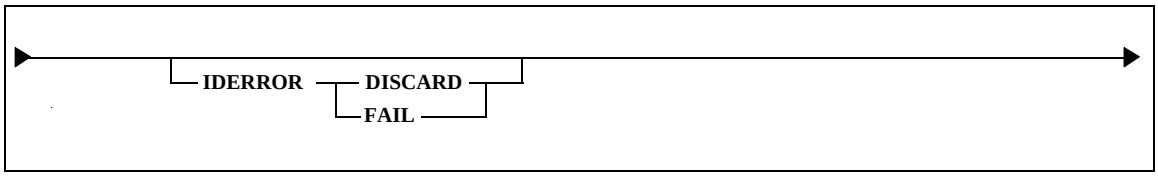
UPDATEMAXA YES tells LOADPLUS to update the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES when loading identity column values from an input file.

LOADPLUS validates the input file values against the range that is defined on the identity column and updates the value of MAXASSIGNEDVAL based on this range and the current value of MAXASSIGNEDVAL. For details about how LOADPLUS updates this field, see “How LOADPLUS Updates MAXASSIGNEDVAL” on page 2-35.

NO

UPDATEMAXA NO tells LOADPLUS not to update the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES when loading identity column values from an input file.

BMC Software, Inc., Confidential and Proprietary Information



IDERROR

The IDERROR option tells LOADPLUS what action to take when encountering a generated identity column value that is outside the range that is defined on the column.

You can specify the default for the IDERROR command option in your installation options module (using the IDERROR installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

DISCARD

If you specify IDERROR DISCARD, LOADPLUS discards any generated identity column values that are outside the range that is defined on the column.

FAIL

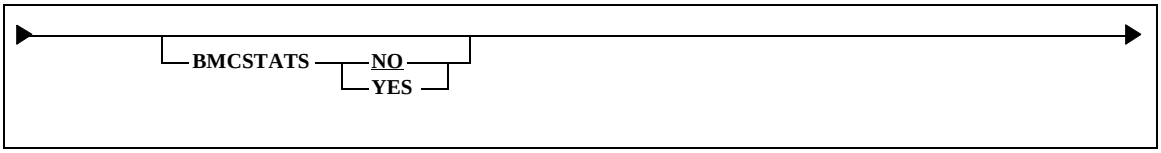
If you specify IDERROR FAIL, LOADPLUS terminates as soon as it encounters a generated identity column value that is outside the range that is defined on the column.

Note: For a SQLAPPLY load, LOADPLUS ignores this option and always discards generated identity column values that are outside the range that is defined on the column.

BMC Software, Inc., Confidential and Proprietary Information

Statistics Options

The statistics options indicate whether you want LOADPLUS to update statistics in the DASD MANAGER PLUS tables, if installed, and in the DB2 catalog.



BMCSTATS

The BMCSTATS option tells LOADPLUS whether to save the BMC Software statistics that it generates to the DASD MANAGER PLUS database tables.

NO

BMCSTATS NO, which is the default, tells LOADPLUS to not save the BMC Software statistics.

YES

If you specify BMCSTATS YES, LOADPLUS saves the BMC Software statistics in the DASD MANAGER PLUS tables. This option requires that you have the BMC Software DASD MANAGER PLUS product installed.

Additional Considerations—The following additional considerations apply to BMCSTATS YES:

- LOADPLUS does not update the BMC Software statistics tables under the following circumstances:
  - — When you specify LOAD RESUME YES, LOADPLUS ignores this option.
  - — For a SQLAPPLY load, LOADPLUS ignores this option and issues message BMC50109I.
  - — For a single-phase load, LOADPLUS does not generate statistics if it finds duplicates in unique secondary indexes.

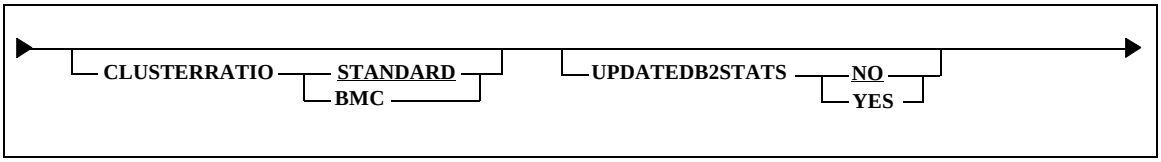
BMC Software, Inc., Confidential and Proprietary Information

— For a two-phase load, if you specify UNIQUECHECK NO and LOADPLUS finds duplicates, LOADPLUS does not generate statistics.

— When you specify LOAD REPLACE with INTO PART and you restart the job, LOADPLUS cannot gather statistics and issues message BMC50503I.

- For any columns that have a FIELDPROC defined, LOADPLUS gathers statistics for the encoded values from the FIELDPROC. LOADPLUS does not gather statistics for column values that are stored in SYSIBM.SYSFIELDS. Additionally, for multicolumn keys, LOADPLUS uses only the first column to update statistics in the DASD MANAGER PLUS tables.

BMC Software, Inc., Confidential and Proprietary Information



CLUSTERRATIO

This option allows you to specify the method for calculating the CLUSTERRATIO value that is updated in the SYSIBM.SYSINDEXES table of the DB2 catalog when you specify UPDATEDB2STATS YES.

STANDARD

STANDARD, which is the default, tells LOADPLUS to use the DB2 CLUSTERRATIO value.

BMC

If you specify BMC, LOADPLUS uses its internal method of calculating the CLUSTERRATIO value. This method takes into account the SYSIBM.SYSINDEXPART NEAROFFPOS and FAROFFPOS values, particularly for nonunique indexes, and the ordering of the row IDs for a particular key value. In general, the higher the NEAROFFPOS and FAROFFPOS values, the lower the cluster ratio.

UPDATEDB2STATS

UPDATEDB2STATS tells LOADPLUS whether to update selected statistical information in the DB2 catalog. The option updates only the DB2 catalog column values that can be updated by means of SQL statements. These column values are the primary values that DB2 uses to determine the access paths that the DB2 optimizer selects during BIND processing.

NO

If you specify UPDATEDB2STATS NO, which is the default, LOADPLUS does not update the statistics in the DB2 catalog.

YES

If you specify UPDATEDB2STATS YES, LOADPLUS updates statistics in the DB2 catalog. The columns that LOADPLUS updates depend on the type of load. Table 3-9 on page 3-78 provides specific information about the tables and statistics that the UPDATEDB2STATS option updates, according to the type of load.

BMC Software, Inc., Confidential and Proprietary Information

Additional Considerations—The following considerations apply to UPDATEDB2STATS YES:

- LOADPLUS does not update statistics in the DB2 catalog under the following circumstances. If you encounter these circumstances, BMC Software recommends that you run the RUNSTATS utility to update statistics in the DB2 catalog.

  — When you specify LOAD RESUME YES, LOADPLUS does not update the DB2 catalog and, therefore, ignores this option.

  — For a SQLAPPLY load, LOADPLUS ignores this option and issues message BMC50109I.

  — LOADPLUS does not gather statistics for column values that are stored in the catalog tables SYSIBM.SYSFIELDS, SYSIBM.SYSCOLDIST, and SYSIBM.SYSCOLDISTSTATS. LOADPLUS also does not currently gather statistics for the catalog tables that were new in DB2 version 7.

  — When you specify LOAD REPLACE with INTO PART and you restart the job, LOADPLUS cannot gather statistics and issues message BMC50503I.

  — For a single-phase load, LOADPLUS does not generate statistics if it finds duplicates in unique secondary indexes.

  — For a two-phase load, if you specify UNIQUECHECK NO and LOADPLUS finds duplicates, LOADPLUS does not generate statistics.

- For multicolumn keys, LOADPLUS uses only the first column to update statistics in the SYSIBM.SYSCOLUMNS table. If the cardinality for the first key column changes dramatically, run the RUNSTATS utility to ensure that the DB2 optimizer selects the appropriate path.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-9 DB2 Catalog Statistics That the UPDATEDB2STATS Option Updates (Part 1 of 2)

DB2 Catalog Table Column Updated by UPDATEDB2STATS YES	Statistics Updated?		
	Nonpartitioned Table Space	Partitioned Table Space, INTO PART Specified	Partitioned Table Space, INTO PART Not Specified
SYSIBM.SYSTABLESPACE: NACTIVE STATSTIME NACTIVEF	updated[a]	never updated	updated[a]
SYSIBM.SYSTABLES: CARD[b] CARDF[b] NPAGES NPAGESF[c] PCTPAGES PCTROWCOMP STATSTIME	updated[a]	never updated	updated[a]
SYSIBM.SYSINDEXES: CLUSTERED FIRSTKEYCARD[b] FIRSTKEYCARDF[b] FULLKEYCARD[b] FULLKEYCARDF[b] NLEAF NLEVELS CLUSTERRATIO STATSTIME CLUSTERRATIOF	updated for participating indexes[a]	updated for participating nonclustering indexes only[a]	updated for participating indexes[a]
SYSIBM.SYSCOLUMNS: (first key column only) COLCARD[b] COLCARDF[b] HIGH2KEY LOW2KEY STATSTIME	updated for participating objects[a]	updated for participating nonclustering indexes only[a]	updated for participating objects[a]
SYSIBM.SYSTABSTATS: CARD NPAGES PCTPAGES NACTIVE PCTROWCOMP STATSTIME CARDF	not applicable	updated[a]	updated[a]

BMC Software, Inc., Confidential and Proprietary Information

Table 3-9 DB2 Catalog Statistics That the UPDATEDB2STATS Option Updates (Part 2 of 2)

DB2 Catalog Table Column Updated by UPDATEDB2STATS YES	Statistics Updated?		
	Nonpartitioned Table Space	Partitioned Table Space, INTO PART Specified	Partitioned Table Space, INTO PART Not Specified
SYSIBM.SYSINDEXSTATS: FIRSTKEYCARD FULLKEYCARD NLEAF NLEVELS CLUSTERRATIO KEYCOUNT STATSTIME FIRSTKEYCARDF FULLKEYCARDF KEYCOUNTF CLUSTERRATIOF	not applicable	updated for participating indexes[a]	updated for participating indexes[a]
SYSIBM.SYSCOLSTATS: (first key column only) HIGHKEY HIGH2KEY LOWKEY LOW2KEY STATSTIME	not applicable	updated for participating objects[a]	updated for participating objects[a]

[a] See the discussion in this section for exceptions.

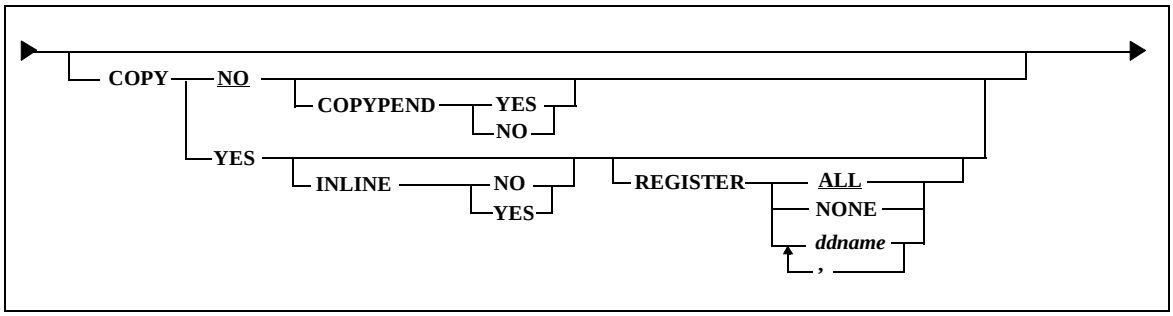
[b] LOADPLUS sets the value of this statistic to -1 when the table space that is involved in the load is LARGE (either by definition or default) or defined with DSSIZE.

[c] This column applies only to DB2 version 7 and later.

BMC Software, Inc., Confidential and Proprietary Information

Copy Options

The copy options indicate whether LOADPLUS is to create a copy of the table space, and, if so, what type of copy. You can have LOADPLUS create an inline image copy *as* it loads the table space, or a standard image copy or DSN1COPY *after* it loads the table space. Additional options control other aspects of the copy process, including the ability to override the default ddnames of the data sets.



COPY

The COPY option tells LOADPLUS whether or not to produce an image copy or a DSN1COPY of the table space.

NO

COPY NO is the default. If you specify COPY NO, LOADPLUS does not make a copy of the table space.

COPYPEND

The COPYPEND option tells LOADPLUS whether to set the COPY pending status when it finishes the load.

You can specify the default for the COPYPEND command option in your installation options module (using the COPYPEND installation option). With the following exception, the command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

If the value of the COPYPEND installation option includes the ENFORCE keyword, you cannot override the installation option with the COPYPEND command option.

BMC Software, Inc., Confidential and Proprietary Information

YES

After loading, LOADPLUS sets the COPY pending status and completes with a return code 4, indicating that you need to make a copy of the table for recovery purposes.

If the value for the COPYPEND installation option is (NO,ENFORCE), specifying COPYPEND YES on the LOAD command causes LOADPLUS to issue message BMC50115E and terminate processing.

Note: For a SQLAPPLY load, LOADPLUS ignores this option.

If the table space that you are loading is in LPL or WEPR status and you specify this option, LOADPLUS terminates.

NO

After loading, LOADPLUS does not set the COPY pending status and completes with a return code 0. You should use this option only for static tables or tables that you can reload if needed, because you will not be able to recover the tables.

If the value for the COPYPEND installation option is (YES,ENFORCE), specifying COPYPEND NO on the LOAD command causes LOADPLUS to issue message BMC50115E and terminate processing.

Note: When you specify this option, LOADPLUS does not reset (remove) COPY pending status if the table that you are loading was in COPY pending status before the load.

YES

If you specify COPY YES, LOADPLUS creates a copy of a nonpartitioned table space, all partitions of a partitioned table space, or selected partitions of a partitioned table space. Depending on your other copy options, LOADPLUS creates a standard DB2 image copy, DSN1COPY, or inline image copy. If you make *any* copies, BMCCPY must be among them. If you make a BMCRCZ copy, you must also make a BMCRCY copy.

If the table space was in COPY pending status prior to the load, LOADPLUS resets (removes) COPY pending status unless you specify REGISTER NONE.

BMC Software, Inc., Confidential and Proprietary Information

Additional Considerations—The following additional considerations apply to COPY YES:

- If you specify this option for a SQLAPPLY load, LOADPLUS issues message BMC50115E and terminates with return code 8.
- If you specify this option and the table space that you are loading is in LPL or WEPR status, you must register at least one copy.
- The difference between an image copy and a DSN1COPY is one of registration: an image copy is registered, a DSN1COPY is not.

INLINE

The INLINE option tells LOADPLUS whether to make an inline image copy as it loads the table space.

You can specify the default for the INLINE command option in your installation options module (using the INLINECP installation option). The command option overrides the default that was established at installation. For details, see Appendix A, "LOADPLUS Installation Options."

NO

INLINE NO tells LOADPLUS not to create an inline image copy as it loads the table space, but to create a DB2 image copy or DSN1COPY after it loads the table space.

YES

INLINE YES tells LOADPLUS to create an inline image copy as it loads the table space. Because LOADPLUS creates the image copy *as* it loads the table space, rather than *after*, specifying INLINE YES can reduce the elapsed time of your load job.

If any of the following conditions apply to your load job, LOADPLUS functions as if you specified INLINE NO, regardless of the option that you specify here or in your installation options:

— You specify LOAD RESUME YES.

However, if you specify LOAD RESUME YES with PART *n* REPLACE for all participating partitions, LOADPLUS creates an inline image copy for the partitions that you are replacing.

BMC Software, Inc., Confidential and Proprietary Information

— The copy data sets are on a stacked tape.

— You restart your load job, with the following exception: If you are running a two-phase load job and you specify RESTART(PHASE), LOADPLUS generates an inline copy if the value of INLINE is YES.

— The size of the table space page is greater than 4 KB, you are loading multiple partitions, and you have one image copy data set.

The inline image copies that LOADPLUS creates have the same characteristics as inline copies that the IBM DB2 LOAD utility creates.

REGISTER

The REGISTER option allows you tell LOADPLUS to register some, none, or all copies with DB2 in the SYSIBM.SYSCOPY table.

Note: LOADPLUS does not register any of the copies if any one is rejected because of a duplicate entry in the SYSIBM.SYSCOPY table.

ALL

ALL, which is the default for REGISTER, registers all copies requested in the JCL by ddname.

NONE

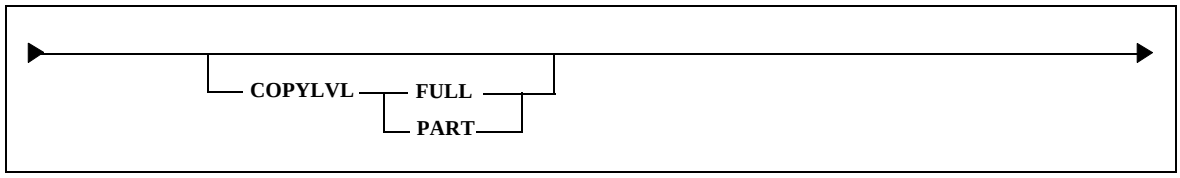
If you specify REGISTER NONE, LOADPLUS does not register any of the copies with DB2. Unless the object was in COPY pending status prior to the load, LOADPLUS does not leave the object in COPY pending status.

Note: If you specify this option and the table space that you are loading is in LPL or WEPR status, LOADPLUS terminates.

ddname

This option allows you to specify by ddname the copy data set that you want to register. LOADPLUS registers only the copies in the data sets that you specify with this option. The ddnames that you specify in this option must also be present in your JCL.

BMC Software, Inc., Confidential and Proprietary Information



COPYLVL

This option is valid only when dynamic allocation is active for copy data sets and you are loading all partitions of a partitioned table space. COPYLVL tells LOADPLUS how you want LOADPLUS to assign the dynamically allocated image copy data sets.

You can improve performance when loading partitioned objects by specifying COPYLVL PART.

You can specify the default for the COPYLVL command option in your installation options module (using the COPYLVL installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

FULL

This option tells LOADPLUS to allocate a single copy data set to contain all the partitions being loaded.

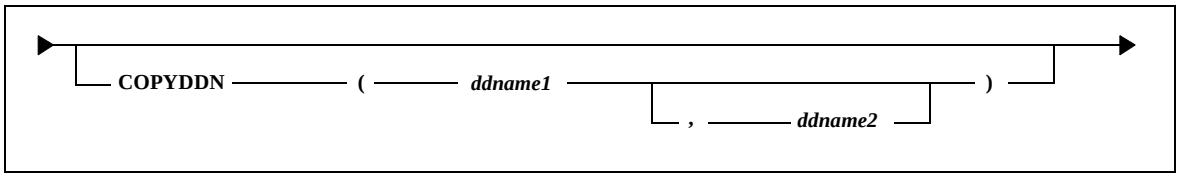
PART

This option tells LOADPLUS to allocate individual copy data sets for each partition being loaded.

Note the following considerations for this option:

- With this option, if you specify a tape device for the UNIT option, LOADPLUS allocates a tape unit for each partition.
- If you are using a GDG name, each partition must have a different GDG base.

BMC Software, Inc., Confidential and Proprietary Information



COPYDDN

COPYDDN allows you to override the default ddnames or ddname prefixes of your local copy data sets in your installation options module. The ddnames correspond to the data sets that receive a full image copy or DSN1COPY either of the table space or of each partition in the table space that you are loading.

If you are making partition-level copies, specify only the ddname prefix (no *nn*) in this option. However, the ddname that you specify in the JCL must have the partition number *nn* appended to this prefix. The length of *nn* can be from one through seven characters, depending on the length of the ddname prefix. The maximum length of the ddname with the prefix must be eight bytes or less. For example, if you are loading partition 157, you could specify *ddname1* as BCOPY, and specify BCOPY157 in your JCL.

Note: If you use this option to override the default name in the installation options module, you must also change the name in your JCL.

Dynamic work file allocation—If dynamic work file allocation is active and you specify more than one ddname prefix for dynamic allocation, the prefix for each ddname must be different enough for LOADPLUS to differentiate one prefix from another. *Different enough* means that these prefixes must be different *and*, if they are different only because one prefix has additional trailing bytes, then these trailing bytes must contain at least one nonnumeric byte. For example, the first set of prefixes that follow is different enough, but the second set is not:

- acceptable:

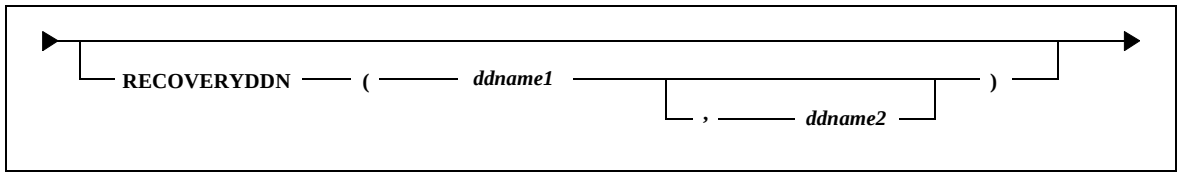
```
BMCRD
BMCRDWK
```

- not acceptable:

```
BMCRD
BMCRD11
```

If you are registering the copies, *ddname1* will be the DB2 local primary and *ddname2* will be the local backup. For information about using these data sets in LOADPLUS, see “Copy Data Sets” on page 4-9.

BMC Software, Inc., Confidential and Proprietary Information



RECOVERYDDN

RECOVERYDDN allows you to override the default ddnames or ddname prefixes of your remote copy data sets in your installation options module. The ddnames correspond to the data sets that receive a full image copy or DSN1COPY either of the table space or of each partition in the table space that you are loading.

If you are making partition-level copies, specify only the ddname prefix (no *nn*) in this option. However, the ddname that you specify in the JCL must have the partition number *nn* appended to this prefix. The length of *nn* can be from one through seven characters, depending on the length of the ddname prefix. The maximum length of the ddname with the prefix must be eight bytes or less. For example, if you are loading partition 203, you could specify *ddname1* as BMCRY, and specify BMCRY203 in your JCL.

Note: If you use this option to override the default name in the installation options module, you must also change the name in your JCL.

Dynamic work file allocation—If dynamic work file allocation is active and you specify more than one ddname prefix for dynamic allocation, the prefix for each ddname must be different enough for LOADPLUS to differentiate one prefix from another. *Different enough* means that these prefixes must be different *and*, if they are different only because one prefix has additional trailing bytes, then these trailing bytes must contain at least one nonnumeric byte. For example, the first set of prefixes that follow is different enough, but the second set is not:

- acceptable:

```
BMCRD
BMCRDWK
```

- not acceptable:

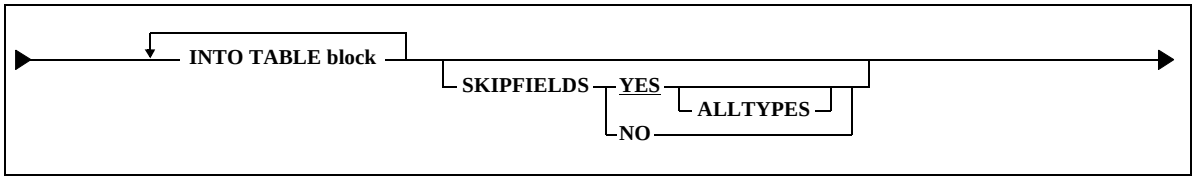
```
BMCRD
BMCRD11
```

If you are registering the copies, *ddname1* will be the DB2 remote primary and *ddname2* will be the remote backup. For information about using these data sets in LOADPLUS, see “Copy Data Sets” on page 4-9.

BMC Software, Inc., Confidential and Proprietary Information

INTO TABLE Options

The options that you specify with INTO TABLE describe the objects and data that you want to load.



INTO TABLE block

Complete details about the keywords that make up the INTO TABLE option begin on page 3-89.

SKIPFIELDS

This option tells LOADPLUS whether to allow fields in your field specifications whose names do not match any column name of any table that is specified in an INTO statement on a single LOAD command. These “skipped” fields are used as placeholders in the input record or are used in predicates, such as NULLIF, for conditional processing.

YES

SKIPFIELDS YES is the default. This option specifies that LOADPLUS not diagnose any CHARACTER, VARCHAR, VARGRAPHIC, or ROWID field whose name does not match any column name of a table that is specified in an INTO statement.

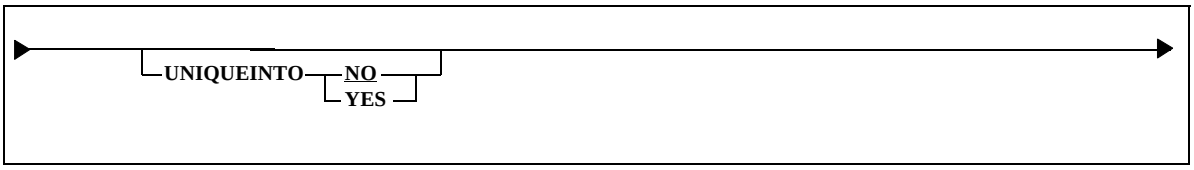
ALLTYPES

ALLTYPES specifies that LOADPLUS not diagnose any field, regardless of type, whose name does not match any column name of a table that is specified in an INTO statement.

NO

SKIPFIELDS NO tells LOADPLUS to fail if any field name that is specified in an INTO statement does not match a column name of the table that you are loading. Specifying this option prevents errors in which misspelled field names in the field specification cause LOADPLUS to load nulls (if the column is nullable) or default values (if the column is defined as NOT NULL WITH DEFAULT).

BMC Software, Inc., Confidential and Proprietary Information



UNIQUEINTO

The UNIQUEINTO option specifies whether each record matches only one INTO specification when you include multiple INTO statements on your LOAD command.

For information about how to enhance LOADPLUS performance when using this option, see “INTO and UNIQUEINTO Command Options” on page 6-25.

NO

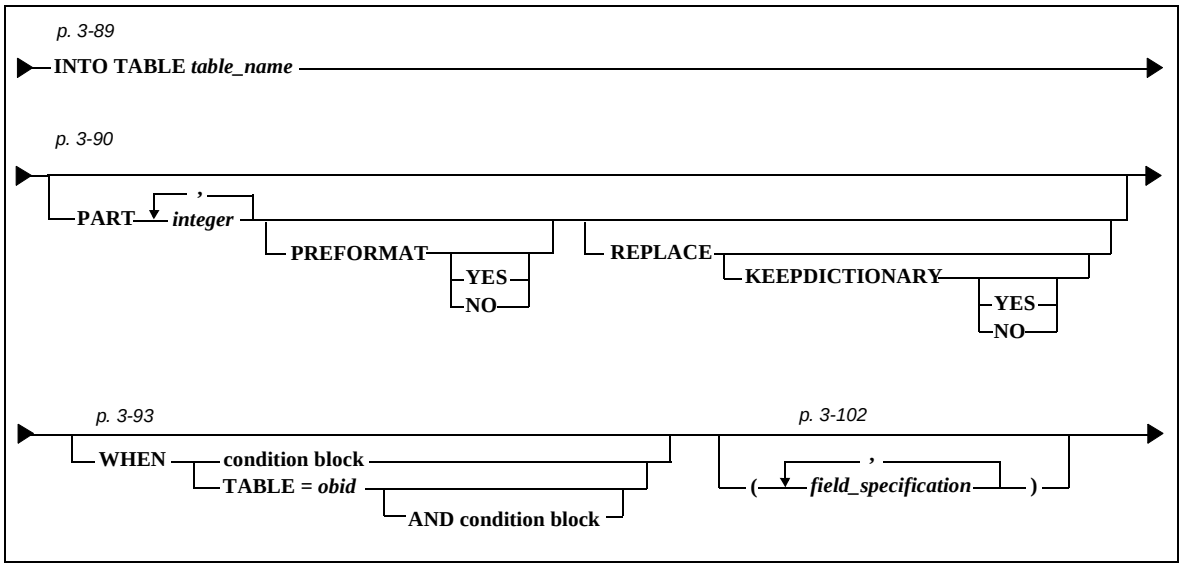
UNIQUEINTO NO is the default. UNIQUEINTO NO tells LOADPLUS to try to match each input record to each INTO specification.

YES

UNIQUEINTO YES tells LOADPLUS that, once an INTO specification selects an input record, LOADPLUS should not test that record against other INTO specifications.

Using UNIQUEINTO YES where appropriate can dramatically reduce execution times for multiple partition loads and multiple table loads if each input record matches at most one INTO specification.

BMC Software, Inc., Confidential and Proprietary Information



INTO TABLE

The INTO TABLE option names a table to be loaded. If you do not qualify the table name with the creator, LOADPLUS uses the creator name that was established when LOADPLUS connected to DB2.

Note: For DB2 systems defined as MIXED=YES, LOADPLUS supports the use of DBCS characters. You can use DBCS characters in DB2 identifiers, such as table or column names, or in the constants and field names specified in your INTO TABLE option.

If the table has a primary key, its primary index must exist. If the table uses a VALIDPROC, an EDITPROC, or both, you must include the routines in your LINKLIB, JOBLIB, or STEPLIB.

You must specify at least one INTO TABLE option for each table that you load. All tables that you specify must reside in the same table space.

Use the following guidelines to determine how many INTO statements to use for your job and when to use the PART option:

- Under either of the following circumstances, specify only one INTO statement without the PART option. The resulting clustering key determines the partition into which the record is loaded, so the PART option is not necessary.

  — if you specify LOAD REPLACE
  — if you specify LOAD RESUME YES and are loading the majority of partitions in a partitioned table space

BMC Software, Inc., Confidential and Proprietary Information

- If you specify LOAD RESUME YES and are loading only a few partitions, specify all partitions on one INTO statement (for example, INTO PART 1, 2, 3). This specification eliminates unloading and loading all partitions of the clustering index and can save considerable time.

- Avoid specifying a separate INTO statement with PART option for each partition.

Note: LOADPLUS optimizes the processing of INTO statements by combining statements that are the same. For example, if you specify INTO PART 1 and INTO PART 1 on the same LOAD command, LOADPLUS combines these statements into a single INTO statement and loads one set of rows. In contrast, DB2 loads two sets of rows into partition 1 in this example.

For information about how to enhance LOADPLUS performance when using this option, see "INTO and UNIQUEINTO Command Options" on page 6-25.

PART

The PART option specifies the partition number of a partitioned table space that you want to load. LOADPLUS loads only data within the limit key range of the specified partition. You can specify one, some, or all of the partitions. If you do not specify PART, LOADPLUS loads all partitions of the table space.

For guidelines about when to use the PART option, see the preceding INTO TABLE discussion.

Additional Information—Note the following additional information about the PART option:

- If you specify multiple partition numbers on one INTO TABLE block, LOADPLUS allows only one INTO TABLE block.

- If data exists in partitions that are not specified on an INTO TABLE option, specify RESUME YES in the LOAD command to save that data. Otherwise, processing terminates.

PREFORMAT

The PREFORMAT option tells LOADPLUS whether to preformat the unused pages of the data set. For compatibility with earlier releases, if you specify PREFORMAT without a value, LOADPLUS assumes PREFORMAT YES.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS also provides PREFORMAT support at the global level. See “PREFORMAT” on page 3-69 for information.

You can specify the default for the PREFORMAT option in your installation options module (using the PREFORMAT installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

YES

If you specify PREFORMAT YES, LOADPLUS preformats the unused portion of the data set. Preformatting writes pages that have been initialized with zeros up to the high-allocated relative byte address (RBA) of the table space part and index part. Preformatting occurs after LOADPLUS loads the data and builds the indexes.

For a SQLAPPLY load, LOADPLUS ignores this option and issues message BMC50109I.

NO

If you specify PREFORMAT NO, LOADPLUS does not preformat the unused pages.

REPLACE

If you specify PART REPLACE, LOADPLUS deletes the existing data in the partition before it loads the data. To delete existing data in all partitions, including those that you do not specify on an INTO option, use LOAD REPLACE.

Warning! If you specify REPLACE, it must immediately follow the PART *integer* option. If anything comes between PART *integer* and REPLACE, LOADPLUS interprets the REPLACE as a global replace of the table space.

LOADPLUS terminates when you specify PART REPLACE under either of the following conditions:

- you are running a SQLAPPLY load
- you are loading a base table defined with LOB columns (BLOB, CLOB, and DBCLOB)

BMC Software, Inc., Confidential and Proprietary Information

KEEPDICTIONARY

This option tells LOADPLUS whether to keep the existing compression dictionary for the partition that you specify with the PART option. For compatibility with earlier releases, if you specify KEEPDICTIONARY without a value, LOADPLUS assumes KEEPDICTIONARY YES.

LOADPLUS also provides KEEPDICTIONARY support at the global level. See “KEEPDICTIONARY” on page 3-17 for information.

You can specify the default for the KEEPDICTIONARY option in your installation options module (using the KEEPDICTIONARY installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

YES

If you specify KEEPDICTIONARY YES, LOADPLUS keeps the existing compression dictionary.

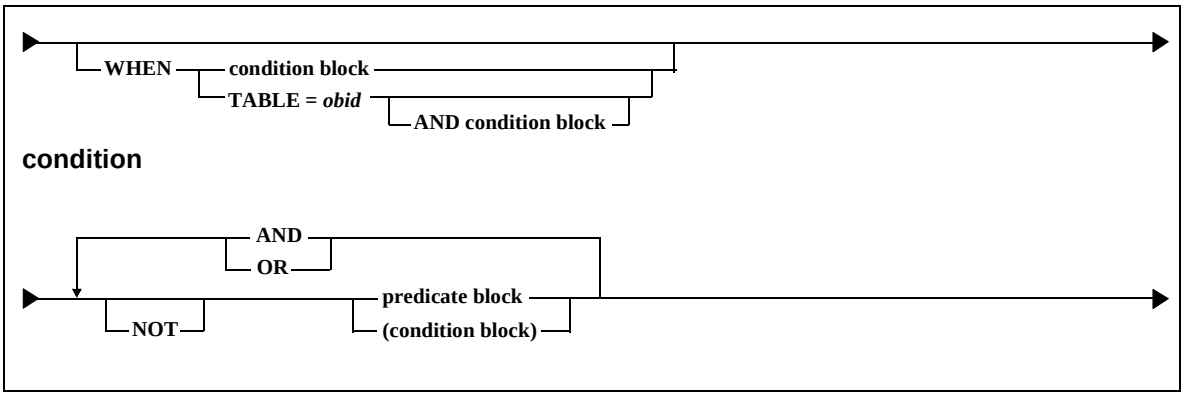
This option is valid only if the table space partition that you are loading has the COMPRESS YES attribute. If a dictionary already exists, LOADPLUS uses it for compression. If a dictionary does not exist, LOADPLUS builds one in the PRELOAD phase or the COMBINED phase. After completely building the dictionary, LOADPLUS compresses the data.

NO

If you specify KEEPDICTIONARY NO, LOADPLUS builds a new compression dictionary.

This option is valid only if the table space partition that you are loading has the COMPRESS YES attribute. LOADPLUS builds the dictionary in the PRELOAD phase or the COMBINED phase. After completely building the dictionary, LOADPLUS compresses the data.

BMC Software, Inc., Confidential and Proprietary Information



WHEN

The WHEN keyword is valid with INTO TABLE options. To see where the WHEN syntax fits in the INTO TABLE option, see the diagram on page 3-89.

You can combine the WHEN keyword with a condition that allows you to specify the condition that must be true for LOADPLUS to select a row. You can also use the WHEN keyword with a TABLE=*obid* option for translating the input table OBID in the input record. This section describes the TABLE=*obid* specification first and follows with a complete discussion of conditions and predicates.

The value of the RULES installation option designated at installation affects the ways in which WHEN, NULLIF, and DEFAULTIF work:

- With RULES=STANDARD, LOADPLUS performs WHEN testing on the input data stream before any NULLIF or DEFAULTIF data conversions.

- With RULES=BMC, LOADPLUS performs WHEN testing after any specified conversion to DB2 internal format.

Therefore, the same LOAD command can produce different results, depending on which option was specified at LOADPLUS installation. See Appendix E, “RULES Installation Option Examples” for details and examples.

For information about how to enhance LOADPLUS performance when using the WHEN option, see “RULES Installation Option” on page 6-24.

BMC Software, Inc., Confidential and Proprietary Information

TABLE=*obid*

If you load data by using the FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD option, you can specify the TABLE=*obid* option. Specify the table OBID as a decimal number. LOADPLUS loads only those records that match the specified table OBID. Table 3-10 describes considerations for using this option with each FORMAT option.

Table 3-10 TABLE=*obid* Option

FORMAT Option	Considerations
FORMAT UNLOAD	Specify TABLE=*obid* to translate the input table OBID in the input record when you are loading data from REORG UNLOAD ONLY output. One application for this option with FORMAT UNLOAD is to load data if the table OBID has changed. This situation might occur, for example, if you • perform a REORG UNLOAD ONLY • then issue a DROP TABLE command • then issue an SQL CREATE TABLE statement (using the same column definitions) To use this option under these circumstances, specify the table's old OBID in TABLE=*obid*.
FORMAT BMC	Specify TABLE=*obid* to translate the input table OBID in the input record when you are loading data from a REORG PLUS SYSARC data set.
FORMAT BMCUNLOAD	You must include TABLE=*obid* when you specify FORMAT BMCUNLOAD (which enables you to load data from UNLOAD PLUS). In this case, the OBID is the object identifier that UNLOAD PLUS assigned when unloading the data, not the table's DB2 object identifier. The LOAD control cards that UNLOAD PLUS generated contain this OBID. For an example, see "Example 14: LOAD REPLACE from UNLOAD PLUS Data in Internal Format" on page 5-106.

condition block

A condition is an expression that combines predicates. Predicates are comparisons of a field name, column name, or *start:end* field to a constant. (For a detailed discussion of predicates, see page 3-96.) You can use conditions and predicates with the WHEN, NULLIF, and DEFAULTIF options.

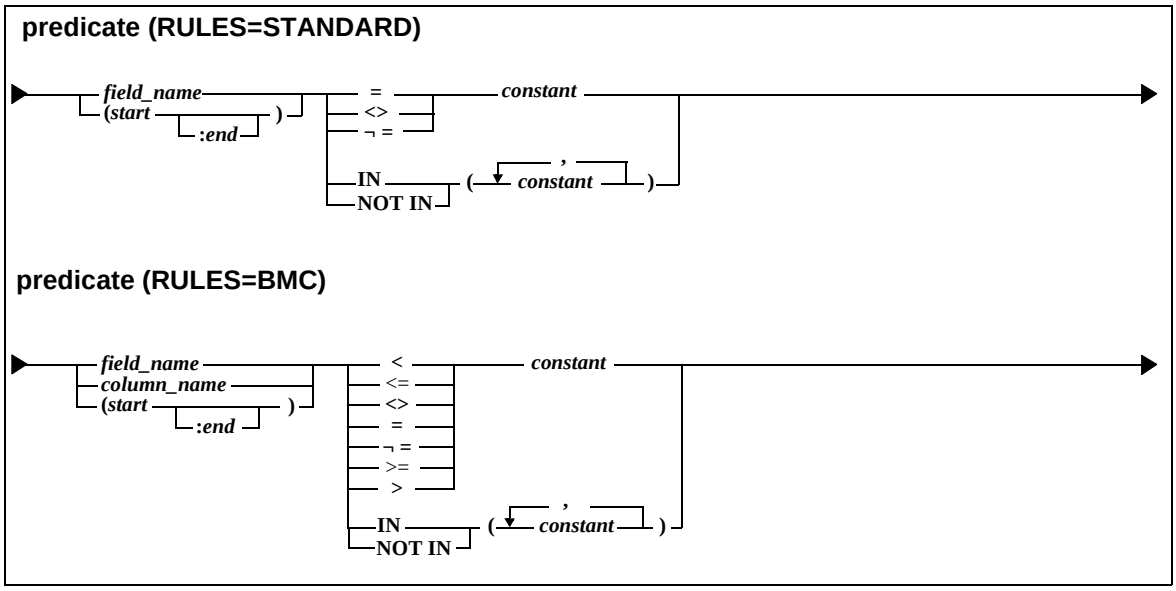
BMC Software, Inc., Confidential and Proprietary Information

A condition uses the parenthetical operators ( ) and the operators AND, OR, and NOT. LOADPLUS evaluates conditions and predicates left to right, with the following operator precedence:

- ( )
- NOT
- AND
- OR

As soon as LOADPLUS determines the resulting value of the overall condition, evaluation of any further predicates stops. If the result is true, the row is selected. If the result is false, the row is not selected.

BMC Software, Inc., Confidential and Proprietary Information



predicates

Predicates are comparisons of a field name, column name, or *start:end* field to a constant. LOADPLUS uses predicates as part of WHEN, NULLIF, and DEFAULTIF syntax. The preceding diagram compares predicate operation when the RULES installation option is set to STANDARD or to BMC.

The predicate operator dictates the nature of the comparison. The following examples illustrate sample predicates:

Example 1: `CITY='AUSTIN'`

Example 2: `(1:6)¬=X'00000000000C'`

Example 3: `INT_COL > 5`

CITY in Example 1 can be a field name or column name. When used with predicates, a name is considered a field name if there is no column with an identical name in the table that you are loading. Table 3-11 on page 3-97 illustrates the differences between specifying field names and column names. The sections that follow this table describe these specifications in more detail.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-11 Field and Column Names in Predicates

Type of Name	Description
field name	• must exist in the field specifications • does not have to have a corresponding column in the table
column name	• must be a column of the table that you are loading • does not have to have a field specification

field_name

If you specify a field name, the field name must be included in your field specification. (For more information about field specifications, see page 3-102.) The length of the field must be less than or equal to 255.

When you specify a name on your predicate and no column of the same name exists in the table that you are loading, LOADPLUS assumes that you have specified a field name. When the value of the RULES installation option is STANDARD, LOADPLUS treats all names that you specify on your predicate as field names, regardless of whether a column of the same name exists in the table that you are loading.

Field names in predicates have the following restrictions:

- A field name cannot refer to a LOB indicator column, a ROWID column, or an identity column.
- If you specify WHEN with FORMAT BMC, FORMAT UNLOAD, or FORMAT BMCUNLOAD, you cannot specify a field name unless it has a corresponding column in the table. You must specify a column name.

***column_name* (RULES=BMC)**

If you specify a column name, it must be a column of the named table and the length of the column must be less than or equal to 255. Column names do not have to have corresponding field specifications. LOADPLUS accepts column names that include DBCS characters.

The column name cannot refer to a LOB indicator column, a ROWID column, or an identity column.

BMC Software, Inc., Confidential and Proprietary Information

(*start*:*end*)

If you specify (*start:end*), the string defined by *start:end* must be contained entirely in the input record. The length of the comparison, that is, the length indicated by (*start:end*), must not be greater than 255 or less than the length of the string. If you do not specify the ending position, LOADPLUS uses the length of the constant for the comparison.

If you specify WHEN with FORMAT BMC, FORMAT UNLOAD, FORMAT BMCUNLOAD, or FORMAT CSV, you cannot specify (*start:end*). For FORMAT BMC, FORMAT UNLOAD, and FORMAT BMCUNLOAD, you must specify a column name. For FORMAT CSV, you must specify a field name or column name.

comparison operators

Use comparison operators in LOADPLUS predicates to indicate how you want LOADPLUS to compare the left and right sides of predicates.

Comparison rules govern the availability of operators and how processing works. Two sets of comparison rules are available with the RULES installation option—STANDARD and BMC. Each set of rules permits different comparison operators as described in Table 3-12. For more information about the RULES option, see Appendix E, “RULES Installation Option Examples.”

Table 3-12 Comparison Operators

Operators		RULES= STANDARD	RULES= BMC
<	less than		X
<=	less than or equal		X
=	equal	X	X
<>	not equal	X	X
¬ =	not equal	X	X
>=	greater than or equal		X
>	greater than		X
IN	equal to any	X	X
NOT IN	not equal to all	X	X

BMC Software, Inc., Confidential and Proprietary Information

constant

The right side argument of a predicate is a constant. Two different sets of rules apply to constants based on the value of the RULES option that is in effect.

Note: The RULES installation option affects the ways in which WHEN, NULLIF, and DEFAULTIF work. See Appendix E, "RULES Installation Option Examples" for details.

Table 3-13 describes the requirements for constants in predicates. In addition, consider the following information:

- Although not prevented or restricted by DB2, you should avoid non-standard comparisons. See rules for constants in the IBM *SQL Reference*.
- If you specify a list of constants, you can use ascending sequential order to improve performance. When you specify the list in this order, you save processing time because LOADPLUS does not sort the list.

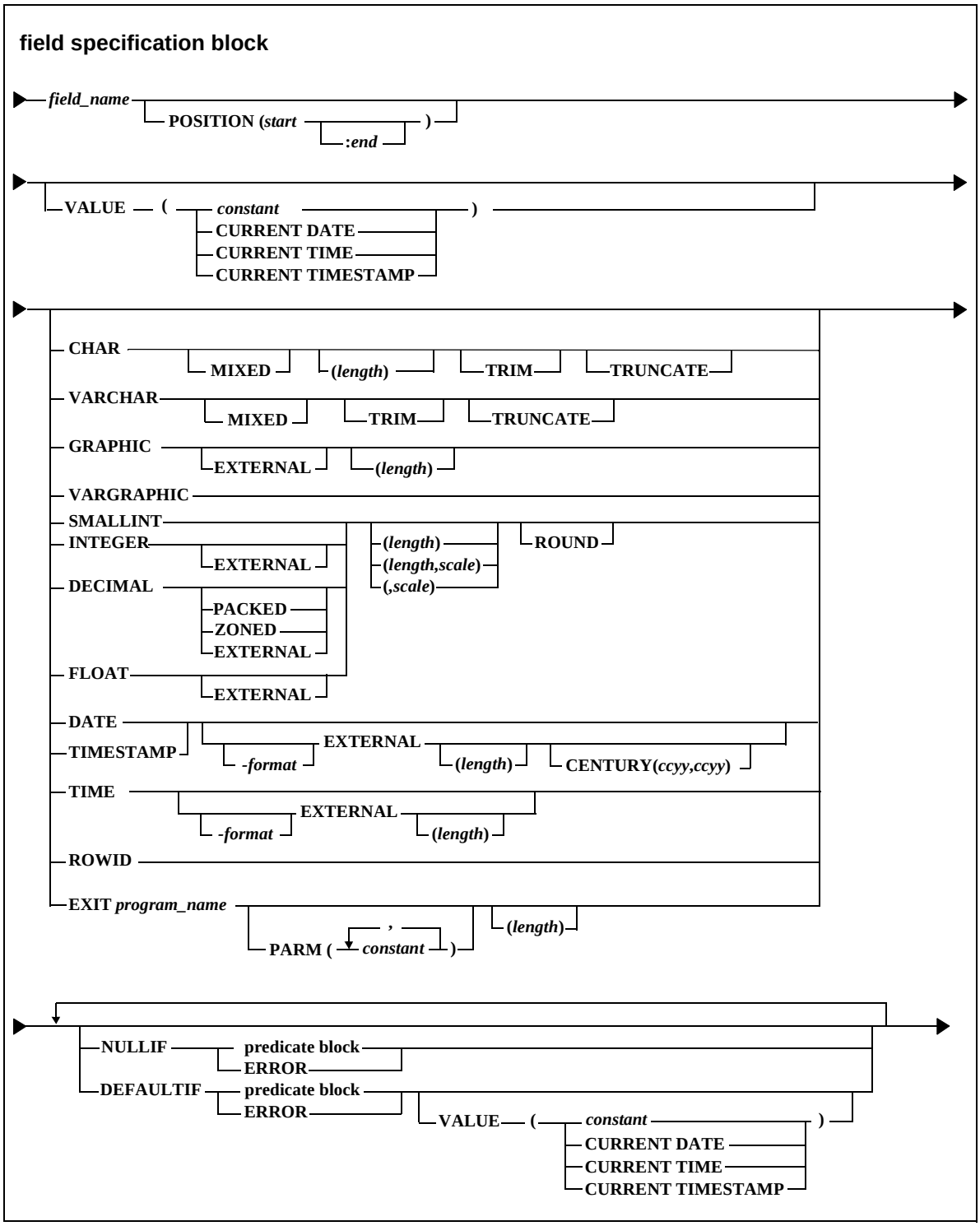
Table 3-13 *constant* Requirements for Predicates (Part 1 of 2)

RULES Value	Left Side of Predicate[a]	Constant Type	Requirements
STANDARD	*field_name*	character string	must be enclosed in single quotes
		hex string	must be enclosed in single quotes (for example, X'000C')
		other	not supported
	(*start:end*)	character string	• must be enclosed in single quotes • length must match the length specified by (*start:end*)
		hex string	• must be enclosed in single quotes (for example, X'000C') • length must match the length specified by (*start:end*)
		other	not supported
	column_name	any	not supported

Table 3-13 *constant* Requirements for Predicates (Part 2 of 2)

RULES Value	Left Side of Predicate[a]	Constant Type	Requirements
BMC	*field_name*	character string	must be enclosed in single quotes
		hex string	must be enclosed in single quotes (for example, X'000C')
		other	not supported
	(*start:end*)	character string	• must be enclosed in single quotes • length must match the length specified by (*start:end*) • corresponding table columns can be any data type
		hex string	• must be enclosed in single quotes (for example, X'000C') • length must match the length specified by (*start:end*) • corresponding table columns can be any data type
		other	not supported
	column_name	integer	column must be a numeric data type
		decimal	column must be a numeric data type
		floating point	not supported
		character string	• column must be a character, date, time, or timestamp data type • must be enclosed in single quotes
		hex string	• must be enclosed in single quotes (for example, X'000C') • column can be any data type • length must match the length of the column's internal data For a VARCHAR column, the length must be the same length as the maximum length of the column. • for a VARCHAR column, must include the two-byte hex length of the data • must include a preceding null indicator if the column is nullable For a VARCHAR column, the length indicator must precede the null indicator. • must be FIELDPROC encoded • must be in DB2 internal format
		graphic string	column must be a graphic data type **Note**: LOADPLUS does not validate graphic input data to determine whether it contains valid graphic coded character set identifier (GCCSID) values.
		NULL	column must be nullable **Note**: If used with an inequality operator, NULL always collates higher than all other values.
		other	not supported

[a] See Table 3-11 on page 3-97 for a description of the difference between field name and column name specifications.


field specification block
field_name
POSITION (start :end)
VALUE (constant CURRENT DATE CURRENT TIME CURRENT TIMESTAMP)
CHAR MIXED (length) TRIM TRUNCATE
VARCHAR MIXED TRIM TRUNCATE
GRAPHIC EXTERNAL (length)
VARGRAPHIC
SMALLINT
INTEGER EXTERNAL
DECIMAL PACKED ZONED EXTERNAL
FLOAT EXTERNAL
(length) (length,scale) (,scale)
ROUND
DATE
TIMESTAMP -format EXTERNAL (length) CENTURY(ccyy,ccyy)
TIME -format EXTERNAL (length)
ROWID
EXIT program_name PARM (, constant) (length)
NULLIF predicate block ERROR
DEFAULTIF predicate block ERROR
VALUE (constant CURRENT DATE CURRENT TIME CURRENT TIMESTAMP)

field_specification

Note: To see where the field specification syntax fits in the INTO TABLE option, see the diagram on page 3-89.

The field specification options determine the format of the input data. They include the following specifications:

- field name, which is usually a column name in the table that you specify
- starting position of the field value in the input record
- data type, data format, and data length of the field
- other processing options

Table 3-14 describes the requirements that apply to the field specification.

Table 3-14 Field Specification Requirements

Condition	Requirement
you are specifying POSITION	You must specify POSITION before the data type.
table contains columns defined as NOT NULL	You must have field specifications for either all of the NOT NULL columns or none of them.
FORMAT UNLOAD FORMAT BMC FORMAT BMCUNLOAD	You cannot use any field specifications.
FORMAT CSV	• You must include a field specification for each input field in the order that the input field occurs in your input file. The exception to this requirement occurs when you do not want to load the last field or fields in the CSV file. LOADPLUS ignores any extra input fields at the end of your input file. **Note**: If you do not want to load a field other than the last field in the CSV file, you must include a placeholder field specification for that input field. • If you want LOADPLUS to interpret two consecutive delimiter characters as a null field, you must include the following NULLIF option in your field specification: `NULLIF` *`field_name`* `= ''` **Note**: Alternatively, you can elect to not include any field specifications. If you do not include any field specifications, LOADPLUS loads the data as if you had included a field specification that maps to every column that you are loading.
LOB indicator field	You cannot include this field in your field specification.
table contains an identity column defined as GENERATED ALWAYS	You cannot include a field specification for the identity column.

BMC Software, Inc., Confidential and Proprietary Information

field_name

In your field specification, you specify a field name for one of the following purposes:

- to identify the data that you want to load into a column of the table
- as a placeholder in the input record
- to identify a field that you want to use with a predicate of a WHEN, NULLIF, or DEFAULTIF option

If the column corresponding to the named field uses a FIELDPROC, the FIELDPROC must be included in your LINKLIB, JOBLIB, or STEPLIB.

The field name specification has the following requirements:

- If you use the named field to specify the data that you want to load into a column and you specify SKIPFIELDS NO, the field name must match the corresponding column name.

- If you use the named field only in the predicate of a WHEN, NULLIF, or DEFAULTIF option (and do not want to load data from that field) or only as a placeholder in the input record, the following requirements apply:

  — You must specify SKIPFIELDS YES.

  — The field name cannot be the name of a column in the table.

  — The data type must be CHARACTER, VARCHAR, VARGRAPHIC, or ROWID unless you specify SKIPFIELDS YES ALLTYPES (in which case the field can be any type).

  — You must supply the length of the field, either with the POSITION(*start*:*end*) option or with the length option. The length must be less than or equal to 255.

    If the data type of the field is CHARACTER, you must explicitly specify a length, even if you use the POSITION option.

    If you use this field specification as a placeholder and you specify FORMAT CSV, you do not need to supply the length of the field. LOADPLUS ignores length specifications with FORMAT CSV.

BMC Software, Inc., Confidential and Proprietary Information

POSITION(*start*:*end*)

This option defines the start (and end) position of the named field. You can abbreviate POSITION to POS. If you specify POSITION, it must appear before the data type on the field specification.

start and *end* designate the locations of the first and last positions of the field. The first position of the record is column one. The start and end positions must be within the bounds of the input record.

Field Length—The end position determines the length of the field, which is calculated as (*end* – *start* + 1). If you specify (*start*:*end*) on the POSITION option and you specify the data type, the length that is derived from (*start*:*end*) must match the explicit or implicit length of the input data type. If there is a conflict between these lengths, LOADPLUS issues message BMC50124E and the job terminates with a return code 8.

If you do not specify POSITION or if you do not specify *end* on the POSITION option when you specify the data type, LOADPLUS uses the field length to determine the length of the input value. BMC Software recommends that, when you specify the data type, you always specify the length or implied length from the associated column rather than specify *end*. To determine the field length from the data type length, see Table 3-24 on page 3-122.

CSV Data—When loading CSV data, LOADPLUS currently starts from the first position of each field, ignoring any POSITION (*start*:*end*) specification. However, BMC Software recommends that you specify POSITION(*).

In the following descriptions, *n* and *m* are integers.

start

This option identifies the starting position of the field. Table 3-15 lists the valid values for *start*.

Table 3-15 Valid *start* Values

Value	Description
n	an actual column number; an absolute position in the input record
*	(default) a relative position one column after the end of the previous field
* + *n*	a relative position *n* + 1 columns after the end of the previous field

BMC Software, Inc., Confidential and Proprietary Information

end

This option identifies the ending position of the field. Table 3-16 lists the valid values for *end*.

Table 3-16 Valid *end* Values

Value	Description
m	an actual column number; an absolute position in the input record You can use an absolute end position only with an absolute start position. The position that *m* represents must be greater than or equal to the position that *n* represents. The length of the field is *m* – *n* + 1.
*	(default) a relative position one column after the end of the current field The length of the field with an end position * and a start position * is 1.
* + *m*	a relative position *m* + 1 columns after the end of the current field The position that *m* represents must be greater than or equal to the position that *n* represents. You can use an end position * + *m* only with a start position of * or * + *n*. The length of the field is *m* – *n* + 1.

VALUE

The VALUE option specifies that you want to load a constant value, CURRENT DATE, CURRENT TIME, or CURRENT TIMESTAMP for the column or field that you specify. You can specify VALUE for a column or field that is defined as any type except ROWID. For ROWID, LOADPLUS uses a unique, generated value as the default.

Note: You can also use the VALUE option with DEFAULTIF, which is discussed on page 3-110.

constant

This value supplies a constant value for the specified column or field. Table 3-17 on page 3-106 describes the requirements for specifying a constant with the VALUE option.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-17 *constant* Requirements for VALUE

Data Type	Requirements
any	You can use a hex string. Hex strings have the following requirements: • Hex strings must be enclosed in single quotes (for example, X'000C'). • The length of the string must match the length of the column's internal data. For a VARCHAR column, the length must be the same length as the maximum length of the column. • For a VARCHAR column, the string must include the two-byte hex length of the data. • The string must include a preceding null indicator if the column is nullable. For a VARCHAR column, the length indicator must precede the null indicator. • The string must be FIELDPROC encoded. • The string must be in DB2 internal format.
numeric	You must use one of the following values: • integer • decimal • hex string See the first row in this table for hex string requirements. • NULL You can specify NULL only if the column is nullable. LOADPLUS does not support floating-point constants.
character	You must use one of the following values: • character string A character string must be enclosed in single quotes. • hex string See the first row in this table for hex string requirements. • NULL You can specify NULL only if the column is nullable.
graphic	You must use one of the following values: • graphic string • hex string See the first row in this table for hex string requirements. • NULL You can specify NULL only if the column is nullable.
date, time, or timestamp	You must use one of the following values: • character string A character string must be enclosed in single quotes. • hex string See the first row in this table for hex string requirements. • NULL You can specify NULL only if the column is nullable.

CURRENT DATE
CURRENT TIME
CURRENT TIMESTAMP

These options specify the current system date, time, and timestamp. The column to which you are loading this value must be a DATE, TIME, or TIMESTAMP column, respectively.

All CURRENT DATE, TIME, or TIMESTAMP column values for a single row are derived from the same system clock value. For each row, LOADPLUS fetches a new system clock value.

For a two-phase load, LOADPLUS assigns the CURRENT DATE, TIME, or TIMESTAMP value to the column during the PRELOAD phase, not when actually loading the row during the LOAD phase.

data_type

The data type option specifies the data type and, in most cases, data format and data length of the input value. For details regarding valid data types and information about using these data types, see the discussion that starts on page 3-110. Table 3-18 describes how the data type specification operates.

Table 3-18 Data Type Specification Operations

Operation or Condition	Details
defaults	For a field with a corresponding column, if you do not specify a data type, LOADPLUS uses the following defaults: • The default data type is the DB2 data type of the column. • The default data format is the internal format of that data type. However, for DATE, TIME, and TIMESTAMP columns, the default date format is the EXTERNAL format. • The default length (or the precision and scale in the case of a DECIMAL column) is the column's length (or precision and scale). For a field with no corresponding column, the default data type is CHARACTER(1).
data conversion	If the field data type, format, and length are not identical to the column data type, format (always internal), and length, LOADPLUS performs a data conversion. See Table 3-23 on page 3-121 for valid conversions. If the corresponding column is defined with a FIELDPROC, LOADPLUS applies the FIELDPROC to the input value after any data conversions.
field length	You can specify a length along with the data type for most data types. In some cases, you must specify a length or POSITION(*start*:*end*) to indicate the length of the input field. In other cases, LOADPLUS uses the default length, or you cannot change the length. See Table 3-24 on page 3-122 for the default lengths of the different data types.
additional processing options	In addition to the data type, format, and length, certain data types allow additional processing options such as TRIM, TRUNCATE, and ROUND. You can find descriptions of these options with the applicable data type descriptions, beginning on page 3-110.
CSV data	For CSV input data, LOADPLUS ignores any data type specification other than a DATE, TIME, or TIMESTAMP external type.

BMC Software, Inc., Confidential and Proprietary Information

NULLIF

NULLIF is a field specification option. It determines whether the value that is assigned to a column is null, based on the condition that you specify. If the condition is satisfied, LOADPLUS assigns null to the column. The column must be nullable. See the diagram on page 3-101 for NULLIF syntax.

For information about how this option affects performance, see "RULES Installation Option" on page 6-24.

Additional Considerations—The following additional considerations apply to the NULLIF option:

- You can use NULLIF only on a field specification that corresponds to a column.
- If you specify multiple NULLIF options, LOADPLUS uses the first matching condition.
- You cannot use an identity column in the predicate of a NULLIF statement on the field specification of another column.
- The RULES installation option designated at installation affects the ways in which WHEN, NULLIF, and DEFAULTIF work. For details, see Appendix E, "RULES Installation Option Examples."

predicate block

See the description in "predicates" on page 3-96 for complete information.

ERROR

ERROR indicates that you want LOADPLUS to assign null to the column if a conversion error occurs. If you specify MSGLEVEL(1) on your EXEC statement or in your installation options, LOADPLUS issues message BMC51501E to indicate the conversion error.

DEFAULTIF

DEFAULTIF is a field specification option that determines whether the value that is assigned to a column is the default, based on the condition that you specify. If the condition is satisfied, LOADPLUS assigns a default value to the column. The column does not have to be defined NOT NULL WITH DEFAULT. See the diagram on page 3-101 for DEFAULTIF syntax.

BMC Software, Inc., Confidential and Proprietary Information

Additional Considerations—The following additional considerations apply to DEFAULTIF:

- To obtain a null value with DEFAULTIF on a nullable column, specify DEFAULTIF *predicate* VALUE(NULL). If you do not specify VALUE with a null indication, the column defaults to the column's data type. This action differs from IBM's DB2 LOAD utility, for which using DEFAULTIF on a nullable column always results in a null value.

  Alternatively, you can use the NULLIF option to obtain the same results.

  Note: You cannot obtain a null value with DEFAULTIF ERROR. See page 3-109 for more details.

- You cannot use an identity column in the predicate of a DEFAULTIF statement on the field specification of another column.

- The RULES installation option designated at installation affects the ways in which WHEN, NULLIF, and DEFAULTIF work. See Appendix E, "RULES Installation Option Examples" for details and examples.

  predicate block

  See the description in "predicates" on page 3-96 for complete information.

  ERROR

  ERROR indicates that you want LOADPLUS to assign the default value to the column if a conversion error occurs. If you specify MSGLEVEL(1) on your EXEC statement or in your installation options, LOADPLUS issues message BMC51501E to indicate the conversion error.

  Note: Use of DEFAULTIF ERROR on a nullable column never results in a null value. The column always defaults to the column's data type. This is different than the IBM DB2 LOAD utility, for which using DEFAULTIF ERROR on a nullable column always results in a null value. If you want a null value for a nullable column when a conversion error occurs, use the NULLIF option.

BMC Software, Inc., Confidential and Proprietary Information

VALUE

VALUE specifies the default value that you want LOADPLUS to use. The default DEFAULTIF VALUE is the same as those described by the DB2 SQL CREATE TABLE statement. The syntax for this value option is the same as the syntax for VALUE as it is used for specifying a constant value for a field or column. For details about using this keyword, see “VALUE” on page 3-105.

Data Types

You can specify the following data types. See the following sections for additional information about data types:

- For common data type information, see “General Rules for Some Types of Data” on page 3-126.
- For valid formats for date, time, and timestamp data types, see the tables that start on page 3-120.
- For valid data type conversions, see Table 3-23 on page 3-121.
- For sample data type conversions, see the tables that start on page 3-124.
- For default input data lengths and field lengths, see the tables that start on page 3-122.

CHAR

CHAR defines a fixed-length character field. You can also specify CHAR as CHARACTER.

MIXED

MIXED defines a mixed character data field.

(*length*)

Length indicates the number of single-byte characters in this field.

TRIM

The TRIM option truncates trailing blanks from the input string value before LOADPLUS attempts to assign the value to a CHAR or VARCHAR column. If the string is still too long, a conversion error occurs. LOADPLUS performs TRIM before TRUNCATE.

TRIM is useful when you are converting from a CHAR field to a VARCHAR column or for converting from CHAR(*m*) to CHAR(*n*), where *m* > *n*, and you want to detect nonblank data beyond position *n*.

TRUNCATE

The TRUNCATE option truncates a string if it is longer than the column length (after TRIMming). LOADPLUS performs TRUNCATE after TRIM. See Table 3-26 on page 3-124 for more information.

VARCHAR

VARCHAR defines a varying-length character string.

MIXED

MIXED defines a mixed character data field.

TRIM

The TRIM option truncates trailing blanks from the input string value before LOADPLUS attempts to assign the value to a CHAR or VARCHAR column. If the string is still too long, a conversion error occurs. LOADPLUS performs TRIM before TRUNCATE.

TRIM is useful when converting from a VARCHAR field to a CHAR(*n*) or VARCHAR(*n*) column and you want to detect nonblank data beyond position *n*.

TRUNCATE

The TRUNCATE option truncates a string if it is longer than the column's maximum length (after TRIMming). LOADPLUS performs TRUNCATE after TRIM. See Table 3-26, on page 3-124, for more information.

GRAPHIC

GRAPHIC defines a fixed-length graphic field. Without the EXTERNAL specification, this data type description indicates an internal graphic value without the shift-out and shift-in characters.

BMC Software, Inc., Confidential and Proprietary Information

GRAPHIC EXTERNAL

This data type description indicates an external graphic value with the shift-out and shift-in characters.

(*length*)

Length indicates the number of double-byte characters. For EXTERNAL, this length does not include the two shift characters.

VARGRAPHIC

VARGRAPHIC indicates a varying-length graphic string. A two-byte length field precedes the string. This length field contains the length of the graphic string. The length is the number of double-byte characters and does not include the two length bytes.

SMALLINT

SMALLINT indicates a two-byte internal integer value. For additional information about numeric fields, see the general rules given on page 3-126.

(*length,scale*)

Length, if specified, must be 2. You can specify scale with or without specifying a length. Specifying scale multiplies or divides the number by a power of 10. For more information about *scale*, see page 3-127.

ROUND

The ROUND option directs LOADPLUS to round the value, if necessary, during conversion. If you do not specify ROUND, LOADPLUS truncates any digits that are not needed for precision.

INTEGER

INTEGER defines an integer field. You can abbreviate INTEGER to INT. Without the EXTERNAL specification, this data type description indicates a four-byte internal integer value. For additional information about numeric fields, see the general rules given on page 3-126.

INTEGER EXTERNAL

This data type description indicates a character string that contains an EBCDIC integer constant.

BMC Software, Inc., Confidential and Proprietary Information

(*length,scale*)

Length, if specified, must be 4. You can specify scale with or without specifying a length. Specifying scale multiplies or divides the number by a power of 10. For more information about *scale*, see page 3-127.

ROUND

The ROUND option directs LOADPLUS to round the value, if necessary, during conversion. If you do not specify ROUND, LOADPLUS truncates any digits that are not needed for precision.

DECIMAL

DECIMAL defines a decimal field. You can abbreviate DECIMAL to DEC. For additional information about numeric fields, see the general rules given on page 3-126.

DECIMAL without an additional keyword functions the same as DECIMAL PACKED.

DECIMAL PACKED

DECIMAL PACKED is the default. It indicates that the input value is packed decimal.

DECIMAL ZONED

This data type description indicates that the input value is zoned decimal.

DECIMAL EXTERNAL

This data type description indicates a character string that contains an EBCDIC decimal constant. If the value includes a decimal point, it overrides the scale specification.

(*length,scale*)

You can specify one or both of these options. *Length* (or precision) is the total number of digits and *scale* is the number of digits in the fractional part of the number. For more information about *scale*, see page 3-127.

ROUND

The ROUND option directs LOADPLUS to round the value, if necessary, during conversion. If you do not specify ROUND, LOADPLUS truncates any digits that are not needed for precision.

BMC Software, Inc., Confidential and Proprietary Information

FLOAT

FLOAT defines a floating-point numeric field. Without the EXTERNAL keyword, this data type description indicates a four- or eight-byte internal floating-point number.

For additional information about numeric fields, see the general rules given on page 3-126. For examples of data type conversions using FLOAT, see Table 3-27 on page 3-124.

Note: LOADPLUS does not support IEEE Binary Floating Point (BFP) format.

FLOAT EXTERNAL

FLOAT EXTERNAL indicates a character string that contains an EBCDIC floating-point constant.

(*length,scale*)

Length, if specified, must be 1 through 21 for a four-byte internal floating-point number. For an eight-byte internal floating-point number, *length* must be 22 through 53. You can specify scale with or without specifying a length. Specifying scale multiplies or divides the number by a power of 10. For more information about *scale*, see page 3-127.

ROUND

The ROUND option directs LOADPLUS to round the value, if necessary, during conversion. If you do not specify ROUND, LOADPLUS truncates any digits that are not needed for precision.

DATE

DATE defines a date field. If you specify this option without EXTERNAL, LOADPLUS does not verify input data.

When loading CSV data (FORMAT CSV), you must specify an EXTERNAL format with your field specification.

DATE

This data type description indicates a DB2 internal date value.

DATE EXTERNAL(*length*)

This data type description indicates a DB2 date string value.

BMC Software, Inc., Confidential and Proprietary Information

DATE-*format* EXTERNAL(*length*)

This data type description indicates an external date value. In addition, it indicates the format of the external date value. See Table 3-20 on page 3-119 for information about the available date formats.

When you specify DATE-*format* EXTERNAL, LOADPLUS loads the current date when the input data equates to spaces or zeros unless the column has a default value of null. If the column has a default value of null, LOADPLUS loads nulls instead of the current date when the input data equates to spaces or zeros.

CENTURY(*ccyy,ccyy*)

CENTURY specifies the 100-year range that determines the century for DATE external formats that contain two-digit year values. The first four-digit year value must be less than the second four-digit year. You must specify both values and these values must span 100 years.

Any two-digit year in the input data that lies between the first *yy* specification and 99 is prefixed with the first *cc* value to create a four-digit year. Any two-digit year in the input data that lies between 00 and the second *yy* specification is prefixed with the second *cc* value to create a four-digit year.

For example, if you specify CENTURY(1950,2049), LOADPLUS places 19 in front of each two-digit year with a value 50 through 99, and places 20 in front of each two-digit year with a value 00 through 49. The date 99/12/31 becomes 1999/12/31 and 00/12/31 becomes 2000/12/31.

For information about specifying CENTURY globally on the LOAD command, see page 3-70. For information about the CENTURY installation option, see Appendix A, “LOADPLUS Installation Options.”

TIMESTAMP

TIMESTAMP defines a timestamp field. If you specify this option without EXTERNAL, LOADPLUS does not verify input data.

When loading CSV data (FORMAT CSV), you must specify an EXTERNAL format with your field specification.

TIMESTAMP

This data type description indicates a DB2 internal timestamp value.

BMC Software, Inc., Confidential and Proprietary Information

TIMESTAMP EXTERNAL(*length*)

This data type description indicates a DB2 timestamp string value.

TIMESTAMP-*format* EXTERNAL(*length*)

This data type description indicates an external timestamp value. In addition, it indicates the format of the external timestamp value. See Table 3-22 on page 3-120 for information about the available timestamp formats.

When you specify TIMESTAMP-*format* EXTERNAL, LOADPLUS loads the current timestamp when the input data equates to spaces or zeros unless the column has a default value of null. If the column has a default value of null, LOADPLUS loads nulls instead of the current date when the input data equates to spaces or zeros.

CENTURY(*ccyy,ccyy*)

CENTURY specifies the 100-year range that determines the century for TIMESTAMP external formats that contain two-digit year values. The first four-digit year value must be less than the second four-digit year. You must specify both values and these values must span 100 years.

Any two-digit year in the input data that lies between the first *yy* specification and 99 is prefixed with the first *cc* value to create a four-digit year. Any two-digit year in the input data that lies between 00 and the second *yy* specification is prefixed with the second *cc* value to create a four-digit year.

For example, if you specify CENTURY(1950,2049), LOADPLUS places 19 in front of each two-digit year with a value 50 through 99, and places 20 in front of each two-digit year with a value 00 through 49. The date 99/12/31 becomes 1999/12/31 and 00/12/31 becomes 2000/12/31.

For information about specifying CENTURY globally on the LOAD command, see page 3-70. For information about the CENTURY installation option, see Appendix A, “LOADPLUS Installation Options.”

TIME

TIME defines a time field. If you specify this option without EXTERNAL, LOADPLUS does not verify input data.

BMC Software, Inc., Confidential and Proprietary Information

When loading CSV data (FORMAT CSV), you must specify an EXTERNAL format with your field specification.

TIME

This data type description indicates a DB2 internal time value.

TIME EXTERNAL(*length*)

This data type description indicates a DB2 time string value.

TIME-*format* EXTERNAL(*length*)

This data type description indicates an external time value. In addition, it indicates the format of the external time value. See Table 3-21 on page 3-120 for information about the available time formats.

When you specify TIME-*format* EXTERNAL(*length*), LOADPLUS loads the current time when the input data equates to spaces or zeros unless the column has a default value of null. If the column has a default value of null, LOADPLUS loads nulls instead of the current date when the input data equates to spaces or zeros.

ROWID

ROWID defines a field that contains generated row IDs. This field must be in internal format—a varying length hex string. The first two bytes of this hex string contain the length of the ROWID string. The length is the number of bytes and does not include the two length bytes.

The following information describes how LOADPLUS processes ROWID fields:

- If the ROWID field is defined as GENERATED ALWAYS, LOADPLUS ignores the value in the field specification and generates its own unique row identifiers.
- If the ROWID field is defined as GENERATED BY DEFAULT, LOADPLUS performs minimal validation on the ROWID field and loads the ROWID data with no conversions.

ROWID fields in LOADPLUS have the following restrictions:

- You cannot reference a ROWID field in a WHEN, DEFAULTIF, or NULLIF clause.
- You cannot specify DEFAULTIF with VALUE or NULLIF for a ROWID field.

BMC Software, Inc., Confidential and Proprietary Information

EXIT

EXIT allows you to request your own data conversion by using a user-written exit routine. You must specify POSITION(*start*:*end*) or the *length* option to define the input field length.

program_name

This value is the name of your conversion exit routine. The program that you name must be included in your LINKLIB, JOBLIB, or STEPLIB.

Table 3-19 shows the return codes that LOADPLUS expects in register 15.

Table 3-19 User Exit Return Codes

Return Code	Description
0	conversion complete
4	conversion error LOADPLUS issues message BMC50289E and rejects this record but continues processing other records.
8	parameter error LOADPLUS issues message BMC50289E and does not process other records. The utility return code is set to 8.

If the exit returns an invalid return code, LOADPLUS issues message BMC50289E and terminates. A sample exit is located in member AMUXDEPT of the *HLQ*.CNTL data set (where *HLQ* is the high-level qualifier specified during installation).

PARM (*constant1, constant2,...*)

You can specify an optional parameter list that contains integer, decimal, string, or hex constants. LOADPLUS does not support floating-point and graphic constants.

The parameter list that you pass to a conversion exit routine is identical to the parameter list that is passed to a DB2 FIELDPROC exit routine; that is, the FPBFCODE in the FPIB is set to FBBFENC (= 0), the CVD is the value from the input record, the FVD is the converted column value, and the FPPVL is the PARM.

Coding requirements for the LOADPLUS conversion exit are the same as those for a DB2 FIELDPROC. For further help with exit requirements, see the IBM *DB2 Administration Guide*.

BMC Software, Inc., Confidential and Proprietary Information

DATE, TIME, and TIMESTAMP Formats

When you specify DATE, TIME, or TIMESTAMP formats, LOADPLUS loads the current date, current time, or current timestamp when the input data equates to spaces or zeros.

Note: For EXTERNAL formats, if the column has a default value of null, LOADPLUS loads nulls instead of the current date, time, or timestamp when the input data equates to spaces or zeros.

The following tables list input date, time, and timestamp formats, other than internal DB2 formats, that LOADPLUS supports.

Note: The format number listed in these tables is the number that you use in place of the *format* in the DATE-*format* EXTERNAL, TIME-*format* EXTERNAL, and TIMESTAMP-*format* EXTERNAL data type specification.

Table 3-20 Date Formats

Non-delimited				Delimited	
Format Number			Format	Format No.	Format[a]
Char	Int	Dec		Char	
1	I1	P1	MMDDYY	D1	MM/DD/YY
2	I2	P2	YYMMDD	D2	YY/MM/DD
3	I3	P3	YYDDD	D3	YY/DDD
4	I4	P4	YYWWD	D4	YY/WW/D
5	I5	P5	DDMMYY	D5	DD/MM/YY
6	I6	P6	DDDDD		
1E	I1E	P1E	MMDDYYYY	D1E	MM/DD/YYYY
2E	I2E	P2E	YYYYMMDD	D2E	YYYY/MM/DD
3E	I3E	P3E	YYYYDDD	D3E	YYYY/DDD
4E	I4E	P4E	YYYYWWD	D4E	YYYY/WW/D
5E	I5E	P5E	DDMMYYYY	D5E	DD/MM/YYYY

[a] **'/'** can be any character.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-21 Time Formats

Non-delimited				Delimited	
Format Number			Format	Format No.	Format[a]
Char	Int	Dec		Char	
1	I1	P1	HHMMSS	D1	HH.MM.SS
2	I2	P2	HHMM	D2	HH.MM
				D3	HH.MM *x*M[b]

[a] **'.'** can be any character.
[b] *x*M is either AM or PM.

Table 3-22 Timestamp Formats

Non-delimited		Delimited[a]	
Format No.	Format	Format No.	Format[b]
Char		Char	
1	YYMMDDHHMMSS	D1	YY-MM-DD-HH.MM.SS
2	YYMMDDHHMMSSNNNNNN	D2	YY-MM-DD-HH.MM.SS.NNNNNN
3	Store clock 8-byte binary		
1E	YYYYMMDDHHMMSS	D1E	YYYY-MM-DD-HH.MM.SS
2E	YYYYMMDDHHMMSSNNNNNN	D2E	YYYY-MM-DD-HH.MM.SS.NNNNNN

[a] Delimited timestamp formats are valid only if you specify FORMAT CSV.
[b] **'-'** and '.' can be any character.

Data Type Conversion

The tables in this section provide the following information:

- allowable data conversions
- default lengths of input values
- field types and lengths
- sample conversion results

Table 3-23 Allowable Data Type Conversions

	Column[a]											
Input	**SMALLINT**	**INTEGER**	**DECIMAL**	**FLOAT**	**CHARACTER**	**VARCHAR**	**GRAPHIC**	**VARGRAPHIC**	**DATE**	**TIME**	**TIMESTAMP**	**ROWID**
SMALLINT	D	X	X	X[b]	X	X						
INTEGER	X	D	X	X[b]	X	X			X	X		
INTEGER EXTERNAL	X	X	X	X[b]	X	X			X	X		
DECIMAL	X	X	D	X[b]	X	X			X	X		
DECIMAL EXTERNAL	X	X	X	X[b]	X	X			X	X		
DECIMAL ZONED	X	X	X	X[b]	X	X						
FLOAT[c]	X	X	X	D	X	X						
FLOAT EXTERNAL[c]	X	X	X	X	X	X						
CHAR	X	X	X	X	D	X			X	X	X	
VARCHAR	X	X	X	X	X	D			X	X	X	
GRAPHIC							D	X				
GRAPHIC EXTERNAL							X	X				
VARGRAPHIC							X	D				
DATE[d]					X	X			X			
DATE EXTERNAL[d]									D			
DATE-*format* EXTERNAL[d,e]					X	X			X			
TIME[d]					X	X				X		
TIME EXTERNAL[d]										D		
TIME-*format* EXTERNAL[d,e]					X	X				X		
TIMESTAMP EXTERNAL[d]									X	X	D	
TIMESTAMP-*format* EXTERNAL[d,e]					X	X			X	X	X	
ROWID												D
EXIT	X	X	X	X	X	X	X	X	X	X	X	

[a]The column values are

X—allowable conversion

D—allowable conversion and indicates the default input data type

[b] This conversion is approximate and the value in the floating-point column might not be identical to the value in the input field.

[c] LOADPLUS does not support IEEE BFP format.

[d] For a date, time, or timestamp column, the default input data format is EXTERNAL. The default input length (or precision and scale for DECIMAL) is the column length (or precision and scale).

[e] The *format* number describes the representation of the input date, time, or timestamp value. If you don't specify the format number, then ISO, USA, EUR, JIS, or LOCAL is assumed.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-24 Default Lengths of Input Values

	Column[a]											
Input	**SMALLINT**	**INTEGER**	**DECIMAL**	**FLOAT**	**CHARACTER**	**VARCHAR**	**GRAPHIC**	**VARGRAPHIC**	**DATE**	**TIME**	**TIMESTAMP**	**ROWID**
SMALLINT	2	2	2	2	2	2						
INTEGER	4	4	4	4	4	4			4	4		
INTEGER EXTERNAL	X	X	X	X	X	X			8	6		
DECIMAL	X	X	C	X	X	X			8	6		
DECIMAL EXTERNAL	X	X	X	X	X	X			8	6		
DECIMAL ZONED	X	X	C	X	X	X			8	6		
FLOAT	X	X	X	C	X	X						
FLOAT EXTERNAL	X	X	X	X	X	X						
CHAR	X	X	X	X	C	C			10	8	26	
VARCHAR	V	V	V	V	V	V			V	V	V	
GRAPHIC							C	C				
GRAPHIC EXTERNAL[b]							C	C				
VARGRAPHIC							V	V				
DATE					4	4			4			
DATE EXTERNAL									L			
DATE-*format* EXTERNAL					T	T			T			
TIME					3	3				3		
TIME EXTERNAL										L		
TIME-*format* EXTERNAL					T	T				T		
TIMESTAMP					10	10			10	10	10	
TIMESTAMP EXTERNAL									26	26	26	
TIMESTAMP-*format* EXTERNAL					T	T			T	T	T	
ROWID												V
EXIT	X	X	X	X	X	X	X	X	X	X	X	

[a]The column values are

numeric values—These values indicate the required input length.

X—These data types require explicit length.

C—The column defines the default length. For DECIMAL, the column defines the precision and scale.

V—The input value defines the length. A two-byte length value precedes the field value.

L—These data types use local DATE and TIME length.

T—See the tables that begin on page 3-120 for length information.

[b] GRAPHIC EXTERNAL includes the shift-out and shift-in characters.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-25 Field Types and Field Lengths

Field Type	Field Length (in bytes)
SMALLINT	always 2
INTEGER	always 4
INTEGER EXTERNAL(*n*)	*n*
DECIMAL(*n*)	integer(*p*/2) + 1
DECIMAL EXTERNAL(*n*)	*n*
DECIMAL ZONED(*n*)	*n*
FLOAT(*n*)	4 if $1 \leq n \leq 21$ 8 if $22 \leq n \leq 53$
FLOAT EXTERNAL(*n*)	*n*
CHAR(*n*)	*n*
VARCHAR	the length is determined by the two-byte length + 2
GRAPHIC(*n*)	2*n*
GRAPHIC EXTERNAL(*n*)	2*n* + 2 for the shift characters
VARGRAPHIC	the length is determined by the two-byte length $\times$ 2 + 2
DATE	always 4
DATE EXTERNAL(*n*)	*n*
DATE-*format* EXTERNAL(*n*)	*n* if CHARACTER always 4 if INTEGER integer(*n*/2) + 1 if DECIMAL
TIME	always 3
TIME EXTERNAL(*n*)	*n*
TIME-*format* EXTERNAL(*n*)	*n* if CHARACTER always 4 if INTEGER integer(*n*/2) + 1 if DECIMAL
TIMESTAMP	always 10
TIMESTAMP EXTERNAL(*n*)	*n*
TIMESTAMP-*format* EXTERNAL(*n*)	*n* if CHARACTER always 4 if INTEGER integer(*n*/2) + 1 if DECIMAL
ROWID	the length is determined by the two-byte length + 2
EXIT(*n*)	*n*

BMC Software, Inc., Confidential and Proprietary Information

Table 3-26 Sample Character Data Conversions

Data Type Input	Data Type Column	Data Input[a]	Data Column[a,b]	Comments
CHAR	CHAR(8)	'ABCDEFbb'	'ABCDEFbb'	input length defaults to column length
CHAR(6)	CHAR(8)	'ABCDEF'	'ABCDEFbb'	result padded with blanks
CHAR(8)	VARCHAR(8)	'ABCDEFbb'	X'0008','ABCDEFbb'	
CHAR(8) TRIM	VARCHAR(6)	'ABCDEFbb'	X'0006','ABCDEF'	trailing blanks truncated
CHAR(8)	VARCHAR(6)	'ABCDEFbb'	conversion error	target too small
VARCHAR	CHAR(8)	X'0004','ABCD'	'ABCDbbbb'	result padded with blanks
VARCHAR	VARCHAR(6)	X'0008','ABCDEFGH'	conversion error	target too small

[a] In the Data Input and Data Column values
Character strings are depicted as '*string*'.
b represents a blank in the character strings.
Hex strings are depicted as X'*hex_string*'.
Noncharacter input and column values are depicted as hex strings.

[b] In the Data Column values
All column values are depicted in non-normalized DB2 internal format.
All columns are defined NOT NULL and no column values contain the null indicator.

Table 3-27 Sample Numeric Data Conversions (Part 1 of 2)

Data Type Input	Data Type Column	Data Input[a]	Data Column[a,b]	Comments
INTEGER EXTERNAL(10)	INTEGER	'bbbbbbbb10'	X'0000000A'	
INTEGER EXTERNAL(10,-1)	INTEGER	'bbbb10bbbb'	X'00000064'	result = 100
INTEGER EXTERNAL(10)	INTEGER	'9999999999'	conversion error	result larger than maximum integer
INTEGER	INTEGER(,2)	X'00000064'	X'00000001'	input value = 100 result = 1
INTEGER[c]	CHAR(10)	X'00000064'	'bbbbbbb100'	input value = 100
INTEGER(,2)	DECIMAL(5,2)	X'0000007D'	X'00125C'	input value = 125 result = 1.25
DECIMAL EXTERNAL	DECIMAL(5,2)	'12345'	X'12345C'	result = 123.45
DECIMAL EXTERNAL(7,-2)	DECIMAL(7,2)	'bb12345'	X'1234500C'	result = 12345.00
DECIMAL EXTERNAL(10,4)	DECIMAL(8,2)	'12345678.b'	conversion error	target too small
DECIMAL EXTERNAL(5,1)	INTEGER	'bb15b'	X'00000001'	fraction truncated
DECIMAL EXTERNAL(5,2)	DECIMAL(5,2)	'-1234'	X'01234D'	result = -12.34
DECIMAL EXTERNAL(5,1) ROUND	DECIMAL(5)	'bbb15'	X'00002C'	result = 2 due to rounding

BMC Software, Inc., Confidential and Proprietary Information

Table 3-27 Sample Numeric Data Conversions (Part 2 of 2)

Data Type Input	Data Type Column	Data Input[a]	Data Column[a,b]	Comments
DECIMAL ZONED(4,0)	DECIMAL(4,0)	X'F1F2F3D3'	X'01233D'	result = -1233.
DECIMAL ZONED(4,0)	DECIMAL(4,0)	X'F1F2F3C4'	X'01234C'	result = 1234.
DECIMAL	DECIMAL(5,2)	X'12345C'	X'12345C'	input and result = 123.45
DECIMAL(5,2)	DECIMAL(4,2)	X'12345C'	conversion error	target too small
FLOAT(53)	DECIMAL(5,1)	X'40DC28F5C28F5C28'	X'00859C'	input = .85999999999999999E+0
FLOAT(53)	DECIMAL(7,4)	X'4239C65B1343EC2E'	X'0577748C'	input = .57774781421740208E+2 ROUND is specified

[a] In the Data Input and Data Column values
Character strings are depicted as '*string*'.
b represents a blank in the character strings.
Hex strings are depicted as X'*hex_string*'.
Noncharacter input and column values are depicted as hex strings.

[b] In the Data Column values
All column values are depicted in non-normalized DB2 internal format.
All columns are defined NOT NULL and no column values contain the null indicator.

[c] For additional examples of INTEGER to CHAR conversions, see “Converting to Character Data” on page 3-127.

Table 3-28 Sample Date, Time, and Timestamp Data Conversions (Part 1 of 2)

Data Type Input	Data Type Column	Data Input[a]	Data Column[a,b]	Comments
No data type	DATE	'01/31/1990'	X'19900131'	input data type defaults to DATE EXTERNAL
DATE	DATE	X'19900131'	X'19900131'	
DATE EXTERNAL	DATE	'1990-2-29b'	conversion error	day out of range
DATE-2 EXTERNAL CENTURY (1950-2049)	DATE	'900131'	X'19900131'	year '90' becomes 1990 with CENTURY specification
DATE-D1 EXTERNAL	DATE	'01/31/90'	X'19900131'	
DATE-D1 EXTERNAL CENTURY (1950-2049)	DATE	'01/31/00'	X'20000131'	year '00' becomes 2000 with CENTURY specification
DATE-P1E EXTERNAL	DATE	X'002291990C'	conversion error	day out of range input value = 2/29/1990
TIME	TIME	X'123000'	'12:30 PMbb’	
TIME-I1 EXTERNAL	TIME	X'0001E078'	X'123000'	

BMC Software, Inc., Confidential and Proprietary Information

Table 3-28 Sample Date, Time, and Timestamp Data Conversions (Part 2 of 2)

Data Type Input	Data Type Column	Data Input[a]	Data Column[a,b]	Comments
TIMESTAMP EXTERNAL(19)	TIMESTAMP	'1990-1-31-12.30.00b'	X'19900131123000000000'	
TIMESTAMP EXTERNAL(26)	TIMESTAMP	'1990-01-31-12.30.00.000000'	X'19900131123000000000'	
TIMESTAMP-3 EXTERNAL	TIMESTAMP	X'A197DC8F1C200000'	X'19900131123000000000'	input is 8-byte store clock value result = 1990-01-31-12.30.00
TIMESTAMP	CHAR(19)	X'19900131123000000000'	'1990-01-31-12.30.00'	

[a] In the Data Input and Data Column values
Character strings are depicted as '*string*'.
b represents a blank in the character strings.
Hex strings are depicted as X'*hex_string*'.
Noncharacter input and column values are depicted as hex strings.

[b] In the Data Column values
All column values are depicted in non-normalized DB2 internal format.
All columns are defined NOT NULL and no column values contain the null indicator.

General Rules for Some Types of Data

The following sections describe common information for some types of input data.

Numeric Data Types

This section describes additional information for numeric input data.

Numeric EXTERNAL

A numeric EXTERNAL field consists of a valid integer, decimal, or floating-point EBCDIC constant as defined by DB2 SQL. LOADPLUS allows leading and trailing blanks. You must specify a length (or POSITION(*start*:*end*)). The length specification indicates the number of characters in the field, not the precision. The scale defaults to 0 if the input data does not contain an explicit decimal point.

For information about converting INTEGER EXTERNAL to CHAR, see “Converting to Character Data” on page 3-127.

BMC Software, Inc., Confidential and Proprietary Information

Scale

You can specify scale on any numeric field (internal, EXTERNAL, or other format) with or without specifying a length. Specifying scale multiplies or divides the number by a power of 10. For example, a scale of 3 divides the number by 1000 while a scale of -2 multiplies the number by 100.

If you do not specify scale, 0 is the default unless the column is DECIMAL and the field is DECIMAL(PACKED) (explicit or defaulted), or DECIMAL ZONED. In these cases, the scale defaults to the scale of the column.

Rounding and Truncation

If you specify ROUND, LOADPLUS rounds the value during conversion to make it fit into the target column. If you don't specify ROUND, LOADPLUS truncates any additional digits that are not needed for precision. LOADPLUS does not consider truncation to be a conversion error.

Converting to Character Data

This section describes how LOADPLUS converts numeric input data to character data.

INTEGER EXTERNAL, INTEGER, and SMALLINT—When converting an INTEGER EXTERNAL, INTEGER, or SMALLINT field to a CHAR column, LOADPLUS places your data in the CHAR column such that the *last* character of your input data is in the following position:

- 16th position—for INTEGER EXTERNAL input to CHAR columns that are defined with a length that is greater than or equal to 16
- 11th position—for INTEGER or SMALLINT input to CHAR columns that are defined with a length that is greater than or equal to 11
- last position—for INTEGER or SMALLINT input to CHAR columns that are defined with a length that is less than 11 and for INTEGER EXTERNAL input to CHAR columns that are defined with a length that is less than 16

The following data conversion results occur:

- When the length of the value of your input data is smaller than the length of your CHAR column, LOADPLUS pads your column with leading blanks. LOADPLUS removes any leading zeros prior to padding with leading blanks.

BMC Software, Inc., Confidential and Proprietary Information

- For INTEGER EXTERNAL input, when the length of your CHAR column is greater than 16, LOADPLUS inserts blanks in positions 17 and greater.

- For INTEGER and SMALLINT input, when the length of your CHAR column is greater than 11, LOADPLUS inserts blanks in positions 12 and greater.

- When the length of your CHAR column is smaller than the length of the value of your input data, LOADPLUS discards the record.

Table 3-29 illustrates how LOADPLUS handles INTEGER EXTERNAL, INTEGER, and SMALLINT input conversion to CHAR columns.

Table 3-29 Sample INTEGER EXTERNAL, INTEGER, and SMALLINT to CHAR Data Conversions

Data Type Input	Data Type Column	Data Input[a]	Data Column[b]	Comments
INTEGER EXT(6)	CHAR(18)	'bbb100'	'bbbbbbbbbbbb100bb'	
INTEGER	CHAR(10)	X'00000064'	'bbbbbbb100'	input value = 100
INTEGER	CHAR(14)	X'00000064'	'bbbbbbbb100bbb'	input value = 100
INTEGER	CHAR(3)	X'000003E8'	error	input value = 1000 target too small record discarded
SMALLINT	CHAR(6)	X'03E8'	'bb1000'	input value = 1000
SMALLINT	CHAR(8)	X'83E8'	'bbb-1000'	input value = -1000

[a] Integer input is shown in hex format.
[b] **b** represents a blank.

DECIMAL—When converting a DECIMAL field to a CHAR column, LOADPLUS performs the following conversion tasks:

- LOADPLUS includes a place for the decimal point in the output column, regardless of whether you specify a value for scale in your input field.

- When the length of the value of your input data is smaller than the length of your CHAR column, LOADPLUS pads your column with leading blanks. LOADPLUS removes any leading zeros prior to padding with leading blanks.

- When the length of your CHAR column is smaller than the length of the value of your input data, LOADPLUS discards the record.

Table 3-30 on page 3-129 illustrates how LOADPLUS handles DECIMAL input to CHAR columns.

BMC Software, Inc., Confidential and Proprietary Information

Table 3-30 Sample DECIMAL to CHAR Data Conversions

Data Type Input	Data Type Column	Data Input[a]	Data Column[b]	Comments
DECIMAL(5,0)	CHAR(6)	X'12345'	'12345.'	
DECIMAL(5,0)	CHAR(5)	X'12345'	error	target too small record discarded
DECIMAL(5,0)	CHAR(5)	X'02345'	'2345.'	
DECIMAL(5,0)	CHAR(6)	X'02345'	'b2345.'	
DECIMAL(5,0)	CHAR(6)	X'2345D'	'-2345.'	

[a] Decimal input is shown in hex format.
[b] **b** represents a blank.

Date, Time, and Timestamp EXTERNAL Data Types

A date, time, or timestamp EXTERNAL field consists of a valid date, time, or timestamp EBCDIC string as defined by DB2 SQL. LOADPLUS allows only trailing blanks.

EXTERNAL is the default for the field data type if you do not specify the field data type for a date, time, or timestamp column. This is the only case where the default data format is not internal.

VARCHAR, VARGRAPHIC, and ROWID Data Types

For these data types, a two-byte length precedes the character, graphic, or hex data. For VARCHAR and ROWID, the length is the number of bytes. For VARGRAPHIC, the length is the number of double-byte characters. The length does not include the two length bytes.

Data Translation

LOADPLUS supports translation of data from the encoding scheme of the input data to the encoding scheme of the table by using the ASCII/EBCDIC command option or the CCSID command option (see page 3-28). This topic presents considerations related to data translation.

BMC Software, Inc., Confidential and Proprietary Information

Restrictions

LOADPLUS has the following restrictions on data translation:

- LOADPLUS does not currently support Unicode encoding schemes.
- LOADPLUS ignores any of the data translation options if you specify them when using FORMAT BMCUNLOAD.

Constants

For comparisons between predicate block character constants and input data, LOADPLUS translates constants from EBCDIC (using the EBCDIC SBCS system default CCSID) to the encoding scheme of the input data.

For constants that are placed in the table data, LOADPLUS translates the constants from EBCDIC (using the EBCDIC SBCS system default CCSID) to the table encoding scheme. LOADPLUS translates the following character constants for output:

- VALUE constants
- DEFAULTIF VALUE constants

SUBBYTE and ERRORBYTE Fields

LOADPLUS uses these fields in the same way that IBM's DB2 LOAD utility uses them.

Note: With IBM's DB2 LOAD utility, if you specify NOSUBS and discard processing is not active, the job fails when it encounters records that require substitution. LOADPLUS *does not fail* under these circumstances. However, LOADPLUS processes only the records that do not require substitution.

MIXED Data

LOADPLUS translates MIXED data types only when both the source and target of the translation are MIXED.

FOR BIT DATA

LOADPLUS does not translate columns that are defined as FOR BIT DATA.

BMC Software, Inc., Confidential and Proprietary Information

Data Type Conversion and Translation

LOADPLUS fully supports data type conversions from and to ASCII data. Table 3-31 details when translation occurs with respect to the data type conversion being performed.

Table 3-31 Translation Processing (Part 1 of 2)

	Column[a]											
Input	**SMALLINT**	**INTEGER**	**DECIMAL**	**FLOAT**	**CHARACTER**	**VARCHAR**	**GRAPHIC**	**VARGRAPHIC**	**DATE**	**TIME**	**TIMESTAMP**	**ROWID**
SMALLINT	x	x	x	x	cpt 2	cpt 2						
INTEGER	x	x	x	x	cpt 2	cpt 2			x	x		
INTEGER EXTERNAL	tc 1	tc 1	tc 1	tc 1	tcp 1,2	tcp 1,2			tc 1	tc 1		
DECIMAL	x	x	x	x	cpt 2	cpt 2			x	x		
DECIMAL EXTERNAL	tc 1	tc 1	tc 1	tc 1	tcp 1,2	tcp 1,2			tc 1	tc 1		
DECIMAL ZONED					tcp 1,2	tcp 1,2			tc 1	tc 1		
FLOAT	x	x	x	x	cpt 2	cpt 2						
FLOAT EXTERNAL	tc 1	tc 1	tc 1	tc 1	tcp 1,2	tcp 1,2						
CHAR	tc 1	tc 1	tc 1	tc 1	cpt 3	cpt 3			tc 1	tc 1	tc 1	
VARCHAR	tc 1	tc 1	tc 1	tc 1	cpt 3	cpt 3			tc 1	tc 1	tc 1	
GRAPHIC							x	x				
GRAPHIC EXTERNAL							x	x				
VARGRAPHIC							x	x				
DATE					cpt 2	cpt 2			x			
DATE EXTERNAL									tc 1			
DATE-*format* EXTERNAL					tcp 1,2	tcp 1,2			tc 1			
TIME					ctp 2	ctp 2				x		

BMC Software, Inc., Confidential and Proprietary Information

Table 3-31 Translation Processing (Part 2 of 2)

	Column[a]											
Input	**SMALLINT**	**INTEGER**	**DECIMAL**	**FLOAT**	**CHARACTER**	**VARCHAR**	**GRAPHIC**	**VARGRAPHIC**	**DATE**	**TIME**	**TIMESTAMP**	**ROWID**
TIME EXTERNAL										tc 1		
TIME-*format* EXTERNAL					tcp 1,2	tcp 1,2				tc 1		
TIMESTAMP EXTERNAL									tc 1	tc 1	tc 1	
TIMESTAMP-*format* EXTERNAL					tcp 1,2	tcp 1,2			tc 1	tc 1	tc 1	
ROWID												x
EXIT	x	x	x	x	x	x	x	x	x	x	x	

[a] The character combinations in this table indicate the order in which data type conversions, data translations, and character padding occur. The numbers represent the type of data translation that occurs when converting from one data type to another.

x—supported conversion, no translation required.
c—convert
t—translate
p—pad with blank
1—translate input SBCS to EBCDIC SBCS (system default)
2—translate EBCDIC SBCS (system default) to output SBCS
3—translate input SBCS to output SBCS

BMC Software, Inc., Confidential and Proprietary Information

Dynamic Work File Allocation Options

Note: LOADPLUS does not use these options to dynamically allocate input (SYSREC) data sets. See “INDSN” on page 3-34 for information about dynamically allocating input data sets.

Dynamic work file allocation options allow you to have LOADPLUS optimally allocate the size and number of work files that are needed to process your load. Dynamic work file allocation provides the following benefits:

- eliminates the need for you to include DD statements for the specified file types in your JCL

  Note: If you activate dynamic allocation *and* you specify DD statements in your JCL, use the IFALLOC installation option or command option to control whether LOADPLUS dynamically allocates your data sets or uses the data sets that you allocated in your JCL.

- eliminates the need for you to calculate work file allocation sizes

- automatically provides an optimal allocation for the highest possible multitasking level

- allows you to use symbolic variables and user-defined variables to assist in creating data set names

The following sections provide the syntax that is required for enabling and using dynamic work file allocation. For additional information, see “Dynamic Work File Allocation” on page 2-13.

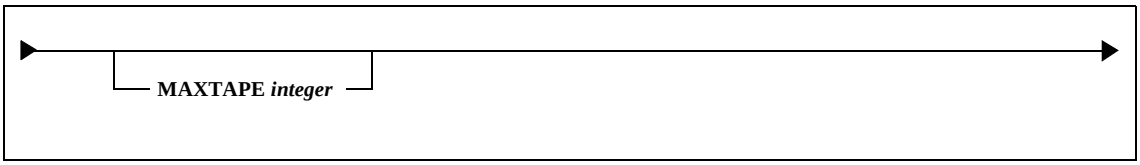
ENUMROWS Option

Dynamic work file allocation requires that you specify ENUMROWS. For more information about the ENUMROWS option, see page 3-46.

SORTOUT and SORTWK Data Sets

For tables with variable length rows, LOADPLUS allocates the load (SORTOUT) and sort (SORTWK) data sets based on the average row length. If this calculation results in a large difference from the actual average, use the SIZEPCT option (page 3-144) to adjust the file allocations.

BMC Software, Inc., Confidential and Proprietary Information



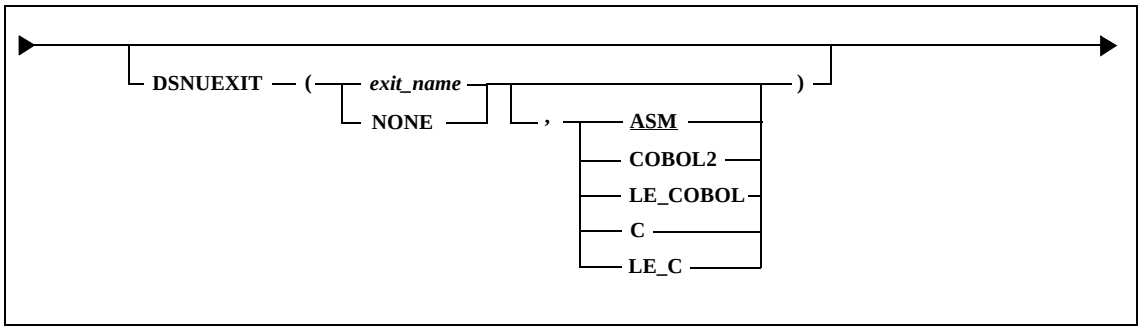
MAXTAPE

The MAXTAPE option allows you to specify the maximum number of tape devices that LOADPLUS can use when dynamically allocating work files.

You can specify the default for the MAXTAPE command option in your installation options module (using the MAXTAPE installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

For details regarding how LOADPLUS handles dynamic work file allocation when it reaches your MAXTAPE limit, see “Reaching the MAXTAPE Limit” on page 2-18.

BMC Software, Inc., Confidential and Proprietary Information



DSNUEXIT

The DSNUEXIT option allows you to specify a user-written exit routine that creates user-defined variables. You can use these variables with the DSNPAT command option to create patterns for data set names for dynamically allocated data sets. For details about the DSNPAT option, see page 3-145.

exit_name

Specify the name of the exit routine. This exit routine must be authorized and present in your system's LINKLIST or your JOBLIB or STEPLIB. For COBOL II and C programs, the appropriate runtime libraries must be authorized and in your LINKLIST, JOBLIB, or STEPLIB. For the IBM Language Environment COBOL and C programs (LE COBOL and LE C), the appropriate language environment runtime libraries must be authorized and in your LINKLIST, JOBLIB, or STEPLIB.

See Appendix F, "LOADPLUS User Exits" for more information about creating an exit routine.

You can also specify a default user exit routine name in your installation options module (using the DSNUEXIT installation option). A user exit routine specified with the DSNUEXIT command option overrides any default that is specified in your installation options. See Appendix A, "LOADPLUS Installation Options" for details.

NONE

Specify NONE in place of an exit name to override a default user exit in your installation options. This tells LOADPLUS that you do not want to call any DSNUEXIT user exit.

BMC Software, Inc., Confidential and Proprietary Information

program language

LOADPLUS supports user-written exit routines written in assembler, COBOL II, LE COBOL, C, or LE C. After you specify the name of your user exit, you can optionally provide the program language of the exit routine. If you do supply this specification, place a comma between the exit name and the program language.

You can also specify the user exit program language in your installation options module (using the DSNUEXIT installation option). A program language specified with the DSNUEXIT command option overrides any default that is specified in your installation options. See Appendix A, “LOADPLUS Installation Options” for details.

Warning! The program language, whether specified here or in your installation option, must match the source language of the specified user exit. If the language does not match, your user exit might not function correctly.

ASM

ASM indicates that the user exit is an assembler language program.

COBOL2

COBOL2 indicates that the routine is a COBOL II language program.

LE_COBOL

LE_COBOL indicates that the routine is an LE COBOL language program.

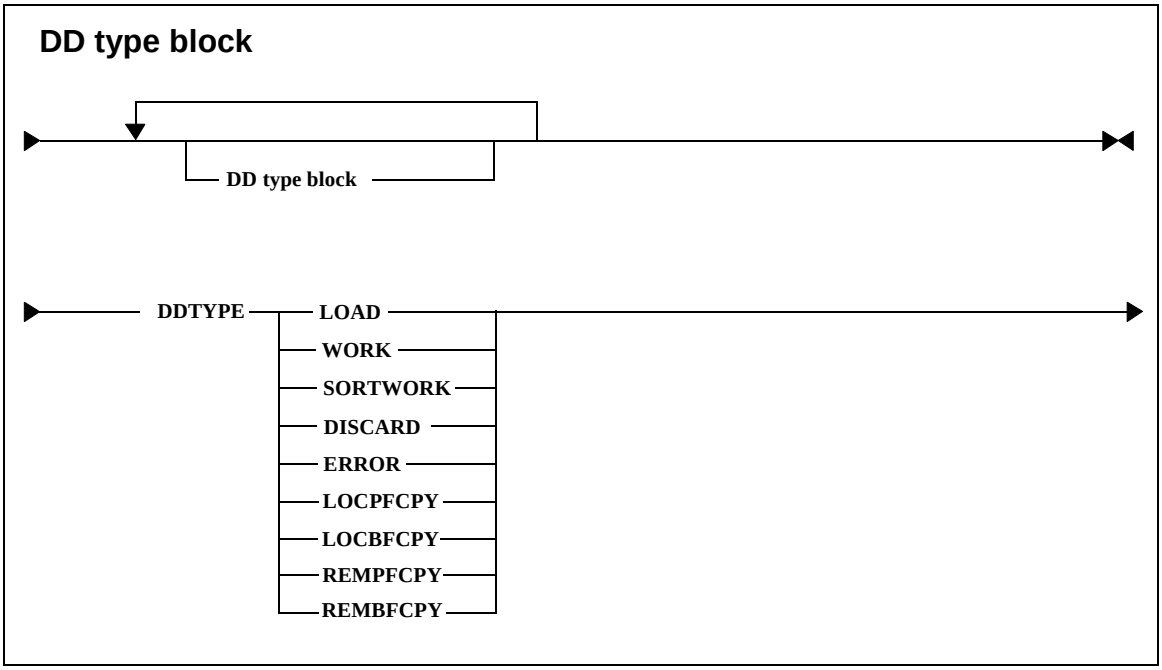
C

C indicates that the program is a C language program.

LE_C

LE_C indicates that the routine is an LE C language program.

BMC Software, Inc., Confidential and Proprietary Information



DDTYPE

This option tells LOADPLUS that you want one or more types of work files dynamically allocated or that you want to override an active dynamic work file allocation default in your installation options. Additional options in the DD type block provide control for various aspects of the dynamic work file allocation process. You can repeat this DD type block for more than one work file type.

You can also specify the work file type in your installation options module (using the DDTYPE installation option). Any dynamic work file allocation options that you specify with the DDTYPE command option override the defaults established at installation for the same DDTYPE. For details, see Appendix A, “LOADPLUS Installation Options.”

LOAD

LOAD tells LOADPLUS to apply the dynamic work file specification to the load data files (SORTOUT*n* data sets).

WORK

WORK tells LOADPLUS to apply the dynamic work file specification to the index work files (SYSUT1*nn* data sets).

BMC Software, Inc., Confidential and Proprietary Information

SORTWORK

SORTWORK tells LOADPLUS to apply the dynamic work file specification to the sort work files (SORTWK*nn* data sets).

DISCARD

DISCARD tells LOADPLUS to apply the dynamic work file specification to the discard file (SYSDISC data set).

ERROR

ERROR tells LOADPLUS to apply the dynamic work file specification to the error file (SYSERR data set).

LOCPFCPY

LOCPFCPY tells LOADPLUS to apply the dynamic work file allocation specifications to the local primary copy data set (BMCCPY).

LOCBFCPY

LOCBFCPY tells LOADPLUS to apply the dynamic work file allocation specifications to the local backup copy data set (BMCCPZ).

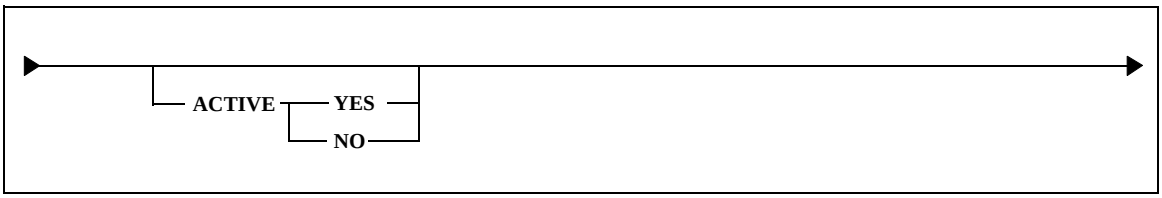
REMPFCPY

REMPFCPY tells LOADPLUS to apply the dynamic work file allocation specifications to the remote primary copy data set (BMCRCY).

REMBFCPY

REMBFCPY tells LOADPLUS to apply the dynamic work file allocation specifications to the remote backup copy data set (BMCRCZ).

BMC Software, Inc., Confidential and Proprietary Information



ACTIVE

This option allows you to specify whether you want dynamic work file allocation active for the specified work file type. If you specify just the ACTIVE keyword, LOADPLUS treats it as ACTIVE YES.

You can specify the default for the ACTIVE command option in your installation options module (using the ACTIVE installation option). The command option overrides the default established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

When running in a worklist environment, LOADPLUS ignores the ACTIVE option in your installation options module. LOADPLUS dynamically allocates your work files only if the invoking product (DASD MANAGER PLUS, CATALOG MANAGER for DB2, or CHANGE MANAGER for DB2) supplies the ACTIVE YES syntax.

YES

This option tells LOADPLUS that you want to activate dynamic work file allocation for the specified work file type.

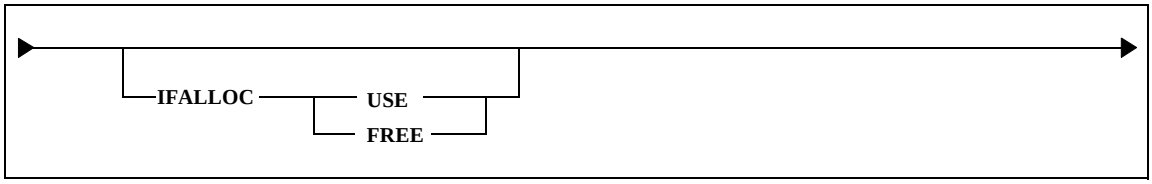
ACTIVE YES must be in effect for LOADPLUS to use the remaining dynamic allocation options.

Note: When determining whether to actually dynamically allocate a work file, LOADPLUS takes into account the values of both the ACTIVE and IFALLOC options for that work file type.

NO

This option tells LOADPLUS that you want to inactivate dynamic work file allocation for the specified work file type. Use this option if dynamic work file allocation is set to active in your installation options but you do not want LOADPLUS to dynamically allocate the specified work files for a particular processing session.

BMC Software, Inc., Confidential and Proprietary Information



IFALLOC

This option directs LOADPLUS to either free the data sets specified in your JCL or use them and, if necessary, allocate additional data sets.

You can specify the default for the IFALLOC command option in your installation options (using the IFALLOC installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

USE

Specifying this option tells LOADPLUS to use the data sets that you allocated in your JCL. If the number of SORTOUT, SYSUT1, SORTWK, SYSDISC, SYSERR, and copy files that you specify in your JCL is insufficient for processing, LOADPLUS (or BMCSORT in the case of SORTWK files) dynamically allocates the additional files that your job needs. LOADPLUS uses both the dynamically allocated work files and those that you specify in your JCL.

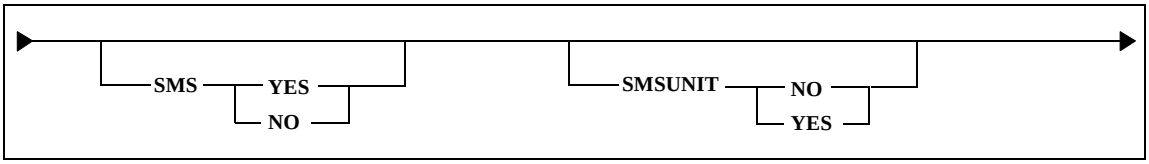
Note: LOADPLUS allocates only additional files, not additional space for a single file.

FREE

This option tells LOADPLUS to free the data sets that you allocated in your JCL and use only dynamically allocated work files.

Warning! If you specify this option for a load job in a worklist environment, the data sets are freed for the remainder of the worklist execution. This can result in an error if a subsequent LOADPLUS job in the worklist requires these data sets.

BMC Software, Inc., Confidential and Proprietary Information



SMS

This option tells LOADPLUS whether to use an SMS allocation.

You can specify the default for the SMS command option in your installation options module (using the SMS installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

YES

This option tells LOADPLUS to perform an SMS allocation.

NO

This option tells LOADPLUS to perform a non-SMS allocation.

SMSUNIT

This option tells LOADPLUS whether to pass the UNIT value in the SMS allocation parameter list during dynamic allocation. LOADPLUS does not modify any other parameters based on this option. If you do not specify SMS YES, LOADPLUS ignores the SMSUNIT option.

You can specify the default for the SMSUNIT command option in your installation options module (using the SMSUNIT installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

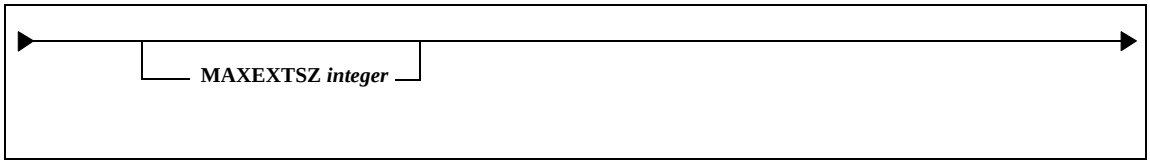
NO

This option tells LOADPLUS not to pass the value for the UNIT option.

YES

This option tells LOADPLUS to pass the value for the UNIT option.

BMC Software, Inc., Confidential and Proprietary Information



MAXEXTSZ

This option allows you to specify the maximum size (in kilobytes) for any extent that LOADPLUS allocates as the primary and secondary quantities for a dynamically allocated file.

LOADPLUS ignores MAXEXTSZ if you are using SMS to allocate the files. It also ignores this option for SORTWK files.

Note: You can specify the default for the MAXEXTSZ command option in your installation options module (using the MAXEXTSZ installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

If you do not want to set a limit for space allocation, specify 0 as the value for MAXEXTSZ. If you specify 0, or if you specify a value greater than the size limit of your physical device, LOADPLUS replaces this value at runtime with the size of your physical device.

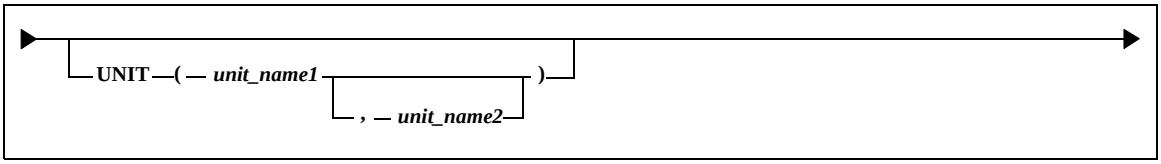
Regardless of the amount of space that LOADPLUS determines that it needs, LOADPLUS will not allocate more than your specified MAXEXTSZ limit for either the primary or the secondary quantity:

- If the amount of required space that LOADPLUS calculates is greater than the MAXEXTSZ limit for the primary quantity, LOADPLUS uses the secondary extents to hold the remainder of the required primary space.

- If the amount of required space that LOADPLUS calculates is greater than the MAXEXTSZ limit for both the primary and secondary quantities, LOADPLUS issues message BMC50451E and the job terminates with a return code 8.

Note: LOADPLUS checks the value of MAXEXTSZ after applying SIZEPCT to the allocation amount.

BMC Software, Inc., Confidential and Proprietary Information



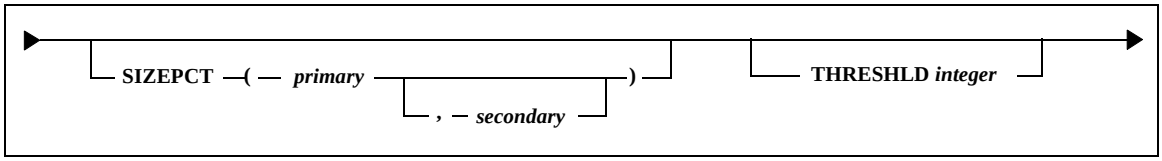
UNIT

This option allows you to specify the primary and secondary unit names that you want LOADPLUS to use during non-SMS dynamic work file allocation. The unit names cannot exceed eight characters. The two unit names are enclosed in parentheses and separated by a comma.

The THRESHLD installation option or command option determines how LOADPLUS uses the unit names that you specify in this option. See the THRESHLD command option on page 3-144 for details.

You can specify default unit names in your installation options module (using the UNIT installation option). Primary and/or secondary unit names specified with the UNIT command option override any default established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

BMC Software, Inc., Confidential and Proprietary Information



SIZEPCT

This option allows you to adjust, by percentages, the allocated work file sizes that LOADPLUS calculated. The numbers that you specify must be greater than zero.

You can specify the default for the SIZEPCT command option in your installation options module (using the SIZEPCT installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

primary

This value indicates the percentage of the primary work file size calculated by LOADPLUS that you actually want allocated.

secondary

This value indicates the percentage of the secondary work file size calculated by LOADPLUS that you actually want allocated.

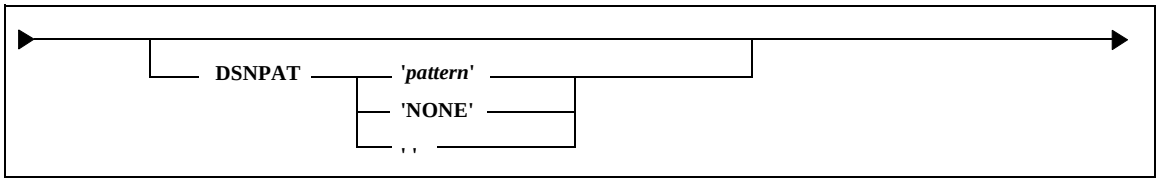
THRESHLD

This option allows you to specify a threshold value, in kilobytes, above which LOADPLUS uses the secondary unit or secondary SMS class for dynamic allocation. LOADPLUS tests this threshold for each data set that it is allocating. If the size for a particular data set is greater than the threshold, LOADPLUS uses the secondary unit or SMS class for that allocation.

A value of 0 tells LOADPLUS not to use the secondary unit name or SMS class.

You can specify the default for the THRESHLD command option in your installation options module (using the THRESHLD installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

BMC Software, Inc., Confidential and Proprietary Information



DSNPAT

This option allows you to direct LOADPLUS to use a particular pattern of variables and text to create data set names for dynamic work file allocation.

You can specify the default for the DSNPAT command option in your installation options module (using the DSNPAT installation option). The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

'pattern'

This option provides the pattern of variables and text that you want LOADPLUS to use to create data set names. The pattern must be specified with surrounding single quotes.

Warning! The pattern specified in your DSNPAT option must allow LOADPLUS to generate unique data set names. With multiple SORTOUT and SYSUT1 files, you must include the &DDNAME variable to generate unique names. For copy data sets, you might need to include additional variables, such as &VCAT, &DATEJ, or &TIME4 to generate unique names across multiple loads. If LOADPLUS encounters non-unique data set names, processing terminates.

Variables—You can use any of the symbolic variables that are described in Table 3-32 on page 3-146 to construct your pattern. In addition, you can use text or provide user-defined variables from a user exit routine. The LOAD command option, DSNUEXIT, is described on page 3-135 and user exits are further discussed in Appendix F, “LOADPLUS User Exits.”

BMC Software, Inc., Confidential and Proprietary Information

Symbols for numeric variables (such as &DATE or &TIME) must be prefixed by a national character (alphabetic, #, @, or $). In the following example, the first statement causes an error, while the second is correct.

```
DSNPAT '&DB.&TS.&DATE'
DSNPAT '&DB.&TS..RP&DATE'
```

Although you can *prefix* a symbolic variable with an alphabetic character, you cannot *append* characters. For example, XX&DB is valid, but &DBXX is invalid. &DB.XX is valid.

User-defined variables must begin with an underscore character, as in _DEPT. For more information, see “User-Defined Variables” on page F-4.

Table 3-32 Symbolic Variables for DSNPAT Command Option (Part 1 of 2)

Symbolic Variable	Definition	Length of Result[a, b]
&DATE	current date (in the form MMDDYY)	6 bytes
&DATEJ	current Julian date (in the form YYYYDDD)	7 bytes
&DB	database containing the space being used for this data set allocation	8 bytes maximum
&DDNAME	ddname being used for this data set allocation	8 bytes maximum
&GRPNM	DB2 data-sharing group name[c]	4 bytes
&JOBNAME	JOB name used in the JCL	8 bytes maximum
&PART	partition being used for this data set allocation	3 bytes
&REPLACE	LOAD REPLACE being done (Y or N)	1 byte
&RESUME	LOAD RESUME being done (Y or N)	1 byte
&SSID	DB2 subsystem ID	4 bytes
&STEPNAME	STEP name used in the JCL[d]	8 bytes maximum
&TIME	current time (in the form HHMMSS)	6 bytes
&TIME4	current time (in the form HHMM)	4 bytes
&TS	table space containing the table specified in your LOAD command	8 bytes maximum
&USERID or &UID [e]	job user ID	8 bytes maximum
&UTIL	BMC Software utility ID	8 bytes maximum[f]
&UTILPFX	BMC Software utility ID prefix	8 bytes maximum

Table 3-32 Symbolic Variables for DSNPAT Command Option (Part 2 of 2)

Symbolic Variable	Definition	Length of Result[a, b]
&UTILSFX	BMC Software utility ID suffix	8 bytes maximum
&VCAT	VCATNAME specified in the DB2 catalog for the table space that you are loading; or, if the table space is partitioned, the VCAT name from the first part that you are loading	8 bytes

[a] LOADPLUS removes any trailing blanks in the result.
[b] The maximum total length that LOADPLUS allows for a data set name is 44 bytes.
[c] In a nondata-sharing environment, &GRPNM contains the DB2 SSID.
[d] LOADPLUS ignores PROC names.
[e] You cannot use this variable unless you have a security package.
[f] LOADPLUS truncates longer utility IDs to 8 characters.

Utility ID Variable—If the utility ID has no special character delimiters, &UTILPFX contains the first eight bytes of text and &UTILSFX contains the remaining eight bytes of text.

If the utility ID has a special character delimiter within the first eight bytes of text, &UTILPFX contains the bytes up to but not including the delimiter. &UTILSFX contains the eight bytes following the first delimiter. The first delimiter is not included in either variable. Any delimiter after the first is treated as normal text and might be included in &UTILSFX. Depending on the utility ID that is specified for this load job, &UTILSFX might be blank.

For example, if the utility ID is USER1/WORK1, the variables contain the following values:

```
&UTIL = USER1/WO
&UTILPFX = USER1
&UTILSFX = WORK1
```

If the utility ID is USER1//WORK1, the variables contain the following values:

```
&UTIL = USER1//W
&UTILPFX = USER1
&UTILSFX = /WORK1
```

The following special delimiting characters tell LOADPLUS to split the utility ID:

```
(SP)(X'40') . + | ; – / ¦ (X'6A') , _: =
```

Name Construction—You can specify any or all nodes of a data set name by using variables or text. The following example generates data set names that contain the ID of the user, the table that is involved in the load, and the name of the load job:

```
DSNPAT '&UID.&TS.&JOBNAME'
```

The following example combines actual text with symbolic variables to generate a data set name.

```
DSNPAT '&DB.&DDNAME..NEW'
```

In certain cases, a period is required in your pattern as a node delimiter or to indicate the end of a variable name. Table 3-33 illustrates concatenation in a data set name pattern and those instances in which a period is needed.

Note: Concatenation is not affected by a period contained in the value of a user variable.

Table 3-33 Variable Concatenation Examples

Task	Code	Result (where &UID=RDAB and _DEPT=DEV)
Concatenate the values of 2 variables—no node delimiter.	&UID_DEPT	RDABDEV
Make 2 nodes from the values of 2 variables.	&UID._DEPT	RDAB.DEV
Concatenate the value of a variable with text—no node delimiter.	&UID.NEW	RDABNEW
Concatenate text with the value of a variable—no node delimiter.	NEW&UID	NEWRDAB
Make 2 nodes from the value of a variable followed by text	&UID..NEW	RDAB.NEW
Make 2 nodes from text followed by the value of a variable	NEW.&UID	NEW.RDAB

LOADPLUS ignores trailing blanks and null value variables. Node-delimiting periods in the pattern, however, are included regardless of the variable's value. This can result in an invalid data set name.

BMC Software, Inc., Confidential and Proprietary Information

For example, given that &UID=RDAB and the value of _DEPT is null, the following pattern results in an invalid data set name of RDAB..NEW:

```
&UID._DEPT..NEW
```

GDG Names—For your dynamically allocated copy data sets, you can also specify a pattern that contains a GDG name. Each DDTYPE must have a different GDG base.

The GDG format that you use to construct data set names is the same as the format that you use in your JCL when you allocate copy data sets with DD statements. Simply append the generation number in parentheses. The open parenthesis tells LOADPLUS that the pattern is a GDG name. The generation number must be an integer from 1 through 255.

If the base does not exist, LOADPLUS creates it for you, using everything in the pattern up to the open parenthesis as the base name. For more information about GDG names and options that affect GDGs, see “Generating Data Set Names” on page 2-16.

The following example shows a valid GDG name:

```
'&UTILPFX.&DDNAME..COPY(+1)'
```

If you are using a substitution variable as the last variable before the open parenthesis, you must include a period before the open parenthesis. The following example shows this usage:

```
'&UTILPFX.&DDNAME.(+1)'
```

If you specify COPYLVL PART on the LOAD command, each partition must have a different GDG base. To specify a pattern that includes a partition, the partition must not be in parentheses. The following example shows a valid name:

```
'&UTILPFX.&DDNAME..P&PART.(+1)'
```

You cannot specify a pattern that contains a PDS name. The following example shows an invalid name:

```
'&UTILPFX.&DDNAME..(P&PART)'
```

BMC Software, Inc., Confidential and Proprietary Information

'NONE' or ''

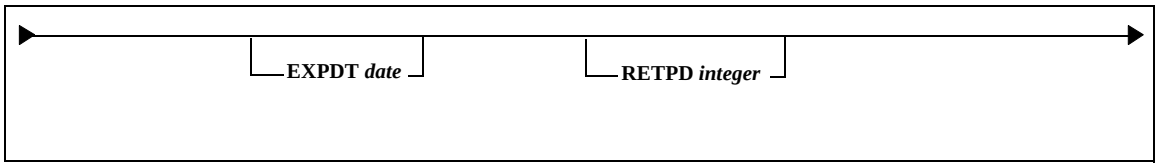
Indicates that you do not want to use any pattern to create data set names during dynamic work file allocation. This option is valid only for use with SORTWK data sets.

Warning! If you specify NONE for SORTOUT, SYSUT1, SYSDISC, SYSERR, or copy data set patterns, LOADPLUS does not allocate your data sets and processing terminates.

You must enclose the keyword NONE with single quotes.

Note: In the DSNPAT *installation* option, the keyword NONE must not be surrounded by single quotes.

BMC Software, Inc., Confidential and Proprietary Information



EXPDT

Specify EXPDT to set an expiration date for the copy data set that you are dynamically allocating. The value of *date* must be in the format *yyddd*, *yyyyddd*, or *yyyy/ddd*. *yy* is the last two digits of the year, *yyyy* is the 4-digit year (1900 through 2155), and *ddd* is the 3-digit Julian day (000 through 366).

For expiration dates of January 1, 2000 and later, you must use the form *yyyyddd* or *yyyy/ddd.*

Note: You can specify the default for the EXPDT command option in your installation options module using the EXPDT installation option. The EXPDT command option overrides the RETPD command option and the EXPDT and RETPD installation options. For details, see Appendix A, “LOADPLUS Installation Options.”

LOADPLUS ignores this option if you specify it for any of the *non-copy* data sets that you are dynamically allocating.

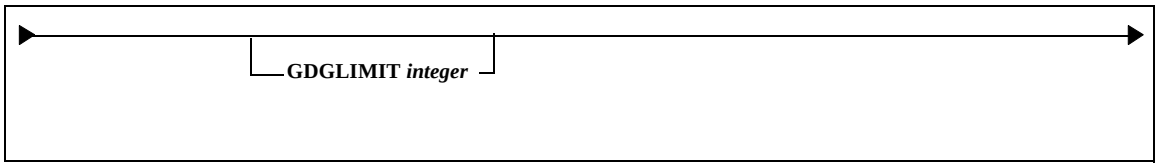
RETPD

Specify RETPD to set a retention period in days for the copy data set that you are dynamically allocating. The value of *integer* must be in the range 0 through 9999.

Note: You can specify the default for the RETPD command option in your installation options module using the RETPD installation option. The RETPD command option overrides the default established at installation for both the RETPD and EXPDT installation options. However, if you specify the EXPDT command option, it takes precedence over the RETPD command option. For information about the installation options, see Appendix A, “LOADPLUS Installation Options.”

LOADPLUS ignores this option if you specify it for any *non-copy* data set that you are dynamically allocating.

BMC Software, Inc., Confidential and Proprietary Information



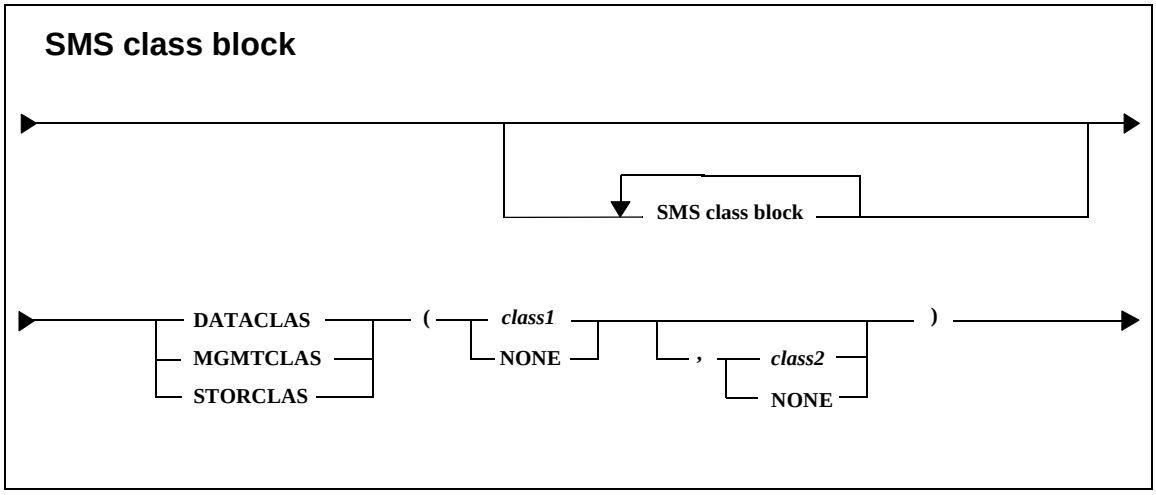
GDGLIMIT

Specify GDGLIMIT to indicate the number of generations to keep for the copy data set that you are dynamically allocating as a GDG data set. The value of *integer* must be in the range 1 through 255.

LOADPLUS honors this option only when creating the GDG base. LOADPLUS ignores this option if you specify it for any *non-copy* data set that you are dynamically allocating.

You can specify the default for the GDGLIMIT command option in your installation options module using the GDGLIMIT installation option. The command option overrides the default that was established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

BMC Software, Inc., Confidential and Proprietary Information



SMS class block

These options allow you to tell LOADPLUS what class to use for SMS allocations. Note that you can repeat this block to specify more than one type of SMS class.

You must enclose the class names for each of the class types in parentheses and you must separate them with a comma. You can use the keyword NONE in place of the class name to indicate that you do not want to specify a class for dynamic work file allocation.

The THRESHLD installation option or command option determines how LOADPLUS selects the classes that you specify here. See the THRESHLD command option on page 3-144 for details.

You can specify the defaults for the SMS class command options in your installation options module (using the DATACLAS, MGMTCLAS, and STORCLAS installation options). The command options override the defaults that were established at installation. For details, see Appendix A, “LOADPLUS Installation Options.”

DATACLAS

DATACLAS indicates the primary and secondary SMS data classes that you want LOADPLUS to use to allocate the specified work file type. *class1* and *class2* must be valid SMS data class names not exceeding eight characters each.

BMC Software, Inc., Confidential and Proprietary Information

MGMTCLAS

This option indicates the primary and secondary SMS management classes that you want LOADPLUS to use to allocate the specified work file type. *class1* and *class2* must be valid SMS management class names not exceeding eight characters each.

STORCLAS

The STORCLAS option indicates the primary and secondary SMS storage classes that you want LOADPLUS to use to allocate the specified work file type. *class1* and *class2* must be valid SMS storage class names not exceeding eight characters each.

BMC Software, Inc., Confidential and Proprietary Information

Chapter 4 How to Build and Execute LOADPLUS Jobs

This chapter presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Building the LOADPLUS Job

Building a job for the LOADPLUS product involves the following steps:

- creating JCL that includes a JOB statement
- specifying the EXEC statement with the appropriate utility parameters
- specifying any STEPLIB or JOBLIB DD statements
- specifying the DD statements as needed for the appropriate number and size of data sets for the data structure that you are loading
- specifying the LOADPLUS control statements using the appropriate command syntax

JOB Statement

Include a LOADPLUS JOB statement that conforms to your site's standards. You can include the REGION parameter on either your JOB statement or your EXEC statement. See "REGION Parameter" on page 4-3 for recommendations.

EXEC Statement

The LOADPLUS EXEC statement specifies the module to be executed for the LOADPLUS utility. The LOADPLUS module name is AMUUMAIN. The EXEC statement also specifies LOADPLUS utility parameters, which are described in "Utility Parameters on the EXEC Statement" on page 4-3.

You can include the REGION parameter on either your EXEC statement or your JOB statement. See "REGION Parameter" on page 4-3 for recommendations.

BMC Software, Inc., Confidential and Proprietary Information

REGION Parameter

Include the REGION parameter on either your JOB statement or your EXEC statement to specify the region size (the amount of virtual storage used by the utility). For the best performance, BMC Software recommends that you specify REGION=0M to allocate all available virtual storage to the LOADPLUS job. If your data center does not permit you to specify REGION=0M, specify the amount that allows the most virtual storage both above and below the 16-megabyte line.

Utility Parameters on the EXEC Statement

The LOADPLUS EXEC statement includes the following utility parameters:

- DB2 subsystem ID or group attachment name
- utility ID
- restart parameter
- TSO user ID for notification of progress made on the load
- message level
- installation options module parameter

The following illustration shows the format of the EXEC statement with all the required and optional utility parameters:

```
//stepname EXEC PGM=AMUUMAIN,
//  PARM='ssid,utilid,restart_parm,userid,MSGLEVEL(n),options_module'
```

Note: The LOADPLUS utility parameters are positional. If you do not specify a value for a parameter (thus allowing the default value), you must substitute a comma for that parameter if additional parameters follow. The comma indicates that a parameter was omitted.

DB2 Subsystem Identifier (SSID)

This parameter specifies the 4-character DB2 subsystem ID that indicates where the table space resides. If you do not specify the SSID, LOADPLUS uses the DB2 installation default found in the DSNHDECP module.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS supports the DB2 group attachment name capability. When you supply a group attachment name as the SSID, LOADPLUS uses the name to connect all plans. LOADPLUS then determines the actual DB2 SSID from within that group to use for the current load job.

Note: When restarting in a data sharing environment, LOADPLUS can use either the same member that was chosen in the original load or any other member in the specified group.

Utility Identifier (UTILID)

This parameter specifies the 16-character utility ID that gives a unique name to a utility job. If you omit this parameter, LOADPLUS uses the default, *userid.jobname*. Each BMC Software utility job should have a unique ID.

Restart Parameter

The restart parameter can have one of the following values. For additional information about specific circumstances for which you should use a particular parameter, see "Restarting LOADPLUS" on page 4-26.

Warning! This warning applies to the parameters RESTART, RESTART(PHASE), NEW/RESTART, and NEW/RESTART(PHASE). If you are restarting your job and you specified DISP=NEW for the SORTOUT, SYSUT1, and SYSERR data sets, you must change your JCL to DISP=OLD before restarting. Alternatively, you can allocate the data sets with DISP=MOD, or allocate the data sets in a prior step and then specify DISP=OLD for the data sets in the load step. For more information, see "Check for Data Set Attributes" on page 2-22.

Blank or Not Specified

By not specifying a restart parameter, the default, LOADPLUS initiates a new BMC Software utility. The utility ID that you specify cannot currently exist in the BMCUTIL table.

RESTART

Specifying this value restarts a BMC Software utility from the last restart sync point. LOADPLUS takes restart sync points as each phase completes and as the processing of each DB2 object completes. The utility ID must exist in the BMCUTIL table. If the utility ID does not exist in the BMCUTIL table, LOADPLUS ends with a return code 8.

BMC Software, Inc., Confidential and Proprietary Information

RESTART(PHASE)

For a two-phase load, specifying this value restarts LOADPLUS at the beginning of the last incomplete phase. The utility ID must exist in the BMCUTIL table. If the utility ID does not exist in the BMCUTIL table, LOADPLUS ends with a return code 8.

In general, for a single-phase load or for a SQLAPPLY load, LOADPLUS treats RESTART(PHASE) as if you specified RESTART.

Note: For additional criteria that affect whether to use the PHASE option, see "Restarting LOADPLUS" on page 4-26.

NEW

Specifying this value initiates a new BMC Software utility or replaces an existing utility ID. Specifying this value allows you to start a utility without having to end the utility ID separately.

Warning! Use care when you specify NEW. When you replace or terminate a utility ID for LOADPLUS, data from the PRELOAD or COMBINED phase is lost and existing data in the table space can also be lost. Similar consequences can occur in other BMC Software utilities. Consult the reference manual for the specific BMC Software utility. See "Terminating or Canceling the Job" on page 4-33 for more information for LOADPLUS.

If you specify NEW and the utility ID has a status of *X* (executing), LOADPLUS issues error message BMC50012E and ends with a return code 8. Refer to this message in Appendix C, "LOADPLUS Messages and Codes" for additional information.

NEW/RESTART

Specifying this value restarts a BMC Software utility from the last restart sync point if the utility ID exists. LOADPLUS takes restart sync points as each phase completes and as the processing of each DB2 object completes. Otherwise, specifying this value starts a utility as NEW.

NEW/RESTART(PHASE)

For a two-phase load, specifying this value restarts LOADPLUS at the beginning of the last incomplete phase if the utility ID exists. Otherwise, for a two-phase load, specifying this value starts LOADPLUS as NEW.

BMC Software, Inc., Confidential and Proprietary Information

In general, for a single-phase load or for a SQLAPPLY load, LOADPLUS treats NEW/RESTART(PHASE) as if you specified NEW/RESTART.

Note: For additional criteria that affect whether to use the PHASE option, see “Restarting LOADPLUS” on page 4-26.

TERM

Specifying this value terminates the existing utility and removes the utility ID from the BMCUTIL table. After removing all sync point and restart information, LOADPLUS terminates without performing a load. If the utility ID does not exist, LOADPLUS terminates with a return code 0.

The minimum JCL that is required when specifying TERM is the STEPLIB to the LOADPLUS load library and the SYSPRINT DD statement.

Warning! Use care when you specify TERM. When you replace or terminate a utility ID, data from the PRELOAD phase or the COMBINED phase is lost and existing data in the table space can also be lost. Similar consequences can occur in other BMC Software utilities. Consult the reference manual for the specific BMC Software utility. See “Terminating or Canceling the Job” on page 4-33 for more information for LOADPLUS.

MAINT

Specifying this value causes LOADPLUS to print additional information in your SYSPRINT. When you specify the MAINT parameter, the job ends without affecting any utility that is running.

LOADPLUS prints the following additional information in the SYSPRINT:

- a summary report of all the product fixes that you have applied
- an options module report that lists the values in the installation options module that you are using for this load job

BMC Software, Inc., Confidential and Proprietary Information

- if the value of the LOADDECP installation option is YES, the values in the DSNHDECP module that LOADPLUS uses

  The following example illustrates the DSNHDECP values that LOADPLUS prints in this case:

```
BMC50471I DB2 DSNHDECP MODULE SETTINGS:
BMC50471I VERSION                 = 710
BMC50471I SUBSYSTEM DEFAULT       = DEBA
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            = (37,65534,65534)
BMC50471I ASCII CCSID             = (437,65534,65534)
BMC50471I UNICODE CCSID           = (367,1208,1200)
```

User Identifier (USERID)

This is the TSO user ID that LOADPLUS notifies after the completion of each phase and at the end of utility command execution.

Message Level (MSGLEVEL)

This parameter controls which messages LOADPLUS returns to the user in the SYSPRINT data set. MSGLEVEL(0) returns minimal messages. MSGLEVEL(1) returns additional messages to help you diagnose problems and fine-tune performance.

You can use the MSGLEVEL installation option to change the default value of this parameter. For details, see Appendix A, “LOADPLUS Installation Options.”

Options Module

This parameter allows you to identify which installation options module to use. If you include this parameter, you must specify the full name of the options module. If you omit this parameter, LOADPLUS uses the default options module, AMU$OPTS.

See Appendix A, “LOADPLUS Installation Options,” for more information about installation options. In addition, the *Utility Products for DB2 Customization Guide* explains how to create multiple installation options modules.

BMC Software, Inc., Confidential and Proprietary Information

STEPLIB DD Statement

The LOADPLUS STEPLIB DD statement specifies the following libraries, unless they are included in your system's LINKLIST or in a JOBLIB statement:

- load library or libraries that contain the files for the following products and components:
  - — LOADPLUS
  - — BMCSORT
  - — Apply Plus, if you are performing a SQLAPPLY load
- libraries that contain any DB2 user exits (EDITPROCs, VALIDPROCs, FIELDPROCs, and user-written conversion routines)
- DB2 load library

All load libraries in the STEPLIB or JOBLIB concatenation must be APF authorized.

LOADPLUS DD Statements

LOADPLUS uses data sets that are specified by ddnames. Use Table 4-1 to quickly find the data set for which you want more information.

Table 4-1 ddname Descriptions

Data Set Name	Found on Page
Copy	4-9
Other	4-24
SORTOUT	4-12
SORTWK	4-14
SYSDISC	4-15
SYSERR	4-17
SYSIDCIN	4-22
SYSIN	4-17
SYSPRINT	4-18
SYSREC	4-18
SYSUT1	4-20
UTPRINT	4-24

BMC Software, Inc., Confidential and Proprietary Information

As an alternative to specifying data sets in DD statements in your JCL, you can have LOADPLUS dynamically allocate the following types of data sets:

- SYSREC
- SORTOUT
- SYSUT1
- SORTWK
- SYSDISC
- SYSERR
- copy data sets

For details, about dynamically allocating SYSREC data sets, see “INDSN” on page 3-34. For details about dynamically allocating the other data sets, see “Dynamic Work File Allocation” on page 2-13 and “Dynamic Work File Allocation Options” on page 3-133.

Copy Data Sets

BMCCPY is required if you specify COPY YES. If you specify BMCRCZ, BMCRCY is also required. The copy ddnames identify the output data sets that will contain an image copy or DSN1COPY either of the table space or of each partition in the table space after the load. The existence of the copy ddnames in your JCL determines the number of copies that LOADPLUS makes when you specify COPY YES. See Table 4-3 on page 4-10 for information about the data sets that you need when you specify COPY YES.

Note: When you are loading all partitions of a partitioned table space, you can also specify how you want LOADPLUS to assign the dynamically allocated copy data sets. For more information, see “COPYLVL” on page 3-84.

Overriding the Default ddnames

To override the default copy ddnames that were specified at installation, use the COPYDDN (page 3-85) and RECOVERYDDN (page 3-86) command options.

Registration

The ddname itself controls the registration information that is placed in the ICBACKUP column of SYSIBM.SYSCOPY when you specify either ALL (the default) or specific ddnames for the REGISTER command option. Table 4-2 on page 4-10 describes the LOADPLUS copy data sets that, when registered, correspond to the DB2 local and remote copies. The table also states when they are required.

BMC Software, Inc., Confidential and Proprietary Information

Table 4-2 Corresponding Image Copy Types

Image Copy Type	Corresponding LOADPLUS Copy Data Set (When Registered)	When the Data Set is Required
local primary	BMCCPY or *ddname1* specified in the COPYDDN command or installation option	always required if you specify COPY YES, unless you are dynamically allocating your copy data sets
local backup	BMCCPZ or *ddname2* specified in the COPYDDN command or installation option	optional
remote primary	BMCRCY or *ddname1* specified in the RECOVERYDDN command option or the RCVYDDN installation option	required only if a BMCRCZ copy is made
remote backup	BMCRCZ or *ddname2* specified in the RECOVERYDDN command option or the RCVYDDN installation option	optional

DD Statements for Partitioned and Nonpartitioned Table Spaces

You should not specify separate data sets for nonpartitioned, multiple data set table spaces.

Table 4-3 lists the copy DD statements that LOADPLUS requires based on the table space and type of load.

Table 4-3 Copy DD Statements Required With COPY YES[a]

Table Space Being Copied	Type of Load	Copy DD Statements Required When You Specify COPY YES
nonpartitioned	any kind	Specify one DD statement for each copy type that you want LOADPLUS to make. Do not use *nn* in the ddname.
partitioned	all partitions (entire table space)	Specify one DD statement for each copy type that you want LOADPLUS to make. Do not use *nn* in the ddname. OR For each partition, specify one DD statement for each copy type that you want LOADPLUS to make. Use *nn* for all ddnames, where *nn* matches the partition number.
	selected partitions (using INTO TABLE PART option)	For each specified partition, specify one DD statement for each copy type that you want LOADPLUS to make. Use *nn* for all ddnames, where *nn* matches the partition number. (The *nn* is not required if you are performing a partial load with only one partition.)

[a] As an alternative to specifying these DD statements in your JCL, you can have LOADPLUS dynamically allocate the copy data sets. Based on the information in this table, specify dynamic allocation options that result in the same number and type of copy data sets.

BMC Software, Inc., Confidential and Proprietary Information

Naming Conventions for Partition Level Copies

Use the *nn* as part of the ddname only if you are loading a partitioned table space and you want a separate data set for each partition. You must use *nn* when making copies for a load of selected partitions (when you specify INTO TABLE PART).

JCL rules limit ddnames to eight characters. If you are making partition-level copies the ddname has two parts—the ddname prefix (minimum of one character) and the *nn* (one to seven characters). The ddname that is specified in the installation options or the command options becomes the prefix. The ddname that you specify in the JCL must have the partition number *nn* appended to this prefix. The *nn* must match the partition that you are copying and you must allow sufficient bytes for the partition number to be added to the prefix and still have a valid ddname.

Allocating the Copy Data Sets

If you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS provides an estimate of the space that is needed for the copy data sets. See “ANALYZE Option for Estimating Data Set Allocation” on page 4-25 for more information. Alternatively, you can use the DDTYPE command option to have LOADPLUS dynamically allocate your copy data sets.

LOADPLUS determines the optimal block size of the copy data sets based on the device type that contains the data set. Copy data sets can be on different device types. However, if additional copy data sets for the same object (for example, remote backup copy data sets) are on different device types, the block size for all copy data sets for that object is the block size that LOADPLUS determines is optimal for the local primary copy.

The size that LOADPLUS needs for the copy data sets depends on the number of pages that are required when the table space is loaded. LOADPLUS calculates the number of required pages and issues message BMC51486I, which includes the page number. If you specify PRELOAD PAUSE, you can use the information from the ANALYZE phase to allocate the primary amount as the total amount. No secondary amount is needed.

LOADPLUS does not allow dynamically allocated copy data sets to be stacked on tape.

BMC Software, Inc., Confidential and Proprietary Information

SORTOUT*n* Data Sets

For a two-phase load, the SORTOUT data set contains DB2 row images for loading in the LOAD phase. LOADPLUS uses the data set during the PRELOAD phase through the end of the LOAD phase. For a single-phase LOAD RESUME YES SHRLEVEL NONE without PART REPLACE but with ORDER YES specified, LOADPLUS writes the clustering index and new DB2 row images to this data set during the COMBINED phase, but only reads from this data set if you restart your job. Refer to Table 4-4 for circumstances under which LOADPLUS requires this data set.

Table 4-4 SORTOUT*n* Usage

Load Type	SORTOUT*n* Requirement
two-phase load	at least one required
single-phase LOAD RESUME YES SHRLEVEL NONE without PART REPLACE and with ORDER YES	at least one required if you want to be able to restart **Note**: If you do not specify a SORTOUT data set and the load job fails, you might have to recover the object. See "Recovering the DB2 Object after Terminating or Canceling the Job" on page 4-34 for more information.
single-phase LOAD RESUME YES SHRLEVEL NONE with ORDER NO	not used
single-phase LOAD RESUME YES SHRLEVEL NONE with PART REPLACE	not used
single-phase LOAD REPLACE or LOAD RESUME NO	not used
LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY	not used

You can use either form of the data set name (SORTOUT or SORTOUT*n*). When specifying multiple data sets, you can use one *or* both forms. The variable *n* can be any valid single-digit national character or a blank. To override the default ddname, use the LOADDN command option as described on page 3-37. If you plan to add more than one digit to the prefix or have LOADPLUS dynamically allocate more than nine SORTOUT data sets, you must override the default name to create a ddname prefix of six or fewer characters. For performance information about this data set, see "SORTOUT Data Set" on page 6-9.

BMC Software, Inc., Confidential and Proprietary Information

Allocating the SORTOUT Data Set

If you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS provides an estimate of the space needed for the SORTOUT data set. See “ANALYZE Option for Estimating Data Set Allocation” on page 4-25 for more information. If you do not specify ANALYZE PAUSE or ANALYZE ONLY, you must estimate the allocation values yourself by using the formulas that are provided in “SORTOUT Data Set” on page D-3. Alternatively, you can use the DDTYPE command option to have LOADPLUS dynamically allocate your SORTOUT data set.

Warning! To restart LOADPLUS any time during the COMBINED phase or after the PRELOAD phase, SORTOUT cannot be a temporary data set. For information about the LOADPLUS treatment of temporary data sets, see “LOADPLUS Work File Validity and Integrity Checks” on page 2-22.

For formulas and information about calculating the size of the SORTOUT data set, see “SORTOUT Data Set” on page D-3.

Specifying Multiple SORTOUT*n* Data Sets

LOADPLUS allows you to specify up to 16 SORTOUT data sets. To determine individual SORTOUT data set sizes when using multiple data sets, divide the total (as calculated in “SORTOUT Data Set” on page D-3) by the number of data sets.

LOADPLUS determines the optimal number of concurrent SORTOUT data sets to use based on such factors as the number of data sets that you specify in your DD statements, the number of sort processes that will fit in memory, and the number of partitions that are involved in the job. See “Using Multiple SORTOUT Data Sets” on page 6-7 for details about how LOADPLUS optimizes the number of SORTOUT data sets and other performance implications of multitasking in LOADPLUS.

If you specify PRELOAD ANALYZE, LOADPLUS stops after PRELOAD optimization and displays a message that indicates the optimal number of SORTOUT and SYSREC data sets that LOADPLUS can process concurrently. You can use this information to adjust the number of data sets in your JCL. There is no advantage to specifying more SORTOUT data sets than the LOADPLUS optimizer selects.

Note: For nonpartitioned table spaces, LOADPLUS uses only the first SORTOUT DD statement. For partitioned table spaces, LOADPLUS uses SORTOUT DD statements up to the number of sort processes that LOADPLUS can run concurrently, based on available memory and other factors.

BMC Software, Inc., Confidential and Proprietary Information

SORTWK*nn* Data Sets

SORTWK data sets are used by BMCSORT. BMCSORT uses the data sets in the PRELOAD and LOAD phases for two-phase load and in the COMBINED phase for single-phase processing.

Any time that LOADPLUS performs a sort, you must allocate SORTWK files in one of the following ways:

- Explicitly specify SORTWK DD statements in your JCL. Use this option when you want to control the allocation of your SORTWK data sets.
- Dynamically allocate SORTWK files with the DDTYPE command or installation option.
- Use the SORTNUM option to specify the maximum number of data sets for BMCSORT to dynamically allocate.

You cannot allocate a SORTWK data set as any of the following data set types:

- a VIO data set
- a tape data set
- a multivolume data set
- a data set in an SMS storage group that specifies EXTENDED FORMAT YES

  BMCSORT does not support SORTWK data sets that extend beyond 65535 tracks on a single volume.

Dynamically Allocating SORTWK Data Sets

You can have your SORTWK data sets dynamically allocated by either LOADPLUS or BMCSORT.

To have LOADPLUS dynamically allocate your SORTWK data sets, you must specify the ENUMROWS command option and the DDTYPE command or installation option.

BMCSORT dynamically allocates SORTWK data sets if they are not already allocated explicitly in your JCL or allocated dynamically by LOADPLUS with the DDTYPE command or installation option. BMCSORT also dynamically allocates SORTWK data sets as needed. To ensure that BMCSORT has enough information to allocate SORTWK space accurately and efficiently, specify the ENUMROWS and SORTNUM command options.

BMC Software, Inc., Confidential and Proprietary Information

For more information, see “Dynamically Allocating Sort Work Data Sets” on page 2-14.

Allocating SORTWK Data Sets

If you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS provides an estimate of the space that is needed for your SORTWK data sets. See “ANALYZE Option for Estimating Data Set Allocation” on page 4-25 for more information. If you do not specify ANALYZE PAUSE or ANALYZE ONLY and you do not want to have your sort work data sets dynamically allocated, you must estimate the allocation values yourself, by using the formulas that are provided in “SORTWK Data Set” on page D-9.

Note: To provide maximum flexibility and optimize performance, BMC Software recommends that you allocate 12 data sets, all of the same size.

When determining the number of concurrent tasks to run, LOADPLUS checks the amount of SORTWK space that is allocated. In this calculation, LOADPLUS uses only the primary allocation. The secondary allocation is not guaranteed.

For information about calculating the size of the SORTWK data sets, see “SORTWK Data Set” on page D-9.

SYSDISC Data Set

LOADPLUS requires the SYSDISC data set if you want to save discarded records or if you specify a DISCARDS limit. If the SYSDISC data set is not allocated, discarded records are lost. The SYSDISC data set contains the input records that are discarded when LOADPLUS cannot load the record for any of the following reasons:

- An input value position is past the end of the input record.
- An input value is not entirely contained on the input record.
- A data conversion error (including a FIELDPROC error) occurs.
- A value generated for an identity column is out of range.
- A verification record does not match the table that is being loaded (for FORMAT BMCUNLOAD).
- LOADPLUS encounters an error byte during data translation.

BMC Software, Inc., Confidential and Proprietary Information

- LOADPLUS encounters a substitution character during data translation, but you specified NOSUBS.
- A VALIDPROC violation occurs.
- A record does not match any WHEN specification or table OBID.
- A record does not match any partition that is being loaded.
- A DB2 table check constraint violation occurred.
- A unique key violation occurred (except violations detected in a two-phase load under UNIQUECHECK NO or for a nonclustering index under UNIQUECHECK CLUSTER).

Note: If your input data set is a pipe, LOADPLUS does not write discards to a SYSDISC data set.

LOADPLUS writes to this data set in the PRELOAD phase of a two-phase load or the COMBINED phase of a single-phase or SQLAPPLY load. To override the default ddname, use the DISCARDDN command option as described on page 3-39.

You can correct this discarded data and use it as input to a subsequent load job specifying RESUME YES.

LOADPLUS frees the SYSDISC data set at the end of the PRELOAD phase or the COMBINED phase. If you specify SYSDISC YES and no records were written to the SYSDISC file during the load process, LOADPLUS automatically deletes the SYSDISC file. However, LOADPLUS does not delete this data set if it contains discarded records. You must manually delete the SYSDISC data set in this case.

If you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS provides an estimate of the space that LOADPLUS needs for the SYSDISC data set. See “ANALYZE Option for Estimating Data Set Allocation” on page 4-25 for more information. If you do not specify ANALYZE PAUSE or ANALYZE ONLY, you must estimate the allocation values yourself by using the formula that is provided in “SYSDISC Data Set” on page D-8. Alternatively, you can use the DDTYPE command option to have LOADPLUS dynamically allocate your SYSDISC data set.

BMC Software, Inc., Confidential and Proprietary Information

SYSERR Data Set

LOADPLUS always requires the SYSERR data set. This data set contains information about errors and is for LOADPLUS internal use only. LOADPLUS uses this data set from the time LOADPLUS starts reading the SYSREC data set until the time LOADPLUS completes loading all the tables. To override the default ddname, use the ERRDDN command option as described on page 3-39.

Note: You can use the DISCARDS option to limit the number of SYSERR records that LOADPLUS generates.

For a SQLAPPLY load, when Apply Plus needs to take a snap dump, Apply Plus always directs the snap dump to the SYSERR data set. To prevent a data set name conflict when you run a SQLAPPLY load, BMC Software recommends that you use the ERRDDN installation or command option to specify a default ddname other than SYSERR for the standard error data set that LOADPLUS uses.

If you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS provides an estimate of the space that LOADPLUS needs for the SYSERR data set. See “ANALYZE Option for Estimating Data Set Allocation” on page 4-25 for more information. If you do not specify ANALYZE PAUSE or ANALYZE ONLY, you must estimate the allocation values yourself by using the formulas that are provided in “SYSERR Data Set” on page D-8. Alternatively, you can use the DDTYPE command option to have LOADPLUS dynamically allocate your SYSERR data set.

SYSIN Data Set

LOADPLUS always requires the SYSIN data set. This input data set contains the LOAD command. The UTILINIT phase reads, parses, and verifies the LOAD command that is provided in this data set. The data set’s attributes must be specified as fixed length and blocked records (RECFM=FB), and the record length must be 80 columns (LRECL=80). LOADPLUS uses only columns 1 through 72.

BMC Software, Inc., Confidential and Proprietary Information

SYSPRINT Data Set

LOADPLUS always requires the SYSPRINT data set. This output data set contains LOADPLUS messages.

Note the following considerations if you direct SYSPRINT to a tape or disk data set:

- The DSN messages from DB2, such as those from a QUIESCE utility, are lost if you direct SYSPRINT to a tape or disk data set.
- If you direct SYSPRINT to a tape or disk data set, BMC Software recommends that you do not specify the BUFNO parameter on the DD statement for this data set.

For information about the level of messages that LOADPLUS displays and how to change the message level, see “Message Level (MSGLEVEL)” on page 4-7.

SYSREC*nn* Data Sets

LOADPLUS always requires at least one SYSREC data set. This input data set contains the input data that you are loading. LOADPLUS uses the data set in the PRELOAD phase for two-phase load or in the COMBINED phase for single-phase processing. At the end of the PRELOAD phase or the COMBINED phase, LOADPLUS frees this data set. However, you must specify a SYSREC DD statement in the JCL (which can be SYSREC DD DUMMY) or ensure that your LOAD command contains the INDSN option if you restart the job, regardless of the phase in which you restart.

You can use either form of the data set name (SYSREC or SYSREC*nn*). When specifying multiple data sets, you can use one *or* both forms. *nn* can be one or two valid national characters or blanks. To override the default ddname, use the INDDN command option as described on page 3-36. For performance information about this data set, see “SYSREC Data Set” on page 6-9.

As an alternative to specifying SYSREC data sets in your JCL, you can have LOADPLUS dynamically allocate your SYSREC data sets by using the INDSN command option. See “INDSN” on page 3-34 for information about this option.

BMC Software, Inc., Confidential and Proprietary Information

Restrictions

SYSREC data sets have the following restrictions:

- If you are using RECFM=U, or have concatenated an uninitialized data set in your SYSREC DD (which results in RECFM=U), LOADPLUS processes only one record per block and processing might not function as you expect.
- If you allocate a block size greater than 32 KB, the load job terminates.
- If you are generating values into an identity column and you want the values ordered in input order, you must use a single SYSREC data set.
- If you have multiple SYSREC*nn* data sets and specify ORDER PRESORTED, the load job terminates.
- You can specify a mixture of DASD files and pipes as input. However, you cannot concatenate these files within one SYSREC DD.
- If you use multiple pipe data sets, you must specify a separate SYSREC DD statement for each pipe data set, and the DCB parameters for all of the pipe data sets must be the same.
- If you specify multiple SYSREC*nn* data sets, they must use the same record format (RECFM). In other words, they must all be either F or FB, or they must all be V, VB, or VBS. Mixing fixed and variable format data sets produces unpredictable results.
- If you specify multiple SYSREC*nn* data sets, they must be defined with the same record length (LRECL).
- If you concatenate your input data sets, specify the data set with the largest block size first.

Specifying Multiple SYSREC*nn* Data Sets

LOADPLUS permits up to 256 SYSREC*nn* DD statements, providing multitasking of the read process in the PRELOAD phase or in the COMBINED phase. LOADPLUS reads the SYSREC*nn* data sets in parallel only when loading data into a partitioned table space. For details about performance implications of multitasking in LOADPLUS, see “Enabling Multitasking” on page 6-5.

Data in the SYSREC*nn* data sets does not have to be grouped by partition. Data for a particular partition can be in one or more SYSREC*nn* data sets.

BMC Software, Inc., Confidential and Proprietary Information

Although not required, BMC Software recommends that you order your SYSREC DD statements from the largest data set to the smallest to optimize performance.

If you specify PRELOAD ANALYZE, LOADPLUS stops at the beginning of the PRELOAD phase and displays a message that indicates the optimal number of SORT and READER tasks that it can process concurrently. You can use this information to adjust the number of SYSREC data sets in your JCL.

For restrictions when using multiple SYSREC data sets, see “Restrictions” on page 4-19.

SYSUT1*nn* Data Sets

The index work data set contains index key entries and becomes the input for the index-building process. For a two-phase load, LOADPLUS uses this data set in the PRELOAD phase and in the LOAD phase. When used with a single-phase LOAD RESUME YES SHRLEVEL NONE job, LOADPLUS writes the index key entries to the index work data set during the COMBINED phase, but only reads from this data set if you restart your job. Refer to Table 4-5 for circumstances under which LOADPLUS requires this data set.

Table 4-5 SYSUT1*nn* Usage

Load Type	SYSUT1*nn* Requirement
two-phase load	at least one required unless you specify ORDER YES and no nonclustering indexes are participating in the load
single-phase LOAD RESUME YES SHRLEVEL NONE	if you want to be able to restart, at least one required unless you specify ORDER YES and no nonclustering indexes are participating in the load **Note**: If you do not specify a SYSUT1 data set and the load job fails, you might have to recover the object. See “Recovering the DB2 Object after Terminating or Canceling the Job” on page 4-34 for more information.
single-phase LOAD REPLACE or LOAD RESUME NO	not used
LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY	not used

You can always run with just one SYSUT1 data set. However, if you want to improve performance, you can specify more than one data set. For more information about specifying multiple SYSUT1 data sets, see page 4-21.

BMC Software, Inc., Confidential and Proprietary Information

To override the default ddname or ddname prefix SYSUT1, use the WORKDDN command option as described on page 3-38. See “SYSUT1 Data Set” on page 6-10 and page 6-27 for additional information about performance and considerations.

Allocating the SYSUT1*nn* Data Set

If you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS provides an estimate of the space that LOADPLUS needs for the SYSUT1 data set. See “ANALYZE Option for Estimating Data Set Allocation” on page 4-25 for more information.

If you do not specify ANALYZE PAUSE or ANALYZE ONLY, you must estimate the allocation values yourself by using the information that is provided in “SYSUT1 Data Set” on page D-5. Alternatively, you can use the DDTYPE command option to have LOADPLUS dynamically allocate your SYSUT1 data set.

Warning! In order to restart LOADPLUS anytime after the PRELOAD or COMBINED phase has begun, the SYSUT1 cannot be a temporary data set. For information about how LOADPLUS treats temporary data sets, see “Check for Data Set Attributes” on page 2-22.

Specifying Multiple Work Data Sets—There is no penalty for specifying more SYSUT1 data sets than LOADPLUS requires. Using multiple work data sets reduces the amount of DASD that is required for loading if the key lengths vary a great deal. Also, if enough sort work space and memory are available, using multiple work data sets allows LOADPLUS to build nonclustering indexes concurrently. See “SYSUT1 Data Set” on page 6-10 for additional information about performance and DASD considerations.

If you specify more than one SYSUT1 data set, specify them in the following way:

- If you want a separate work data set for each participating nonclustering index, specify one DD statement named SYSUT1*nn* for index.
- If you do not specify ORDER YES, also include a SYSUT1*nn* DD statement for any clustering index.

Specifying *nn* creates a unique ddname; the *nn* has no relation to the index name.

BMC Software, Inc., Confidential and Proprietary Information

When using multiple SYSUT1 data sets, LOADPLUS assigns the index with the largest key length to the first SYSUT1*nn* DD statement in the JCL, the index with the second-largest key length to the second SYSUT1*nn* DD statement in the JCL, and so on. This assignment of indexes to data set by key length can assist you in allocating the sizes of your data sets and allows you to place data sets on different devices when needed.

Due to this assignment, you should allocate the space for the participating index with the largest key length in the first SYSUT1*nn* DD statement in your JCL, space for the index with the second-largest key length in the second SYSUT1*nn* DD statement, and so on. Follow the instructions in "SYSUT1 Data Set" on page D-5 to determine the amount of space that you need for the primary and secondary allocations.

Specifying a Single Work Data Set—If you want a single work data set for all nonclustering indexes, specify one SYSUT1 DD statement. To calculate the space that LOADPLUS needs for a single SYSUT1 data set, see "SYSUT1 Data Set" on page D-5.

SYSIDCIN Data Set

SYSIDCIN is the input data set that contains your IDCAMS command statements. LOADPLUS uses these statements to delete and redefine VCAT-defined data sets or the staging data sets for your VCAT-defined data sets. LOADPLUS issues the commands that the SYSIDCIN data set contains. You are responsible for command specifications and results.

LOADPLUS requires the SYSIDCIN data set if the value of the REDEFINE option is YES and one of the following conditions applies:

- You want to delete and redefine your existing VCAT-defined VSAM data sets as part of the load.
- You specify LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE and you are loading VCAT-defined data sets.

If you specify REDEFINE YES and do not provide the SYSIDCIN data set, LOADPLUS reuses the existing data sets.

Note: You still have the option of deleting and redefining VCAT-defined data sets when you specify PRELOAD PAUSE. However, when you specify PRELOAD CONTINUE, the only way to delete and redefine VCAT-defined data sets as part of the load is to specify REDEFINE YES and provide the SYSIDCIN data set that contains the necessary IDCAMS command statements.

BMC Software, Inc., Confidential and Proprietary Information

To override the default ddname, SYSIDCIN, use the IDCDDN command option as described on page 3-57.

If any referenced object is VCAT-defined, the UTILINIT phase reads, parses, and performs minimal verification checks on the commands in this data set. LOADPLUS issues the commands during load processing.

SYSIDCIN can either be a single or concatenated list of sequential data sets, partitioned data set members, or both. You must define this data set as fixed length with blocked records (RECFM=FB), and the record length must be 80 bytes (LRECL=80). LOADPLUS uses only columns 1 through 72.

Note: See "Restarting LOADPLUS" on page 4-26 for important restart information.

Guidelines for Providing IDCAMS Commands in the SYSIDCIN Data Set

LOADPLUS supports the following IDCAMS commands and their associated parameters as defined in the IBM *DFSMS Access Method Services for Catalogs* document:

- DELETE
- DEFINE
- SET
- IF-THEN-ELSE command sequence
- DO/END (with restrictions)

LOADPLUS does not allow the following IDCAMS command specifications. (This can differ from the specification guidelines given in the *DFSMS Access Method Services for Catalogs* document.) LOADPLUS checks for these specifications during the UTILINIT phase and terminates with an error message if it finds any of the following items:

- an IF statement without a DEFINE, DELETE, or SET command
- a nested IF statement
- different data set names specified in an IF statement
- more than one DO/END statement within a single THEN or ELSE clause
- a DO/END statement existing outside of an IF statement
- an embedded comment (a comment within a comment)
- a comment on a DO statement that continues to the next line
- a keyword that continues to the next line

To avoid a failure during processing, ensure that each DEFINE has a corresponding DELETE specified before it and that each DELETE has a corresponding DEFINE specified after it. Ensure that both commands specify the same data set name.

BMC Software, Inc., Confidential and Proprietary Information

You can use the SET command to reset the IDCAMS condition code if a failure occurs in the LOAD phase or the COMBINED phase and you do not want LOADPLUS to terminate.

When You Specify SHRLEVEL NONE

Using the guidelines in the preceding section, specify commands only for those VCAT-defined data sets that you want to delete and redefine. LOADPLUS reuses any data sets participating in the load that have no corresponding IDCAMS commands. LOADPLUS ignores any command that references a data set that does *not* participate in the load and issues message BMC50604I. LOADPLUS ignores any empty SYSIDCIN data set and continues processing.

When You Specify LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE

Using the guidelines in “Guidelines for Providing IDCAMS Commands in the SYSIDCIN Data Set” on page 4-23, specify commands that operate only on the staging data sets for your VCAT-defined data sets. Always use the naming conventions outlined in “Staging Data Sets” on page 2-31.

To define the staging data sets and avoid definition errors, provide a DELETE statement followed by a SET MAXCC=0 command (in case the delete process fails) before each DEFINE statement. LOADPLUS processes this set of statements (DELETE, SET MAXCC, DEFINE) for each object as the LOAD or COMBINED phase begins for that object. LOADPLUS ignores any command that references one of the original VCAT-defined data sets and issues message BMC50604I.

UTPRINT Data Set

LOADPLUS always requires the UTPRINT data set if sorting is necessary. The presence of this data set tells LOADPLUS to report sort messages. However, the actual messages for each sort process appear in separate SYS*nnnnn* data sets, where *nnnnn* is a system-assigned sequential number. You cannot specify a sequential or partitioned data set for UTPRINT. UTPRINT supports only SYSOUT data sets.

Other Data Sets

Because it uses BMCSORT, LOADPLUS ignores any traditional sort routine DD statements (such as $ORTPARM and DFSPARM) that you specify.

BMC Software, Inc., Confidential and Proprietary Information

ANALYZE Option for Estimating Data Set Allocation

If you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS gathers information about the table space and the index space. In addition to cardinality and average row size, the ANALYZE phase provides estimated data set sizes for the following data sets:

- load (SORTOUT)
- work (SYSUT1)
- sort work (SORTWK)
- discard (SYSDISC)
- error (SYSERR)
- image copy (BMCCPY, BMCCPZ, BMCRCY, and BMCRCZ)

Note: When you specify ANALYZE ONLY, LOADPLUS provides estimates for all data sets, regardless of whether you need them for the type of load that you are running. These estimates give you the information that you need to run any type of load.

LOADPLUS writes these statistics to the SYSPRINT data set. For information about other statistical information messages that the ANALYZE phase issues, see “Messages from the ANALYZE Phase” on page 6-18.

As an alternative to using ANALYZE to help you estimate sizes for work files, you can have LOADPLUS dynamically allocate your work files for you. To use dynamic allocation, you must specify ENUMROWS and have dynamic work file allocation active, either in your installation options or with the DDTYPE command option.

If you do not use the PAUSE or ONLY keywords with ANALYZE, LOADPLUS also gathers the information described in this section. However, instead of pausing or stopping, LOADPLUS continues processing. If dynamic allocation is enabled, LOADPLUS uses the ANALYZE phase information to dynamically allocate your work files. In this case, the ANALYZE phase does not write the statistics to SYSPRINT.

Note: When you use the ANALYZE function, you must specify the ENUMROWS option. For information about the ENUMROWS option, see “ENUMROWS” on page 3-46.

ANALYZE PAUSE or ANALYZE ONLY provides the estimated information in table format. Messages BMC51530I, BMC51531I, and BMC51532I provide the heading information and multiple BMC51533I messages provide the estimates. LOADPLUS issues a separate BMC51533I message for each data set and provides the following information:

- data set name
- number of kilobytes

BMC Software, Inc., Confidential and Proprietary Information

- primary and secondary 3380 cylinder quantities
- primary and secondary 3390 cylinder quantities
- index name, where applicable

Note: For rows that contain VARCHAR columns or tables that contain EDITPROCs, ANALYZE ONLY might report a secondary quantity for SORTOUT that is larger than the reported primary quantity because LOADPLUS bases the primary quantity on the average row length and the secondary quantity on the maximum row length from the DB2 catalog. In this instance, BMC Software recommends that you provide a secondary quantity of approximately 25 percent of the primary quantity.

Running LOADPLUS Jobs

You can invoke LOADPLUS as a batch job, restart it from any phase, and terminate or cancel the job.

Invoking LOADPLUS

You normally invoke LOADPLUS as a batch job by specifying execution of the module AMUUMAIN on the EXEC statement of your JCL, along with the required EXEC statement parameters. You must also specify any DD statements that LOADPLUS requires, as described in preceding sections.

Ensure that all required libraries are available and APF-authorized as described in “STEPLIB DD Statement” on page 4-8.

Restarting LOADPLUS

You can restart LOADPLUS from any phase. In general, when you encounter a failure, you can correct the problem and restart the load with either RESTART or RESTART(PHASE). LOADPLUS issues messages as it loads and rebuilds each DB2 object. The BMCSYNC table contains an entry for each DB2 object that is involved in the load and its current status. The BMCUTIL table contains an entry for each load job and its current status.

BMC Software, Inc., Confidential and Proprietary Information

When you restart a LOADPLUS job, you must use the same input data that you used originally and it must be in the same order. For example, if you are loading concatenated data sets, these data sets must be in the same order as they were originally, and you must not include any additional data sets. Similarly, if LOADPLUS encountered duplicates during the original job, do not remove them before you restart. Data that is missing or that is in a different order in the restarted job than in the original job produces unpredictable results.

The following sections describe information that you need to know when restarting or recovering from specific situations or when a particular environment, type of load job, or object is involved. Use Table 4-6 to help locate this information.

Table 4-6 Restart Situations

Environment or Object	Page Reference
Data sharing environment	4-28
Dynamic allocation	4-28
Two-phase load	4-29
Single-phase load	4-29
SQLAPPLY load	4-29
SORTOUT data sets	4-30
SYSUT1 data sets	4-30
SYSERR data set	4-30, 4-32
Inline image copies	4-30
Nonparticipating secondary indexes (SKIPIX option)	4-30
Duplicate unique keys	4-30
IDCAMS DEFINE	4-30
INDEX UPDATE	4-31
ORDER option	4-31
DELETEFILES	4-31
DEFINE NO objects	4-31
Inadequate space failure—SYSERR data set	4-32
Inadequate space failure—DB2 object	4-32

BMC Software, Inc., Confidential and Proprietary Information

Data Sharing Environment

On restart in a data sharing environment, LOADPLUS can use either the same member that was chosen in the original load or any other member in the specified group.

Dynamic Allocation

On restart, LOADPLUS automatically re-allocates dynamically allocated data sets.

Changing any dynamic work file allocation option on restart such that it results in different ddnames or a different number of DDs than the original option can produce an error. If you need to change the number of SORTOUT or SYSUT1 work files, resubmit the job with a parameter of NEW.

You cannot change the value for the ENUMROWS or ACTIVE option on any restart.

To change the value of other dynamic work file allocation options, specify RESTART(PHASE).

If you run out of space when dynamically allocating copy data sets, you can perform one of the following actions:

- Specify COPY NO and restart the job.

  Note: If you choose this option, ensure that you specify the appropriate COPYPEND option.

- Alternatively, you can perform the following steps:

  1. Delete and reallocate the data sets.
  2. Specify DD statements for the data sets in your JCL.
  3. Change your syntax to IFALLOC USE for the data sets.
  4. Restart your job.

BMC Software, Inc., Confidential and Proprietary Information

Two-Phase Load

In general, you restart a two-phase load depending on whether you want to restart from the last restart sync point or at the beginning of the last incomplete phase:

- To restart from the last restart sync point, specify RESTART without (PHASE). LOADPLUS takes restart sync points as each phase completes and as the processing of each DB2 object completes.

  Note: The SYNC option (on the LOAD command) controls the frequency with which rows are updated in the BMCSYNC table. It does not control sync points that LOADPLUS uses when restarting.

- To restart at the beginning of the last incomplete phase, specify RESTART(PHASE). The utility ID must exist in the BMCUTIL table.

Single-Phase Load

In general, the restart parameter RESTART(PHASE) works the same as RESTART when you restart a single-phase load. For additional criteria that might affect which RESTART option to use, refer to the other topics in this section.

For a single-phase LOAD RESUME YES SHRLEVEL NONE load job, you cannot restart the job under either of the following circumstances:

- You did not allocate a SYSUT1 data set (or specify dynamic allocation for SYSUT1) on the original job.

- You did not allocate a SORTOUT data set (or specify dynamic allocation for SORTOUT) on the original job, and you specified ORDER YES but not PART REPLACE.

See “Recovering the DB2 Object after Terminating or Canceling the Job” on page 4-34 for information about recovering your load job.

SQLAPPLY Load

As with other LOADPLUS jobs, if you restart a SQLAPPLY job, you must use the same input data that you used originally and it must be in the same order. This is particularly critical for a SQLAPPLY job because of the way that Apply Plus handles restart processing.

BMC Software, Inc., Confidential and Proprietary Information

In general, the restart parameter RESTART(PHASE) works the same as RESTART when you restart a SQLAPPLY job. For additional criteria that might affect which RESTART option to use, refer to the other topics in this section.

SORTOUT, SYSUT1, and SYSERR Data Sets

If you specified DISP=NEW for the SORTOUT, SYSUT1, and SYSERR data sets, you must change your JCL to DISP=OLD before restarting. Alternatively, you can allocate the data sets with DISP=MOD, or allocate the data sets in a prior step and then specify DISP=OLD for the data sets in the load step. For more information, see “Check for Data Set Attributes” on page 2-22.

Inline Image Copies

If you want an inline image copy and you restart your two-phase job in the LOAD phase, you must specify RESTART(PHASE).

Note: If you are restarting a two-phase load job and *do not* specify RESTART(PHASE) or you are restarting a single-phase load job, LOADPLUS changes the value of the INLINE command to NO, regardless of the value that you specified for the INLINE command or the INLINECP installation option.

Nonparticipating Secondary Indexes (SKIPIX Option)

If you change the SKIPIX option on restart after the load job terminated in any phase other than UTILINIT, LOADPLUS issues message BMC50113E and terminates with return code 8.

Duplicate Unique Keys

If you are restarting because the job terminated in discard processing while handling duplicate unique keys, LOADPLUS issues message BMC51449W if any indexes are left in a RECOVER pending status. If this situation occurs, you can recover the affected indexes by running RECOVER PLUS or the IBM recovery utility.

IDCAMS DEFINE

If you are restarting because a failure occurred during IDCAMS DEFINE processing, you must manually define the data set.

BMC Software, Inc., Confidential and Proprietary Information

INDEX UPDATE

If you specified INDEX UPDATE and a failure occurred during index update processing, you must recover your table space and indexes. If you attempt to restart, LOADPLUS terminates and issues message BMC51435S.

When you specify INDEX UPDATE and index update processing completes, but the job fails in the LOAD or COMBINED phase, you must restart the job by specifying RESTART without PHASE. If you specify RESTART(PHASE), LOADPLUS terminates and issues message BMC51436E.

ORDER Option

You cannot change the value of the ORDER command option when restarting your load job.

DELETEFILES

To restart during DELETEFILES processing, specify RESTART without (PHASE).

DEFINE NO Objects

For table spaces or index spaces that are created with DEFINE NO, if LOADPLUS is unable to materialize a data set that is associated with the tables that you are loading, LOADPLUS attempts to materialize the underlying data sets for *all* of the tables before terminating.

If you are loading a table whose table space or index spaces are created with DEFINE NO and there is a check constraint or VALIDPROC on the table, LOADPLUS attempts to materialize the underlying data sets. However, if LOADPLUS cannot resolve the constraint or if the VALIDPROC prevents LOADPLUS from materializing the data sets, LOADPLUS terminates. To load the table in this situation, you must manually materialize the associated data sets before you restart the job.

BMC Software, Inc., Confidential and Proprietary Information

Inadequate Space Failure on SYSERR Data Set

When allocating your SYSERR data set, underestimating its size can result in an *x*37 abend, indicating an out-of-space condition. Perform the following steps to recover and restart your job:

Step 1 Allocate a larger SYSERR data set and make sure that you use the same DCB parameters as your original data set. These might not be the same DCB parameters that were coded in your JCL.

Note: If you are dynamically allocating your SYSERR data set, you must use the same data set name as the original. Therefore you must perform this step after you delete the original data set (see step 2).

Step 2 Because LOADPLUS verifies your SYSERR data set on restart, you must copy the data from the original data set to the new data set.

Note: If you are dynamically allocating your SYSERR data set, use a temporary data set to copy your data from the original data set, delete the original data set, allocate a larger SYSERR data set with the same data set name and DCB parameters as the original, then copy the data to the new SYSERR data set.

Step 3 Restart your job.

Inadequate Space Failure on DB2 Object

Failure during the LOAD phase can result in an unusable table space. If space is inadequate in the DB2 data set when you specify REPLACE, complete the actions that Table 4-7 on page 4-33 describes to help eliminate the cause of the failure.

BMC Software, Inc., Confidential and Proprietary Information

Table 4-7 Reallocation Process for Inadequate Space Failure

Data Set Definition	Action
The table space or index data sets are VCAT-defined and no SYSIDCIN DD statement is provided.	Allocate new data sets for the data sets that failed, specifying a larger primary and secondary space or both.
The table space or index data sets are VCAT-defined and you included a SYSIDCIN DD statement for redefining them as part of the load.	Either specify REDEFINE NO and allocate new data sets for the data sets that caused the failure, or alter the primary and secondary space quantities that are specified in the SYSIDCIN data set.
The table space or index data sets are STOGROUP-defined.	Specify REDEFINE YES and alter the table space and index primary and secondary quantity values.

Note: After the PRELOAD phase, LOADPLUS does not use changes to FREEPAGE, PCTFREE, or PIECESIZE values. LOADPLUS uses other values, such as STOGROUP definitions and data set passwords, if they have changed.

After correcting the problem that caused the inadequate space failure, restart the load job with the RESTART option. LOADPLUS does not reload or rebuild DB2 objects that were already loaded or built successfully. However, if you reallocate any data sets that were successfully loaded or built, restart the load process at the beginning of the LOAD phase with RESTART(PHASE).

Warning! If you reallocate data sets that were successfully loaded or built in the COMBINED phase, you will lose the data contained in those data sets.

Terminating or Canceling the Job

If you want to end the utility immediately and want the ability to restart your job, cancel the job by using the OS/390 or TSO CANCEL command.

If you want to end the job and do not intend to restart, terminate the load job by deleting the corresponding row from the BMCUTIL table or by specifying TERM on the restart parameter of the utility. If the job is currently running, it terminates at the next sync point. If you terminate a job, you cannot restart it. The minimum JCL that is required when specifying TERM is the STEPLIB for the LOADPLUS load library and the SYSPRINT DD statement.

BMC Software, Inc., Confidential and Proprietary Information

You can also terminate a LOADPLUS job by issuing the TERM command through BMCDSN—the BMC Software command processor for the BMC Software utilities for DB2. For information about using BMCDSN, see the *BMCDSN Command Processor Reference Manual*. For information about installing BMCDSN, see the *Utility Products for DB2 Customization Guide*.

Recovering the DB2 Object after Terminating or Canceling the Job

You must perform certain recovery steps under the following circumstances:

- You terminated the job by

  — issuing the TERM command through BMCDSN
  — specifying the TERM parameter on the EXEC statement
  — deleting the row from the BMCUTIL table that corresponds to the utility job

- The job ended for a reason other than you terminating it, and you do not intend to restart it (or you cannot restart it due to omitted SORTOUT or SYSUT1 data sets).

- You canceled the job, and you do not intend to restart it (or you cannot restart it due to omitted SORTOUT or SYSUT1 data sets).

Table 4-8 on page 4-35 describes the recovery steps that you need to take.

Additional Considerations—The following additional considerations apply to recovery:

- If you canceled your LOADPLUS job, or the job ended for a reason other than you terminating it, you can restart the job (with the exception described in the following note). When you restart the job, you do not need to perform any additional recovery steps other than under the special circumstances that are described in “Restarting LOADPLUS” on page 4-26.

  Note: If you did not allocate SORTOUT or SYSUT1 data sets and LOADPLUS completed sort processing before the cancellation or termination, you cannot restart your load job.

- If LOADPLUS dynamically allocated your work files, and you cancel your job or your job terminates, you must manually delete the dynamically allocated files.

BMC Software, Inc., Confidential and Proprietary Information

Table 4-8 Recovering After Termination (Part 1 of 2)

Phase in Which Job Terminated	Command Option Specified	Other Conditions	Is Object Usable?	Recovery Steps
any phase other than UTILTERM	LOAD REPLACE SHRLEVEL REFERENCE *or* LOAD REPLACE SHRLEVEL CHANGE		The object is usable (with original data).	No recovery steps are required.
UTILINIT	any		The object is usable.	No recovery steps are required.
PRELOAD	LOAD REPLACE		The object is usable.	No recovery steps are required.
PRELOAD	any option other than LOAD REPLACE		The object is usable.	Start the table space and index spaces.
LOAD	LOAD RESUME YES SHRLEVEL NONE	The table space contains data before the load job.	The table space is *not* usable. Associated participating indexes are *not* usable until recovered.	Complete the following steps: 1. Start the table space and index spaces. 2. Recover the table space. 3. Recover the indexes. 4. Rerun the job.
LOAD or COMBINED	LOAD REPLACE SHRLEVEL NONE		The object is *not* usable.	Complete the following steps: 1. Start the table and index spaces. 2. Either rerun the job *or* recover the table space and then recover the indexes.
COMBINED	LOAD RESUME YES SHRLEVEL NONE, single-phase		The object is *not* usable.	Complete the following steps: 1. Start the table and index spaces. 2. Recover the table space and then recover the indexes.

BMC Software, Inc., Confidential and Proprietary Information

Table 4-8 Recovering After Termination (Part 2 of 2)

Phase in Which Job Terminated	Command Option Specified	Other Conditions	Is Object Usable?	Recovery Steps
LOAD	LOAD RESUME YES PART REPLACE		The object is *not* usable.	Complete the following steps: 1. Start the table and index spaces. 2. Recover the partition. 3. Recover the partition's index and all nonclustering indexes. 4. Rerun the job.
LOAD or COMBINED	any	The table space has been loaded and all discard processing has completed.	It might be only the indexes that are *not* usable.	Complete the following steps: 1. Determine whether the table space was loaded by checking message BMC51475I or by looking in the BMCSYNC table. 2. Start the table and index spaces. 3. Recover the indexes. (You must perform this last step before you can use any of the loaded data.)
UTILTERM	SHRLEVEL NONE		The object is usable.	No recovery steps are required.
UTILTERM	LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY		The object is usable, but might be partially loaded.	No recovery steps are required.
UTILTERM	LOAD REPLACE SHRLEVEL REFERENCE *or* LOAD REPLACE SHRLEVEL CHANGE	The renaming process started but did not complete.	The object is *not* usable.	Manually complete or back out the renaming process and remove pending statuses. **Note:** BMC Software recommends that you restart your job rather than attempt to rename the staging data sets manually.

BMC Software, Inc., Confidential and Proprietary Information

Chapter 5 Examples of LOADPLUS Jobs

This chapter presents several examples of jobs that were run by using the LOADPLUS product. Each example includes a description of the load job and shows sample LOADPLUS job streams. Use Table 5-1 on page 5-2 to locate an example with a specific load type. Then use Table 5-2 on page 5-4 to locate the starting page for that example.

All of these examples have the following common properties:

- With the exception of the SQLAPPLY example, these examples use a single BMC Software product load library, assuming that this library contains both the LOADPLUS and BMCSORT load files.

- These examples use minimal data set space allocations in the JCL. For help in calculating allocation sizes for your data sets, see Appendix D, “Calculating LOADPLUS Work Data Set Sizes”. If you want LOADPLUS to calculate the size and dynamically allocate the data sets for you, see “Dynamic Work File Allocation Options” on page 3-133.

- The value for the FILECHK installation option is set to WARN. This option prevents the jobs from failing when they encounter temporary data sets.

You can find copies of the JCL for these examples in members AMUEX*nn* (where *nn* is the example number) in the *HLQ*.CNTL installation data set (where *HLQ* is the high-level qualifier set during installation).

BMC Software, Inc., Confidential and Proprietary Information

Table 5-1 Cross Reference of Examples by Function (Part 1 of 3)

Function	Relevant Examples
Load Type	
REPLACE	1, 2, 4–6, 8, 11, 13–15
RESUME YES	3, 9, 12
RESUME NO	7, 10
single phase	1, 3, 13
two phase	1, 2, 4–11, 14, 15
CSV input	13
UNLOAD PLUS input	14
dynamic allocation, sort only	7, 10
dynamic allocation, SYSREC only	2
dynamic allocation	11, 12, 14
online load	1, 12
partition parallelism	2, 3, 5, 8, 11
user exit	5
Object Type	
simple table space	1, 6, 7, 10
partitioned table space	2, 3, 5, 8, 9, 11, 12, 14, 15
segmented table space	4, 13
multitable table space	4
compressed table space	8, 9
clustering index, unique	2–4, 7–9, 10, 11, 13–15
clustering index, nonunique	5
nonclustering index, unique	5, 6, 15
nonclustering index, nonunique	2–4, 8, 9, 11, 14, 15
multiple indexes per table	4, 5, 8, 9, 11, 14, 15
Command Option	
ACTIVE	11
ANALYZE PAUSE	8
AND/OR condition	4
BMCSTATS YES	2, 8, 11
CENTURY	8
CHECK CONSTRAINTS	10
COPYDDN	3, 9

BMC Software, Inc., Confidential and Proprietary Information

Table 5-1 Cross Reference of Examples by Function (Part 2 of 3)

Function	Relevant Examples
COPY NO	7, 10, 13, 15
COPY YES	2–5, 8, 9, 11
COPYPEND NO	13
DDTYPE	11, 12
DEFAULTIF	5, 6
DELETEFILES	12–14
DISCARDDN	3, 9
DISCARDS	7, 10
DSNPAT	11, 12
ENFORCE CHECK CONSTRAINTS	10
ENUMROWS	7, 8, 10–12, 14
ERRDDN	3, 9
EXIT	5
FORMAT BMCUNLOAD	14
FORMAT CSV	13
GDGLIMIT	11
IFALLOC	11
IGNORE	7, 10
INDDN	3, 9
INDEX UPDATE	9
INDSN	2
INLINE YES	2
INTO TABLE (multiple)	4, 6, 9
INTO TABLE (one)	1–3, 5, 7, 8, 10–15
LOADDN	3, 9
LOG NO	2–4, 8, 9, 11, 15
NULLIF	5, 7, 10, 13
ORDER NO	3, 4, 6, 9, 11, 12
ORDER YES	2, 5, 8, 12, 15
PART REPLACE	9
POSITION (*start*:*end*)	2, 3, 7–12, 15
POSITION (*start*)	1, 5–7, 10
POSITION (*)	4, 7, 10, 13
PRELOAD LOAD	1, 3, 13

BMC Software, Inc., Confidential and Proprietary Information

Table 5-1 Cross Reference of Examples by Function (Part 3 of 3)

Function	Relevant Examples
RECOVERYDDN	3, 9
REDEFINE NO	2, 8, 11
REDEFINE YES	6, 7, 10
REGISTER ddname	5
REGISTER NONE	2, 3, 8, 9
REPORT	7, 10
SHRLEVEL CHANGE SQLAPPLY	12
SHRLEVEL REFERENCE	1
SIZEPCT	11
SKIPFIELDS YES	7, 10
SKIPIX	15
SORTDEVT	7, 10
SORTNUM	7, 10
SQLAPPLY	12
THRESHLD	11
TRIM	1, 5
UNIQUECHECK NO	9
UNIQUEINTO YES	4
UNIT	11
UPDATEDB2STATS	2, 8, 11
VALUE	5, 6
WHEN *condition*	4, 5, 7, 10, 15
WHEN TABLE=*obid*	14
WORKDDN	3, 9

Table 5-2 Page References for Each Example (Part 1 of 2)

Example	Starts on Page
Example 1	5-6
Example 2	5-16
Example 3	5-22
Example 4	5-29
Example 5	5-36
Example 6	5-44
Example 7	5-50

BMC Software, Inc., Confidential and Proprietary Information

Table 5-2 Page References for Each Example (Part 2 of 2)

Example	Starts on Page
Example 8	5-55
Example 9	5-66
Example 10	5-72
Example 11	5-77
Example 12	5-84
Example 13	5-101
Example 14	5-106
Example 15	5-115

BMC Software, Inc., Confidential and Proprietary Information

Example 1: LOAD REPLACE of a Simple Table Space (Two Phase, Single Phase, and SHRLEVEL REFERENCE)

This example illustrates the following three runs of a LOAD REPLACE of a simple table space with no indexes and shows the use of data conversions. With the exception of the changes to the LOAD command statement, the JCL is the same for all jobs.

- SYSPRINT A (starting on page 5-8) shows the output when the job is run as a two-phase load.

- The PRELOAD LOAD option is added to the LOAD command statement and the job is rerun as a single-phase load. SYSPRINT B (starting on page 5-10) shows the single-phase load output.

- In the third version of this job, the SHRLEVEL REFERENCE option is added to the command statement to improve the accessibility of the data and the job is rerun. SYSPRINT C (starting on page 5-13) shows the SHRLEVEL REFERENCE load output.

In this example, LOADPLUS loads the data into a simple, STOGROUP-defined table space that contains only one table with no indexes. LOADPLUS replaces all existing rows and deletes and redefines the VSAM data set for the space.

No sort work data sets are allocated because there are no indexes and LOADPLUS is not sorting the data.

The following data conversions take place:

- character data to small integer format
- timestamp in character format to DB2 timestamp format
- date in character format to DB2 date format

Because no COPY option is included on the LOAD command, LOADPLUS does not make a copy. After loading the data, LOADPLUS places the table space in COPY pending status.

The SYSPRINT output indicates that conversion errors occurred. Because the EXEC statement designates MSGLEVEL(1), LOADPLUS issues message BMC51501E to identify the erroneous input fields.

Note that the JCL defines the SORTOUT, SYSERR, and SYSDISC data sets as DISP=(MOD,CATLG,CATLG), and uses the NEW/RESTART restart parameter.

BMC Software, Inc., Confidential and Proprietary Information

The JCL for example 1A follows:

```
//        JOB
//EXAMPLE1 EXEC PGM=AMUUMAIN,
//         PARM='&SSID,EXAMPLE1,NEW/RESTART,,MSGLEVEL(1),AMU$OPTO'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//SYSREC   DD DSN=AMU.QA.RGRTESTS(EX1DATA),
//            DISP=SHR,UNIT=DISK
//SORTOUT  DD DSN=AMU.EXAMPL1.SORTOUT,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSERR   DD DSN=AMU.EXAMPL1.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSDISC  DD DSN=AMU.EXAMPL1.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
//SYSIN    DD *
   LOAD REPLACE
     INTO TABLE EXAMPLE1.TBL1
                (SMINT_COL1      POSITION(1)      CHARACTER(3)
                ,TIMESTAMP_COL2  POSITION(4)      TIMESTAMP EXTERNAL(19)
                ,DATE_COL3       POSITION(23)     DATE EXTERNAL
                ,VARCHAR_COL4    POSITION(33)     CHARACTER(40) TRIM
                ,CHAR_COL5       POSITION(73)     CHARACTER(7)
                )
/*
```

The LOAD command changes for example 1B:

```
//SYSIN    DD *
   LOAD REPLACE
     PRELOAD LOAD
     INTO TABLE EXAMPLE1.TBL1
                (SMINT_COL1      POSITION(1)      CHARACTER(3)
                ,TIMESTAMP_COL2  POSITION(4)      TIMESTAMP EXTERNAL(19)
                ,DATE_COL3       POSITION(23)     DATE EXTERNAL
                ,VARCHAR_COL4    POSITION(33)     CHARACTER(40) TRIM
                ,CHAR_COL5       POSITION(73)     CHARACTER(7)
                )
```

The LOAD command changes again for example 1C:

```
//SYSIN    DD *
   LOAD REPLACE SHRLEVEL REFERENCE
     PRELOAD LOAD
     INTO TABLE EXAMPLE1.TBL1
                (SMINT_COL1      POSITION(1)      CHARACTER(3)
                ,TIMESTAMP_COL2  POSITION(4)      TIMESTAMP EXTERNAL(19)
                ,DATE_COL3       POSITION(23)     DATE EXTERNAL
                ,VARCHAR_COL4    POSITION(33)     CHARACTER(40) TRIM
                ,CHAR_COL5       POSITION(73)     CHARACTER(7)
                )
```

BMC Software, Inc., Confidential and Proprietary Information

Output from a Two-Phase Load

The SYSPRINT A output for example 1 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                  COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   11:16:14 ...
BMC50002I UTILITY ID = 'EXAMPLE1'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7848K,ABOVE 16M=1664064K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST
```

(continued on following page)

(example 1A continued from preceding page)

```
BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD REPLACE
BMC50102I     INTO TABLE EXAMPLE1.TBL1
BMC50102I                (SMINT_COL1      POSITION(1)      CHARACTER(3)
BMC50102I                ,TIMESTAMP_COL2  POSITION(4)     TIMESTAMP EXTERNAL(19)
BMC50102I                ,DATE_COL3       POSITION(23)     DATE EXTERNAL
BMC50102I                ,VARCHAR_COL4    POSITION(33)     CHARACTER(40) TRIM
BMC50102I                ,CHAR_COL5       POSITION(73)     CHARACTER(7)
BMC50102I                )

BMC51422I FOR TABLE 'EXAMPLE1.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'SMINT_COL1        ' HAS POSITION(1:3)
BMC51423I FIELD 'TIMESTAMP_COL2    ' HAS POSITION(4:22)
BMC51423I FIELD 'DATE_COL3         ' HAS POSITION(23:32)
BMC51423I FIELD 'VARCHAR_COL4      ' HAS POSITION(33:72)
BMC51423I FIELD 'CHAR_COL5         ' HAS POSITION(73:79)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50394I UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'
BMC50395I MAXSORTS SET TO 1
BMC50474I BELOW 16M = 7380K, ABOVE 16M = 1650656K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 0, TIME = 72704
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 0
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 1 OF SYSREC, DATA = C'1990-02-29'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 2 OF SYSREC, DATA = C'1990-13-1 '
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 3 OF SYSREC, DATA = C'1990-9-31 '
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 5 OF SYSREC, DATA = C'1990-02-31-12.32.00'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 7 OF SYSREC, DATA = C'1990-04-31-12.34.00'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 8 OF SYSREC, DATA = C'1990-11-31-12.35.00'
```

(continued on following page)

(example 1A continued from preceding page)

```
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  26 PHYSICAL (26 LOGICAL) RECORDS READ FROM SYSREC
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 2: SORT TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 2
BMC50476I DDNAME = SYSREC, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 5, XFERS = 1, EMPTY WAITS = 0, FULL WAITS =2

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC, ID# 1

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                        FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                         CONSTRAINT NAME

BMC51524E         1         1         1  CONVERSION            0:         7  EXAMPLE1.TBL1                DATE_COL3
BMC51524E         2         2         2  CONVERSION            0:         7  EXAMPLE1.TBL1                DATE_COL3
BMC51524E         3         3         3  CONVERSION            0:         7  EXAMPLE1.TBL1                DATE_COL3
BMC51524E         5         5         4  CONVERSION            0:         7  EXAMPLE1.TBL1                TIMESTAMP_COL2
BMC51524E         7         7         5  CONVERSION            0:         7  EXAMPLE1.TBL1                TIMESTAMP_COL2
BMC51524E         8         8         6  CONVERSION            0:         7  EXAMPLE1.TBL1                TIMESTAMP_COL2
BMC50476I DDNAME = SYSREC, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0

BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX1DB.EX1TS.I0001.A001' WILL REQUIRE 3 PAGES
BMC51472I PRELOAD PHASE STATISTICS:  20 ROWS SELECTED FOR SPACE 'AMUEX1DB.EX1TS', 6 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  6 PHYSICAL (6 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7472K, ABOVE 16M = 1663288K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 0, #INDEX TASKS = 0
BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX1DB.EX1TS' DELETED
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX1DB.EX1TS.I0001.A001'
BMC50477I 0: PARTITION =   0, ROWS/KEYS = 20, I/O WAITS = 5, DDNAME = SYS00019
BMC50476I DDNAME = SORTOUT, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51475I LOAD STATISTICS:  20 ROWS LOADED INTO TABLE SPACE 'AMUEX1DB.EX1TS'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX1DB.EX1TS
BMC50513I             PARTS   = 0         TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 3             SPACE   = 2
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   0         3         20          0          0        2        0   33    0       2    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE1.TBL1                         20          1         0      58      33       0
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

Output from a Single-Phase Load

The SYSPRINT B output for example 1 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   11:16:20 ...
BMC50002I UTILITY ID = 'EXAMPLE1'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7848K,ABOVE 16M=1664064K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
```

(continued on following page)

(example 1B continued from preceding page)

```
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO
```

(continued on following page)

(example 1B continued from preceding page)

```
BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD REPLACE
BMC50102I    PRELOAD LOAD
BMC50102I     INTO TABLE EXAMPLE1.TBL1
BMC50102I                (SMINT_COL1      POSITION(1)      CHARACTER(3)
BMC50102I                ,TIMESTAMP_COL2  POSITION(4)     TIMESTAMP EXTERNAL(19)
BMC50102I                ,DATE_COL3       POSITION(23)     DATE EXTERNAL
BMC50102I                ,VARCHAR_COL4    POSITION(33)     CHARACTER(40) TRIM
BMC50102I                ,CHAR_COL5       POSITION(73)     CHARACTER(7)
BMC50102I                )

BMC51422I FOR TABLE 'EXAMPLE1.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'SMINT_COL1        ' HAS POSITION(1:3)
BMC51423I FIELD 'TIMESTAMP_COL2    ' HAS POSITION(4:22)
BMC51423I FIELD 'DATE_COL3         ' HAS POSITION(23:32)
BMC51423I FIELD 'VARCHAR_COL4      ' HAS POSITION(33:72)
BMC51423I FIELD 'CHAR_COL5         ' HAS POSITION(73:79)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7372K, ABOVE 16M = 1662296K, CPUS = 3
BMC51495I COMBINED OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 0, TIME = 36864
BMC51496I COMBINED ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 0
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX1DB.EX1TS' DELETED
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 1 OF SYSREC, DATA = C'1990-02-29'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 2 OF SYSREC, DATA = C'1990-13-1 '
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 3 OF SYSREC, DATA = C'1990-9-31 '
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 5 OF SYSREC, DATA = C'1990-02-31-12.32.00'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 7 OF SYSREC, DATA = C'1990-04-31-12.34.00'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 8 OF SYSREC, DATA = C'1990-11-31-12.35.00'
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  26 PHYSICAL (26 LOGICAL) RECORDS READ FROM SYSREC
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX1DB.EX1TS.I0001.A001'
BMC50477I 2: PARTITION =   0, ROWS/KEYS = 20, I/O WAITS = 5, DDNAME = SYS00022
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:01
BMC51510I 2: SORT TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 6, XFERS = 1, EMPTY WAITS = 0, FULL WAITS =1

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC, ID# 1

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                         FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                          CONSTRAINT NAME

BMC51524E         1         1         1  CONVERSION            0:         7  EXAMPLE1.TBL1                 DATE_COL3
BMC51524E         2         2         2  CONVERSION            0:         7  EXAMPLE1.TBL1                 DATE_COL3
BMC51524E         3         3         3  CONVERSION            0:         7  EXAMPLE1.TBL1                 DATE_COL3
BMC51524E         5         5         4  CONVERSION            0:         7  EXAMPLE1.TBL1                 TIMESTAMP_COL2
BMC51524E         7         7         5  CONVERSION            0:         7  EXAMPLE1.TBL1                 TIMESTAMP_COL2
BMC51524E         8         8         6  CONVERSION            0:         7  EXAMPLE1.TBL1                 TIMESTAMP_COL2
BMC50476I DDNAME = SYSREC, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0

BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51472I COMBINED PHASE STATISTICS:  20 ROWS SELECTED FOR SPACE 'AMUEX1DB.EX1TS', 6 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I COMBINED PHASE STATISTICS:  6 PHYSICAL (6 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC51475I LOAD STATISTICS:  20 ROWS LOADED INTO TABLE SPACE 'AMUEX1DB.EX1TS'
BMC50004I COMBINED PHASE COMPLETE.  ELAPSED TIME = 00:00:02
```

(continued on following page)

(example 1B continued from preceding page)

```
BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX1DB.EX1TS
BMC50513I             PARTS   = 0         TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 3             SPACE   = 2
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL   DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED  PSAVED
BMC50512I   0         3         20          0          0        2        0   33    0       2    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE1.TBL1                         20          1         0      58      33       0
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

Output from a SHRLEVEL REFERENCE Load

The SYSPRINT C output for example 1 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                 COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   11:16:28 ...
BMC50002I UTILITY ID = 'EXAMPLE1'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7852K,ABOVE 16M=1664060K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
```

(continued on following page)

(example 1C continued from preceding page)

```
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                 WORK                 SORTWORK             ERROR                DISCARD
BMC50470I ACTIVE   = NO                   NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    N/A                  0                    0

BMC50470I DDTYPE   = LOCPFCPY             LOCBFCPY             REMPFCPY             REMBFCPY
BMC50470I ACTIVE   = NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    0                    0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                    5                    5                    5
BMC50470I GDGEMPTY = NO                   NO                   NO                   NO
BMC50470I GDGSCRAT = NO                   NO                   NO                   NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD REPLACE SHRLEVEL REFERENCE
BMC50102I    PRELOAD LOAD
BMC50102I     INTO TABLE EXAMPLE1.TBL1
BMC50102I                (SMINT_COL1      POSITION(1)      CHARACTER(3)
BMC50102I                ,TIMESTAMP_COL2  POSITION(4)     TIMESTAMP EXTERNAL(19)
BMC50102I                ,DATE_COL3       POSITION(23)     DATE EXTERNAL
BMC50102I                ,VARCHAR_COL4    POSITION(33)     CHARACTER(40) TRIM
BMC50102I                ,CHAR_COL5       POSITION(73)     CHARACTER(7)
BMC50102I                )

BMC51422I FOR TABLE 'EXAMPLE1.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'SMINT_COL1        ' HAS POSITION(1:3)
BMC51423I FIELD 'TIMESTAMP_COL2    ' HAS POSITION(4:22)
BMC51423I FIELD 'DATE_COL3         ' HAS POSITION(23:32)
BMC51423I FIELD 'VARCHAR_COL4      ' HAS POSITION(33:72)
BMC51423I FIELD 'CHAR_COL5         ' HAS POSITION(73:79)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:00
```

(continued on following page)

(example 1C continued from preceding page)

```
BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50394I UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'
BMC50395I MAXSORTS SET TO 1
BMC50474I BELOW 16M = 7384K, ABOVE 16M = 1662320K, CPUS = 3
BMC51495I COMBINED OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 0, TIME = 36864
BMC51496I COMBINED ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 0
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX1DB.EX1TS' DELETED
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 1 OF SYSREC, DATA = C'1990-02-29'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 2 OF SYSREC, DATA = C'1990-13-1 '
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'DATE_COL3', RECORD NO. 3 OF SYSREC, DATA = C'1990-9-31 '
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 5 OF SYSREC, DATA = C'1990-02-31-12.32.00'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 7 OF SYSREC, DATA = C'1990-04-31-12.34.00'
BMC51501E DATE/TIME VALUE OUT-OF-RANGE, COLUMN = 'TIMESTAMP_COL2', RECORD NO. 8 OF SYSREC, DATA = C'1990-11-31-12.35.00'
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  26 PHYSICAL (26 LOGICAL) RECORDS READ FROM SYSREC
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.BMCDBD.AMUEX1DB.EX1TS.I0001.A001'
BMC50477I 2: PARTITION =   0, ROWS/KEYS = 20, I/O WAITS = 5, DDNAME = SYS00024
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:02
BMC51510I 2: SORT TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 6, XFERS = 1, EMPTY WAITS = 0, FULL WAITS =1

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC, ID# 1

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                         FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                          CONSTRAINT NAME

BMC51524E         1         1         1  CONVERSION            0:         7  EXAMPLE1.TBL1                 DATE_COL3
BMC51524E         2         2         2  CONVERSION            0:         7  EXAMPLE1.TBL1                 DATE_COL3
BMC51524E         3         3         3  CONVERSION            0:         7  EXAMPLE1.TBL1                 DATE_COL3
BMC51524E         5         5         4  CONVERSION            0:         7  EXAMPLE1.TBL1                 TIMESTAMP_COL2
BMC51524E         7         7         5  CONVERSION            0:         7  EXAMPLE1.TBL1                 TIMESTAMP_COL2
BMC51524E         8         8         6  CONVERSION            0:         7  EXAMPLE1.TBL1                 TIMESTAMP_COL2
BMC50476I DDNAME = SYSREC, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0

BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51472I COMBINED PHASE STATISTICS:  20 ROWS SELECTED FOR SPACE 'AMUEX1DB.EX1TS', 6 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I COMBINED PHASE STATISTICS:  6 PHYSICAL (6 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC51475I LOAD STATISTICS:  20 ROWS LOADED INTO TABLE SPACE 'AMUEX1DB.EX1TS'
BMC50004I COMBINED PHASE COMPLETE.  ELAPSED TIME = 00:00:03

BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEBACAT.DSNDBC.AMUEX1DB.EX1TS.I0001.A001'...
BMC50891I 1: DATASET 'DEBACAT.DSNDBC.AMUEX1DB.EX1TS.I0001.A001' IS RENAMED
BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEBACAT.DSNDBD.AMUEX1DB.EX1TS.I0001.A001'...
BMC50891I 1: DATASET 'DEBACAT.DSNDBD.AMUEX1DB.EX1TS.I0001.A001' IS RENAMED
BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEBACAT.BMCDBC.AMUEX1DB.EX1TS.I0001.A001'...
BMC50891I 1: DATASET 'DEBACAT.BMCDBC.AMUEX1DB.EX1TS.I0001.A001' IS RENAMED
BMC50890I 1: ATTEMPTING TO RENAME DATASET 'DEBACAT.BMCDBD.AMUEX1DB.EX1TS.I0001.A001'...
BMC50891I 1: DATASET 'DEBACAT.BMCDBD.AMUEX1DB.EX1TS.I0001.A001' IS RENAMED

BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
BMC50890I 1: ATTEMPTING TO DELETE DATASET 'DEBACAT.OLDDBC.AMUEX1DB.EX1TS.I0001.A001'...
BMC50891I 1: DATASET 'DEBACAT.OLDDBC.AMUEX1DB.EX1TS.I0001.A001' IS DELETED

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX1DB.EX1TS
BMC50513I             PARTS   = 0         TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 3             SPACE   = 2
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL   DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED  PSAVED
BMC50512I   0         3         20          0          0        2        0   33    0       2    1           0    0         0     0
BMC50521I TABLE NAME                           CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE1.TBL1                          20          1         0      58      33       0
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

Example 2: LOAD REPLACE of a Partitioned Table Space

This example shows a LOAD REPLACE of a 10-part table with a unique clustering index and a nonunique secondary index. Only the secondary index is processed by using a SYSUT1 work file, so the JCL includes only one SYSUT1 DD statement.

Seven of ten partitions are loaded in this example. Individual INTO PART commands are not specified because processing multiple INTO PART commands is less efficient and offers no benefit on a REPLACE run.

The LOAD command includes the INDSN option, which tells LOADPLUS to dynamically allocate the input data set. Notice that the JCL does not include a SYSREC DD statement.

Because the LOAD command includes the ORDER YES option, LOADPLUS sorts the data rows and clustering index keys. Because the LOAD command does not include the UNIQUECHECK NO option, LOADPLUS defaults to UNIQUECHECK YES, which tells LOADPLUS to check the clustering index for uniqueness.

COPY YES INLINE YES REGISTER NONE creates inline copies. The copy DD statements in the JCL direct LOADPLUS to create a single copy of each partition, with each copy in a separate data set.

LOADPLUS updates statistics in both the DB2 catalog and the DASD MANAGER PLUS statistics tables.

The JCL for example 2 follows:

```
//        JOB
//EXAMPLE2 EXEC PGM=AMUUMAIN,
//             PARM='&SSID,EXAMPLE2,NEW/RESTART,,MSGLEVEL(1),AMU$OPTO'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//SORTOUT  DD DSN=AMU.EXAMPLE2.SORTOUT,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSERR   DD DSN=AMU.EXAMPLE2.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSDISC  DD DSN=AMU.EXAMPLE2.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SYSUT101 DD DSN=AMU.EXAMPLE2.SYSUT101,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//SORTWK01 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK02 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK03 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK04 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK05 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK06 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK07 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK08 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK09 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK10 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK11 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK12 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//*
//BMCCPY01 DD DSN=AMU.EXAMPLE2.PART1.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY02 DD DSN=AMU.EXAMPLE2.PART2.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY03 DD DSN=AMU.EXAMPLE2.PART3.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY04 DD DSN=AMU.EXAMPLE2.PART4.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY05 DD DSN=AMU.EXAMPLE2.PART5.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY06 DD DSN=AMU.EXAMPLE2.PART6.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY07 DD DSN=AMU.EXAMPLE2.PART7.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY08 DD DSN=AMU.EXAMPLE2.PART8.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY09 DD DSN=AMU.EXAMPLE2.PART9.COPY,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY10 DD DSN=AMU.EXAMPLE2.PART10.COPY,,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
//SYSIN DD *
   LOAD REPLACE
     INDSN('AMU.QA.RGRTESTS(EX2DATA)')
     ORDER YES
     UPDATEDB2STATS YES BMCSTATS YES
     REDEFINE NO
   INTO TABLE EXAMPLE2.TBL1
       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
       ,SELLER_ID           POSITION(62:71) CHARACTER
        )
     LOG NO
     COPY YES INLINE YES REGISTER NONE
/*
```

BMC Software, Inc., Confidential and Proprietary Information

The SYSPRINT output for example 2 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                  COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   13:20:56 ...
BMC50002I UTILITY ID = 'EXAMPLE2'.  DB2 SUBSYSTEM ID = 'DEAH'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=6.1.0
BMC50471I REGION=0M,BELOW 16M=7828K,ABOVE 16M=1666132K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0
```

(continued on following page)

(example 2 continued from preceding page)

```
BMC50470I DDTYPE   = LOCPFCPY             LOCBFCPY             REMPFCPY             REMBFCPY
BMC50470I ACTIVE   = NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    0                    0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                    5                    5                    5
BMC50470I GDGEMPTY = NO                   NO                   NO                   NO
BMC50470I GDGSCRAT = NO                   NO                   NO                   NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY3'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD REPLACE
BMC50102I     INDSN('AMU.QA.RGRTESTS(EX2DATA)')
BMC50102I     ORDER YES
BMC50102I     UPDATEDB2STATS YES BMCSTATS YES
BMC50102I     REDEFINE NO
BMC50102I   INTO TABLE EXAMPLE2.TBL1
BMC50102I       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(62:71) CHARACTER
BMC50102I       )
BMC50102I   LOG NO
BMC50102I   COPY YES INLINE YES REGISTER NONE

BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7344K, ABOVE 16M = 1654108K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 74240
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 1
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  854 PHYSICAL (854 LOGICAL) RECORDS READ FROM SYSREC01
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 2: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSREC01, I/OS = 87, I/O WAITS = 10, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT, I/OS = 4, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 5, XFERS = 2, EMPTY WAITS = 0, FULL WAITS =1
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51471I PRELOAD STATISTICS:  454 ROWS SELECTED FOR PARTITION 1
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A001' WILL REQUIRE 12 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A001' WILL REQUIRE 5 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  59 ROWS SELECTED FOR PARTITION 2
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A002' WILL REQUIRE 4 PAGES
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 2 continued from preceding page)

```
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A002' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  60 ROWS SELECTED FOR PARTITION 3
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A003' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A003' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 4
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A004' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A004' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  62 ROWS SELECTED FOR PARTITION 5
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A005' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A005' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 6
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A006' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A006' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  19 ROWS SELECTED FOR PARTITION 7
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A007' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A007' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  100 ROWS SELECTED FOR PARTITION 8
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A008' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A008' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  100 ROWS SELECTED FOR PARTITION 9
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A009' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A009' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 10
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A010' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A010' WILL REQUIRE 3 PAGES (APPROX.)
BMC51488I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX2DB.IX1.I0001.A001' MAY REQUIRE 8 PAGES
BMC51472I PRELOAD PHASE STATISTICS:  854 ROWS SELECTED FOR SPACE 'AMUEX2DB.EX2TS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50474I BELOW 16M = 7432K, ABOVE 16M = 1664688K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 0, #INDEX TASKS = 1
BMC51508I MAX INDEX TASKS = 1, INDEXES PER TASK = 1, SORTWKS PER TASK = 1, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 10, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX2DB.EX2TS' DELETED
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A001'
BMC50476I DDNAME = SYSUT101, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 1: SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A001'
BMC50477I 0: PARTITION =   1, ROWS/KEYS = 454, I/O WAITS = 7, DDNAME = SYS00004
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A001'
BMC50477I 0: PARTITION =   1, ROWS/KEYS = 454, I/O WAITS = 5, DDNAME = SYS00003
BMC51474I LOAD STATISTICS:  454 ROWS LOADED INTO PARTITION 1
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX1.I0001.A001'
BMC50477I 1: PARTITION =   0, ROWS/KEYS = 854, I/O WAITS = 7, DDNAME = SYS00005
BMC51476I BUILD STATISTICS:  854 KEYS LOADED INTO INDEX 'EXAMPLE2.IX1'
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A002'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 59, I/O WAITS = 7, DDNAME = SYS00007
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 59, I/O WAITS = 5, DDNAME = SYS00006
BMC51474I LOAD STATISTICS:  59 ROWS LOADED INTO PARTITION 2
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A003'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A003'
BMC50477I 0: PARTITION =   3, ROWS/KEYS = 60, I/O WAITS = 7, DDNAME = SYS00009
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A003'
BMC50477I 0: PARTITION =   3, ROWS/KEYS = 60, I/O WAITS = 5, DDNAME = SYS00008
BMC51474I LOAD STATISTICS:  60 ROWS LOADED INTO PARTITION 3
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A005'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A005'
BMC50477I 0: PARTITION =   5, ROWS/KEYS = 62, I/O WAITS = 7, DDNAME = SYS00011
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A005'
BMC50477I 0: PARTITION =   5, ROWS/KEYS = 62, I/O WAITS = 5, DDNAME = SYS00010
BMC51474I LOAD STATISTICS:  62 ROWS LOADED INTO PARTITION 5
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A007'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A007'
BMC50477I 0: PARTITION =   7, ROWS/KEYS = 19, I/O WAITS = 7, DDNAME = SYS00013
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A007'
BMC50477I 0: PARTITION =   7, ROWS/KEYS = 19, I/O WAITS = 5, DDNAME = SYS00012
BMC51474I LOAD STATISTICS:  19 ROWS LOADED INTO PARTITION 7
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A008'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A008'
BMC50477I 0: PARTITION =   8, ROWS/KEYS = 100, I/O WAITS = 7, DDNAME = SYS00015
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A008'
BMC50477I 0: PARTITION =   8, ROWS/KEYS = 100, I/O WAITS = 5, DDNAME = SYS00014
BMC51474I LOAD STATISTICS:  100 ROWS LOADED INTO PARTITION 8
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A009'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A009'
BMC50477I 0: PARTITION =   9, ROWS/KEYS = 100, I/O WAITS = 7, DDNAME = SYS00017
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A009'
BMC50477I 0: PARTITION =   9, ROWS/KEYS = 100, I/O WAITS = 5, DDNAME = SYS00016
BMC51474I LOAD STATISTICS:  100 ROWS LOADED INTO PARTITION 9
BMC50476I DDNAME = SORTOUT, I/OS = 5, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00018
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A004'
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A004'
```

(continued on following page)

(example 2 continued from preceding page)

```
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00019
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 4
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A006'
BMC50477I 0: PARTITION =   6, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00020
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A006'
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A006'
BMC50477I 0: PARTITION =   6, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00021
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 6
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.IX0.I0001.A010'
BMC50477I 0: PARTITION =  10, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00022
BMC50375I INLINE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A010'
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEAHCAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A010'
BMC50477I 0: PARTITION =  10, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00023
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 10
BMC51476I BUILD STATISTICS:  854 KEYS LOADED INTO INDEX 'EXAMPLE2.IX0'
BMC51475I LOAD STATISTICS:  854 ROWS LOADED INTO TABLE SPACE 'AMUEX2DB.EX2TS'
BMC50476I DDNAME = BMCCPY01, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50376I 14 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART1.COPY'
BMC50476I DDNAME = BMCCPY02, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 6 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART2.COPY'
BMC50476I DDNAME = BMCCPY03, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 6 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART3.COPY'
BMC50476I DDNAME = BMCCPY04, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART4.COPY'
BMC50476I DDNAME = BMCCPY05, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 6 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART5.COPY'
BMC50476I DDNAME = BMCCPY06, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART6.COPY'
BMC50476I DDNAME = BMCCPY07, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 5 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART7.COPY'
BMC50476I DDNAME = BMCCPY08, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 6 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART8.COPY'
BMC50476I DDNAME = BMCCPY09, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 6 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART9.COPY'
BMC50476I DDNAME = BMCCPY10, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE2.PART10.COPY'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:07

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX2DB.EX2TS
BMC50513I             PARTS   = 10        TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 41          SPACE   = 150
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   1        12        454          0          0        2        0   85    0      15    1           0    0         0     0
BMC50512I   2         4         59          0          0        2        0   55    0      15    1           0    0         0     0
BMC50512I   3         4         60          0          0        2        0   56    0      15    1           0    0         0     0
BMC50512I   4         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   5         4         62          0          0        2        0   58    0      15    1           0    0         0     0
BMC50512I   6         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   7         3         19          0          0        2        0   36    0      15    1           0    0         0     0
BMC50512I   8         4        100          0          0        2        0   93    0      15    1           0    0         0     0
BMC50512I   9         4        100          0          0        2        0   93    0      15    1           0    0         0     0
BMC50512I  10         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE2.TBL1                        854         21         0      68      51       0
BMC50510I INDEX       EXAMPLE2.IX0  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE2.TBL1     COLUMN  DATE_OF_SALE
BMC50514I             NACTIVE = 51            SPACE   = 10
BMC50542I             FIRSTKEY= 12            FULLKEY = 854           NLEAF   = 11
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 9                                     COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'1990012500800000'  LOW2KEY= X'1982052400800000'
BMC50547I             HIGH2KEY=        LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   1         6        454       50          0          1        0        4        2     2         1       1
BMC50532I   2         5         59        0          1          0        0        6        1     2         1       1
BMC50532I   3         5         60        0          1          0        0        6        1     2         1       1
BMC50532I   4         5          0        0          0          0        0        7        1     2         1       1
BMC50532I   5         5         62        0          1          0        0        6        1     2         1       1
BMC50532I   6         5          0        0          0          0        0        7        1     2         1       1
BMC50532I   7         5         19        0          1          0        0        7        1     2         1       1
BMC50532I   8         5        100        0          1          0        0        6        1     2         1       1
BMC50532I   9         5        100        0          1          0        0        6        1     2         1       1
BMC50532I  10         5          0        0          0          0        0        7        1     2         1       1
BMC50510I INDEX       EXAMPLE2.IX1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE2.TBL1     COLUMN  SELLER_ID
BMC50514I             NACTIVE = 6             SPACE   = 1
BMC50542I             FIRSTKEY= 3             FULLKEY = 3             NLEAF   = 2
BMC50543I             LEVELS  = 2                                     PCTCLUST= 58
BMC50544I             KEYLEN  = 10                                    COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = D
BMC50546I             HIGH2KEY= X'E64BC14BD9C5C5C4'  LOW2KEY= X'E64BC14BD9C5C5C4'
BMC50547I             HIGH2KEY= W.A.REED  LOW2KEY= W.A.REED
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         6        854       50         12          2        0        7        2     2         1       1
BMC50505I BMCSTATS UPDATED IN THE DASD MANAGER PLUS DATABASE
BMC50506I STATISTICS COLUMNS UPDATED IN THE DB2 CATALOG
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

BMC Software, Inc., Confidential and Proprietary Information

Example 3: Single-Phase LOAD RESUME YES of a Partitioned Table Space

This example shows a single-phase LOAD RESUME YES job for the same table that was loaded in example 2. LOADPLUS adds new data to two of the ten partitions.

In this example, the SYSREC ddname, work file ddnames, and copy ddnames are all overridden by using the appropriate command options in addition to specifying the override ddnames in the JCL. Note the following override names:

- INDATA overrides SYSREC
- DATAWK overrides SORTOUT
- ERRWK overrides SYSERR
- IXWK overrides SYSUT1
- DISCARDS overrides SYSDISC
- CPY overrides BMCCPY
- RCY overrides BMCRCY

Assuming that the data was sorted prior to this run, the LOAD command includes the ORDER NO option. The clustering index is processed as a secondary index, which allows LOADPLUS to use two SYSUT1 data sets to process the two indexes concurrently.

The DD statements in the JCL direct LOADPLUS to create two copies of each partition.

The JCL for example 3 follows:

```
//        JOB
//EXAMPLE3 EXEC PGM=AMUUMAIN,
//             PARM='&SSID,EXAMPLE3,NEW/RESTART,,MSGLEVEL(1),AMU$OPTO'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//INDATA   DD DSN=AMU.QA.RGRTESTS(EX3DATA),DISP=SHR
//DATAWK   DD DSN=AMU.EXAMPLE3.SORTOUT,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//ERRWK    DD DSN=AMU.EXAMPLE3.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//DISCARDS DD DSN=AMU.EXAMPLE3.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
//IXWK01   DD DSN=AMU.EXAMPLE3.SYSUT1,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//IXWK02   DD DSN=AMU.EXAMPLE3.SYSUT2,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//SORTWK01 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK02 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK03 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK04 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK05 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK06 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK07 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK08 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK09 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK10 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK11 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK12 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//*
//CPY00001 DD DSN=AMU.EXAMPLE3.COPY01,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00002 DD DSN=AMU.EXAMPLE3.COPY02,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00003 DD DSN=AMU.EXAMPLE3.COPY03,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00004 DD DSN=AMU.EXAMPLE3.COPY04,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00005 DD DSN=AMU.EXAMPLE3.COPY05,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00006 DD DSN=AMU.EXAMPLE3.COPY06,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00007 DD DSN=AMU.EXAMPLE3.COPY07,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00008 DD DSN=AMU.EXAMPLE3.COPY08,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00009 DD DSN=AMU.EXAMPLE3.COPY09,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY00010 DD DSN=AMU.EXAMPLE3.COPY10,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00001 DD DSN=AMU.EXAMPLE3.RCOPY01,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00002 DD DSN=AMU.EXAMPLE3.RCOPY02,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00003 DD DSN=AMU.EXAMPLE3.RCOPY03,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00004 DD DSN=AMU.EXAMPLE3.RCOPY04,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00005 DD DSN=AMU.EXAMPLE3.RCOPY05,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00006 DD DSN=AMU.EXAMPLE3.RCOPY06,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00007 DD DSN=AMU.EXAMPLE3.RCOPY07,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00008 DD DSN=AMU.EXAMPLE3.RCOPY08,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00009 DD DSN=AMU.EXAMPLE3.RCOPY09,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY00010 DD DSN=AMU.EXAMPLE3.RCOPY10,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SYSIN    DD *
  LOAD RESUME YES
    PRELOAD LOAD
    INDDN       INDATA
    LOADDN      DATAWK
    ERRDDN      ERRWK
    WORKDDN     IXWK
    DISCARDDN   DISCARDS
    COPYDDN     CPY
    RECOVERYDDN RCY
    ORDER NO
  INTO TABLE EXAMPLE2.TBL1
      (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
      ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
      ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
      ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
      ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
      ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
      ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
      ,SELLER_ID           POSITION(62:71) CHARACTER
      )
  LOG NO
  COPY YES REGISTER NONE
/*
```

The SYSPRINT output for example 3 follows:

```
                         *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   13:31:11 ...
BMC50002I UTILITY ID = 'EXAMPLE3'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7900K,ABOVE 16M=1665744K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
```

(continued on following page)

(example 3 continued from preceding page)

```
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                 WORK                 SORTWORK             ERROR                DISCARD
BMC50470I ACTIVE   = NO                   NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    N/A                  0                    0

BMC50470I DDTYPE   = LOCPFCPY             LOCBFCPY             REMPFCPY             REMBFCPY
BMC50470I ACTIVE   = NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    0                    0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                    5                    5                    5
BMC50470I GDGEMPTY = NO                   NO                   NO                   NO
BMC50470I GDGSCRAT = NO                   NO                   NO                   NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 3 continued from preceding page)

```
BMC50102I   LOAD RESUME YES
BMC50102I     PRELOAD LOAD
BMC50102I     INDDN       INDATA
BMC50102I     LOADDN      DATAWK
BMC50102I     ERRDDN      ERRWK
BMC50102I     WORKDDN     IXWK
BMC50102I     DISCARDDN   DISCARDS
BMC50102I     COPYDDN     CPY
BMC50102I     RECOVERYDDN RCY
BMC50102I     ORDER NO
BMC50102I   INTO TABLE EXAMPLE2.TBL1
BMC50102I       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(62:71) CHARACTER
BMC50102I       )
BMC50102I   LOG NO
BMC50102I   COPY YES REGISTER NONE

BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7400K, ABOVE 16M = 1662756K, CPUS = 3
BMC51495I COMBINED OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 2, TIME = 279296
BMC51495I COMBINED OPTIMIZATION, RC = 14, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 284672
BMC51496I COMBINED ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 2
BMC50477I 3: PARTITION =   1, ROWS/KEYS = 454, I/O WAITS = 1, DDNAME = SYS00003
BMC50477I 3: PARTITION =   2, ROWS/KEYS = 59, I/O WAITS = 1, DDNAME = SYS00005
BMC50477I 4: PARTITION =   0, ROWS/KEYS = 854, I/O WAITS = 1, DDNAME = SYS00004
BMC50477I 3: PARTITION =   3, ROWS/KEYS = 60, I/O WAITS = 1, DDNAME = SYS00008
BMC50477I 3: PARTITION =   4, ROWS/KEYS = 0, I/O WAITS = 1, DDNAME = SYS00010
BMC50477I 3: PARTITION =   5, ROWS/KEYS = 62, I/O WAITS = 1, DDNAME = SYS00012
BMC50477I 3: PARTITION =   6, ROWS/KEYS = 0, I/O WAITS = 1, DDNAME = SYS00014
BMC50477I 3: PARTITION =   7, ROWS/KEYS = 19, I/O WAITS = 1, DDNAME = SYS00016
BMC50477I 3: PARTITION =   8, ROWS/KEYS = 100, I/O WAITS = 1, DDNAME = SYS00018
BMC50477I 3: PARTITION =   9, ROWS/KEYS = 100, I/O WAITS = 1, DDNAME = SYS00020
BMC50477I 3: PARTITION =  10, ROWS/KEYS = 0, I/O WAITS = 1, DDNAME = SYS00022
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  131 PHYSICAL (131 LOGICAL) RECORDS READ FROM INDATA
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A009'
BMC50477I 2: PARTITION =   9, ROWS/KEYS = 27, I/O WAITS = 8, DDNAME = SYS00024
BMC51474I LOAD STATISTICS:  27 ROWS LOADED INTO PARTITION 9
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A010'
BMC50477I 2: PARTITION =  10, ROWS/KEYS = 104, I/O WAITS = 8, DDNAME = SYS00025
BMC51474I LOAD STATISTICS:  104 ROWS LOADED INTO PARTITION 10
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:02
BMC51510I 2: SORT TASK, XBLK XFERS = 3, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = IXWK01, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = IXWK02, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50481I 3: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:03
BMC50481I 4: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:03
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A001'
BMC50477I 3: PARTITION =   1, ROWS/KEYS = 454, I/O WAITS = 7, DDNAME = SYS00030
BMC51474I BUILD STATISTICS:  454 KEYS LOADED INTO PARTITION 1
BMC50482I 4: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX1.I0001.A001'
BMC50477I 4: PARTITION =   0, ROWS/KEYS = 985, I/O WAITS = 7, DDNAME = SYS00031
BMC51476I BUILD STATISTICS:  985 KEYS LOADED INTO INDEX 'EXAMPLE2.IX1'
BMC50481I 4: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:05
BMC51510I 4: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 2
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A002'
BMC50477I 3: PARTITION =   2, ROWS/KEYS = 59, I/O WAITS = 7, DDNAME = SYS00034
BMC51474I BUILD STATISTICS:  59 KEYS LOADED INTO PARTITION 2
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A003'
BMC50477I 3: PARTITION =   3, ROWS/KEYS = 60, I/O WAITS = 7, DDNAME = SYS00037
BMC51474I BUILD STATISTICS:  60 KEYS LOADED INTO PARTITION 3
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A005'
BMC50477I 3: PARTITION =   5, ROWS/KEYS = 62, I/O WAITS = 7, DDNAME = SYS00040
BMC51474I BUILD STATISTICS:  62 KEYS LOADED INTO PARTITION 5
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A007'
BMC50477I 3: PARTITION =   7, ROWS/KEYS = 19, I/O WAITS = 7, DDNAME = SYS00043
BMC51474I BUILD STATISTICS:  19 KEYS LOADED INTO PARTITION 7
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE2.IX0', RECORD NO. = 1 OF INDATA
BMC51506E     KEY VALUE = ' 1/25/1990,1', RID-1 = X'80000201', RID-2 = X'80000401'
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A008'
BMC50477I 3: PARTITION =   8, ROWS/KEYS = 100, I/O WAITS = 7, DDNAME = SYS00046
BMC51474I BUILD STATISTICS:  100 KEYS LOADED INTO PARTITION 8
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE2.IX0', RECORD NO. = 2 OF INDATA
BMC51506E     KEY VALUE = ' 1/25/1990,2', RID-1 = X'80000202', RID-2 = X'80000402'
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A009'
```

(continued on following page)

(example 3 continued from preceding page)

```
BMC50477I 3: PARTITION =   9, ROWS/KEYS = 125, I/O WAITS = 6, DDNAME = SYS00049
BMC51474I BUILD STATISTICS:  125 KEYS LOADED INTO PARTITION 9
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A010'
BMC50477I 3: PARTITION =  10, ROWS/KEYS = 104, I/O WAITS = 7, DDNAME = SYS00052
BMC51474I BUILD STATISTICS:  104 KEYS LOADED INTO PARTITION 10
BMC50481I 3: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:14
BMC51510I 3: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 2
BMC50476I DDNAME = ERRWK, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50482I 6: EXTRACT DUP KEYS COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A009'
BMC50477I 6: PARTITION =   9, ROWS/KEYS = 2, I/O WAITS = 3, DDNAME = SYS00055
BMC50476I DDNAME = ERRWK, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 6: DISCARD TASK 1 COMPLETE.  ELAPSED TIME = 00:00:06
BMC50481I 5: DISCARD SORT 2 COMPLETE.  ELAPSED TIME = 00:00:02
BMC50482I 5: DISCARD INDEX KEYS COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX0.I0001.A009'
BMC50477I 5: PARTITION =   9, ROWS/KEYS = 0, I/O WAITS = 6, DDNAME = SYS00056
BMC50482I 5: DISCARD INDEX KEYS COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.IX1.I0001.A001'
BMC50477I 5: PARTITION =   0, ROWS/KEYS = 2, I/O WAITS = 6, DDNAME = SYS00057
BMC50481I 5: DISCARD TASK 2 COMPLETE.  ELAPSED TIME = 00:00:07
BMC51510I 5: DISCARD TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 2
BMC50481I 5: DISCARD SORT 3 COMPLETE.  ELAPSED TIME = 00:00:00
BMC50482I 5: DISCARD DATA ROWS COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A009'
BMC50477I 5: PARTITION =   9, ROWS/KEYS = 2, I/O WAITS = 5, DDNAME = SYS00059
BMC50476I DDNAME = ERRWK, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 5: DISCARD TASK 3 COMPLETE.  ELAPSED TIME = 00:00:01
BMC51510I 5: DISCARD TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC50476I DDNAME = INDATA, I/OS = 15, I/O WAITS = 3, RDB LOCK WAITS = 0
BMC50476I DDNAME = DATAWK, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51507I XBLKS = 14, XFERS = 4, EMPTY WAITS = 0, FULL WAITS =7

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR INDATA, ID# 1

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                        FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                         CONSTRAINT NAME

BMC51524E         1         1         1  DUPLICATE KEY         0:         0  EXAMPLE2.TBL1                EXAMPLE2.IX0
BMC51524E         2         2         2  DUPLICATE KEY         0:         0  EXAMPLE2.TBL1                EXAMPLE2.IX0
BMC50476I DDNAME = INDATA, I/OS = 10, I/O WAITS = 2, RDB LOCK WAITS = 0

BMC50476I DDNAME = ERRWK, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = DISCARDS, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51472I COMBINED PHASE STATISTICS:  129 ROWS SELECTED FOR SPACE 'AMUEX2DB.EX2TS', 2 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I COMBINED PHASE STATISTICS:  2 PHYSICAL (2 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 1
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 2
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 3
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 4
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 5
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 6
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 7
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 8
BMC51474I LOAD STATISTICS:  25 ROWS LOADED INTO PARTITION 9
BMC51474I LOAD STATISTICS:  104 ROWS LOADED INTO PARTITION 10
BMC51473I LOAD STATISTICS:  129 ROWS LOADED INTO TABLE 'EXAMPLE2.TBL1'
BMC51476I BUILD STATISTICS:  983 KEYS LOADED INTO INDEX 'EXAMPLE2.IX0'
BMC51476I BUILD STATISTICS:  983 KEYS LOADED INTO INDEX 'EXAMPLE2.IX1'
BMC51475I LOAD STATISTICS:  129 ROWS LOADED INTO TABLE SPACE 'AMUEX2DB.EX2TS'
BMC50370I STARTING 10 IMAGE COPY TASKS
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A001'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A006'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A008'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A009'
BMC50476I DDNAME = CPY00008, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY08'
BMC50476I DDNAME = CPY00001, I/OS = 4, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY08'
BMC50376I 12 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY01'
BMC50376I 12 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY01'
BMC50476I DDNAME = CPY00006, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 2 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY06'
BMC50376I 2 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY06'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A004'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A003'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A002'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A010'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A005'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX2DB.EX2TS.I0001.A007'
BMC50476I DDNAME = CPY00009, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 5 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY09'
BMC50376I 5 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY09'
BMC50476I DDNAME = CPY00003, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = CPY00004, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = CPY00002, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY03'
BMC50376I 2 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY04'
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY02'
BMC50476I DDNAME = CPY00010, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY02'
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 3 continued from preceding page)

```
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY03'
BMC50376I 5 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY10'
BMC50376I 5 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY10'
BMC50376I 2 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY04'
BMC50476I DDNAME = CPY00005, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY05'
BMC50376I 4 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY05'
BMC50476I DDNAME = CPY00007, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 3 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.COPY07'
BMC50376I 3 PAGES COPIED TO DATASET = 'AMU.EXAMPLE3.RCOPY07'
BMC50004I COMBINED PHASE COMPLETE.  ELAPSED TIME = 00:00:31

BMC50501I DB2 OBJECT STATISTICS
BMC50503I STATISTICS FOR TABLESPACE 'AMUEX2DB.EX2TS' NOT AVAILABLE
BMC50503I STATISTICS FOR TABLE      'EXAMPLE2.TBL1' NOT AVAILABLE
BMC50503I STATISTICS FOR INDEX      'EXAMPLE2.IX0' NOT AVAILABLE
BMC50503I STATISTICS FOR INDEX      'EXAMPLE2.IX1' NOT AVAILABLE
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

BMC Software, Inc., Confidential and Proprietary Information

Example 4: LOAD REPLACE of Three Tables

This example illustrates a LOAD REPLACE of three tables—all with multiple indexes.

This job loads three tables in a single, segmented table space. Each table has a unique clustering index and a nonunique nonclustering index. LOADPLUS replaces all rows in the table space and deletes and redefines the VSAM data sets.

The six SYSUT1*nn* files are allocated to process the six indexes on the three tables. Because the LOAD command includes the ORDER NO option, the three clustering indexes are processed as if they were nonclustering indexes. Specifying multiple SYSUT1*nn* files enables LOADPLUS to process the indexes concurrently.

Because each input record is used to create no more than one row, the LOAD command includes the UNIQUEINTO YES option for faster processing.

The COPY YES option directs LOADPLUS to make two image copies of the table space. By default, LOADPLUS registers the copies in the DB2 catalog.

RULES=BMC is specified in the options module so that the job can use extended operators, such as greater than (>) and less than (<), on the LOAD command.

The JCL for example 4 follows:

```
//        JOB
//EXAMPLE4 EXEC PGM=AMUUMAIN,
//             PARM='&SSID,EXAMPLE4,NEW/RESTART,,MSGLEVEL(1),AMU$OPTS'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//SYSREC   DD DSN=AMU.QA.RGRTESTS(EX4DATA),DISP=SHR
//SORTOUT  DD DSN=AMU.EXAMPLE4.SORTOUT,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSERR   DD DSN=AMU.EXAMPLE4.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSDISC  DD DSN=AMU.EXAMPLE4.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SYSUT101 DD DSN=AMU.EXAMPLE4.SYSUT101,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSUT102 DD DSN=AMU.EXAMPLE4.SYSUT102,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSUT103 DD DSN=AMU.EXAMPLE4.SYSUT103,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSUT104 DD DSN=AMU.EXAMPLE4.SYSUT104,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSUT105 DD DSN=AMU.EXAMPLE4.SYSUT105,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSUT106 DD DSN=AMU.EXAMPLE4.SYSUT106,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//BMCCPY   DD DSN=AMU.EXAMPLE4.COPY1,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPZ   DD DSN=AMU.EXAMPLE4.COPY2,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//SORTWK01 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK02 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK03 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK04 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK05 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK06 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK07 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK08 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK09 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK10 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK11 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK12 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//*
//SYSIN    DD *
   LOAD REPLACE ORDER NO
    UNIQUEINTO YES
     INTO TABLE EXAMPLE4.RET_EMPLS
        WHEN STATUS_IND = 'RETIRED'
          AND (EMPL_HIREDT >= '1960-01-01'
           OR EMPL_DOB < '1936-01-01')
                (EMPL_ID     POSITION(*)  INTEGER EXTERNAL(4),
                 EMPL_LNAME  POSITION(*)    CHARACTER(20),
                 EMPL_FNAME  POSITION(*)  CHARACTER(10),
                 EMPL_DOB    POSITION(*)  DATE-2E EXTERNAL,
                 EMPL_SALARY POSITION(*)  DEC EXTERNAL(10),
                 EMPL_HIREDT POSITION(*)  DATE-2E EXTERNAL,
                 STATUS_IND  POSITION(*)  CHARACTER(7)
                 )
     INTO TABLE EXAMPLE4.SAL_EMPLS
        WHEN STATUS_IND = 'SALARY'
                (EMPL_ID     POSITION(*)  INTEGER EXTERNAL(4),
                 EMPL_LNAME  POSITION(*)    CHARACTER(20),
                 EMPL_FNAME  POSITION(*)  CHARACTER(10),
                 EMPL_DOB    POSITION(*)  DATE-2E EXTERNAL,
                 EMPL_SALARY POSITION(*)  DEC EXTERNAL(10),
                 EMPL_HIREDT POSITION(*)  DATE-2E EXTERNAL,
                 STATUS_IND  POSITION(*)  CHARACTER(7)
                 )
     INTO TABLE EXAMPLE4.HRLY_EMPLS
        WHEN STATUS_IND = 'HOURLY'
                (EMPL_ID     POSITION(*)  INTEGER EXTERNAL(4),
                 EMPL_LNAME  POSITION(*)    CHARACTER(20),
                 EMPL_FNAME  POSITION(*)  CHARACTER(10),
                 EMPL_DOB    POSITION(*)  DATE-2E EXTERNAL,
                 EMPL_SALARY POSITION(*)  DEC EXTERNAL(10),
                 EMPL_HIREDT POSITION(*)  DATE-2E EXTERNAL,
                 STATUS_IND  POSITION(*)  CHARACTER(7)
                 )
   LOG NO
   COPY YES
/*
```

BMC Software, Inc., Confidential and Proprietary Information

The SYSPRINT output for example 4 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   13:35:25 ...
BMC50002I UTILITY ID = 'EXAMPLE4'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTS'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7836K,ABOVE 16M=1666252K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 4 continued from preceding page)

```
BMC50470I DDTYPE   = LOCPFCPY              LOCBFCPY              REMPFCPY              REMBFCPY
BMC50470I ACTIVE   = NO                    NO                    NO                    NO
BMC50470I IFALLOC  = USE                   USE                   USE                   USE
BMC50470I SMS      = NO                    NO                    NO                    NO
BMC50470I SMSUNIT  = NO                    NO                    NO                    NO
BMC50470I SIZEPCT  = (100,100)             (100,100)             (100,100)             (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)           (NONE,NONE)           (NONE,NONE)           (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)           (NONE,NONE)           (NONE,NONE)           (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)           (NONE,NONE)           (NONE,NONE)           (NONE,NONE)
BMC50470I THRESHLD = 0                     0                     0                     0
BMC50470I MAXEXTSZ = 0                     0                     0                     0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                     5                     5                     5
BMC50470I GDGEMPTY = NO                    NO                    NO                    NO
BMC50470I GDGSCRAT = NO                    NO                    NO                    NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD REPLACE ORDER NO
BMC50102I    UNIQUEINTO YES
BMC50102I     INTO TABLE EXAMPLE4.RET_EMPLS
BMC50102I        WHEN STATUS_IND = 'RETIRED'
BMC50102I          AND (EMPL_HIREDT >= '1960-01-01'
BMC50102I           OR EMPL_DOB < '1936-01-01')
BMC50102I                (EMPL_ID     POSITION(*)  INTEGER EXTERNAL(4),
BMC50102I                 EMPL_LNAME  POSITION(*)    CHARACTER(20),
BMC50102I                 EMPL_FNAME  POSITION(*)  CHARACTER(10),
BMC50102I                 EMPL_DOB    POSITION(*)  DATE-2E EXTERNAL,
BMC50102I                 EMPL_SALARY POSITION(*)  DEC EXTERNAL(10),
BMC50102I                 EMPL_HIREDT POSITION(*)  DATE-2E EXTERNAL,
BMC50102I                 STATUS_IND  POSITION(*)  CHARACTER(7)
BMC50102I                 )
BMC50102I     INTO TABLE EXAMPLE4.SAL_EMPLS
BMC50102I        WHEN STATUS_IND = 'SALARY'
BMC50102I                (EMPL_ID     POSITION(*)  INTEGER EXTERNAL(4),
BMC50102I                 EMPL_LNAME  POSITION(*)    CHARACTER(20),
BMC50102I                 EMPL_FNAME  POSITION(*)  CHARACTER(10),
BMC50102I                 EMPL_DOB    POSITION(*)  DATE-2E EXTERNAL,
BMC50102I                 EMPL_SALARY POSITION(*)  DEC EXTERNAL(10),
BMC50102I                 EMPL_HIREDT POSITION(*)  DATE-2E EXTERNAL,
BMC50102I                 STATUS_IND  POSITION(*)  CHARACTER(7)
BMC50102I                 )
BMC50102I     INTO TABLE EXAMPLE4.HRLY_EMPLS
BMC50102I        WHEN STATUS_IND = 'HOURLY'
BMC50102I                (EMPL_ID     POSITION(*)  INTEGER EXTERNAL(4),
BMC50102I                 EMPL_LNAME  POSITION(*)    CHARACTER(20),
BMC50102I                 EMPL_FNAME  POSITION(*)  CHARACTER(10),
BMC50102I                 EMPL_DOB    POSITION(*)  DATE-2E EXTERNAL,
BMC50102I                 EMPL_SALARY POSITION(*)  DEC EXTERNAL(10),
BMC50102I                 EMPL_HIREDT POSITION(*)  DATE-2E EXTERNAL,
BMC50102I                 STATUS_IND  POSITION(*)  CHARACTER(7)
BMC50102I                 )
BMC50102I   LOG NO
BMC50102I   COPY YES
```

(continued on following page)

(example 4 continued from preceding page)

```
BMC51422I FOR TABLE 'EXAMPLE4.RET_EMPLS', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'EMPL_ID           ' HAS POSITION(1:4)
BMC51423I FIELD 'EMPL_LNAME        ' HAS POSITION(5:24)
BMC51423I FIELD 'EMPL_FNAME        ' HAS POSITION(25:34)
BMC51423I FIELD 'EMPL_DOB          ' HAS POSITION(35:42)
BMC51423I FIELD 'EMPL_SALARY       ' HAS POSITION(43:52)
BMC51423I FIELD 'EMPL_HIREDT       ' HAS POSITION(53:60)
BMC51423I FIELD 'STATUS_IND        ' HAS POSITION(61:67)
BMC51422I FOR TABLE 'EXAMPLE4.SAL_EMPLS', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'EMPL_ID           ' HAS POSITION(1:4)
BMC51423I FIELD 'EMPL_LNAME        ' HAS POSITION(5:24)
BMC51423I FIELD 'EMPL_FNAME        ' HAS POSITION(25:34)
BMC51423I FIELD 'EMPL_DOB          ' HAS POSITION(35:42)
BMC51423I FIELD 'EMPL_SALARY       ' HAS POSITION(43:52)
BMC51423I FIELD 'EMPL_HIREDT       ' HAS POSITION(53:60)
BMC51423I FIELD 'STATUS_IND        ' HAS POSITION(61:67)
BMC51422I FOR TABLE 'EXAMPLE4.HRLY_EMPLS', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'EMPL_ID           ' HAS POSITION(1:4)
BMC51423I FIELD 'EMPL_LNAME        ' HAS POSITION(5:24)
BMC51423I FIELD 'EMPL_FNAME        ' HAS POSITION(25:34)
BMC51423I FIELD 'EMPL_DOB          ' HAS POSITION(35:42)
BMC51423I FIELD 'EMPL_SALARY       ' HAS POSITION(43:52)
BMC51423I FIELD 'EMPL_HIREDT       ' HAS POSITION(53:60)
BMC51423I FIELD 'STATUS_IND        ' HAS POSITION(61:67)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7344K, ABOVE 16M = 1651588K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 3, TIME = 48640
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 1, #READERS = 1, INDEX TASKS = 2, TIME = 54144
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 70656
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 3
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  181 PHYSICAL (181 LOGICAL) RECORDS READ FROM SYSREC
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 2: SORT TASK, XBLK XFERS = 4, EMPTY WAITS = 0, FULL WAITS = 2
BMC50481I 4: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:02
BMC50481I 3: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:02
BMC50481I 4: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:02
BMC50481I 5: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:02
BMC51510I 4: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC50481I 3: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:02
BMC50481I 5: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:02
BMC51510I 3: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC51510I 5: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSREC, I/OS = 20, I/O WAITS = 3, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT104, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT105, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT102, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT106, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT103, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 16, XFERS = 4, EMPTY WAITS = 0, FULL WAITS =5
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX4DB.EX4TS.I0001.A001' WILL REQUIRE 5 PAGES
BMC51470I PRELOAD STATISTICS:  15 ROWS SELECTED FOR TABLE 'EXAMPLE4.RET_EMPLS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX4DB.RETIX1.I0001.A001' WILL REQUIRE 4 PAGES (APPROX.)
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX4DB.RETIX2.I0001.A001' MAY REQUIRE 4 PAGES
BMC51470I PRELOAD STATISTICS:  120 ROWS SELECTED FOR TABLE 'EXAMPLE4.SAL_EMPLS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX4DB.SALIX1.I0001.A001' WILL REQUIRE 4 PAGES (APPROX.)
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX4DB.SALIX2.I0001.A001' MAY REQUIRE 5 PAGES
BMC51470I PRELOAD STATISTICS:  46 ROWS SELECTED FOR TABLE 'EXAMPLE4.HRLY_EMPLS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX4DB.HRLYIX1.I0001.A001' WILL REQUIRE 4 PAGES (APPROX.)
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX4DB.HRLYIX2.I0001.A001' MAY REQUIRE 4 PAGES
BMC51472I PRELOAD PHASE STATISTICS:  181 ROWS SELECTED FOR SPACE 'AMUEX4DB.EX4TS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:03

BMC50474I BELOW 16M = 7432K, ABOVE 16M = 1662028K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 1, #INDEX TASKS = 6
BMC50370I STARTING 1 IMAGE COPY TASKS
BMC51508I MAX INDEX TASKS = 6, INDEXES PER TASK = 1, SORTWKS PER TASK = 1, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX4DB.EX4TS' DELETED
BMC50476I DDNAME = SYSUT103, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 6: SORT COMPLETE.  ELAPSED TIME = 00:00:03
BMC50476I DDNAME = SYSUT101, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:03
BMC50476I DDNAME = SYSUT102, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 4: SORT COMPLETE.  ELAPSED TIME = 00:00:02
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:08  DSN = 'DEBACAT.DSNDBD.AMUEX4DB.EX4TS.I0001.A001'
BMC50477I 0: PARTITION =   0, ROWS/KEYS = 181, I/O WAITS = 5, DDNAME = SYS00017
BMC50476I DDNAME = SORTOUT, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
```

(continued on following page)

(example 4 continued from preceding page)

```
BMC51473I LOAD STATISTICS:  15 ROWS LOADED INTO TABLE 'EXAMPLE4.RET_EMPLS'
BMC51473I LOAD STATISTICS:  120 ROWS LOADED INTO TABLE 'EXAMPLE4.SAL_EMPLS'
BMC51473I LOAD STATISTICS:  46 ROWS LOADED INTO TABLE 'EXAMPLE4.HRLY_EMPLS'
BMC51475I LOAD STATISTICS:  181 ROWS LOADED INTO TABLE SPACE 'AMUEX4DB.EX4TS'
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:09  DSN = 'DEBACAT.DSNDBD.AMUEX4DB.SALIX1.I0001.A001'
BMC50477I 3: PARTITION =   0, ROWS/KEYS = 120, I/O WAITS = 7, DDNAME = SYS00021
BMC50482I 5: BUILD COMPLETE.  ELAPSED TIME = 00:00:09  DSN = 'DEBACAT.DSNDBD.AMUEX4DB.HRLYIX1.I0001.A001'
BMC50477I 5: PARTITION =   0, ROWS/KEYS = 46, I/O WAITS = 7, DDNAME = SYS00023
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:09  DSN = 'DEBACAT.DSNDBD.AMUEX4DB.RETIX1.I0001.A001'
BMC50477I 1: PARTITION =   0, ROWS/KEYS = 15, I/O WAITS = 7, DDNAME = SYS00022
BMC50476I DDNAME = SYSUT105, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  120 KEYS LOADED INTO INDEX 'EXAMPLE4.SALIX1'
BMC50476I DDNAME = SYSUT104, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  15 KEYS LOADED INTO INDEX 'EXAMPLE4.RETIX1'
BMC50476I DDNAME = SYSUT106, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  46 KEYS LOADED INTO INDEX 'EXAMPLE4.HRLYIX1'
BMC50482I 2: BUILD COMPLETE.  ELAPSED TIME = 00:00:05  DSN = 'DEBACAT.DSNDBD.AMUEX4DB.RETIX2.I0001.A001'
BMC50477I 2: PARTITION =   0, ROWS/KEYS = 15, I/O WAITS = 7, DDNAME = SYS00024
BMC50482I 6: BUILD COMPLETE.  ELAPSED TIME = 00:00:06  DSN = 'DEBACAT.DSNDBD.AMUEX4DB.HRLYIX2.I0001.A001'
BMC50477I 6: PARTITION =   0, ROWS/KEYS = 46, I/O WAITS = 7, DDNAME = SYS00025
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX4DB.EX4TS.I0001.A001'
BMC51476I BUILD STATISTICS:  46 KEYS LOADED INTO INDEX 'EXAMPLE4.HRLYIX2'
BMC51476I BUILD STATISTICS:  15 KEYS LOADED INTO INDEX 'EXAMPLE4.RETIX2'
BMC50476I DDNAME = BMCCPY, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 5 PAGES COPIED TO DATASET = 'AMU.EXAMPLE4.COPY1'
BMC50376I 5 PAGES COPIED TO DATASET = 'AMU.EXAMPLE4.COPY2'
BMC50482I 4: BUILD COMPLETE.  ELAPSED TIME = 00:00:04  DSN = 'DEBACAT.DSNDBD.AMUEX4DB.SALIX2.I0001.A001'
BMC50477I 4: PARTITION =   0, ROWS/KEYS = 120, I/O WAITS = 7, DDNAME = SYS00028
BMC51476I BUILD STATISTICS:  120 KEYS LOADED INTO INDEX 'EXAMPLE4.SALIX2'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:11

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX4DB.EX4TS
BMC50513I             PARTS   = 0         TABLES  = 3         SEGSIZE = 0
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   0         5        181          0          0        2        0   93    0       1    1           0    0         0     0
BMC50521I TABLE NAME                        CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE4.RET_EMPLS                  15          3         0      55      60       0
BMC50510I INDEX       EXAMPLE4.RETIX1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE4.RET_EMPLS     COLUMN  EMPL_ID
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 15            FULLKEY = 15            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 73
BMC50544I             KEYLEN  = 2                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'808E404040404040'  LOW2KEY= X'800C404040404040'
BMC50547I             HIGH2KEY=           LOW2KEY=

BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5         15       0          0          6        0        7        1     2         1       1
BMC50510I INDEX       EXAMPLE4.RETIX2  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE4.RET_EMPLS     COLUMN  EMPL_LNAME
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 12            FULLKEY = 12            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 91
BMC50544I             KEYLEN  = 32                                    COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = D
BMC50546I             HIGH2KEY= X'00E6D6D9C4E2E6D6'  LOW2KEY= X'00C2C9C4C4D3E840'
BMC50547I             HIGH2KEY=  WORDSWO  LOW2KEY=  BIDDLY
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5         15       0          0          2        0        7        1     2         1       1
BMC50521I TABLE NAME                        CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE4.SAL_EMPLS                 120          3         0      55      60       0
BMC50510I INDEX       EXAMPLE4.SALIX1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE4.SAL_EMPLS     COLUMN  EMPL_ID
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 120           FULLKEY = 120           NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 83
BMC50544I             KEYLEN  = 2                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'80B2404040404040'  LOW2KEY= X'8004404040404040'
BMC50547I             HIGH2KEY=           LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5        120       0          0         23        0        6        1     2         1       1
BMC50510I INDEX       EXAMPLE4.SALIX2  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE4.SAL_EMPLS     COLUMN  EMPL_LNAME
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 30            FULLKEY = 30            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 8
BMC50544I             KEYLEN  = 32                                    COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = D
BMC50546I             HIGH2KEY= X'00E6C8C9E3E3D3C5'  LOW2KEY= X'00C4C1D5C6D6D9E3'
BMC50547I             HIGH2KEY=  WHITTLE  LOW2KEY=  DANFORT
```

(continued on following page)

(example 4 continued from preceding page)

```
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5        120        0          0         30        0        6        1     2         1        1
BMC50521I TABLE NAME                           CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE4.HRLY_EMPLS                    46          1         0      55      20       0
BMC50510I INDEX       EXAMPLE4.HRLYIX1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE4.HRLY_EMPLS     COLUMN  EMPL_ID
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 46            FULLKEY = 46            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 2                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'80B4404040404040'  LOW2KEY= X'800A404040404040'
BMC50547I             HIGH2KEY=           LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5         46        0          0          1        0        7        1     2         1        1
BMC50510I INDEX       EXAMPLE4.HRLYIX2  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE4.HRLY_EMPLS     COLUMN  EMPL_LNAME
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 16            FULLKEY = 16            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 32                                    COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = D
BMC50546I             HIGH2KEY= X'00E9C5C9C7D3C1D9'  LOW2KEY= X'00C4C9E2C8E6C1C9'
BMC50547I             HIGH2KEY=  ZEIGLAR  LOW2KEY=  DISHWAI
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5         46        0          0          1        0        7        1     2         1        1
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

BMC Software, Inc., Confidential and Proprietary Information

Example 5: LOAD REPLACE of a Partitioned Table Space with Data Conversion

This example shows a LOAD REPLACE of a partitioned table space with four partitions and a clustering and nonclustering index. LOADPLUS replaces all existing rows.

The ORDER YES option tells LOADPLUS to sort the input data in clustering index order. Although LOADPLUS makes four copies of the table space, only two of the copies are registered as specified by the REGISTER option.

Options on the WHEN statement direct LOADPLUS to trim excess space on VARCHAR columns, and convert dates in MM/DD/YY format to the DB2 date format.

This example shows use of a conversion exit that converts a four-byte alphanumeric DEPT_NO to a four-byte numeric field. The STEPLIB DD statement includes the exit library. Member AMUXDEPT of the *HLQ*.CNTL data set contains the exit.

This job also illustrates how you can use LOADPLUS to handle problems with invalid input data. It shows use of the NULLIF (*start:end*) clause to bypass the format check in the case of a date input field and a character field. If the HIRE_DATE field contains data that is not in valid date format (***** or 00/00/00), the record is not discarded, but the invalid data is translated to nulls. Likewise, using the NOT IN operand with the PART_401K field directs LOADPLUS to convert invalid data in that field to nulls. The DEFAULTIF clause tells LOADPLUS to convert input values in the PART_401K field to user-assigned default values.

RULES=BMC is specified in the options module so that you can use extended operators, such as greater than (>) and less than (<), on the LOAD command.

Note the use of comments in this example. LOADPLUS treats as a comment any line in the LOADPLUS command syntax that has an asterisk in column one.

This example provides an image of the DDL that was used to create the DB2 objects being loaded, the JCL that was used to run the job, an image of the SYSREC data set, SYSPRINT output from the execution of the LOADPLUS utility, and output showing the loaded table.

BMC Software, Inc., Confidential and Proprietary Information

The DDL to create the DB2 objects being loaded in example 5 follows:

```
CREATE TABLESPACE EX5TS IN AMUEX5DB
    BUFFERPOOL  BP0
    CLOSE NO
    NUMPARTS 4
    USING STOGROUP AMUQASTO
                          PRIQTY 12
                          SECQTY 12
                          ERASE NO
    FREEPAGE 005
    PCTFREE 25;

CREATE TABLE EXAMPLE5.EMPLS
       (EMP_ID                DECIMAL(5) NOT NULL
       ,SSN                   CHAR(9)    NOT NULL
       ,HIRE_DATE             DATE
       ,DEPT_NO               CHAR(4)    NOT NULL
       ,JOB_CODE              SMALLINT
       ,SALARY                DECIMAL(9,2)
       ,PART_401K             CHAR(1)
       ,EMP_LNAME             VARCHAR(20) NOT NULL
       ,EMP_FNAME             VARCHAR(15) NOT NULL
       ,EMP_MNAME             VARCHAR(15)
       )
    IN AMUEX5DB.EX5TS;

CREATE INDEX EXAMPLE5.EMPLS_CLUST_IX
    ON EXAMPLE5.EMPLS (DEPT_NO)
        USING STOGROUP AMUQASTO
        CLUSTER  (PART 1 VALUES ('3999')
                 ,PART 2 VALUES ('5999')
                 ,PART 3 VALUES ('7999')
                 ,PART 4 VALUES ('9999'))
        SUBPAGES 4 ;

CREATE UNIQUE INDEX EXAMPLE5.EMPLS_NC_IX
    ON EXAMPLE5.EMPLS (EMP_ID)
        USING STOGROUP AMUQASTO;
```

The JCL for example 5 follows:

```
//        JOB
//EXAMPLE5 EXEC PGM=AMUUMAIN,
//             PARM='&SSID,EXAMPLE5,NEW/RESTART,,MSGLEVEL(1),AMU$OPTS'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//*
//SYSPRINT  DD SYSOUT=*
//SYSOUT    DD SYSOUT=*
//UTPRINT   DD SYSOUT=*
//SYSUDUMP  DD SYSOUT=*
//*
//SYSREC    DD DSN=AMU.QA.RGRTESTS.DTA163,DISP=SHR
//*
//SORTOUT   DD DSN=AMU.EXAMPLE5.SORTOUT,
//             UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSERR    DD DSN=AMU.EXAMPLE5.SYSERR,
//             UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSDISC   DD DSN=AMU.EXAMPLE5.SYSDISC,
//             UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//*
//SYSUT101  DD DSN=AMU.EXAMPLE5.SYSUT1,
//             UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSUT102  DD DSN=AMU.EXAMPLE5.SYSUT2,
//             UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//SORTWK01  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK02  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK03  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK04  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK05  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK06  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK07  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK08  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK09  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK10  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK11  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//SORTWK12  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=NEW
//*
//BMCCPY    DD DSN=AMU.EXAMPLE5.BMCCPY,
//             UNIT=WORK,SPACE=(TRK,(3,1)),DISP=(NEW,CATLG)
//BMCCPZ    DD DSN=AMU.EXAMPLE5.BMCCPZ,
//             UNIT=WORK,SPACE=(TRK,(3,1)),DISP=(NEW,CATLG)
//BMCRCY    DD DSN=AMU.EXAMPLE5.BMCRCY,
//             UNIT=WORK,SPACE=(TRK,(3,1)),DISP=(NEW,CATLG)
//BMCRCZ    DD DSN=AMU.EXAMPLE5.BMCRCZ,
//             UNIT=WORK,SPACE=(TRK,(3,1)),DISP=(NEW,CATLG)
//*
//SYSIN     DD  *
  *     LOAD OF EMPLOYEE TABLE, REPLACE ANY EXISTING ROWS,
  *     SORT ON DEPT_NO BEFORE LOADING, BUILD IMAGE COPY;
  *       CONVERTING 4-BYTE ALPHA TO 4 DIGITS VIA EXIT;
  *       CONVERTING "Y" & "N" IN PART_401K TO 1 & 0;
  *       TRIMMING EXCESS SPACES ON VARCHAR COLUMNS;
  *       CONVERTING MM/DD/YY TO DB2'S: X'YYYYMMDD';
    LOAD DATA REPLACE
      ORDER YES
      COPY YES REGISTER BMCCPY,BMCRCY
        INTO TABLE EXAMPLE5.EMPLS
          WHEN DEPT_NO > '1000'
               (PART_401K  POSITION(34)  CHAR(1)
                   NULLIF    PART_401K NOT IN ('Y','N','1','0')
                   DEFAULTIF PART_401K = 'Y' VALUE('1')
                   DEFAULTIF PART_401K = 'N' VALUE('0')
               ,EMP_ID     POSITION(1)    CHAR(5)
               ,SSN        POSITION(7)  CHAR(9)
               ,JOB_CODE   POSITION(22)  INTEGER EXTERNAL(3)
                   NULLIF  ERROR
               ,SALARY     POSITION(26)  DEC EXTERNAL(7,0)
               ,DEPT_NO    POSITION(17)  EXIT AMUXDEPT(4)
               ,EMP_LNAME  POSITION(36) CHAR(20) TRIM
               ,EMP_FNAME  POSITION(57) CHAR(15) TRIM
               ,EMP_MNAME  POSITION(73) CHAR(15) TRIM
               ,HIRE_DATE  POSITION(89) DATE-D1 EXTERNAL
                   NULLIF (89:93) = '*****'
                   NULLIF (89:96) = '00/00/00'
               )
/*
```

BMC Software, Inc., Confidential and Proprietary Information

An image of the SYSREC data set for example 5 follows:

```
00101.451846211.A00 .066.0052750.Y.HAAS              .CHRISTINE     .I          .01/01/65
00120.326759081.B01 .061.0041250.Y.THOMPSON          .MICHAEL       .L           .*****
00138.238755558.C01 .060.0038250.Y.KWAN              .SALLY         .A           .*****
00240.456679040.E01 .058.0040175.Y.GEYER             .JOHN          .B          .08/17/49
00258.873402211.D11 .055.0032250.Y.STERN             .IRVING        .F          .09/14/73
00266.433358923.D21 .056.0036170. .PULASKI           .EVA           .D          .00/00/00
00307.345270812.E11 .055.0029750.N.HENDERSON         .EILEEN        .W          .08/15/70
00380.453809376.E21 .054.0026150.N.SPENSER           .THEODORE      .Q          .06/19/80
00390.176834690.A00 .058.0046500.N.LUCCHESI          .VINCENZO      .G          .05/16/58
00400.611800345.A00 .058.0029250.N.O'CONNELL         .SEAN          .           .00/00/00
00419.112377650.C01 .055.0023800.N.QUINTANA          .DOLORES       .M          .07/28/71
00428.231544001.C01 .056.0028420.N.NICHOLLS          .HEATHER       .A          .12/15/76
00433.157919155.D11 .055.0025280.N.ADAMSON           .BRUCE         .           .02/12/72
00540.268025274.D11 .056.0022250.N.PIANKA            .ELIZABETH     .R          .10/11/77
00559.379136386.D11 .055.0024680.N.YOSHIMURA         .MASATOSHI     .J          .09/15/76
00567.480247497.D11 .054.0021340.0.SCOUTEN           .MARILYN       .S          .07/07/78
00672.591358508.D11 .054.0020450.N.WALKER            .JAMES         .H          .07/26/73
00682.602469819.D11 .053.0027740.Y.BROWN             .DAVID         .           .03/03/74
00690.713571723.D11 .053.0018270.N.JONES             .WILLIAM       .T          .04/11/66
00701.824682632.D11 .055.0029840. .LUTZ              .JENNIFER      .K          .08/29/79
00718.935793541.D11 .052.0022180. .JEFFERSON         .JAMES         .J          .11/21/69
00724.146824450.D11 .055.0028760. .MARINO            .SALVATORE     .M          .12/05/75
00830.257935361.D21 .053.0019180.N.SMITH             .DANIEL        .S          .10/13/80
00844.368046272.D21 .055.0017250.0.JOHNSON           .SYBIL         .V          .09/11/67
00852.479157183.D21 .052.0027380.N.PEREZ             .MARIA         .L          .09/29/80
00862.580268294.D21 .052.0026250.N.SCHNEIDER         .ETHEL         .R          .03/21/72
00876.691379385.D21 .055.0015340. .PARKER            .JOHN          .R          .05/05/64
00888.702480479.E11 .054.0017750. .SMITH             .PHILLIP       .X          .06/30/65
00898.813593568.E11 .042.0015900.N.SETRIGHT          .MAUDE         .F          .09/24/76
00909.924644657.E11 .048.0019950.N.MEHTA             .RAMLAL        .V          .07/30/77
00918.135755746.E11 .043.0025370.Y.LEE               .WING          .           .02/19/88
00923.246866835.E21 .052.0023840.1.GROUNOT           .JASON         .R          .05/12/86
```

The SYSPRINT output for example 5 follows:

```
                          *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   13:38:17 ...
BMC50002I UTILITY ID = 'EXAMPLE5'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTS'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7844K,ABOVE 16M=1665616K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 5 continued from preceding page)

```
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
```

(continued on following page)

(example 5 continued from preceding page)

```
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I  *     LOAD OF EMPLOYEE TABLE, REPLACE ANY EXISTING ROWS,
BMC50102I  *     SORT ON DEPT_NO BEFORE LOADING, BUILD IMAGE COPY;
BMC50102I  *       CONVERTING 4-BYTE ALPHA TO 4 DIGITS VIA EXIT;
BMC50102I  *       CONVERTING "Y" & "N" IN PART_401K TO 1 & 0;
BMC50102I  *       TRIMMING EXCESS SPACES ON VARCHAR COLUMNS;
BMC50102I  *       CONVERTING MM/DD/YY TO DB2'S: X'YYYYMMDD';
BMC50102I     LOAD DATA REPLACE
BMC50102I       ORDER YES
BMC50102I       COPY YES REGISTER BMCCPY,BMCRCY
BMC50102I         INTO TABLE EXAMPLE5.EMPLS
BMC50102I           WHEN DEPT_NO > '1000'
BMC50102I                (PART_401K  POSITION(34)  CHAR(1)
BMC50102I                    NULLIF    PART_401K NOT IN ('Y','N','1','0')
BMC50102I                    DEFAULTIF PART_401K = 'Y' VALUE('1')
BMC50102I                    DEFAULTIF PART_401K = 'N' VALUE('0')
BMC50102I                ,EMP_ID     POSITION(1)    CHAR(5)
BMC50102I                ,SSN        POSITION(7)  CHAR(9)
BMC50102I                ,JOB_CODE   POSITION(22)  INTEGER EXTERNAL(3)
BMC50102I                    NULLIF  ERROR
BMC50102I                ,SALARY     POSITION(26)  DEC EXTERNAL(7,0)
BMC50102I                ,DEPT_NO    POSITION(17)  EXIT AMUXDEPT(4)
BMC50102I                ,EMP_LNAME  POSITION(36) CHAR(20) TRIM
BMC50102I                ,EMP_FNAME  POSITION(57) CHAR(15) TRIM
BMC50102I                ,EMP_MNAME  POSITION(73) CHAR(15) TRIM
BMC50102I                ,HIRE_DATE  POSITION(89) DATE-D1 EXTERNAL
BMC50102I                    NULLIF (89:93) = '*****'
BMC50102I                    NULLIF (89:96) = '00/00/00'
BMC50102I                )

BMC51422I FOR TABLE 'EXAMPLE5.EMPLS', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'PART_401K         ' HAS POSITION(34:34)
BMC51423I FIELD 'EMP_ID            ' HAS POSITION(1:5)
BMC51423I FIELD 'SSN               ' HAS POSITION(7:15)
BMC51423I FIELD 'JOB_CODE          ' HAS POSITION(22:24)
BMC51423I FIELD 'SALARY            ' HAS POSITION(26:32)
BMC51423I FIELD 'DEPT_NO           ' HAS POSITION(17:20)
BMC51423I FIELD 'EMP_LNAME         ' HAS POSITION(36:55)
BMC51423I FIELD 'EMP_FNAME         ' HAS POSITION(57:71)
BMC51423I FIELD 'EMP_MNAME         ' HAS POSITION(73:87)
BMC51423I FIELD 'HIRE_DATE         ' HAS POSITION(89:96)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7364K, ABOVE 16M = 1653116K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 80896
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 1
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  32 PHYSICAL (32 LOGICAL) RECORDS READ FROM SYSREC
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:01
BMC51510I 2: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 1
BMC50481I 3: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:01
BMC50481I 3: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:02
BMC51510I 3: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 2
BMC50476I DDNAME = SYSREC, I/OS = 5, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 10, XFERS = 2, EMPTY WAITS = 0, FULL WAITS =3

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC, ID# 1

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD           RELATED       TABLE                       FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE             ID: RECORD     NAME                        CONSTRAINT NAME

BMC51524E         1         1         1  WHEN                 0:         0
BMC51524E         9         9         2  WHEN                 0:         0
BMC51524E        10        10         3  WHEN                 0:         0
BMC50476I DDNAME = SYSREC, I/OS = 5, I/O WAITS = 1, RDB LOCK WAITS = 0
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 5 continued from preceding page)

```
BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51471I PRELOAD STATISTICS:  4 ROWS SELECTED FOR PARTITION 1
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A001' WILL REQUIRE 3 PAGES
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A001' MAY REQUIRE 4 PAGES
BMC51471I PRELOAD STATISTICS:  12 ROWS SELECTED FOR PARTITION 2
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A002' WILL REQUIRE 3 PAGES
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A002' MAY REQUIRE 4 PAGES
BMC51471I PRELOAD STATISTICS:  11 ROWS SELECTED FOR PARTITION 3
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A003' WILL REQUIRE 3 PAGES
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A003' MAY REQUIRE 4 PAGES
BMC51471I PRELOAD STATISTICS:  2 ROWS SELECTED FOR PARTITION 4
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A004' WILL REQUIRE 3 PAGES
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A004' MAY REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRNC.I0001.A001' WILL REQUIRE 4 PAGES (APPROX.)
BMC51472I PRELOAD PHASE STATISTICS:  29 ROWS SELECTED FOR SPACE 'AMUEX5DB.EX5TS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  3 PHYSICAL (3 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:03

BMC50474I BELOW 16M = 7440K, ABOVE 16M = 1664212K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 1, #INDEX TASKS = 1
BMC50370I STARTING 1 IMAGE COPY TASKS
BMC51508I MAX INDEX TASKS = 1, INDEXES PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 4, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX5DB.EX5TS' DELETED
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRNC.I0001.A001'
BMC50477I 1: PARTITION =   0, ROWS/KEYS = 29, I/O WAITS = 7, DDNAME = SYS00018
BMC50476I DDNAME = SYSUT101, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  29 KEYS LOADED INTO INDEX 'EXAMPLE5.EMPLS_NC_IX'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A001'
BMC50477I 0: PARTITION =   1, ROWS/KEYS = 4, I/O WAITS = 7, DDNAME = SYS00021
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:04  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A001'
BMC50477I 0: PARTITION =   1, ROWS/KEYS = 4, I/O WAITS = 5, DDNAME = SYS00017
BMC51474I LOAD STATISTICS:  4 ROWS LOADED INTO PARTITION 1
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A001'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 12, I/O WAITS = 7, DDNAME = SYS00028
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 12, I/O WAITS = 5, DDNAME = SYS00025
BMC51474I LOAD STATISTICS:  12 ROWS LOADED INTO PARTITION 2
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A002'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A003'
BMC50477I 0: PARTITION =   3, ROWS/KEYS = 11, I/O WAITS = 7, DDNAME = SYS00035
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:10  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A003'
BMC50477I 0: PARTITION =   3, ROWS/KEYS = 11, I/O WAITS = 5, DDNAME = SYS00032
BMC51474I LOAD STATISTICS:  11 ROWS LOADED INTO PARTITION 3
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A003'
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EMPLSRCL.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 2, I/O WAITS = 7, DDNAME = SYS00042
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:12  DSN = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 2, I/O WAITS = 5, DDNAME = SYS00039
BMC51474I LOAD STATISTICS:  2 ROWS LOADED INTO PARTITION 4
BMC50476I DDNAME = SORTOUT, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  29 KEYS LOADED INTO INDEX 'EXAMPLE5.EMPLS_CLUST_IX'
BMC51475I LOAD STATISTICS:  29 ROWS LOADED INTO TABLE SPACE 'AMUEX5DB.EX5TS'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX5DB.EX5TS.I0001.A004'
BMC50476I DDNAME = BMCCPY, I/OS = 8, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50376I 12 PAGES COPIED TO DATASET = 'AMU.EXAMPLE5.BMCCPY'
BMC50376I 12 PAGES COPIED TO DATASET = 'AMU.EXAMPLE5.BMCCPZ'
BMC50376I 12 PAGES COPIED TO DATASET = 'AMU.EXAMPLE5.BMCRCY'
BMC50376I 12 PAGES COPIED TO DATASET = 'AMU.EXAMPLE5.BMCRCZ'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:32

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX5DB.EX5TS
BMC50513I             PARTS   = 4         TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 12            SPACE   = 4
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   1         3          4          0          0        2        0    7    0       1    1           0    0         0     0
BMC50512I   2         3         12          0          0        2        0   18    0       1    1           0    0         0     0
BMC50512I   3         3         11          0          0        2        0   16    0       1    1           0    0         0     0
BMC50512I   4         3          2          0          0        2        0    4    0       1    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE5.EMPLS                        29          4         0      52      33       0
BMC50510I INDEX       EXAMPLE5.EMPLS_CLUST_IX  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE5.EMPLS     COLUMN  DEPT_NO
BMC50514I             NACTIVE = 20            SPACE   = 4
BMC50542I             FIRSTKEY= 7             FULLKEY = 7             NLEAF   = 4
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 4                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = D
BMC50546I             HIGH2KEY= X'F9F0F0F040404040'  LOW2KEY= X'F3F0F0F040404040'
BMC50547I             HIGH2KEY= 9000      LOW2KEY= 3000
```

(continued on following page)

(example 5 continued from preceding page)

```
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   1         5          4        0          0          1        0        7        1     2         1       1
BMC50532I   2         5         12        0          1          0        0        7        1     2         1       1
BMC50532I   3         5         11        0          1          0        0        7        1     2         1       1
BMC50532I   4         5          2        0          1          0        0        7        1     2         1       1
BMC50510I INDEX       EXAMPLE5.EMPLS_NC_IX  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE5.EMPLS     COLUMN  EMP_ID
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 29            FULLKEY = 29            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 89
BMC50544I             KEYLEN  = 3                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = U
BMC50546I             HIGH2KEY= X'F009184040404040'  LOW2KEY= X'F001384040404040'
BMC50547I             HIGH2KEY=          LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5         29        0          7          1        0        7        1     2         1       1
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

The output that shows the loaded table for example 5 follows:

```
PAGE    1
 ***INPUT STATEMENT:
    SELECT EMP_ID,SSN,HIRE_DATE,DEPT_NO,JOB_CODE,SALARY,PART_401K,
      EMP_LNAME FROM EXAMPLE5.EMPLS ;
   +-------------------------------------------------------------------------------------------------------+
   |  EMP_ID   |    SSN    | HIRE_DATE  |DEPT_NO |JOB_CODE |  SALARY     | PART_401K |      EMP_LNAME       |
   +-------------------------------------------------------------------------------------------------------+
 1_|       120 | 326759081 | ?          | 2000   |      61 |    41250.00 | 1         | THOMPSON             |
 2_|       138 | 238755558 | ?          | 3000   |      60 |    38250.00 | 1         | KWAN                 |
 3_|       419 | 112377650 | 07/28/1971 | 3000   |      55 |    23800.00 | 0         | QUINTANA             |
 4_|       428 | 231544001 | 12/15/1976 | 3000   |      56 |    28420.00 | 0         | NICHOLLS             |
 5_|       240 | 456679040 | 08/17/1949 | 4000   |      58 |    40175.00 | 1         | GEYER                |
 6_|       258 | 873402211 | 09/14/1973 | 5000   |      55 |    32250.00 | 1         | STERN                |
 7_|       433 | 157919155 | 02/12/1972 | 5000   |      55 |    25280.00 | 0         | ADAMSON              |
 8_|       540 | 268025274 | 10/11/1977 | 5000   |      56 |    22250.00 | 0         | PIANKA               |
 9_|       559 | 379136386 | 09/15/1976 | 5000   |      55 |    24680.00 | 0         | YOSHIMURA            |
10_|       567 | 480247497 | 07/07/1978 | 5000   |      54 |    21340.00 | 0         | SCOUTEN              |
11_|       672 | 591358508 | 07/26/1973 | 5000   |      54 |    20450.00 | 0         | WALKER               |
12_|       682 | 602469819 | 03/03/1974 | 5000   |      53 |    27740.00 | 1         | BROWN                |
13_|       690 | 713571723 | 04/11/1966 | 5000   |      53 |    18270.00 | 0         | JONES                |
14_|       701 | 824682632 | 08/29/1979 | 5000   |      55 |    29840.00 | ?         | LUTZ                 |
15_|       718 | 935793541 | 11/21/1969 | 5000   |      52 |    22180.00 | ?         | JEFFERSON            |
16_|       724 | 146824450 | 12/05/1975 | 5000   |      55 |    28760.00 | ?         | MARINO               |
17_|       266 | 433358923 | ?          | 6000   |      56 |    36170.00 | ?         | PULASKI              |
18_|       830 | 257935361 | 10/13/1980 | 6000   |      53 |    19180.00 | 0         | SMITH                |
19_|       844 | 368046272 | 09/11/1967 | 6000   |      55 |    17250.00 | 0         | JOHNSON              |
20_|       852 | 479157183 | 09/29/1980 | 6000   |      52 |    27380.00 | 0         | PEREZ                |
21_|       862 | 580268294 | 03/21/1972 | 6000   |      52 |    26250.00 | 0         | SCHNEIDER            |
22_|       876 | 691379385 | 05/05/1964 | 6000   |      55 |    15340.00 | ?         | PARKER               |
23_|       307 | 345270812 | 08/15/1970 | 7000   |      55 |    29750.00 | 0         | HENDERSON            |
24_|       888 | 702480479 | 06/30/1965 | 7000   |      54 |    17750.00 | ?         | SMITH                |
25_|       898 | 813593568 | 09/24/1976 | 7000   |      42 |    15900.00 | 0         | SETRIGHT             |
26_|       909 | 924644657 | 07/30/1977 | 7000   |      48 |    19950.00 | 0         | MEHTA                |
27_|       918 | 135755746 | 02/19/1988 | 7000   |      43 |    25370.00 | 1         | LEE                  |
28_|       380 | 453809376 | 06/19/1980 | 9000   |      54 |    26150.00 | 0         | SPENSER              |
29_|       923 | 246866835 | 05/12/1986 | 9000   |      52 |    23840.00 | 1         | GROUNOT              |
   +-------------------------------------------------------------------------------------------------------+
```

Example 6: LOAD REPLACE of a Simple Table Space with Multiple INTO Statements

This example illustrates a LOAD REPLACE of a simple table space that contains one table and one unique, nonclustering index. This example uses multiple INTO statements to create a large test table from a small input file and to override input data values.

This job uses an input data set that contains 213,000 records and loads a table that contains 852,000 rows.

This LOADPLUS job creates a unique index key by assigning a unique value in KEYFLD1 in each INTO statement. KEYFLD2 is unique for each record of the input file and has values 1 through 213,000.

This example illustrates methods for putting specific values in columns as needed for testing. In this example, the DEFAULTIF option is used to create multiple values in the same column. Different VALUE options are used for the same column in different INTO statements to create multiple values for the same column.

The UNIQUEINTO option is not specified in this job, causing LOADPLUS to use the default, UNIQUEINTO NO. Specifying UNIQUEINTO YES with this job would cause LOADPLUS to stop processing each record after the first INTO statement—thus creating only 213,000 rows instead of 852,000.

RULES=BMC is specified in the options module so that the job can use extended operators, such as greater than (>) and less than (<), on the LOAD command.

Because the LOAD command does not include the COPY YES option, LOADPLUS does not make a copy. After loading the data, LOADPLUS places the table space in COPY pending status.

The JCL for example 6 follows:

```
//        JOB
//EXAMPLE6 EXEC PGM=AMUUMAIN,
//             PARM='&SSID,EXAMPLE6,NEW,,MSGLEVEL(1)'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SORTOUT  DD DSN=AMU.EXAMPLE6.SORTOUT,
//            DISP=(MOD,CATLG),UNIT=CART,
//            VOL=(,,,10)
//SYSUT101 DD DSN=AMU.EXAMPLE6.SYSUT1,
//            UNIT=WORK,SPACE=(CYL,(10,10)),DISP=(MOD,CATLG)
//SORTWK01 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK02 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK03 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK04 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK05 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK06 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK07 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK08 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK09 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK10 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK11 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SORTWK12 DD UNIT=WORK,SPACE=(CYL,(25,10)),DISP=NEW
//SYSREC   DD DSN=AMU.QA.RGRTESTS.DTA111,DISP=SHR
//SYSERR   DD DSN=AMU.EXAMPLE6.SYSERR,
//            DISP=(MOD,CATLG),UNIT=WORK,
//            SPACE=(CYL,(5,1))
//SYSDISC  DD DSN=AMU.EXAMPLE6.SYSDISC,
//            DISP=(MOD,CATLG),UNIT=WORK,
//            SPACE=(CYL,(5,1))
//SYSIN    DD *
   LOAD ORDER NO
     REPLACE
     REDEFINE YES
   INTO TABLE EXAMPLE6.TBL1
       (KEYFLD1_CHAR        VALUE('01')
       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
       ,FLD5_SMALLINT       VALUE(1)
       ,FLD6_INT            POSITION(4) INTEGER
          DEFAULTIF KEYFLD2_CHAR < '001000' VALUE(2311)
          DEFAULTIF KEYFLD2_CHAR >= '001000' VALUE(-500)
       ,FLD10_DEC           VALUE(123.4567)
       ,FLD11_CHAR          VALUE('B')
       ,FLD12_CHAR          VALUE('C')
       )
   INTO TABLE EXAMPLE6.TBL1
       (KEYFLD1_CHAR        VALUE('02')
       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
       ,FLD5_SMALLINT       VALUE(1)
       ,FLD6_INT            POSITION(4) INTEGER
       DEFAULTIF KEYFLD2_CHAR < '002000' VALUE(2311)
       DEFAULTIF KEYFLD2_CHAR >= '002000' VALUE(-500)
       ,FLD10_DEC           VALUE(123)
       ,FLD11_CHAR          VALUE('H')
       ,FLD12_CHAR          VALUE('J')
       )
   INTO TABLE EXAMPLE6.TBL1
       (KEYFLD1_CHAR        VALUE('03')
       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
       ,FLD5_SMALLINT       VALUE(1)
       ,FLD6_INT            POSITION(4) INTEGER
       DEFAULTIF KEYFLD2_CHAR < '050000' VALUE(1406)
       DEFAULTIF KEYFLD2_CHAR >= '050000' VALUE(-409)
       ,FLD10_DEC           VALUE(-3.45)
       ,FLD11_CHAR          VALUE('D')
       ,FLD12_CHAR          VALUE('E')
       )
   INTO TABLE EXAMPLE6.TBL1
       (KEYFLD1_CHAR        VALUE('04')
       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
       ,FLD5_SMALLINT       VALUE(1)
       ,FLD6_INT            POSITION(4) INTEGER
       DEFAULTIF KEYFLD2_CHAR < '002000' VALUE(1405)
       DEFAULTIF KEYFLD2_CHAR >= '002000' VALUE(3409)
       ,FLD10_DEC           VALUE(-.4567)
       ,FLD11_CHAR          VALUE('A')
       ,FLD12_CHAR          VALUE('Z')
       )
/*
```

BMC Software, Inc., Confidential and Proprietary Information

The SYSPRINT output for example 6 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                  COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   13:55:39 ...
BMC50002I UTILITY ID = 'EXAMPLE6'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTS'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7848K,ABOVE 16M=1665840K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0
```

(continued on following page)

(example 6 continued from preceding page)

```
BMC50470I DDTYPE   = LOCPFCPY             LOCBFCPY             REMPFCPY             REMBFCPY
BMC50470I ACTIVE   = NO                    NO                   NO                   NO
BMC50470I IFALLOC  = USE                   USE                  USE                  USE
BMC50470I SMS      = NO                    NO                   NO                   NO
BMC50470I SMSUNIT  = NO                    NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)             (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)           (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)           (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)           (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                     0                    0                    0
BMC50470I MAXEXTSZ = 0                     0                    0                    0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                     5                    5                    5
BMC50470I GDGEMPTY = NO                    NO                   NO                   NO
BMC50470I GDGSCRAT = NO                    NO                   NO                   NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD ORDER NO
BMC50102I     REPLACE
BMC50102I     REDEFINE YES
BMC50102I   INTO TABLE EXAMPLE6.TBL1
BMC50102I       (KEYFLD1_CHAR        VALUE('01')
BMC50102I       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
BMC50102I       ,FLD5_SMALLINT       VALUE(1)
BMC50102I       ,FLD6_INT            POSITION(4) INTEGER
BMC50102I          DEFAULTIF KEYFLD2_CHAR < '001000' VALUE(2311)
BMC50102I          DEFAULTIF KEYFLD2_CHAR >= '001000' VALUE(-500)
BMC50102I       ,FLD10_DEC           VALUE(123.4567)
BMC50102I       ,FLD11_CHAR          VALUE('B')
BMC50102I       ,FLD12_CHAR          VALUE('C')
BMC50102I       )
BMC50102I   INTO TABLE EXAMPLE6.TBL1
BMC50102I       (KEYFLD1_CHAR        VALUE('02')
BMC50102I       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
BMC50102I       ,FLD5_SMALLINT       VALUE(1)
BMC50102I       ,FLD6_INT            POSITION(4) INTEGER
BMC50102I       DEFAULTIF KEYFLD2_CHAR < '002000' VALUE(2311)
BMC50102I       DEFAULTIF KEYFLD2_CHAR >= '002000' VALUE(-500)
BMC50102I       ,FLD10_DEC           VALUE(123)
BMC50102I       ,FLD11_CHAR          VALUE('H')
BMC50102I       ,FLD12_CHAR          VALUE('J')
BMC50102I       )
BMC50102I   INTO TABLE EXAMPLE6.TBL1
BMC50102I       (KEYFLD1_CHAR        VALUE('03')
BMC50102I       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
BMC50102I       ,FLD5_SMALLINT       VALUE(1)
BMC50102I       ,FLD6_INT            POSITION(4) INTEGER
BMC50102I       DEFAULTIF KEYFLD2_CHAR < '050000' VALUE(1406)
BMC50102I       DEFAULTIF KEYFLD2_CHAR >= '050000' VALUE(-409)
BMC50102I       ,FLD10_DEC           VALUE(-3.45)
BMC50102I       ,FLD11_CHAR          VALUE('D')
BMC50102I       ,FLD12_CHAR          VALUE('E')
BMC50102I       )
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 6 continued from preceding page)

```
BMC50102I   INTO TABLE EXAMPLE6.TBL1
BMC50102I       (KEYFLD1_CHAR        VALUE('04')
BMC50102I       ,KEYFLD2_CHAR        POSITION(75) CHAR(6)
BMC50102I       ,FLD5_SMALLINT       VALUE(1)
BMC50102I       ,FLD6_INT            POSITION(4) INTEGER
BMC50102I       DEFAULTIF KEYFLD2_CHAR < '002000' VALUE(1405)
BMC50102I       DEFAULTIF KEYFLD2_CHAR >= '002000' VALUE(3409)
BMC50102I       ,FLD10_DEC           VALUE(-.4567)
BMC50102I       ,FLD11_CHAR          VALUE('A')
BMC50102I       ,FLD12_CHAR          VALUE('Z')
BMC50102I       )

BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD3_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD4_TSTAMP       '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD7_FLOAT        '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD8_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD9_SMALLINT     '.  VALUE IS DEFAULTED
BMC51422I FOR TABLE 'EXAMPLE6.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'KEYFLD2_CHAR      ' HAS POSITION(75:80)
BMC51423I FIELD 'FLD6_INT          ' HAS POSITION(4:7)
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD3_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD4_TSTAMP       '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD7_FLOAT        '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD8_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD9_SMALLINT     '.  VALUE IS DEFAULTED
BMC51422I FOR TABLE 'EXAMPLE6.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'KEYFLD2_CHAR      ' HAS POSITION(75:80)
BMC51423I FIELD 'FLD6_INT          ' HAS POSITION(4:7)
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD3_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD4_TSTAMP       '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD7_FLOAT        '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD8_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD9_SMALLINT     '.  VALUE IS DEFAULTED
BMC51422I FOR TABLE 'EXAMPLE6.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'KEYFLD2_CHAR      ' HAS POSITION(75:80)
BMC51423I FIELD 'FLD6_INT          ' HAS POSITION(4:7)
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD3_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD4_TSTAMP       '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD7_FLOAT        '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD8_DATE         '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'FLD9_SMALLINT     '.  VALUE IS DEFAULTED
BMC51422I FOR TABLE 'EXAMPLE6.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'KEYFLD2_CHAR      ' HAS POSITION(75:80)
BMC51423I FIELD 'FLD6_INT          ' HAS POSITION(4:7)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:02

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SORTOUT'
BMC50474I BELOW 16M = 7372K, ABOVE 16M = 1653600K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 41472
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 1
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:29
BMC51510I 1: READ TASK, XBLK XFERS = 873, EMPTY WAITS = 3, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  213000 PHYSICAL (213000 LOGICAL) RECORDS READ FROM SYSREC
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:29
BMC51510I 2: SORT TASK, XBLK XFERS = 1147, EMPTY WAITS = 8, FULL WAITS = 866
BMC50481I 3: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:33
BMC50481I 3: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:38
BMC51510I 3: INDEX TASK, XBLK XFERS = 274, EMPTY WAITS = 0, FULL WAITS = 266
BMC50476I DDNAME = SYSREC, I/OS = 728, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT, I/OS = 1862, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 489, I/O WAITS = 12, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 10, XFERS = 1147, EMPTY WAITS = 11, FULL WAITS =1132
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX6DB.EX6TS.I0001.A001' WILL REQUIRE 14203 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX6DB.TBL1IX.I0001.A001' WILL REQUIRE 4022 PAGES (APPROX.)
BMC51472I PRELOAD PHASE STATISTICS:  852000 ROWS SELECTED FOR SPACE 'AMUEX6DB.EX6TS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:39

BMC50474I BELOW 16M = 7460K, ABOVE 16M = 1664464K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 0, #INDEX TASKS = 1
BMC51508I MAX INDEX TASKS = 1, INDEXES PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX6DB.EX6TS' DELETED
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:16  DSN = 'DEBACAT.DSNDBD.AMUEX6DB.EX6TS.I0001.A001'
BMC50477I 0: PARTITION =   0, ROWS/KEYS = 852000, I/O WAITS = 63, DDNAME = SYS00006
BMC50476I DDNAME = SORTOUT, I/OS = 1863, I/O WAITS = 87, RDB LOCK WAITS = 0
BMC51475I LOAD STATISTICS:  852000 ROWS LOADED INTO TABLE SPACE 'AMUEX6DB.EX6TS'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:46  DSN = 'DEBACAT.DSNDBD.AMUEX6DB.TBL1IX.I0001.A001'
BMC50477I 1: PARTITION =   0, ROWS/KEYS = 852000, I/O WAITS = 20, DDNAME = SYS00007
BMC50476I DDNAME = SYSUT101, I/OS = 490, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  852000 KEYS LOADED INTO INDEX 'EXAMPLE6.TBL1IX'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:47
```

(continued on following page)

(example 6 continued from preceding page)

```
BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX6DB.EX6TS
BMC50513I             PARTS   = 0         TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 14203         SPACE   = 1365
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   0     14203     852000          0          0        3        0   90    0    1365    3           0    0         0     0
BMC50521I TABLE NAME                        CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE6.TBL1                    852000      14200         0      53       99       0
BMC50510I INDEX       EXAMPLE6.TBL1IX  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE6.TBL1     COLUMN  KEYFLD1_CHAR
BMC50514I             NACTIVE = 4023          SPACE   = 336
BMC50542I             FIRSTKEY= 4             FULLKEY = 852000        NLEAF   = 4000
BMC50543I             LEVELS  = 3                                     PCTCLUST= 99
BMC50544I             KEYLEN  = 10                                    COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = U
BMC50546I             HIGH2KEY= X'F0F3F0F0F0F0F0F4'  LOW2KEY= X'F0F2F0F0F0F0F0F4'
BMC50547I             HIGH2KEY= 03000004  LOW2KEY= 02000004
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0      4023     852000        0          3          5        0     1643     4000     3       336      50
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

BMC Software, Inc., Confidential and Proprietary Information

Example 7: LOAD RESUME NO of a Simple Table Space Using IGNORE and REPORT Options

In this example, LOADPLUS loads data into a simple table space that contains only one table with one unique index.

BMCSORT dynamically allocates sort work files. The JCL does not specify any SORTWK DD statements and LOADPLUS dynamic allocation for sort work data sets is not active.

This example uses the DISCARDS IGNORE option to ignore all duplicate key errors for processing and to keep LOADPLUS from writing errors to the SYSERR and SYSDISC data sets. The LOAD command includes the DISCARDS REPORT NO WHEN option. Using this option reduces the number of lines in the SYSPRINT because WHEN discards are not reported in the LOADPLUS error summary report. Not reporting the discarded WHEN records is beneficial if the input data set (SYSREC) contains large amounts of data that will be discarded. Note messages BMC51462I and BMC51463I regarding the DISCARDS IGNORE and REPORT options.

RESUME NO indicates that the table space being loaded is empty. Specifying ENUMROWS 50 estimates that LOADPLUS will load 50 rows.

The 3390 device type will be used for the sort work file that BMCSORT dynamically allocates (as specified by SORTDEVT 3390). Specifying SORTNUM 2 provides the following benefits:

- tells BMCSORT to allocate the sort work files that it needs, up to 32 per sort task, regardless of the values specified in the BMCSORT DYNALOC installation option
- tells LOADPLUS that it can use multiple sort tasks if necessary

SKIPFIELDS YES enables processing of input record fields that do not have corresponding columns in the table. In this case, NULL1 is used in NULLIF condition processing.

The JCL for example 7 follows:

```
//        JOB
//EXAMPLE7 EXEC PGM=AMUUMAIN,REGION=4M,
//             PARM='&SSID,EXAMPLE7,NEW,,MSGLEVEL(1),AMU$OPTO'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSREC   DD DSN=AMU.QA.RGRTESTS(EX7DATA),
//            DISP=SHR
//*
//SORTOUT  DD DSN=AMU.EXAMPLE7.SORTOUT,
//            SPACE=(CYL,(20,10)),UNIT=WORK,DISP=(MOD,CATLG,DELETE)
//SYSUT101 DD DSN=AMU.EXAMPLE7.SYSUT1,
//            SPACE=(CYL,(20,10)),UNIT=WORK,DISP=(MOD,CATLG,DELETE)
//SYSERR   DD DSN=AMU.EXAMPLE7.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,DELETE)
//SYSDISC  DD DSN=AMU.EXAMPLE7.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,DELETE)
//SYSIN    DD *
     LOAD
     RESUME NO
     DISCARDS 50 IGNORE DUPKEY REPORT NO WHEN
     ENUMROWS 50
     SORTDEVT 3390 SORTNUM 2
     COPY NO
     REDEFINE YES
     SKIPFIELDS YES
     INTO TABLE EXAMPLE7.TBL1
        WHEN RECNO <> '90000'
                (RECNO       POSITION(*)     CHARACTER
                ,INPUT_A     POSITION(*)     CHARACTER
                ,NULL1       POSITION(9)     CHARACTER(1)
                ,INPUT_3     POSITION(10:12) CHARACTER
                   NULLIF NULL1='?'
                ,INPUT_7     POSITION(13:19) CHARACTER
                ,INPUT_10    POSITION(30:39) CHARACTER
                ,INPUT_12    POSITION(25:36) CHARACTER
                ,MLT_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,-1)
                ,DIV_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,1)
                ,MLT_INT2    POSITION(13)    DECIMAL EXTERNAL (7,-2)
                ,DIV_INT2    POSITION(13)    DECIMAL EXTERNAL (7,2)
                ,MLT_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,-3)
                ,DIV_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,3)
                ,MLT_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,-40)
                ,DIV_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,40)
      )
/*
```

The SYSPRINT output for example 7 follows:

```
                          *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
               COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   14:35:16 ...
BMC50002I UTILITY ID = 'EXAMPLE7'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7876K,ABOVE 16M=1665928K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
```

(continued on following page)

(example 7 continued from preceding page)

```
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME
```

(continued on following page)

(example 7 continued from preceding page)

```
BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I     LOAD
BMC50102I     RESUME NO
BMC50102I     DISCARDS 50 IGNORE DUPKEY REPORT NO WHEN
BMC50102I     ENUMROWS 50
BMC50102I     SORTDEVT 3390 SORTNUM 2
BMC50102I     COPY NO
BMC50102I     REDEFINE YES
BMC50102I     SKIPFIELDS YES
BMC50102I     INTO TABLE EXAMPLE7.TBL1
BMC50102I        WHEN RECNO <> '90000'
BMC50102I                (RECNO       POSITION(*)     CHARACTER
BMC50102I                ,INPUT_A     POSITION(*)     CHARACTER
BMC50102I                ,NULL1       POSITION(9)     CHARACTER(1)
BMC50102I                ,INPUT_3     POSITION(10:12) CHARACTER
BMC50102I                   NULLIF NULL1='?'
BMC50102I                ,INPUT_7     POSITION(13:19) CHARACTER
BMC50102I                ,INPUT_10    POSITION(30:39) CHARACTER
BMC50102I                ,INPUT_12    POSITION(25:36) CHARACTER
BMC50102I                ,MLT_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,-1)
BMC50102I                ,DIV_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,1)
BMC50102I                ,MLT_INT2    POSITION(13)    DECIMAL EXTERNAL (7,-2)
BMC50102I                ,DIV_INT2    POSITION(13)    DECIMAL EXTERNAL (7,2)
BMC50102I                ,MLT_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,-3)
BMC50102I                ,DIV_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,3)
BMC50102I                ,MLT_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,-40)
BMC50102I                ,DIV_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,40)
BMC50102I       )

BMC51422I FOR TABLE 'EXAMPLE7.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'RECNO             ' HAS POSITION(1:5)
BMC51423I FIELD 'INPUT_A           ' HAS POSITION(6:8)
BMC51423I FIELD 'NULL1             ' HAS POSITION(9:9)
BMC51423I FIELD 'MLT_SMINT1        ' HAS POSITION(10:12)
BMC51423I FIELD 'DIV_SMINT1        ' HAS POSITION(10:12)
BMC51423I FIELD 'MLT_INT2          ' HAS POSITION(13:19)
BMC51423I FIELD 'DIV_INT2          ' HAS POSITION(13:19)
BMC51423I FIELD 'MLT_DEC3          ' HAS POSITION(25:36)
BMC51423I FIELD 'DIV_DEC3          ' HAS POSITION(25:36)
BMC51423I FIELD 'MLT_FLOAT40       ' HAS POSITION(30:39)
BMC51423I FIELD 'DIV_FLOAT40       ' HAS POSITION(30:39)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SORTOUT'
BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SYSERR'
BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SYSDISC'
BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SYSUT101'
BMC50394I UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'
BMC50474I BELOW 16M = 7388K, ABOVE 16M = 1653688K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 46080
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 1
BMC50477I 4: PARTITION =   0, ROWS/KEYS = 0, I/O WAITS = 2, DDNAME = SYS00001
BMC51501E INVALID NUMERIC DATA, COLUMN = 'INPUT_7', RECORD NO. 22 OF SYSREC, DATA = C'65S÷220'
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  50 PHYSICAL (50 LOGICAL) RECORDS READ FROM SYSREC
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 2: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC50481I 3: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:01
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE7.TBL1IX', RECORD NO. = 5 OF SYSREC
BMC51506E     KEY VALUE = ''14800'', RID-1 = X'00000204', RID-2 = X'00000204'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE7.TBL1IX', RECORD NO. = 10 OF SYSREC
BMC51506E     KEY VALUE = ''14800'', RID-1 = X'00000204', RID-2 = X'00000209'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE7.TBL1IX', RECORD NO. = 9 OF SYSREC
BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000208', RID-2 = X'00000208'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE7.TBL1IX', RECORD NO. = 25 OF SYSREC
BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000208', RID-2 = X'00000215'
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 7 continued from preceding page)

```
 BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE7.TBL1IX', RECORD NO. = 32 OF SYSREC
 BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000208', RID-2 = X'0000021B'
 BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLE7.TBL1IX', RECORD NO. = 41 OF SYSREC
 BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000208', RID-2 = X'00000307'
 BMC50481I 3: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:01
 BMC51510I 3: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
 BMC50476I DDNAME = SYSREC, I/OS = 6, I/O WAITS = 2, RDB LOCK WAITS = 0
 BMC50476I DDNAME = SORTOUT, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
 BMC50476I DDNAME = SYSUT101, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
 BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
 BMC51507I XBLKS = 10, XFERS = 2, EMPTY WAITS = 0, FULL WAITS =3

 BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC, ID# 1

 BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                        FIELD, INDEX, OR
 BMC51523I  RECORD     RECORD    RECORD      TYPE             ID: RECORD     NAME                         CONSTRAINT NAME

 BMC51524E        22        22         3  CONVERSION            0:         1  EXAMPLE7.TBL1                INPUT_7
 BMC50476I DDNAME = SYSREC, I/OS = 6, I/O WAITS = 1, RDB LOCK WAITS = 0

 BMC51463I ONE OR MORE DISCARDS NOT REPORTED DUE TO DISCARD REPORT OPTION
 BMC51462I ONE OR MORE DISCARDS IGNORED DUE TO DISCARD IGNORE OPTION
 BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
 BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
 BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX7DB.EX7TS.I0001.A001' WILL REQUIRE 4 PAGES
 BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX7DB.TBL1IX.I0001.A001' WILL REQUIRE 4 PAGES (APPROX.)
 BMC51472I PRELOAD PHASE STATISTICS:  38 ROWS SELECTED FOR SPACE 'AMUEX7DB.EX7TS', 7 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
 BMC51479I PRELOAD PHASE STATISTICS:  6 PHYSICAL (6 LOGICAL) RECORDS DISCARDED TO SYSDISC
 BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:04

 BMC50474I BELOW 16M = 7480K, ABOVE 16M = 1664508K, CPUS = 3
 BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 0, #INDEX TASKS = 1
 BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
 BMC51508I MAX INDEX TASKS = 1, INDEXES PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
 BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
 BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX7DB.EX7TS.I0001.A001'
 BMC50477I 0: PARTITION =   0, ROWS/KEYS = 38, I/O WAITS = 5, DDNAME = SYS00009
 BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX7DB.TBL1IX.I0001.A001'
 BMC50477I 1: PARTITION =   0, ROWS/KEYS = 38, I/O WAITS = 7, DDNAME = SYS00010
 BMC50476I DDNAME = SYSUT101, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
 BMC51476I BUILD STATISTICS:  38 KEYS LOADED INTO INDEX 'EXAMPLE7.TBL1IX'
 BMC50476I DDNAME = SORTOUT, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
 BMC51475I LOAD STATISTICS:  38 ROWS LOADED INTO TABLE SPACE 'AMUEX7DB.EX7TS'
 BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:04

 BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
 BMC50501I DB2 OBJECT STATISTICS
 BMC50510I TABLESPACE  AMUEX7DB.EX7TS
 BMC50513I             PARTS   = 0         TABLES  = 1         SEGSIZE = 0
 BMC50514I             NACTIVE = 4               SPACE   = 255
 BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
 BMC50512I   0         4         38          0          0        2        0   39    0     255    2           0    0         0     0
 BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
 BMC50522I EXAMPLE7.TBL1                         38          2         0      74      50       0
 BMC50510I INDEX       EXAMPLE7.TBL1IX  (TYPE 2)
 BMC50541I             ON TABLE  EXAMPLE7.TBL1     COLUMN  RECNO
 BMC50514I             NACTIVE = 5               SPACE   = 1
 BMC50542I             FIRSTKEY= 38              FULLKEY = 38              NLEAF   = 1
 BMC50543I             LEVELS  = 2                                         PCTCLUST= 89
 BMC50544I             KEYLEN  = 5                                         COLCOUNT= 1
 BMC50545I             SUBPAGE = 1               CLUSTER = Y               UNIQUE  = U
 BMC50546I             HIGH2KEY= X'F8F4F8F4F0404040'  LOW2KEY= X'F1F1F8F1F0404040'
 BMC50547I             HIGH2KEY= 84840     LOW2KEY= 11810
 BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
 BMC50532I   0         5         38       0          0          5        0        7        1     2         1       1
0BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

Example 8: LOAD REPLACE of a Partitioned Table Space Using ANALYZE PAUSE

This example shows a LOAD REPLACE of a 10-part table in a compressed table space with a unique clustering index and a nonunique secondary index.

To obtain the optimal number and size of the data sets to use, the job is run first with ANALYZE PAUSE and ENUMROWS. SYSPRINT A (page 5-57) shows the report that LOADPLUS generates. The numbers from the ANALYZE report can be used to modify the space allocations on the work file DD statements. The job is run again, with the NEW/RESTART option, to perform the actual load. SYSPRINT B (page 5-60) shows this output. There is no change to the JCL.

Seven of ten partitions are loaded in this example. Individual INTO PART commands are not specified because processing multiple INTO PART commands is less efficient and offers no benefit on a REPLACE run.

The JCL contains multiple SYSREC and SORTOUT DD statements, enabling LOADPLUS to multitask in both the PRELOAD and LOAD phases. Only the secondary index is processed by using a SYSUT1 work file, so the JCL includes only one SYSUT1.

Because the LOAD command includes the ORDER YES option, LOADPLUS sorts the data rows and clustering index keys. Because the LOAD command does not include the UNIQUECHECK NO option, LOADPLUS defaults to UNIQUECHECK YES, which tells LOADPLUS to check the clustering index for uniqueness.

COPY YES REGISTER NONE creates unregistered copies (DSN1COPYs). The copy DD statements in the JCL direct LOADPLUS to create a single copy of each partition, with each copy in a separate data set.

Specifying CENTURY(1900,1999) overrides the default of (1950,2049) that was specified in the installation option. This override loads the proper century value for records that contain two-digit years for dates earlier than 1950.

This example illustrates how LOADPLUS automatically provides partition parallelism to achieve optimal performance. Message BMC51508I shows that LOADPLUS performs five data tasks concurrently, with up to two parts per task.

LOADPLUS updates statistics in both the DB2 catalog and the DASD MANAGER PLUS statistics tables.

BMC Software, Inc., Confidential and Proprietary Information

The JCL for example 8 follows:

```
//        JOB
//EXAMPLE8 EXEC PGM=AMUUMAIN,
//             PARM='&SSID ,EXAMPLE8,NEW,,MSGLEVEL(1),AMU$OPTO'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//SYSREC1  DD DISP=SHR,DSN=AMU.QA.RGRTESTS.DTA112
//SYSREC2  DD DISP=SHR,DSN=AMU.QA.RGRTESTS.DTA113
//SYSREC3  DD DISP=SHR,DSN=AMU.QA.RGRTESTS.DTA114
//SYSREC4  DD DISP=SHR,DSN=AMU.QA.RGRTESTS.DTA115
//*
//SORTOUT1 DD DSN=AMU.EXAMPLE8.SORTOUT1,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SORTOUT2 DD DSN=AMU.EXAMPLE8.SORTOUT2,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SORTOUT3 DD DSN=AMU.EXAMPLE8.SORTOUT3,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SORTOUT4 DD DSN=AMU.EXAMPLE8.SORTOUT4,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SORTOUT5 DD DSN=AMU.EXAMPLE8.SORTOUT5,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//SYSERR   DD DSN=AMU.EXAMPLE8.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSDISC  DD DSN=AMU.EXAMPLE8.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
//SYSUT101 DD DSN=AMU.EXAMPLE8.SYSUT1,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//SORTWK01 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK02 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK03 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK04 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK05 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK06 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK07 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK08 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK09 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK10 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK11 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//SORTWK12 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(5,1))
//*
//BMCCPY01 DD DSN=AMU.EXAMPLE8.COPY01,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY02 DD DSN=AMU.EXAMPLE8.COPY02,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY03 DD DSN=AMU.EXAMPLE8.COPY03,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY04 DD DSN=AMU.EXAMPLE8.COPY04,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY05 DD DSN=AMU.EXAMPLE8.COPY05,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY06 DD DSN=AMU.EXAMPLE8.COPY06,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY07 DD DSN=AMU.EXAMPLE8.COPY07,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY08 DD DSN=AMU.EXAMPLE8.COPY08,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY09 DD DSN=AMU.EXAMPLE8.COPY09,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//BMCCPY10 DD DSN=AMU.EXAMPLE8.COPY10,,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
```

(continued on following page)

(continued from preceding page)

```
//SYSIN    DD *
   LOAD REPLACE ORDER YES
     UPDATEDB2STATS YES BMCSTATS YES
     CENTURY(1900,1999)
     REDEFINE NO
   INTO TABLE EXAMPLE8.TBL1
       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
       ,SELLER_ID           POSITION(62:71) CHARACTER
       )
   LOG NO
   COPY YES REGISTER NONE
   ENUMROWS(,17000,5)
   ANALYZE PAUSE
/*
```

Output from ANALYZE Phase

The SYSPRINT A output for example 8 follows:

```
                          *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   14:45:40 ...
BMC50002I UTILITY ID = 'EXAMPLE8'.  DB2 SUBSYSTEM ID = 'DEAH'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=6.1.0
BMC50471I REGION=0M,BELOW 16M=7764K,ABOVE 16M=1665544K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
```

(continued on following page)

(example 8A continued from preceding page)

```
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                 WORK                 SORTWORK             ERROR                DISCARD
BMC50470I ACTIVE   = NO                   NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    N/A                  0                    0

BMC50470I DDTYPE   = LOCPFCPY             LOCBFCPY             REMPFCPY             REMBFCPY
BMC50470I ACTIVE   = NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    0                    0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                    5                    5                    5
BMC50470I GDGEMPTY = NO                   NO                   NO                   NO
BMC50470I GDGSCRAT = NO                   NO                   NO                   NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY3'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'
```

(continued on following page)

(example 8A continued from preceding page)

```
BMC50102I   LOAD REPLACE ORDER YES
BMC50102I     UPDATEDB2STATS YES BMCSTATS YES
BMC50102I     CENTURY(1900,1999)
BMC50102I     REDEFINE NO
BMC50102I   INTO TABLE EXAMPLE8.TBL1
BMC50102I       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(62:71) CHARACTER
BMC50102I       )
BMC50102I   LOG NO
BMC50102I   COPY YES REGISTER NONE
BMC50102I   ENUMROWS(,17000,5)
BMC50102I   ANALYZE PAUSE

BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:12


BMC51530I LOADPLUS DASD REQUIREMENT ESTIMATES
BMC51531I                                      3380 CYLS                3390 CYLS
BMC51532I DDNAME                  KBYTES          PRI          SEC         PRI         SEC  INDEX

BMC51533I SORTOUT                   1620            3            1           2           1
BMC51533I SORTOUT8                   203            1            1           1           1
BMC51533I SORTOUT7                   203            1            1           1           1
BMC51533I SORTOUT6                   203            1            1           1           1
BMC51533I SORTOUT5                   203            1            1           1           1
BMC51533I SORTOUT4                   203            1            1           1           1
BMC51533I SORTOUT3                   203            1            1           1           1
BMC51533I SORTOUT2                   203            1            1           1           1
BMC51533I SORTOUT1                   203            1            1           1           1

BMC51533I SYSUT101                   266            1            1           1           1  EXAMPLE8.IX1

BMC51533I SYSUT1                     266            1            1           1           1

BMC51533I SORTWK                    2640            4            1           4           1

BMC51533I BMCCPY01                   160            1            1           1           1
BMC51533I BMCCPY02                   160            1            1           1           1
BMC51533I BMCCPY03                   160            1            1           1           1
BMC51533I BMCCPY04                   160            1            1           1           1
BMC51533I BMCCPY05                   160            1            1           1           1
BMC51533I BMCCPY06                   160            1            1           1           1
BMC51533I BMCCPY07                   160            1            1           1           1
BMC51533I BMCCPY08                   160            1            1           1           1
BMC51533I BMCCPY09                   160            1            1           1           1
BMC51533I BMCCPY10                   160            1            1           1           1

BMC51533I BMCCPY                    1600            3            1           2           1

BMC51533I BMCCPZ01                   160            1            1           1           1
BMC51533I BMCCPZ02                   160            1            1           1           1
BMC51533I BMCCPZ03                   160            1            1           1           1
BMC51533I BMCCPZ04                   160            1            1           1           1
BMC51533I BMCCPZ05                   160            1            1           1           1
BMC51533I BMCCPZ06                   160            1            1           1           1
BMC51533I BMCCPZ07                   160            1            1           1           1
BMC51533I BMCCPZ08                   160            1            1           1           1
BMC51533I BMCCPZ09                   160            1            1           1           1
BMC51533I BMCCPZ10                   160            1            1           1           1

BMC51533I BMCCPZ                    1600            3            1           2           1

BMC51533I BMCRCY01                   160            1            1           1           1
BMC51533I BMCRCY02                   160            1            1           1           1
BMC51533I BMCRCY03                   160            1            1           1           1
BMC51533I BMCRCY04                   160            1            1           1           1
BMC51533I BMCRCY05                   160            1            1           1           1
BMC51533I BMCRCY06                   160            1            1           1           1
BMC51533I BMCRCY07                   160            1            1           1           1
BMC51533I BMCRCY08                   160            1            1           1           1
BMC51533I BMCRCY09                   160            1            1           1           1
BMC51533I BMCRCY10                   160            1            1           1           1

BMC51533I BMCRCY                    1600            3            1           2           1
```

(continued on following page)

(example 8A continued from preceding page)

```
BMC51533I BMCRCZ01                  160           1           1           1           1
BMC51533I BMCRCZ02                  160           1           1           1           1
BMC51533I BMCRCZ03                  160           1           1           1           1
BMC51533I BMCRCZ04                  160           1           1           1           1
BMC51533I BMCRCZ05                  160           1           1           1           1
BMC51533I BMCRCZ06                  160           1           1           1           1
BMC51533I BMCRCZ07                  160           1           1           1           1
BMC51533I BMCRCZ08                  160           1           1           1           1
BMC51533I BMCRCZ09                  160           1           1           1           1
BMC51533I BMCRCZ10                  160           1           1           1           1

BMC51533I BMCRCZ                   1600           3           1           2           1

BMC51533I SYSERR                      1           1           1           1           1

BMC51533I SYSDISC                     1           1           1           1           1

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:03

BMC50005I LOAD COMMAND EXECUTION PAUSING BEFORE PHASE = 'PRELOAD'
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

Output from Restarting the Job

The SYSPRINT B output for example 8 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                  COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   14:55:39 ...
BMC50002I UTILITY ID = 'EXAMPLE8'.  DB2 SUBSYSTEM ID = 'DEAH'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=6.1.0
BMC50471I REGION=0M,BELOW 16M=7764K,ABOVE 16M=1665544K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
```

(continued on following page)

(example 8B continued from preceding page)

```
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY3'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'
BMC50003I RESTARTING UTILITY: COMMAND = 'LOAD', PHASE = 'PRELOAD'

BMC50102I   LOAD REPLACE ORDER YES
BMC50102I     UPDATEDB2STATS YES BMCSTATS YES
BMC50102I     CENTURY(1900,1999)
BMC50102I     REDEFINE NO
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 8B continued from preceding page)

```
BMC50102I   INTO TABLE EXAMPLE8.TBL1
BMC50102I       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(62:71) CHARACTER
BMC50102I       )
BMC50102I   LOG NO
BMC50102I   COPY YES REGISTER NONE
BMC50102I   ENUMROWS(,17000,5)
BMC50102I   ANALYZE PAUSE

BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:07

BMC50474I BELOW 16M = 7264K, ABOVE 16M = 1649824K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 5, #READERS = 4, INDEX TASKS = 1, TIME = 78848
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 5, #READERS = 3, INDEX TASKS = 1, TIME = 85648
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 5, #READERS = 2, INDEX TASKS = 1, TIME = 99328
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 5, #READERS = 1, INDEX TASKS = 1, TIME = 140288
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 4, #READERS = 4, INDEX TASKS = 1, TIME = 93440
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 4, #READERS = 3, INDEX TASKS = 1, TIME = 100240
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 4, #READERS = 2, INDEX TASKS = 1, TIME = 113920
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 4, #READERS = 1, INDEX TASKS = 1, TIME = 154880
BMC51496I PRELOAD ANALYZE, #SORTS = 5, #READERS = 4, INDEX TASKS = 1
BMC50481I 3: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 4: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 2: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 3: READ TASK, XBLK XFERS = 5, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  1735 PHYSICAL (1735 LOGICAL) RECORDS READ FROM SYSREC3
BMC51510I 2: READ TASK, XBLK XFERS = 6, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  3442 PHYSICAL (3442 LOGICAL) RECORDS READ FROM SYSREC2
BMC51510I 4: READ TASK, XBLK XFERS = 4, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  854 PHYSICAL (854 LOGICAL) RECORDS READ FROM SYSREC4
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 17, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  10000 PHYSICAL (10000 LOGICAL) RECORDS READ FROM SYSREC1
BMC50481I 9: SORT COMPLETE.  ELAPSED TIME = 00:00:12
BMC50481I 7: SORT COMPLETE.  ELAPSED TIME = 00:00:10
BMC50481I 5: SORT COMPLETE.  ELAPSED TIME = 00:00:11
BMC50481I 6: SORT COMPLETE.  ELAPSED TIME = 00:00:11
BMC50481I 8: SORT COMPLETE.  ELAPSED TIME = 00:00:17
BMC51503E DUPLICATE KEY FOUND FOR CLUSTER INDEX 'EXAMPLE8.IX0', RECORD NO. = 1 OF SYSREC3
BMC51504E     KEY VALUE = ' 1/25/1990,100'
BMC51503E DUPLICATE KEY FOUND FOR CLUSTER INDEX 'EXAMPLE8.IX0', RECORD NO. = 100 OF SYSREC4
BMC51504E     KEY VALUE = ' 1/25/1990,100'
BMC50481I 5: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:30
BMC50481I 8: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:30
BMC51510I 5: SORT TASK, XBLK XFERS = 4, EMPTY WAITS = 0, FULL WAITS = 3
BMC50481I 7: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:31
BMC51510I 8: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 1
BMC50481I 9: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:34
BMC51510I 7: SORT TASK, XBLK XFERS = 4, EMPTY WAITS = 0, FULL WAITS = 3
BMC51510I 9: SORT TASK, XBLK XFERS = 4, EMPTY WAITS = 0, FULL WAITS = 3
BMC50481I 6: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:55
BMC51510I 6: SORT TASK, XBLK XFERS = 18, EMPTY WAITS = 0, FULL WAITS = 7
BMC50476I DDNAME = SYSREC1, I/OS = 36, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC2, I/OS = 13, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC3, I/OS = 7, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC4, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT1, I/OS = 5, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT2, I/OS = 27, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT3, I/OS = 6, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT4, I/OS = 4, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT5, I/OS = 6, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 12, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 38, XFERS = 32, EMPTY WAITS = 0, FULL WAITS =17

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC3, ID# 3

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                         FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                          CONSTRAINT NAME

BMC51524E         1         1         1  DUPLICATE KEY         3:         1  EXAMPLE8.TBL1                 EXAMPLE8.IX0
BMC50476I DDNAME = SYSREC3, I/OS = 7, I/O WAITS = 2, RDB LOCK WAITS = 0
```

(continued on following page)

(example 8B continued from preceding page)

```
BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC4, ID# 4

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                          FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                           CONSTRAINT NAME

BMC51524E       100       100         2  DUPLICATE KEY         3:         1  EXAMPLE8.TBL1                  EXAMPLE8.IX0
BMC50476I DDNAME = SYSREC4, I/OS = 4, I/O WAITS = 2, RDB LOCK WAITS = 0

BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51471I PRELOAD STATISTICS:  11031 ROWS SELECTED FOR PARTITION 1
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A001' WILL REQUIRE 163 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A001' WILL REQUIRE 52 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  522 ROWS SELECTED FOR PARTITION 2
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A002' WILL REQUIRE 29 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A002' WILL REQUIRE 6 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  529 ROWS SELECTED FOR PARTITION 3
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A003' WILL REQUIRE 29 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A003' WILL REQUIRE 6 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  577 ROWS SELECTED FOR PARTITION 4
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A004' WILL REQUIRE 30 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A004' WILL REQUIRE 6 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  543 ROWS SELECTED FOR PARTITION 5
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A005' WILL REQUIRE 29 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A005' WILL REQUIRE 6 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  109 ROWS SELECTED FOR PARTITION 6
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A006' WILL REQUIRE 21 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A006' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  236 ROWS SELECTED FOR PARTITION 7
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A007' WILL REQUIRE 23 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A007' WILL REQUIRE 5 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  793 ROWS SELECTED FOR PARTITION 8
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A008' WILL REQUIRE 34 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A008' WILL REQUIRE 7 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  789 ROWS SELECTED FOR PARTITION 9
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A009' WILL REQUIRE 34 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A009' WILL REQUIRE 7 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  900 ROWS SELECTED FOR PARTITION 10
BMC51486I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A010' WILL REQUIRE 36 PAGES
BMC51490I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A010' WILL REQUIRE 7 PAGES (APPROX.)
BMC51488I LOADING OF DATASET 'DEAHCAT.DSNDBD.AMUEX8DB.IX1.I0001.A001' MAY REQUIRE 87 PAGES
BMC51472I PRELOAD PHASE STATISTICS:  16029 ROWS SELECTED FOR SPACE 'AMUEX8DB.EX8TS', 2 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  2 PHYSICAL (2 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:01:51

BMC50474I BELOW 16M = 7344K, ABOVE 16M = 1663888K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 5, #COPY TASKS = 5, #INDEX TASKS = 1
BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50370I STARTING 5 IMAGE COPY TASKS
BMC51508I MAX INDEX TASKS = 1, INDEXES PER TASK = 1, SORTWKS PER TASK = 1, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 5, MAX PARTS PER TASK = 2, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX8DB.EX8TS' DELETED
BMC50476I DDNAME = SYSUT101, I/OS = 13, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 1: SORT COMPLETE.  ELAPSED TIME = 00:00:12
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:11  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A010'
BMC50477I 3: PARTITION =  10, ROWS/KEYS = 900, I/O WAITS = 7, DDNAME = SYS00016
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:11  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 522, I/O WAITS = 7, DDNAME = SYS00017
BMC50482I 2: BUILD COMPLETE.  ELAPSED TIME = 00:00:11  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A003'
BMC50477I 2: PARTITION =   3, ROWS/KEYS = 529, I/O WAITS = 7, DDNAME = SYS00014
BMC50482I 3: LOAD COMPLETE.  ELAPSED TIME = 00:00:19  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A010'
BMC50477I 3: PARTITION =  10, ROWS/KEYS = 900, I/O WAITS = 5, DDNAME = SYS00009
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:19  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A003'
BMC50477I 2: PARTITION =   3, ROWS/KEYS = 529, I/O WAITS = 5, DDNAME = SYS00010
BMC51474I LOAD STATISTICS:  900 ROWS LOADED INTO PARTITION 10
BMC50482I 4: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A005'
BMC50477I 4: PARTITION =   5, ROWS/KEYS = 543, I/O WAITS = 7, DDNAME = SYS00018
BMC51474I LOAD STATISTICS:  529 ROWS LOADED INTO PARTITION 3
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:19  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 522, I/O WAITS = 5, DDNAME = SYS00008
BMC51474I LOAD STATISTICS:  522 ROWS LOADED INTO PARTITION 2
BMC50476I DDNAME = SORTOUT4, I/OS = 5, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50482I 4: LOAD COMPLETE.  ELAPSED TIME = 00:00:28  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A005'
BMC50477I 4: PARTITION =   5, ROWS/KEYS = 543, I/O WAITS = 5, DDNAME = SYS00013
BMC51474I LOAD STATISTICS:  543 ROWS LOADED INTO PARTITION 5
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A010'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A002'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A003'
BMC50476I DDNAME = BMCCPY10, I/OS = 6, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC50376I 36 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY10'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A005'
BMC50476I DDNAME = BMCCPY02, I/OS = 5, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 29 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY02'
BMC50476I DDNAME = BMCCPY05, I/OS = 5, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = BMCCPY03, I/OS = 5, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50376I 29 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY03'
BMC50376I 29 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY05'
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 8B continued from preceding page)

```
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:16  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX1.I0001.A001'
BMC50477I 1: PARTITION =   0, ROWS/KEYS = 16029, I/O WAITS = 7, DDNAME = SYS00019
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 577, I/O WAITS = 7, DDNAME = SYS00028
BMC50482I 4: BUILD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A006'
BMC50477I 4: PARTITION =   6, ROWS/KEYS = 109, I/O WAITS = 7, DDNAME = SYS00029
BMC50482I 2: BUILD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A008'
BMC50477I 2: PARTITION =   8, ROWS/KEYS = 793, I/O WAITS = 7, DDNAME = SYS00027
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:15  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 577, I/O WAITS = 5, DDNAME = SYS00023
BMC50482I 4: LOAD COMPLETE.  ELAPSED TIME = 00:00:05  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A006'
BMC50477I 4: PARTITION =   6, ROWS/KEYS = 109, I/O WAITS = 5, DDNAME = SYS00025
BMC51474I LOAD STATISTICS:  577 ROWS LOADED INTO PARTITION 4
BMC51476I BUILD STATISTICS:  16029 KEYS LOADED INTO INDEX 'EXAMPLE8.IX1'
BMC51474I LOAD STATISTICS:  109 ROWS LOADED INTO PARTITION 6
BMC50476I DDNAME = SORTOUT1, I/OS = 6, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:21  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A008'
BMC50477I 2: PARTITION =   8, ROWS/KEYS = 793, I/O WAITS = 5, DDNAME = SYS00020
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:34  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A001'
BMC50477I 1: PARTITION =   1, ROWS/KEYS = 11031, I/O WAITS = 7, DDNAME = SYS00015
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A004'
BMC51474I LOAD STATISTICS:  793 ROWS LOADED INTO PARTITION 8
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A006'
BMC50482I 1: LOAD COMPLETE.  ELAPSED TIME = 00:00:44  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A001'
BMC50477I 1: PARTITION =   1, ROWS/KEYS = 11031, I/O WAITS = 5, DDNAME = SYS00011
BMC50476I DDNAME = BMCCPY04, I/OS = 5, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51474I LOAD STATISTICS:  11031 ROWS LOADED INTO PARTITION 1
BMC50476I DDNAME = SORTOUT3, I/OS = 7, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 30 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY04'
BMC50476I DDNAME = BMCCPY06, I/OS = 4, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50376I 21 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY06'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A008'
BMC50476I DDNAME = BMCCPY08, I/OS = 6, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC50376I 34 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY08'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A001'
BMC50476I DDNAME = BMCCPY01, I/OS = 28, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 163 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY01'
BMC50482I 4: BUILD COMPLETE.  ELAPSED TIME = 00:00:08  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A009'
BMC50477I 4: PARTITION =   9, ROWS/KEYS = 789, I/O WAITS = 7, DDNAME = SYS00034
BMC50482I 4: LOAD COMPLETE.  ELAPSED TIME = 00:00:18  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A009'
BMC50477I 4: PARTITION =   9, ROWS/KEYS = 789, I/O WAITS = 5, DDNAME = SYS00031
BMC51474I LOAD STATISTICS:  789 ROWS LOADED INTO PARTITION 9
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.IX0.I0001.A007'
BMC50477I 1: PARTITION =   7, ROWS/KEYS = 236, I/O WAITS = 7, DDNAME = SYS00037
BMC50476I DDNAME = SORTOUT5, I/OS = 7, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50482I 1: LOAD COMPLETE.  ELAPSED TIME = 00:00:09  DSN = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A007'
BMC50477I 1: PARTITION =   7, ROWS/KEYS = 236, I/O WAITS = 5, DDNAME = SYS00035
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A009'
BMC51474I LOAD STATISTICS:  236 ROWS LOADED INTO PARTITION 7
BMC50476I DDNAME = BMCCPY09, I/OS = 6, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT2, I/OS = 28, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 34 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY09'
BMC51476I BUILD STATISTICS:  16029 KEYS LOADED INTO INDEX 'EXAMPLE8.IX0'
BMC51475I LOAD STATISTICS:  16029 ROWS LOADED INTO TABLE SPACE 'AMUEX8DB.EX8TS'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEAHCAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A007'
BMC50476I DDNAME = BMCCPY07, I/OS = 4, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50376I 23 PAGES COPIED TO DATASET = 'AMU.EXAMPLE8.COPY07'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:01:05

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX8DB.EX8TS  (HARDWARE COMPRESSION IN EFFECT)
BMC50513I             PARTS   = 10        TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 428           SPACE   = 150
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   1       163      11031          0          0       18        0   95    0      15    1        1917   83       277    37
BMC50512I   2        29        522          0          0       18        0   95    0      15    1         522    0         0     0
BMC50512I   3        29        529          0          0       18        0   95    0      15    1         529    0         0     0
BMC50512I   4        30        577          0          0       18        0   95    0      15    1         577    0         0     0
BMC50512I   5        29        543          0          0       18        0   96    0      15    1         543    0         0     0
BMC50512I   6        21        109          0          0       18        0   52    0      15    1         109    0         0     0
BMC50512I   7        23        236          0          0       18        0   58    0      15    1         236    0         0     0
BMC50512I   8        34        793          0          0       18        0   95    0      15    1         793    0         0     0
BMC50512I   9        34        789          0          0       18        0   95    0      15    1         789    0         0     0
BMC50512I  10        36        900          0          0       18        0   95    0      15    1         900    0         0     0
BMC50521I TABLE NAME                        CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLE8.TBL1                     16029        408         0      50        95      57
BMC50510I INDEX       EXAMPLE8.IX0  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE8.TBL1     COLUMN  DATE_OF_SALE
BMC50514I             NACTIVE = 116           SPACE   = 14
BMC50542I             FIRSTKEY= 132           FULLKEY = 16029         NLEAF   = 76
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 9                                     COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'1998030600800004'  LOW2KEY= X'1901030100800000'
BMC50547I             HIGH2KEY=         LOW2KEY=
```

(continued on following page)

(example 8B continued from preceding page)

```
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   1        53      11031        2          1          0        0       24       49     2          5       5
BMC50532I   2         7        522       33          1          0        0        7        3     2          1       1
BMC50532I   3         7        529       33          1          0        0        7        3     2          1       1
BMC50532I   4         7        577       33          1          0        0        6        3     2          1       1
BMC50532I   5         7        543       33          1          0        0        7        3     2          1       1
BMC50532I   6         5        109        0          1          0        0        6        1     2          1       1
BMC50532I   7         6        236       50          1          0        0        8        2     2          1       1
BMC50532I   8         8        793       25          1          0        0        7        4     2          1       1
BMC50532I   9         8        789       25          1          0        0        7        4     2          1       1
BMC50532I  10         8        900       25          1          0        0        5        4     2          1       1
BMC50510I INDEX       EXAMPLE8.IX1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLE8.TBL1     COLUMN  SELLER_ID
BMC50514I             NACTIVE = 27            SPACE   = 3
BMC50542I             FIRSTKEY= 66            FULLKEY = 66            NLEAF   = 23
BMC50543I             LEVELS  = 2                                     PCTCLUST= 21
BMC50544I             KEYLEN  = 10                                    COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = D
BMC50546I             HIGH2KEY= X'F8F9404A4BF8F940'  LOW2KEY= X'4A4B4C4D4E4A4B4C'
BMC50547I             HIGH2KEY=         LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0        27      16029        4        653        557        0       14       23     2          3       3
BMC50505I BMCSTATS UPDATED IN THE DASD MANAGER PLUS DATABASE
BMC50506I STATISTICS COLUMNS UPDATED IN THE DB2 CATALOG
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

BMC Software, Inc., Confidential and Proprietary Information

Example 9: LOAD RESUME YES of a Partitioned Table Space with INDEX UPDATE

This example shows a LOAD RESUME YES of the same table that was loaded in example 8. Data in partition 10 is replaced (PART REPLACE) and additional data is added to partition 9.

This example specifies one INTO with the PART option for each partition being loaded because only a few partitions are to be updated with new data and RESUME YES is specified. This eliminates building all partitions of the clustering index and can save considerable time.

Because the LOAD command includes the INDEX UPDATE option, the index entries for the clustering index for partition 9 and the entire secondary index are updated. The index entries for the clustering index for partition 10 are completely rebuilt, because PART REPLACE is specified for that partition.

The JCL contains multiple INDATA (SYSREC) and DATAWK (SORTOUT) DD statements, allowing multitasking of the data in both the PRELOAD and LOAD phases.

The SYSREC ddname, work file ddnames, and copy ddnames are all overridden by using the appropriate command options in addition to specifying the override ddnames in the JCL. Note the following override names:

- INDATA overrides SYSREC
- DATAWK overrides SORTOUT
- ERRWK overrides SYSERR
- IXWK overrides SYSUT1
- DISCARDS overrides SYSDISC
- CPY overrides BMCCPY
- RCY overrides BMCRCY

Assuming that the data was sorted prior to this run, the LOAD command includes the ORDER NO option. The clustering index is processed as a secondary index, which allows LOADPLUS to use two IXWK (SYSUT1) data sets to process the indexes concurrently.

The DD statements in the JCL direct LOADPLUS to create two copies of each partition. Note that LOADPLUS makes copies only for the partitions that are participating in the load.

BMC Software, Inc., Confidential and Proprietary Information

The JCL for example 9 follows:

```
//        JOB
//EXAMPLE9 EXEC PGM=AMUUMAIN,
//             PARM='&SSID,EXAMPLE9,NEW/RESTART,,MSGLEVEL(1)'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//INDATA1  DD DISP=SHR,DSN=AMU.QA.EXAMPLE9.DATA02
//INDATA2  DD DISP=SHR,DSN=AMU.QA.EXAMPLE9.DATA01
//*
//DATAWK1  DD DSN=AMU.QA.EXAMPLE9.SORTOUT1,
//            UNIT=WORK,SPACE=(CYL,(5,1)),DISP=(MOD,CATLG,CATLG)
//DATAWK2  DD DSN=AMU.QA.EXAMPLE9.SORTOUT2,
//            UNIT=WORK,SPACE=(CYL,(5,1)),DISP=(MOD,CATLG,CATLG)
//ERRWK    DD DSN=AMU.QA.EXAMPLE9.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//DISCARDS DD DSN=AMU.QA.EXAMPLE9.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
//IXWK01   DD DSN=AMU.QA.EXAMPLE9.SYSUT1,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//IXWK02   DD DSN=AMU.QA.EXAMPLE9.SYSUT2,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//SORTWK01 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK02 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK03 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK04 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK05 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK06 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK07 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK08 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK09 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK10 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK11 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SORTWK12 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//*
//CPY09    DD DSN=AMU.QA.EXAMPLE9.COPY09,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY09    DD DSN=AMU.QA.EXAMPLE9.RCOPY09,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//CPY10    DD DSN=AMU.QA.EXAMPLE9.COPY10,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//RCY10    DD DSN=AMU.QA.EXAMPLE9.RCOPY10,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SYSIN    DD *
   LOAD RESUME YES
    INDEX UPDATE  UNIQUECHECK NO
     INDDN INDATA  LOADDN DATAWK  ERRDDN ERRWK  WORKDDN IXWK
     DISCARDDN DISCARDS
     ORDER NO
   INTO TABLE EXAMPLE8.TBL1 PART 10 REPLACE
       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
       ,SELLER_ID           POSITION(62:71) CHARACTER
       )
   INTO TABLE EXAMPLE8.TBL1 PART 09
       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
       ,SELLER_ID           POSITION(62:71) CHARACTER
       )
   LOG NO
   COPY YES REGISTER NONE
   COPYDDN CPY  RECOVERYDDN RCY
/*
```

The SYSPRINT output for example 9 follows:

```
                         *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   15:11:42 ...
BMC50002I UTILITY ID = 'EXAMPLE9'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTS'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7848K,ABOVE 16M=1665640K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
```

(continued on following page)

(example 9 continued from preceding page)

```
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 9 continued from preceding page)

```
BMC50102I   LOAD RESUME YES
BMC50102I    INDEX UPDATE  UNIQUECHECK NO
BMC50102I     INDDN INDATA  LOADDN DATAWK  ERRDDN ERRWK  WORKDDN IXWK
BMC50102I     DISCARDDN DISCARDS
BMC50102I     ORDER NO
BMC50102I   INTO TABLE EXAMPLE8.TBL1 PART 10 REPLACE
BMC50102I       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(62:71) CHARACTER
BMC50102I       )
BMC50102I   INTO TABLE EXAMPLE8.TBL1 PART 09
BMC50102I       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(62:71) CHARACTER
BMC50102I       )
BMC50102I   LOG NO
BMC50102I   COPY YES REGISTER NONE
BMC50102I   COPYDDN CPY  RECOVERYDDN RCY

BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7360K, ABOVE 16M = 1649468K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 2, #READERS = 2, INDEX TASKS = 1, TIME = 53760
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 2, #READERS = 1, INDEX TASKS = 1, TIME = 67840
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 1, #READERS = 2, INDEX TASKS = 1, TIME = 107520
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 107520
BMC51496I PRELOAD ANALYZE, #SORTS = 2, #READERS = 2, INDEX TASKS = 1
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 2: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  286 PHYSICAL (286 LOGICAL) RECORDS READ FROM INDATA1
BMC51510I 2: READ TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  129 PHYSICAL (129 LOGICAL) RECORDS READ FROM INDATA2
BMC50481I 4: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 4: SORT TASK, XBLK XFERS = 3, EMPTY WAITS = 0, FULL WAITS = 3
BMC50481I 3: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 3: SORT TASK, XBLK XFERS = 3, EMPTY WAITS = 0, FULL WAITS = 1
BMC50481I 5: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 5: INDEX TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 3
BMC50476I DDNAME = INDATA1, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = INDATA2, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = DATAWK1, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = DATAWK2, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = IXWK02, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = IXWK01, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = ERRWK, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 20, XFERS = 6, EMPTY WAITS = 0, FULL WAITS =7
BMC50476I DDNAME = DISCARDS, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51471I PRELOAD STATISTICS:  67 ROWS SELECTED FOR PARTITION 9
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A009' WILL REQUIRE 35 PAGES
BMC51471I PRELOAD STATISTICS:  348 ROWS SELECTED FOR PARTITION 10
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A010' WILL REQUIRE 25 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX8DB.IX0.I0001.A010' WILL REQUIRE 5 PAGES (APPROX.)
BMC51472I PRELOAD PHASE STATISTICS:  415 ROWS SELECTED FOR SPACE 'AMUEX8DB.EX8TS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:03

BMC50474I BELOW 16M = 7452K, ABOVE 16M = 1663584K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 2, #COPY TASKS = 2, #INDEX TASKS = 2
BMC51508I MAX INDEX TASKS = 2, INDEXES PER TASK = 1, SORTWKS PER TASK = 1, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 2, MAX PARTS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51454I EXISTING ROWS IN TABLESPACE 'AMUEX8DB.EX8TS' PARTITION 10 DELETED
BMC50482I 1: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A009'
BMC50477I 1: PARTITION =   9, ROWS/KEYS = 67, I/O WAITS = 24, DDNAME = SYS00002
BMC51474I LOAD STATISTICS:  67 ROWS LOADED INTO PARTITION 9
BMC50476I DDNAME = DATAWK2, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = IXWK01, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:01
BMC50476I DDNAME = IXWK02, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 1: SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX8DB.IX0.I0001.A009'
```

(continued on following page)

(example 9 continued from preceding page)

```
BMC50477I 1: PARTITION =   9, ROWS/KEYS = 67, I/O WAITS = 8, DDNAME = SYS00008
BMC51474I BUILD STATISTICS:  67 KEYS LOADED INTO PARTITION 9
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A010'
BMC50477I 0: PARTITION =  10, ROWS/KEYS = 348, I/O WAITS = 5, DDNAME = SYS00009
BMC51474I LOAD STATISTICS:  348 ROWS LOADED INTO PARTITION 10
BMC50476I DDNAME = DATAWK1, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51475I LOAD STATISTICS:  415 ROWS LOADED INTO TABLE SPACE 'AMUEX8DB.EX8TS'
BMC50482I 2: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEX8DB.IX1.I0001.A001'
BMC50477I 2: PARTITION =   0, ROWS/KEYS = 415, I/O WAITS = 34, DDNAME = SYS00007
BMC51476I BUILD STATISTICS:  415 KEYS LOADED INTO INDEX 'EXAMPLE8.IX1'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX8DB.IX0.I0001.A010'
BMC50477I 1: PARTITION =  10, ROWS/KEYS = 348, I/O WAITS = 7, DDNAME = SYS00012
BMC51474I BUILD STATISTICS:  348 KEYS LOADED INTO PARTITION 10
BMC51476I BUILD STATISTICS:  415 KEYS LOADED INTO INDEX 'EXAMPLE8.IX0'
BMC50476I DDNAME = ERRWK, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC50370I STARTING 2 IMAGE COPY TASKS
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A009'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEX8DB.EX8TS.I0001.A010'
BMC50476I DDNAME = CPY09, I/OS = 12, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC50376I 35 PAGES COPIED TO DATASET = 'AMU.EXAMPLE9.COPY09'
BMC50376I 35 PAGES COPIED TO DATASET = 'AMU.EXAMPLE9.RCOPY09'
BMC50476I DDNAME = CPY10, I/OS = 10, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50376I 25 PAGES COPIED TO DATASET = 'AMU.EXAMPLE9.COPY10'
BMC50376I 25 PAGES COPIED TO DATASET = 'AMU.EXAMPLE9.RCOPY10'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:07

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX8DB.EX8TS  (HARDWARE COMPRESSION IN EFFECT)
BMC50513I             PARTS   = 10        TABLES  = 1         SEGSIZE = 0
BMC50503I STATISTICS FOR TABLESPACE 'AMUEX8DB.EX8TS' NOT AVAILABLE
BMC50503I STATISTICS FOR TABLE      'EXAMPLE8.TBL1' NOT AVAILABLE
BMC50503I STATISTICS FOR INDEX      'EXAMPLE8.IX0' NOT AVAILABLE
BMC50503I STATISTICS FOR INDEX      'EXAMPLE8.IX1' NOT AVAILABLE
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

BMC Software, Inc., Confidential and Proprietary Information

Example 10: LOAD RESUME NO of a Simple Table Space Using CHECK CONSTRAINTS

In this example, LOADPLUS loads data into a simple table space that contains only one table with check constraints and with one unique index. The ENFORCE CHECK CONSTRAINTS option tells LOADPLUS to report any check constraint violations. Message BMC51514E displays these violations.

BMCSORT dynamically allocates sort work files. The JCL does not specify any SORTWK DD statements and LOADPLUS dynamic allocation for sort work data sets is not active.

This example uses the DISCARDS IGNORE DUPKEY option to ignore all duplicate key errors for processing and to keep LOADPLUS from writing these errors to the SYSERR and SYSDISC data sets. The LOAD command includes the DISCARDS REPORT NO WHEN option. Using this option reduces the number of lines in the SYSPRINT because WHEN discards are not reported in the LOADPLUS error summary report. Not reporting the discarded WHEN records is beneficial if the input data set (SYSREC) contains large amounts of data that will be discarded. Note messages BMC51462I and BMC51463I regarding the DISCARDS IGNORE and REPORT options.

RESUME NO indicates that the table space being loaded is empty. Specifying ENUMROWS 50 estimates that LOADPLUS will load 50 rows.

The 3390 device type will be used for the sort work file that BMCSORT dynamically allocates (as specified by SORTDEVT 3390). Specifying SORTNUM 2 provides the following benefits:

- tells BMCSORT to allocate the sort work files that it needs, up to 32 per sort task, regardless of the values specified in the BMCSORT DYNALOC installation option

- tells LOADPLUS that it can use multiple sort tasks if necessary

SKIPFIELDS YES enables processing of input record fields that are not actually in the table. In this case, NULL1 is used in NULLIF condition processing.

BMC Software, Inc., Confidential and Proprietary Information

The JCL for example 10 follows:

```
//        JOB
//EXAMPLEA EXEC PGM=AMUUMAIN,REGION=4M,
//            PARM='&SSID,EXAMPLEA,NEW,,MSGLEVEL(1),AMU$OPTS'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT  DD  SYSOUT=*
//SYSOUT    DD  SYSOUT=*
//UTPRINT   DD  SYSOUT=*
//SYSUDUMP  DD  SYSOUT=*
//SYSREC   DD DSN=AMU.QA.RGRTESTS(EX7DATA),
//            DISP=SHR
//SORTOUT  DD DSN=AMU.EXAMPLEA.SORTOUT,
//            SPACE=(CYL,(20,10)),UNIT=WORK,DISP=(MOD,CATLG,CATLG)
//SYSUT101 DD DSN=AMU.EXAMPLEA.SYSUT1,
//            SPACE=(CYL,(20,10)),UNIT=WORK,DISP=(MOD,CATLG,CATLG)
//SYSERR   DD DSN=AMU.EXAMPLEA.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSDISC  DD DSN=AMU.EXAMPLEA.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//SYSIN    DD *
     LOAD
     RESUME NO
     ENFORCE CHECK CONSTRAINTS
     DISCARDS 50 IGNORE DUPKEY REPORT NO WHEN
     ENUMROWS 50
     SORTDEVT 3390 SORTNUM 2
     COPY NO
     REDEFINE YES
     SKIPFIELDS YES
     INTO TABLE EXAMPLEA.TBL1
        WHEN RECNO <> '90000'
                (RECNO       POSITION(*)     CHARACTER
                ,INPUT_A     POSITION(*)     CHARACTER
                ,NULL1       POSITION(9)     CHARACTER(1)
                ,INPUT_3     POSITION(10:12) CHARACTER
                   NULLIF NULL1='?'
                ,INPUT_7     POSITION(13:19) CHARACTER
                ,INPUT_10    POSITION(30:39) CHARACTER
                ,INPUT_12    POSITION(25:36) CHARACTER
                ,MLT_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,-1)
                ,DIV_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,1)
                ,MLT_INT2    POSITION(13)    DECIMAL EXTERNAL (7,-2)
                ,DIV_INT2    POSITION(13)    DECIMAL EXTERNAL (7,2)
                ,MLT_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,-3)
                ,DIV_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,3)
                ,MLT_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,-40)
                ,DIV_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,40)
      )
/*
```

The SYSPRINT output for example 10 follows:

```
                          *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
               COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   15:14:46 ...
BMC50002I UTILITY ID = 'EXAMPLEA'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTS'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7872K,ABOVE 16M=1665996K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
```

(continued on following page)

(example 10 continued from preceding page)

```
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                 WORK                 SORTWORK             ERROR                DISCARD
BMC50470I ACTIVE   = NO                   NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    N/A                  0                    0

BMC50470I DDTYPE   = LOCPFCPY             LOCBFCPY             REMPFCPY             REMBFCPY
BMC50470I ACTIVE   = NO                   NO                   NO                   NO
BMC50470I IFALLOC  = USE                  USE                  USE                  USE
BMC50470I SMS      = NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    0                    0
BMC50470I EXPDT    =
BMC50470I RETPD    =
```

(continued on following page)

(example 10 continued from preceding page)

```
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I     LOAD
BMC50102I     RESUME NO
BMC50102I     ENFORCE CHECK CONSTRAINTS
BMC50102I     DISCARDS 50 IGNORE DUPKEY REPORT NO WHEN
BMC50102I     ENUMROWS 50
BMC50102I     SORTDEVT 3390 SORTNUM 2
BMC50102I     COPY NO
BMC50102I     REDEFINE YES
BMC50102I     SKIPFIELDS YES
BMC50102I     INTO TABLE EXAMPLEA.TBL1
BMC50102I        WHEN RECNO <> '90000'
BMC50102I                (RECNO       POSITION(*)     CHARACTER
BMC50102I                ,INPUT_A     POSITION(*)     CHARACTER
BMC50102I                ,NULL1       POSITION(9)     CHARACTER(1)
BMC50102I                ,INPUT_3     POSITION(10:12) CHARACTER
BMC50102I                   NULLIF NULL1='?'
BMC50102I                ,INPUT_7     POSITION(13:19) CHARACTER
BMC50102I                ,INPUT_10    POSITION(30:39) CHARACTER
BMC50102I                ,INPUT_12    POSITION(25:36) CHARACTER
BMC50102I                ,MLT_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,-1)
BMC50102I                ,DIV_SMINT1  POSITION(10)    INTEGER EXTERNAL (3,1)
BMC50102I                ,MLT_INT2    POSITION(13)    DECIMAL EXTERNAL (7,-2)
BMC50102I                ,DIV_INT2    POSITION(13)    DECIMAL EXTERNAL (7,2)
BMC50102I                ,MLT_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,-3)
BMC50102I                ,DIV_DEC3    POSITION(25)    DECIMAL EXTERNAL (12,3)
BMC50102I                ,MLT_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,-40)
BMC50102I                ,DIV_FLOAT40 POSITION(30)    FLOAT EXTERNAL (10,40)
BMC50102I       )


BMC51422I FOR TABLE 'EXAMPLEA.TBL1', THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED
BMC51423I FIELD 'RECNO            ' HAS POSITION(1:5)
BMC51423I FIELD 'INPUT_A          ' HAS POSITION(6:8)
BMC51423I FIELD 'NULL1            ' HAS POSITION(9:9)
BMC51423I FIELD 'MLT_SMINT1       ' HAS POSITION(10:12)
BMC51423I FIELD 'DIV_SMINT1       ' HAS POSITION(10:12)
BMC51423I FIELD 'MLT_INT2         ' HAS POSITION(13:19)
BMC51423I FIELD 'DIV_INT2         ' HAS POSITION(13:19)
BMC51423I FIELD 'MLT_DEC3         ' HAS POSITION(25:36)
BMC51423I FIELD 'DIV_DEC3         ' HAS POSITION(25:36)
BMC51423I FIELD 'MLT_FLOAT40      ' HAS POSITION(30:39)
BMC51423I FIELD 'DIV_FLOAT40      ' HAS POSITION(30:39)
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50394I UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'
BMC50474I BELOW 16M = 7388K, ABOVE 16M = 1653824K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 46080
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 1
BMC50477I 4: PARTITION =   0, ROWS/KEYS = 0, I/O WAITS = 1, DDNAME = SYS00008
BMC51514E CHECK CONSTRAINT VIOLATION.  TABLE = 'EXAMPLEA.TBL1', CONSTRAINT = 'VAL_INPUT_A', RECORD NO. 7 OF SYSREC
BMC51514E CHECK CONSTRAINT VIOLATION.  TABLE = 'EXAMPLEA.TBL1', CONSTRAINT = 'VAL_INPUT_A', RECORD NO. 14 OF SYSREC
BMC51501E INVALID NUMERIC DATA, COLUMN = 'INPUT_7', RECORD NO. 22 OF SYSREC, DATA = C'65S÷220'
BMC51514E CHECK CONSTRAINT VIOLATION.  TABLE = 'EXAMPLEA.TBL1', CONSTRAINT = 'VAL_INPUT_A', RECORD NO. 26 OF SYSREC
BMC51514E CHECK CONSTRAINT VIOLATION.  TABLE = 'EXAMPLEA.TBL1', CONSTRAINT = 'VAL_INPUT_A', RECORD NO. 44 OF SYSREC
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  50 PHYSICAL (50 LOGICAL) RECORDS READ FROM SYSREC
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 10 continued from preceding page)

```
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 2: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC50481I 3: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLEA.TBL1X1', RECORD NO. = 5 OF SYSREC
BMC51506E     KEY VALUE = ''14800'', RID-1 = X'00000204', RID-2 = X'00000204'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLEA.TBL1X1', RECORD NO. = 10 OF SYSREC
BMC51506E     KEY VALUE = ''14800'', RID-1 = X'00000204', RID-2 = X'00000208'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLEA.TBL1X1', RECORD NO. = 9 OF SYSREC
BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000207', RID-2 = X'00000207'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLEA.TBL1X1', RECORD NO. = 25 OF SYSREC
BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000207', RID-2 = X'00000213'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLEA.TBL1X1', RECORD NO. = 32 OF SYSREC
BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000207', RID-2 = X'00000218'
BMC51505E DUPLICATE KEY FOUND FOR INDEX 'EXAMPLEA.TBL1X1', RECORD NO. = 41 OF SYSREC
BMC51506E     KEY VALUE = ''90800'', RID-1 = X'00000207', RID-2 = X'00000304'
BMC50481I 3: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 3: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSREC, I/OS = 6, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51507I XBLKS = 10, XFERS = 2, EMPTY WAITS = 0, FULL WAITS =3

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC, ID# 1

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                        FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                         CONSTRAINT NAME

BMC51524E         7         7         2  CHECK CONSTRAINT      0:         0  EXAMPLEA.TBL1                VAL_INPUT_A
BMC51524E        14        14         4  CHECK CONSTRAINT      0:         0  EXAMPLEA.TBL1                VAL_INPUT_A
BMC51524E        22        22         5  CONVERSION            0:         1  EXAMPLEA.TBL1                INPUT_7
BMC51524E        26        26         7  CHECK CONSTRAINT      0:         0  EXAMPLEA.TBL1                VAL_INPUT_A
BMC51524E        44        44         9  CHECK CONSTRAINT      0:         0  EXAMPLEA.TBL1                VAL_INPUT_A
BMC50476I DDNAME = SYSREC, I/OS = 6, I/O WAITS = 1, RDB LOCK WAITS = 0

BMC51463I ONE OR MORE DISCARDS NOT REPORTED DUE TO DISCARD REPORT OPTION
BMC51462I ONE OR MORE DISCARDS IGNORED DUE TO DISCARD IGNORE OPTION
BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXADB.EXATS.I0001.A001' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXADB.TBL1X1.I0001.A001' WILL REQUIRE 4 PAGES (APPROX.)
BMC51472I PRELOAD PHASE STATISTICS:  34 ROWS SELECTED FOR SPACE 'AMUEXADB.EXATS', 11 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  10 PHYSICAL (10 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:03

BMC50474I BELOW 16M = 7480K, ABOVE 16M = 1664576K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 0, #INDEX TASKS = 1
BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51508I MAX INDEX TASKS = 1, INDEXES PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEXADB.EXATS.I0001.A001'
BMC50477I 0: PARTITION =   0, ROWS/KEYS = 34, I/O WAITS = 5, DDNAME = SYS00016
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEXADB.TBL1X1.I0001.A001'
BMC50477I 1: PARTITION =   0, ROWS/KEYS = 34, I/O WAITS = 7, DDNAME = SYS00017
BMC50476I DDNAME = SYSUT101, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  34 KEYS LOADED INTO INDEX 'EXAMPLEA.TBL1X1'
BMC50476I DDNAME = SORTOUT, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51475I LOAD STATISTICS:  34 ROWS LOADED INTO TABLE SPACE 'AMUEXADB.EXATS'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:04

BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEXADB.EXATS
BMC50513I             PARTS   = 0         TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 4             SPACE   = 255
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   0         4         34          0          0        2        0   35    0     255    3           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLEA.TBL1                         34          2         0      74       50       0
BMC50510I INDEX       EXAMPLEA.TBL1X1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLEA.TBL1     COLUMN  RECNO
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 34            FULLKEY = 34            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 88
BMC50544I             KEYLEN  = 5                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'F8F4F8F4F0404040'  LOW2KEY= X'F1F1F8F1F0404040'
BMC50547I             HIGH2KEY= 84840     LOW2KEY= 11810
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5         34        0          0          5        0        7        1     2         1       1
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

Example 11: LOAD REPLACE of a Partitioned Table Space with Dynamic Allocation

This example shows a LOAD REPLACE of a 10-part table with a unique clustering index and a nonunique secondary index. This example also illustrates the use of dynamic allocation.

This example loads seven of ten partitions. Individual INTO PART commands are not specified because processing multiple INTO PART commands is less efficient and offers no benefit on a REPLACE run.

The EXEC statement in the JCL includes an options module override. This options module activates dynamic work file allocation for the load data (SORTOUT), index work (SYSUT1), sort work (SORTWK), discard (SYSDISC), and copy files by specifying ACTIVE YES for each corresponding DDTYPE. The options module de-activates dynamic allocation for the error (SYSERR) data set by specifying ACTIVE NO for DDTYPE=ERROR.

This example specifies ORDER NO to demonstrate an additional override capability with the IFALLOC option. Because the table has two indexes and ORDER NO is specified, dynamic allocation would normally allocate two work (SYSUT1) files. However, because IFALLOC=USE is specified for SYSUT1 in the options module and the JCL contains a DD statement for a SYSUT101 data set, LOADPLUS uses that data set and dynamically allocates only one additional SYSUT1*nn* data set.

The command statement shows overrides of various options in the options module, including turning on dynamic allocation for the error data set. Also, as a result of specifying IFALLOC=FREE for the error data set, LOADPLUS ignores the DD statement for SYSERR. The LOAD command statement specifies the ENUMROWS option, which is required for dynamic allocation.

The additional dynamic allocation options perform the following functions:

- The DSNPAT option creates unique data set names.
- The SIZEPCT option tells LOADPLUS to allocate 500 percent of the primary and 50 percent of the secondary data set size calculation.
- The UNIT option tells LOADPLUS what type of device to use for allocation.
- The copy data sets are allocated with GDG names. The GDGLIMIT option tells LOADPLUS to keep five generations of each copy data set.

BMC Software, Inc., Confidential and Proprietary Information

This job allocates multiple SORTOUT and SORTWK data sets, demonstrating how LOADPLUS automatically multitasks for optimal performance when you use dynamic allocation.

LOADPLUS updates statistics in both the DB2 catalog and the DASD MANAGER PLUS statistics tables.

The JCL for example 11 follows:

```
//        JOB
//EXAMPLEB EXEC PGM=AMUUMAIN,
//          PARM='&SSID,EXAMPLEB,NEW,,MSGLEVEL(1),AMU$OP6E'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//SYSREC   DD DSN=AMU.QA.RGRTESTS(EX2DATA),DISP=SHR
//SYSERR   DD DSN=AMU.EXAMPLEB.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,DELETE)
//SYSUT101 DD DSN=AMU.EXAMPLEB.SYSUT101,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,DELETE)
//SYSIN    DD *
   LOAD REPLACE ORDER NO
     UPDATEDB2STATS YES BMCSTATS YES
     COPY YES
     REDEFINE NO
   INTO TABLE EXAMPLEB.TBL1
       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
       ,SELLER_ID           POSITION(62:71) CHARACTER
       )
   LOG NO
    ENUMROWS(900,900,100)
    DDTYPE LOAD     THRESHLD 50000  UNIT(WORK,CART)
    DDTYPE SORTWORK DSNPAT 'NONE'
    DDTYPE DISCARD  DSNPAT 'AMU.&UTIL..T&TIME.&DDNAME'
    DDTYPE ERROR    ACTIVE YES  IFALLOC FREE  SIZEPCT(500,50)
    DDTYPE LOCPFCPY ACTIVE YES GDGLIMIT(5)
        DSNPAT 'AMU.EXAMPLEB.LOCPFCPY.&DDNAME.(+1)'
    DDTYPE LOCBFCPY ACTIVE YES GDGLIMIT(5)
        DSNPAT 'AMU.EXAMPLEB.LOCBFCPY.&DDNAME.(+1)'
    DDTYPE REMPFCPY ACTIVE YES GDGLIMIT(5)
        DSNPAT 'AMU.EXAMPLEB.REMPFCPY.&DDNAME.(+1)'
    DDTYPE REMBFCPY ACTIVE YES GDGLIMIT(5)
        DSNPAT 'AMU.EXAMPLEB.REMBFCPY.&DDNAME.(+1)'
/*
```

BMC Software, Inc., Confidential and Proprietary Information

The SYSPRINT for example 11 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                 COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/10/2003    9:28:59 ...
BMC50002I UTILITY ID = 'EXAMPLEB'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTH'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7876K,ABOVE 16M=1666996K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = YES                 YES                 YES                 NO                  YES
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0
```

(continued on following page)

(example 11 continued from preceding page)

```
BMC50470I DDTYPE   = LOCPFCPY              LOCBFCPY              REMPFCPY              REMBFCPY
BMC50470I ACTIVE   = YES                   YES                   YES                   YES
BMC50470I IFALLOC  = USE                   USE                   USE                   USE
BMC50470I SMS      = NO                    NO                    NO                    NO
BMC50470I SMSUNIT  = NO                    NO                    NO                    NO
BMC50470I SIZEPCT  = (100,100)             (100,100)             (100,100)             (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)           (NONE,NONE)           (NONE,NONE)           (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)           (NONE,NONE)           (NONE,NONE)           (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)           (NONE,NONE)           (NONE,NONE)           (NONE,NONE)
BMC50470I THRESHLD = 0                     0                     0                     0
BMC50470I MAXEXTSZ = 0                     0                     0                     0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                     5                     5                     5
BMC50470I GDGEMPTY = NO                    NO                    NO                    NO
BMC50470I GDGSCRAT = NO                    NO                    NO                    NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD REPLACE ORDER NO
BMC50102I     UPDATEDB2STATS YES BMCSTATS YES
BMC50102I     COPY YES
BMC50102I     REDEFINE NO
BMC50102I   INTO TABLE EXAMPLEB.TBL1
BMC50102I       (DATE_OF_SALE        POSITION(72:77) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(52:61) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:21) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(22:30) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(31:39) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(42:50) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(62:71) CHARACTER
BMC50102I       )
BMC50102I   LOG NO
BMC50102I    ENUMROWS(900,900,100)
BMC50102I    DDTYPE LOAD     THRESHLD 50000  UNIT(WORK,CART)
BMC50102I    DDTYPE SORTWORK DSNPAT 'NONE'
BMC50102I    DDTYPE DISCARD  DSNPAT 'AMU.&UTIL..T&TIME.&DDNAME'
BMC50102I    DDTYPE ERROR    ACTIVE YES  IFALLOC FREE  SIZEPCT(500,50)
BMC50102I    DDTYPE LOCPFCPY ACTIVE YES GDGLIMIT(5)
BMC50102I        DSNPAT 'AMU.EXAMPLEB.LOCPFCPY.&DDNAME.(+1)'
BMC50102I    DDTYPE LOCBFCPY ACTIVE YES GDGLIMIT(5)
BMC50102I        DSNPAT 'AMU.EXAMPLEB.LOCBFCPY.&DDNAME.(+1)'
BMC50102I    DDTYPE REMPFCPY ACTIVE YES GDGLIMIT(5)
BMC50102I        DSNPAT 'AMU.EXAMPLEB.REMPFCPY.&DDNAME.(+1)'
BMC50102I    DDTYPE REMBFCPY ACTIVE YES GDGLIMIT(5)
BMC50102I        DSNPAT 'AMU.EXAMPLEB.REMBFCPY.&DDNAME.(+1)'

BMC51424I ANALYZE PHASE WILL BE PERFORMED DUE TO DYNAMIC WORK FILE ALLOCATION
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50425I &JOBNAME  = AMUEXPB   &STEPNAME = EXAMPLEB  &DB       = AMUEXBDB  &TS       = EXBTS     &UID      = RDAMZL2
BMC50425I &RESUME   = N         &REPLACE  = Y         &DATE     = 091003    &TIME     = 092859    &SSID     = DEBA
BMC50425I &UTIL     = EXAMPLEB  &UTILPFX  = EXAMPLEB  &UTILSFX  =           &DATE8    = 09102003  &GRPNM    = DEBA
BMC50425I &VCAT     = DEBACAT   &TIME4    = 0928      &DATEJ    = 2003253
```

(continued on following page)

(example 11 continued from preceding page)

```
BMC50445I LOADPLUS DYNAMIC FILE ALLOCATION REPORT

BMC50446I                                                    UNIT OR                      KBYTES   KBYTES     ALOC      ALOC
BMC50447I DDNAME   DSNAME                                    DATACLAS MGMTCLAS STORCLAS      PRI      SEC      PRI       SEC

BMC50448I BMCCPY   AMU.EXAMPLEB.LOCPFCPY.BMCCPY.G0001V00     SYSALLDA                        120       12        3         1  TRK
BMC50448I BMCCPZ   AMU.EXAMPLEB.LOCBFCPY.BMCCPZ.G0001V00     SYSALLDA                        120       12        3         1  TRK
BMC50448I BMCRCY   AMU.EXAMPLEB.REMPFCPY.BMCRCY.G0001V00     SYSALLDA                        120       12        3         1  TRK
BMC50448I BMCRCZ   AMU.EXAMPLEB.REMBFCPY.BMCRCZ.G0001V00     SYSALLDA                        120       12        3         1  TRK
BMC50448I SORTOUT1 RDAMZL2.EXAMPLEB.SORTOUT1                 WORK                             11        2        1         1  TRK
BMC50448I SORTOUT2 RDAMZL2.EXAMPLEB.SORTOUT2                 WORK                             11        2        1         1  TRK
BMC50448I SORTOUT3 RDAMZL2.EXAMPLEB.SORTOUT3                 WORK                             11        2        1         1  TRK
BMC50448I SORTOUT4 RDAMZL2.EXAMPLEB.SORTOUT4                 WORK                             11        2        1         1  TRK
BMC50448I SORTOUT5 RDAMZL2.EXAMPLEB.SORTOUT5                 WORK                             11        2        1         1  TRK
BMC50448I SORTOUT6 RDAMZL2.EXAMPLEB.SORTOUT6                 WORK                             11        2        1         1  TRK
BMC50448I SORTOUT7 RDAMZL2.EXAMPLEB.SORTOUT7                 WORK                             11        2        1         1  TRK
BMC50448I SORTOUT8 RDAMZL2.EXAMPLEB.SORTOUT8                 WORK                             11        2        1         1  TRK
BMC50448I SORTWK01                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK02                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK03                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK04                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK05                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK06                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK07                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK08                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK09                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK10                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK11                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SORTWK12                                           SYSALLDA                          5        1        1         1  TRK
BMC50448I SYSDISC  AMU.EXAMPLEB.T092859.SYSDISC              SYSALLDA                          7        1        1         1  TRK
BMC50448I SYSERR   RDAMZL2.EXAMPLEB.SYSERR                   SYSALLDA                         10        0        1         0  TRK
BMC50448I SYSUT102 RDAMZL2.EXAMPLEB.SYSUT102                 SYSALLDA                         14        2        1         1  TRK

BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SYSUT101'
BMC50474I BELOW 16M = 7304K, ABOVE 16M = 1655048K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 8, #READERS = 1, INDEX TASKS = 1, TIME = 59136
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 7, #READERS = 1, INDEX TASKS = 1, TIME = 66696
BMC51496I PRELOAD ANALYZE, #SORTS = 8, #READERS = 1, INDEX TASKS = 1
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 6, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  854 PHYSICAL (854 LOGICAL) RECORDS READ FROM SYSREC
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 6: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 5: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 4: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 3: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 7: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 8: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 9: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 2: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 6: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 7: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 8: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 9: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 3: SORT TASK, XBLK XFERS = 0, EMPTY WAITS = 0, FULL WAITS = 1
BMC51510I 4: SORT TASK, XBLK XFERS = 0, EMPTY WAITS = 0, FULL WAITS = 1
BMC51510I 5: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC50481I 10: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 10: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:01
BMC51510I 10: INDEX TASK, XBLK XFERS = 6, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSREC, I/OS = 87, I/O WAITS = 9, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT1, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT2, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT3, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT4, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT5, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT6, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT7, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT8, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT102, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 52, XFERS = 12, EMPTY WAITS = 0, FULL WAITS =15
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51471I PRELOAD STATISTICS:  454 ROWS SELECTED FOR PARTITION 1
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A001' WILL REQUIRE 12 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A001' WILL REQUIRE 5 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  59 ROWS SELECTED FOR PARTITION 2
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A002' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A002' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  60 ROWS SELECTED FOR PARTITION 3
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A003' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A003' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 4
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A004' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A004' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  62 ROWS SELECTED FOR PARTITION 5
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 11 continued from preceding page)

```
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A005' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A005' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 6
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A006' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A006' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  19 ROWS SELECTED FOR PARTITION 7
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A007' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A007' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  100 ROWS SELECTED FOR PARTITION 8
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A008' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A008' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  100 ROWS SELECTED FOR PARTITION 9
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A009' WILL REQUIRE 4 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A009' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 10
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A010' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A010' WILL REQUIRE 3 PAGES (APPROX.)
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXBDB.IX1.I0001.A001' MAY REQUIRE 8 PAGES
BMC51472I PRELOAD PHASE STATISTICS:  854 ROWS SELECTED FOR SPACE 'AMUEXBDB.EXBTS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:02

BMC50474I BELOW 16M = 7376K, ABOVE 16M = 1664780K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 8, #COPY TASKS = 1, #INDEX TASKS = 2
BMC50370I STARTING 1 IMAGE COPY TASKS
BMC51508I MAX INDEX TASKS = 2, INDEXES PER TASK = 1, SORTWKS PER TASK = 1, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 8, MAX PARTS PER TASK = 2, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 2
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEXBDB.EXBTS' DELETED
BMC50476I DDNAME = SORTOUT2, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT3, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:01
BMC50482I 4: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A009'
BMC50477I 4: PARTITION =   9, ROWS/KEYS = 100, I/O WAITS = 5, DDNAME = SYS00022
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A001'
BMC50477I 1: PARTITION =   1, ROWS/KEYS = 454, I/O WAITS = 7, DDNAME = SYS00020
BMC51474I BUILD STATISTICS:  454 KEYS LOADED INTO PARTITION 1
BMC51474I LOAD STATISTICS:  100 ROWS LOADED INTO PARTITION 9
BMC50482I 5: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A001'
BMC50477I 5: PARTITION =   1, ROWS/KEYS = 454, I/O WAITS = 5, DDNAME = SYS00025
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A008'
BMC50477I 0: PARTITION =   8, ROWS/KEYS = 100, I/O WAITS = 5, DDNAME = SYS00024
BMC50482I 3: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A003'
BMC50477I 3: PARTITION =   3, ROWS/KEYS = 60, I/O WAITS = 5, DDNAME = SYS00023
BMC51474I LOAD STATISTICS:  454 ROWS LOADED INTO PARTITION 1
BMC51474I LOAD STATISTICS:  100 ROWS LOADED INTO PARTITION 8
BMC51474I LOAD STATISTICS:  60 ROWS LOADED INTO PARTITION 3
BMC50476I DDNAME = SORTOUT5, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT6, I/OS = 4, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50482I 6: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A007'
BMC50477I 6: PARTITION =   7, ROWS/KEYS = 19, I/O WAITS = 5, DDNAME = SYS00026
BMC50482I 7: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A002'
BMC50477I 7: PARTITION =   2, ROWS/KEYS = 59, I/O WAITS = 5, DDNAME = SYS00027
BMC50476I DDNAME = SORTOUT1, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51474I LOAD STATISTICS:  59 ROWS LOADED INTO PARTITION 2
BMC51474I LOAD STATISTICS:  19 ROWS LOADED INTO PARTITION 7
BMC50476I DDNAME = SORTOUT4, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50482I 2: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX1.I0001.A001'
BMC50477I 2: PARTITION =   0, ROWS/KEYS = 854, I/O WAITS = 7, DDNAME = SYS00028
BMC50476I DDNAME = SORTOUT8, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  854 KEYS LOADED INTO INDEX 'EXAMPLEB.IX1'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A002'
BMC50477I 1: PARTITION =   2, ROWS/KEYS = 59, I/O WAITS = 7, DDNAME = SYS00029
BMC51474I BUILD STATISTICS:  59 KEYS LOADED INTO PARTITION 2
BMC50482I 6: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A005'
BMC50477I 6: PARTITION =   5, ROWS/KEYS = 62, I/O WAITS = 5, DDNAME = SYS00030
BMC51474I LOAD STATISTICS:  62 ROWS LOADED INTO PARTITION 5
BMC50476I DDNAME = SORTOUT7, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A001'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A003'
BMC50477I 1: PARTITION =   3, ROWS/KEYS = 60, I/O WAITS = 7, DDNAME = SYS00031
BMC51474I BUILD STATISTICS:  60 KEYS LOADED INTO PARTITION 3
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A002'
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00033
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 4
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A003'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A005'
BMC50477I 1: PARTITION =   5, ROWS/KEYS = 62, I/O WAITS = 7, DDNAME = SYS00035
BMC51474I BUILD STATISTICS:  62 KEYS LOADED INTO PARTITION 5
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A004'
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A006'
BMC50477I 0: PARTITION =   6, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00037
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 6
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A005'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A007'
BMC50477I 1: PARTITION =   7, ROWS/KEYS = 19, I/O WAITS = 7, DDNAME = SYS00039
```

(continued on following page)

(example 11 continued from preceding page)

```
BMC51474I BUILD STATISTICS:  19 KEYS LOADED INTO PARTITION 7
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A010'
BMC50477I 0: PARTITION =  10, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00041
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 10
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A006'
BMC51475I LOAD STATISTICS:  854 ROWS LOADED INTO TABLE SPACE 'AMUEXBDB.EXBTS'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A007'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A008'
BMC50477I 1: PARTITION =   8, ROWS/KEYS = 100, I/O WAITS = 7, DDNAME = SYS00043
BMC51474I BUILD STATISTICS:  100 KEYS LOADED INTO PARTITION 8
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A008'
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A009'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A009'
BMC50477I 1: PARTITION =   9, ROWS/KEYS = 100, I/O WAITS = 7, DDNAME = SYS00045
BMC51474I BUILD STATISTICS:  100 KEYS LOADED INTO PARTITION 9
BMC50476I DDNAME = SYSUT102, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50375I FULL IMAGE COPY STARTED FOR DATASET = 'DEBACAT.DSNDBD.AMUEXBDB.EXBTS.I0001.A010'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A004'
BMC50477I 1: PARTITION =   4, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00049
BMC51474I BUILD STATISTICS:  0 KEYS LOADED INTO PARTITION 4
BMC50476I DDNAME = BMCCPY, I/OS = 28, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC50376I 41 PAGES COPIED TO DATASET = 'AMU.EXAMPLEB.LOCPFCPY.BMCCPY.G0001V00'
BMC50376I 41 PAGES COPIED TO DATASET = 'AMU.EXAMPLEB.LOCBFCPY.BMCCPZ.G0001V00'
BMC50376I 41 PAGES COPIED TO DATASET = 'AMU.EXAMPLEB.REMPFCPY.BMCRCY.G0001V00'
BMC50376I 41 PAGES COPIED TO DATASET = 'AMU.EXAMPLEB.REMBFCPY.BMCRCZ.G0001V00'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A006'
BMC50477I 1: PARTITION =   6, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00050
BMC51474I BUILD STATISTICS:  0 KEYS LOADED INTO PARTITION 6
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXBDB.IX0.I0001.A010'
BMC50477I 1: PARTITION =  10, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00051
BMC51474I BUILD STATISTICS:  0 KEYS LOADED INTO PARTITION 10
BMC51476I BUILD STATISTICS:  854 KEYS LOADED INTO INDEX 'EXAMPLEB.IX0'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:06

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEXBDB.EXBTS
BMC50513I             PARTS   = 10        TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 41            SPACE   = 150
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP    KSAVED PSAVED
BMC50512I   1        12        454          0          0        2        0   85    0      15    1           0    0         0     0
BMC50512I   2         4         59          0          0        2        0   55    0      15    1           0    0         0     0
BMC50512I   3         4         60          0          0        2        0   56    0      15    1           0    0         0     0
BMC50512I   4         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   5         4         62          0          0        2        0   58    0      15    1           0    0         0     0
BMC50512I   6         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   7         3         19          0          0        2        0   36    0      15    1           0    0         0     0
BMC50512I   8         4        100          0          0        2        0   93    0      15    1           0    0         0     0
BMC50512I   9         4        100          0          0        2        0   93    0      15    1           0    0         0     0
BMC50512I  10         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLEB.TBL1                        854         21         0      68      51       0
BMC50510I INDEX       EXAMPLEB.IX0  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLEB.TBL1     COLUMN  DATE_OF_SALE
BMC50514I             NACTIVE = 51            SPACE   = 10
BMC50542I             FIRSTKEY= 12            FULLKEY = 854           NLEAF   = 11
BMC50543I             LEVELS  = 2                                     PCTCLUST= 99
BMC50544I             KEYLEN  = 9                                     COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'1990012500800000'  LOW2KEY= X'1982052400800000'
BMC50547I             HIGH2KEY=        LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   1         6        454       50          0          3        0        4        2     2         1       1
BMC50532I   2         5         59        0          1          0        0        6        1     2         1       1
BMC50532I   3         5         60        0          1          0        0        6        1     2         1       1
BMC50532I   4         5          0        0          0          0        0        7        1     2         1       1
BMC50532I   5         5         62        0          1          1        0        6        1     2         1       1
BMC50532I   6         5          0        0          0          0        0        7        1     2         1       1
BMC50532I   7         5         19        0          1          0        0        7        1     2         1       1
BMC50532I   8         5        100        0          1          0        0        6        1     2         1       1
BMC50532I   9         5        100        0          1          0        0        6        1     2         1       1
BMC50532I  10         5          0        0          0          0        0        7        1     2         1       1
BMC50510I INDEX       EXAMPLEB.IX1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLEB.TBL1     COLUMN  SELLER_ID
BMC50514I             NACTIVE = 6             SPACE   = 1
BMC50542I             FIRSTKEY= 3             FULLKEY = 3             NLEAF   = 2
BMC50543I             LEVELS  = 2                                     PCTCLUST= 58
BMC50544I             KEYLEN  = 10                                    COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = D
BMC50546I             HIGH2KEY= X'E64BC14BD9C5C5C4'  LOW2KEY= X'E64BC14BD9C5C5C4'
BMC50547I             HIGH2KEY= W.A.REED  LOW2KEY= W.A.REED
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         6        854       50         12          3        0        7        2     2         1       1
BMC50505I BMCSTATS UPDATED IN THE DASD MANAGER PLUS DATABASE
BMC50506I STATISTICS COLUMNS UPDATED IN THE DB2 CATALOG
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

BMC Software, Inc., Confidential and Proprietary Information

Example 12: LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY of a Partitioned Table Space

In this example, LOADPLUS uses one of the online load options to load data into a partitioned table space that has five partitions. SYSPRINT A (starting on page 5-86) shows the output when this job is run with ORDER NO. Because ORDER NO is specified, LOADPLUS sorts only error information for this type of load. Therefore, a single small SORTWK file is allocated.

For the second part of this example, the LOAD command is changed to specify ORDER YES. In addition, the EXEC statement in the JCL includes an options module override. The new options module activates dynamic work file allocation for all work files by specifying ACTIVE YES for each DDTYPE. The LOAD command provides additional overrides for dynamic allocation by using the DSNPAT command option to specify a different pattern for the work file data set names. The LOAD command also includes the DELETEFILES YES SYSDISC YES option to delete the work files and SYSDISC file after the load completes. SYSPRINT B (starting on page 5-93) shows the output from the ORDER YES job.

In these examples, the Apply Plus load library is included in the STEPLIB. However, the Apply Plus load library could have been included in the JOBLIB or LINKLIST instead.

In the output, note the BMC51576 messages, which return information from Apply Plus.

The JCL for example 12A follows:

```
//        JOB
//EXAMPLEC EXEC PGM=AMUUMAIN,
//        PARM='&SSID,EXAMPLEC,NEW,,MSGLEVEL(1),AMU$OPTO'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB1
//         DD DISP=SHR,DSN=&APLIB1
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSREC   DD DSN=AMU.QA.RGRTESTS.DTAEX12,DISP=SHR
//SYSERR   DD DSN=AMU.EXAMPLC.SYSERR,
//            DISP=(MOD,CATLG),
//            UNIT=WORK,SPACE=(CYL,(5,2),RLSE)
//SYSDISC  DD DSN=AMU.EXAMPLC.SYSDISC,
//            DISP=(MOD,CATLG),
//            UNIT=WORK,SPACE=(CYL,(10,01),RLSE)
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SORTWK01 DD DSN=AMU.EXAMPLC.SORTWK01,
//            DISP=(NEW,DELETE,DELETE),
//            UNIT=WORK,SPACE=(CYL,(10,3),RLSE)
//SYSIN    DD *
  LOAD DATA RESUME YES
   SHRLEVEL CHANGE SQLAPPLY
   ORDER NO
   INTO TABLE EXAMPLEC.TBL1
   (RECNO        POS(1:4)   INTEGER,
    INPUT_A      POS(5:14)  CHAR(10),
    RECNO2       POS(15:18) INTEGER,
    INPUT_B      POS(19:28) CHAR(10),
    INPUT_C      POS(29:43) CHAR(15),
    INPUT_D      POS(44:58) CHAR(15)
    )
/*
```

The JCL for example 12B follows:

```
//        JOB
//EXAMPLCB EXEC PGM=AMUUMAIN,
//        PARM='&SSID,EXAMPLCB,NEW,,MSGLEVEL(1),AMU$OPTE'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB1
//         DD DISP=SHR,DSN=&APLIB1
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSREC   DD DSN=AMU.QA.RGRTESTS.DTAEX12,DISP=SHR
//SYSIN    DD *
  LOAD DATA RESUME YES
   SHRLEVEL CHANGE
   ENUMROWS(100000)
   ORDER YES
   DDTYPE LOAD     DSNPAT '&UID..RESUME&RESUME.&DDNAME'
   DDTYPE WORK     DSNPAT '&UID..RESUME&RESUME.&DDNAME'
   DDTYPE SORTWORK DSNPAT '&UID..RESUME&RESUME.&DDNAME'
   DDTYPE DISCARD  DSNPAT '&UID..RESUME&RESUME.&DDNAME'
   DDTYPE ERROR    DSNPAT '&UID..RESUME&RESUME.&DDNAME'
   DELETEFILES YES SYSDISC YES
   INTO TABLE EXAMPLEC.TBL1
   (RECNO        POS(1:4)   INTEGER,
    INPUT_A      POS(5:14)  CHAR(10),
    RECNO2       POS(15:18) INTEGER,
    INPUT_B      POS(19:28) CHAR(10),
    INPUT_C      POS(29:43) CHAR(15),
    INPUT_D      POS(44:58) CHAR(15)
    )
/*
```

BMC Software, Inc., Confidential and Proprietary Information

Output from ORDER NO

The SYSPRINT for example 12A follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                  COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   15:36:42 ...
BMC50002I UTILITY ID = 'EXAMPLEC'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTO'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7856K,ABOVE 16M=1665236K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=STANDARD
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST
```

(continued on following page)

(example 12A continued from preceding page)

```
BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I  LOAD DATA RESUME YES
BMC50102I   SHRLEVEL CHANGE SQLAPPLY
BMC50102I   ORDER NO
BMC50102I   INTO TABLE EXAMPLEC.TBL1
BMC50102I   (RECNO        POS(1:4)   INTEGER,
BMC50102I    INPUT_A      POS(5:14)  CHAR(10),
BMC50102I    RECNO2       POS(15:18) INTEGER,
BMC50102I    INPUT_B      POS(19:28) CHAR(10),
BMC50102I    INPUT_C      POS(29:43) CHAR(15),
BMC50102I    INPUT_D      POS(44:58) CHAR(15)
BMC50102I    )

BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:02

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7380K, ABOVE 16M = 1663896K, CPUS = 3
BMC51495I COMBINED OPTIMIZATION, RC = 0, #SORTS = 5, #READERS = 1, INDEX TASKS = 0, TIME = 83712
BMC51495I COMBINED OPTIMIZATION, RC = 14, #SORTS = 4, #READERS = 1, INDEX TASKS = 0, TIME = 83712
BMC51496I COMBINED ANALYZE, #SORTS = 5, #READERS = 1, INDEX TASKS = 0
BMC51576I 15:36:54 000 APT0000I APPLY PLUS FOR DB2 FOR OS/390 V1.6.00
BMC51576I                       Copyright (c) 1999-2003 BMC Software, Inc. as an unpublished licensed work.  All rights reserved.
BMC51576I 15:36:54 000 APT0001I Date: Thu Sep  4, 2003
BMC51576I 15:36:55 000 APT0085I /STARTUP/
BMC51576I                       SSID=DEBA
BMC51576I                       INPUTTYPE={3895D640-0BF1-11D4-8027-00104B39B10F}
BMC51576I                       PLANNAME=AMUZDIST
BMC51576I                       DBMSTYPE=DB2
BMC51576I                       DISTRIBUTIONTYPE=OBJECT
BMC51576I 15:36:55 000 APT0085I /CONFIGURATION/
BMC51576I                       FILENAME=NONE
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 12A continued from preceding page)

```
BMC51576I 15:36:55 000 APT0085I /SQL/
BMC51576I                       FORMAT=PACKED72
BMC51576I 15:36:55 000 APT0085I /COMMITTRIGGERS/
BMC51576I                       STATEMENTCOUNT=2500
BMC51576I                       COMMITONDEMAND=NO
BMC51576I 15:36:55 000 APT0085I /OBJECTMAP/
BMC51576I                       SOURCETABLE=0X0386.0X0003
BMC51576I                       TARGETTABLE=EXAMPLEC.TBL1
BMC51576I 15:36:55 000 APT0085I /AGENT/
BMC51576I                       MAXAGENTS=10
BMC51576I                       INITIALAGENTS=5
BMC51576I                       MAXPREPARES=50
BMC51576I 15:36:55 000 APT0085I /TEMPORARYFILES/
BMC51576I                       UNIT=SYSALLDA
BMC51576I 15:36:55 000 APT0085I /DIAGNOSTICOUTPUT/
BMC51576I                       FILENAME=APTDIAG
BMC51576I 15:36:55 000 APT0085I /RESTART/
BMC51576I                       RESTARTID=EXAMPLEC
BMC51576I                       RESTARTTYPE=NEW
BMC51576I                       TABLENAME=BMCAPT_APTRSCL
BMC51576I                       RETAINTIME=7
BMC51576I                       RETRY=3
BMC51576I 15:36:55 000 APT0085I /DISTRIBUTIONTUNING/
BMC51576I                       SIMPLETSCLUSTERING=NO
BMC51576I                       RICLUSTERING=NO
BMC51576I                       PARTITIONCLUSTERING=NO
BMC51576I                       URCHECKRI=YES
BMC51576I 15:36:55 000 APT0085I /ANYCONFLICT/
BMC51576I                       CODE=-803
BMC51576I                       ACTION=CALLBACK
BMC51576I                       ACTION=CONTINUE
BMC51576I                       CODE=-530
BMC51576I                       ACTION=CALLBACK
BMC51576I                       ACTION=CONTINUE
BMC51576I                       CODE=-359
BMC51576I                       ACTION=CALLBACK
BMC51576I                       ACTION=CONTINUE
BMC51576I                       CODE=SQLWARN0
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN1
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN2
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN3
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN4
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN5
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN6
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN7
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN8
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN9
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARNA
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=NEGATIVE
BMC51576I                       ACTION=ABORT
BMC51576I                       CODE=POSITIVE
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=MULTIPLEROWS
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=NOROWS
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=TIMEOUT
BMC51576I                       ACTION=RETRY
BMC51576I                       CODE=RICONFLICT
BMC51576I                       ACTION=ABORT
BMC51576I 15:36:55 000 APT0085I /CONFLICT/
BMC51576I                       RETRYLIMIT=COUNT
BMC51576I                       RETRYVALUE=5
BMC51576I                       DATATYPE=ID
BMC51576I                       DATATYPE=SQLCODE
BMC51576I                       DATATYPE=MESSAGE
BMC51576I                       RETRYFAIL=ABORT
BMC51576I                       RETRYDELAY=0
BMC51576I 15:36:55 000 APT0085I /CONFLICTFILE/
BMC51576I                       SPACE=TRACK
BMC51576I                       SPACEPRI=15
BMC51576I                       SPACESEC=5
BMC51576I                       FILENAMEMODEL=&SYSUID..BMCAPT.&REQUESTID..D&DATE..T&TIME.
BMC51576I                       SINGLEFILE=NO
BMC51576I                       DISP=NEW
```

(continued on following page)

(example 12A continued from preceding page)

```
BMC51576I                       MEDIATYPE=FILE
BMC51576I                       RELEASE=YES
BMC51576I 15:36:55 000 APT0085I /BIND/
BMC51576I                       FREEOPTION=ALL
BMC51576I                       PLANPREFIX=APT
BMC51576I 15:36:55 000 APT0085I /BINDTUNING/
BMC51576I                       STATEMENTCOUNT=1
BMC51576I                       MAXPACKAGES=25
BMC51576I                       MAXFAILEDBINDS=5
BMC51576I                       MAXRETRIES=5
BMC51576I                       PACKAGEUSAGE=500
BMC51576I                       STATEMENTUSAGE=200
BMC51576I                       SYNCHRONOUS=NO
BMC51576I 15:36:58 001 APT0097I Request ID: A826      Restart ID: EXAMPLEC
BMC51576I 15:36:59 000 APT0019I File DBRMLIB :SYS03247.T153658.RA000.AMUEX12.R0251388 allocated - elapsed time: 00:00:00
BMC51576I 15:36:59 001 APT0003I  DSN SYSTEM(DEBA)
BMC51576I 15:36:59 001 APT0003I  BIND PLAN(APTA826) +
BMC51576I 15:36:59 001 APT0003I       OWNER(RDAMZL) +
BMC51576I 15:36:59 001 APT0003I       ENCODING(EBCDIC) +
BMC51576I 15:36:59 001 APT0003I       KEEPDYNAMIC(YES) +
BMC51576I 15:36:59 001 APT0003I       PKLIST(AMUZDIST.APTREB2, APTA826.*) +
BMC51576I 15:36:59 001 APT0003I       RELEASE(COMMIT) +
BMC51576I 15:36:59 001 APT0003I       ACTION(REPLACE) +
BMC51576I 15:36:59 001 APT0003I      RETAIN +
BMC51576I 15:36:59 001 APT0003I       ISOLATION(CS) +
BMC51576I 15:36:59 001 APT0003I       VALIDATE(RUN)
BMC51576I 15:36:59 001 APT0003I  BIND PACKAGE(APTA826) +
BMC51576I 15:36:59 001 APT0003I       OWNER(RDAMZL) +
BMC51576I 15:36:59 001 APT0003I       ENCODING(EBCDIC) +
BMC51576I 15:36:59 001 APT0003I       KEEPDYNAMIC(YES) +
BMC51576I 15:36:59 001 APT0003I       MEMBER(A8260000) +
BMC51576I 15:36:59 001 APT0003I       RELEASE(COMMIT) +
BMC51576I 15:36:59 001 APT0003I       ACTION(REPLACE) +
BMC51576I 15:36:59 001 APT0003I       ISOLATION(CS) +
BMC51576I 15:36:59 001 APT0003I       VALIDATE(RUN)
BMC51576I 15:36:59 001 APT0003I  END
BMC51576I 15:37:03 001 APT0032I DSN output follows:
BMC51576I                       1READY
BMC51576I                         DSN SYSTEM(DEBA)
BMC51576I                        DSN
BMC51576I                         BIND PLAN(APTA826) OWNER(RDAMZL) ENCODING(EBCDIC) KEEPDYNAMIC(YES) PKLIST(AMUZ
BMC51576I                       DIST.APTREB2, APTA826.*) RELEASE(COMMIT) ACTION(REPLA
BMC51576I                        CE) RETAIN ISOLATION(CS) VALIDATE(RUN)
BMC51576I                        DSNT252I  *DEBA DSNTBCM1 BIND OPTIONS FOR PLAN APTA826
BMC51576I                                   ACTION        REPLACE RETAIN
BMC51576I                                   OWNER         RDAMZL
BMC51576I                                   VALIDATE      RUN
BMC51576I                                   ISOLATION     CS
BMC51576I                                   ACQUIRE       USE
BMC51576I                                   RELEASE       COMMIT
BMC51576I                                   EXPLAIN       NO
BMC51576I                                   DYNAMICRULES  RUN
BMC51576I                        DSNT253I  *DEBA DSNTBCM1 BIND OPTIONS FOR PLAN APTA826
BMC51576I                                   NODEFER       PREPARE
BMC51576I                                   CACHESIZE     1024
BMC51576I                                   QUALIFIER     RDAMZL
BMC51576I                                   CURRENTSERVER
BMC51576I                                   CURRENTDATA   YES
BMC51576I                                   DEGREE        1
BMC51576I                                   SQLRULES      DB2
BMC51576I                                   DISCONNECT    EXPLICIT
BMC51576I                                   NOREOPT       VARS
BMC51576I                                   KEEPDYNAMIC   YES
BMC51576I                                   IMMEDWRITE    NO
BMC51576I                                   DBPROTOCOL    DRDA
BMC51576I                                   OPTHINT
BMC51576I                                   ENCODING      EBCDIC(00037)
BMC51576I                                   PATH
BMC51576I                        DSNT200I  *DEBA  BIND FOR PLAN APTA826  SUCCESSFUL
BMC51576I                        DSN
BMC51576I                         BIND PACKAGE(APTA826) OWNER(RDAMZL) ENCODING(EBCDIC) KEEPDYNAMIC(YES) MEMBER(A
BMC51576I                       8260000) RELEASE(COMMIT) ACTION(REPLACE) ISOLATION(CS
BMC51576I                        ) VALIDATE(RUN)
BMC51576I                        DSNT254I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                                   ACTION        REPLACE
BMC51576I                                   OWNER         RDAMZL
BMC51576I                                   QUALIFIER     RDAMZL
BMC51576I                                   VALIDATE      RUN
BMC51576I                                   EXPLAIN       NO
BMC51576I                                   ISOLATION     CS
BMC51576I                                   RELEASE       COMMIT
BMC51576I                                   COPY
```

(continued on following page)

(example 12A continued from preceding page)

```
BMC51576I                        DSNT255I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                                   SQLERROR      NOPACKAGE
BMC51576I                                   CURRENTDATA   YES
BMC51576I                                   DEGREE        1
BMC51576I                                   DYNAMICRULES
BMC51576I                                   DEFER
BMC51576I                                   NOREOPT       VARS
BMC51576I                                   KEEPDYNAMIC   YES
BMC51576I                                   IMMEDWRITE    NO
BMC51576I                                   DBPROTOCOL    DRDA
BMC51576I                                   OPTHINT
BMC51576I                       1           ENCODING      EBCDIC(00037)
BMC51576I                                   PATH
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL BIND FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                        DSN
BMC51576I                         END
BMC51576I                        READY
BMC51576I                        END
BMC51576I 15:37:05 006 APT0079I Apply Processing Started - Agent: 1
BMC51576I 15:37:05 007 APT0079I Apply Processing Started - Agent: 2
BMC51576I 15:37:05 008 APT0079I Apply Processing Started - Agent: 3
BMC51576I 15:37:05 009 APT0079I Apply Processing Started - Agent: 4
BMC51576I 15:37:05 010 APT0079I Apply Processing Started - Agent: 5
BMC51576I 15:37:06 001 APT0148I Bind started for package A8260001 - Trigger statistics:
BMC51576I                          STATEMENTCOUNT.........................1
BMC51576I                          PACKAGEUSAGE...........................1
BMC51576I                          STATEMENTUSAGE.........................1
BMC51576I 15:37:06 001 APT0003I  DSN SYSTEM(DEBA)
BMC51576I 15:37:06 001 APT0003I  BIND PACKAGE(APTA826) +
BMC51576I 15:37:06 001 APT0003I       OWNER(RDAMZL) +
BMC51576I 15:37:06 001 APT0003I       ENCODING(EBCDIC) +
BMC51576I 15:37:06 001 APT0003I       KEEPDYNAMIC(YES) +
BMC51576I 15:37:06 001 APT0003I       MEMBER(A8260001) +
BMC51576I 15:37:06 001 APT0003I       RELEASE(COMMIT) +
BMC51576I 15:37:06 001 APT0003I       ACTION(REPLACE) +
BMC51576I 15:37:06 001 APT0003I       ISOLATION(CS) +
BMC51576I 15:37:06 001 APT0003I       VALIDATE(RUN)
BMC51576I 15:37:06 001 APT0003I  END
BMC51576I 15:37:10 001 APT0185I Statements committed: 5000
BMC51576I 15:37:12 001 APT0032I DSN output follows:
BMC51576I                       1READY
BMC51576I                         DSN SYSTEM(DEBA)
BMC51576I                        DSN
BMC51576I                         BIND PACKAGE(APTA826) OWNER(RDAMZL) ENCODING(EBCDIC) KEEPDYNAMIC(YES) MEMBER(A
BMC51576I                       8260001) RELEASE(COMMIT) ACTION(REPLACE) ISOLATION(CS
BMC51576I                        ) VALIDATE(RUN)
BMC51576I                        DSNT254I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                                   ACTION        REPLACE
BMC51576I                                   OWNER         RDAMZL
BMC51576I                                   QUALIFIER     RDAMZL
BMC51576I                                   VALIDATE      RUN
BMC51576I                                   EXPLAIN       NO
BMC51576I                                   ISOLATION     CS
BMC51576I                                   RELEASE       COMMIT
BMC51576I                                   COPY
BMC51576I                        DSNT255I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                                   SQLERROR      NOPACKAGE
BMC51576I                                   CURRENTDATA   YES
BMC51576I                                   DEGREE        1
BMC51576I                                   DYNAMICRULES
BMC51576I                                   DEFER
BMC51576I                                   NOREOPT       VARS
BMC51576I                                   KEEPDYNAMIC   YES
BMC51576I                                   IMMEDWRITE    NO
BMC51576I                                   DBPROTOCOL    DRDA
BMC51576I                                   OPTHINT
BMC51576I                                   ENCODING      EBCDIC(00037)
BMC51576I                                   PATH
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL BIND FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                        DSN
BMC51576I                         END
BMC51576I                        READY
BMC51576I                        END
BMC51576I 15:37:14 001 APT0185I Statements committed: 12500
BMC51576I 15:37:16 001 APT0118I Input statements processed: 61362
BMC51576I 15:37:19 001 APT0185I Statements committed: 17500
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:26
BMC51510I 1: READ TASK, XBLK XFERS = 129, EMPTY WAITS = 106, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  100000 PHYSICAL (100000 LOGICAL) RECORDS READ FROM SYSREC
BMC51480I 3: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:27.  20000 ROWS PASSED TO APPLY PLUS
BMC51480I 6: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:27.  0 ROWS PASSED TO APPLY PLUS
BMC51480I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:27.  20000 ROWS PASSED TO APPLY PLUS
```

(continued on following page)

(example 12A continued from preceding page)

```
BMC51576I 15:37:20 001 APT0185I Statements committed: 22500
BMC51480I 4: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:27.  20000 ROWS PASSED TO APPLY PLUS
BMC51510I 3: SORT TASK, XBLK XFERS = 26, EMPTY WAITS = 0, FULL WAITS = 4
BMC51510I 3: SORT TASK, APBUF XFERS = 43, EMPTY WAITS = 43, FULL WAITS = 0
BMC51510I 6: SORT TASK, XBLK XFERS = 0, EMPTY WAITS = 0, FULL WAITS = 1
BMC51510I 6: SORT TASK, APBUF XFERS = 0, EMPTY WAITS = 0, FULL WAITS = 0
BMC51510I 2: SORT TASK, XBLK XFERS = 26, EMPTY WAITS = 0, FULL WAITS = 4
BMC51510I 2: SORT TASK, APBUF XFERS = 43, EMPTY WAITS = 43, FULL WAITS = 0
BMC51510I 4: SORT TASK, XBLK XFERS = 26, EMPTY WAITS = 0, FULL WAITS = 3
BMC51510I 4: SORT TASK, APBUF XFERS = 43, EMPTY WAITS = 28, FULL WAITS = 0
BMC51480I 5: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:29.  40000 ROWS PASSED TO APPLY PLUS
BMC51510I 5: SORT TASK, XBLK XFERS = 51, EMPTY WAITS = 0, FULL WAITS = 3
BMC51510I 5: SORT TASK, APBUF XFERS = 86, EMPTY WAITS = 86, FULL WAITS = 0
BMC51576I 15:37:23 001 APT0185I Statements committed: 27500
BMC51576I 15:37:24 001 APT0118I Input statements processed: 100000
BMC51576I 15:37:24 001 APT0066I Input processing complete
BMC51576I 15:37:25 001 APT0185I Statements committed: 35000
BMC51576I 15:37:26 001 APT0185I Statements committed: 40000
BMC51576I 15:37:28 001 APT0185I Statements committed: 47500
BMC51576I 15:37:29 001 APT0185I Statements committed: 52500
BMC51576I 15:37:31 001 APT0185I Statements committed: 60000
BMC51576I 15:37:33 001 APT0185I Statements committed: 67500
BMC51576I 15:37:35 001 APT0185I Statements committed: 72500
BMC51576I 15:37:38 001 APT0185I Statements committed: 77500
BMC51576I 15:37:38 006 APT0080I Apply Processing Ended - Agent: 1
BMC51576I 15:37:40 001 APT0185I Statements committed: 82500
BMC51576I 15:37:40 009 APT0080I Apply Processing Ended - Agent: 4
BMC51576I 15:37:42 001 APT0185I Statements committed: 90000
BMC51576I 15:37:43 010 APT0080I Apply Processing Ended - Agent: 5
BMC51576I 15:37:44 007 APT0080I Apply Processing Ended - Agent: 2
BMC51576I 15:37:45 001 APT0185I Statements committed: 97500
BMC51576I 15:37:46 008 APT0080I Apply Processing Ended - Agent: 3
BMC51576I 15:37:46 001 APT0047I TABLE STATISTICS - EXAMPLEC.TBL1
BMC51576I                              INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                              100000         0         0         0     100000
BMC51576I 15:37:46 001 APT0048I PARTITION STATISTICS - EXAMPLEC.TBL1
BMC51576I                        PART    INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                           1     20000         0         0         0      20000
BMC51576I                           2     20000         0         0         0      20000
BMC51576I                           3     20000         0         0         0      20000
BMC51576I                           4     20000         0         0         0      20000
BMC51576I                           5     20000         0         0         0      20000
BMC51576I 15:37:46 001 APT0003I  DSN SYSTEM(DEBA)
BMC51576I 15:37:46 001 APT0003I  FREE PACKAGE(APTA826.*)
BMC51576I 15:37:46 001 APT0003I  FREE PLAN(APTA826)
BMC51576I 15:37:46 001 APT0003I  END
BMC51576I 15:38:57 001 APT0032I DSN output follows:
BMC51576I                       1READY
BMC51576I                         DSN SYSTEM(DEBA)
BMC51576I                       DSN
BMC51576I                         FREE PACKAGE(APTA826.*)
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260002.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260003.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260004.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260005.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260006.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260007.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260008.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260009.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260010.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260011.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260012.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260013.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260014.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260015.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260016.()
BMC51576I                       DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                  PACKAGE = DEBA.APTA826.A8260017.()
```

(continued on following page)

(example 12A continued from preceding page)

```
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260018.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260019.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260020.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260021.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260022.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260023.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260024.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260025.()
BMC51576I                       1DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260026.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260027.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260028.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260029.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260030.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260031.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260032.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260033.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260034.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260035.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260036.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260037.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260038.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260039.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260040.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260041.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260042.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260043.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260044.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260045.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260046.()
BMC51576I                        DSN
BMC51576I                         FREE PLAN(APTA826)
BMC51576I                        DSNT200I  *DEBA  FREE FOR PLAN APTA826  SUCCESSFUL
BMC51576I                        DSN
BMC51576I                         END
BMC51576I                        READY
BMC51576I                        END
BMC51576I 15:38:57 000 APT0020I File DBRMLIB :SYS03247.T153658.RA000.AMUEX12.R0251388 deallocated
BMC51576I 15:38:58 001 APT0111I INPUT PROCESSING STATISTICS
BMC51576I                                INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                                100000         0         0         0     100000
BMC51576I 15:38:58 001 APT0166I STATEMENTS COMMITTED STATISTICS
BMC51576I                        AGENT    INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                            1     20000         0         0         0      20000
BMC51576I                            2     20000         0         0         0      20000
BMC51576I                            3     20000         0         0         0      20000
BMC51576I                            4     20000         0         0         0      20000
BMC51576I                            5     20000         0         0         0      20000
BMC51576I                        ----- --------- --------- --------- --------- ----------
BMC51576I                        TOTAL    100000         0         0         0     100000
BMC51576I 15:38:58 001 APT0082I DYNAMIC SQL PROCESSING STATISTICS
BMC51576I                        AGENT    INSERT    UPDATE    DELETE    COMMIT  ROLLBACK      TOTAL
BMC51576I                            1     19270         0         0         0         0      19270
BMC51576I                            2         0         0         0         0         0          0
BMC51576I                            3         0         0         0         0         0          0
BMC51576I                            4     18330         0         0         0         0      18330
BMC51576I                            5      7676         0         0         0         0       7676
BMC51576I                        ----- --------- --------- --------- --------- --------- ----------
BMC51576I                        TOTAL     45276         0         0         0         0      45276
```

(continued on following page)

(example 12A continued from preceding page)

```
BMC51576I 15:38:58 001 APT0081I STATIC SQL PROCESSING STATISTICS
BMC51576I                       AGENT    INSERT    UPDATE    DELETE   MASSDEL    COMMIT  ROLLBACK   PREPARE   EXECUTE      TOTAL
BMC51576I                           1       730         0         0         0         8         0         1     19270      20009
BMC51576I                           2     20000         0         0         0         8         0         0         0      20008
BMC51576I                           3     20000         0         0         0         8         0         0         0      20008
BMC51576I                           4      1670         0         0         0         8         0         1     18330      20009
BMC51576I                           5     12324         0         0         0         8         0         1      7676      20009
BMC51576I                       ----- --------- --------- --------- --------- --------- --------- --------- ---------- ----------
BMC51576I                       TOTAL     54724         0         0         0        40         0         3     45276     100043
BMC51576I 15:38:58 001 APT0200I EXECUTE IMMEDIATE PROCESSING STATISTICS
BMC51576I                       AGENT    INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                           1         0         0         0         0          0
BMC51576I                           2         0         0         0         0          0
BMC51576I                           3         0         0         0         0          0
BMC51576I                           4         0         0         0         0          0
BMC51576I                           5         0         0         0         0          0
BMC51576I                       ----- --------- --------- --------- --------- ----------
BMC51576I                       TOTAL         0         0         0         0          0
BMC51576I 15:38:58 001 APT0182I OTHER STATISTICS
BMC51576I                       AGENT FAILPREPS   RETRIES DBMS ELAPSED TIME
BMC51576I                           1         0         0   00:00:26.909705
BMC51576I                           2         0         0   00:00:27.035152
BMC51576I                           3         0         0   00:00:23.319619
BMC51576I                           4         0         0   00:00:27.874438
BMC51576I                           5         0         0   00:00:28.610234
BMC51576I                       ----- --------- --------- -----------------
BMC51576I                       TOTAL         0         0   00:02:13.749152
BMC51576I 15:38:58 001 APT0183I Request elapsed time: 00:02:01.767247
BMC51576I 15:38:58 000 APT0067I Request completed Return code: 0
BMC50481I 7: SQLAPPLY TASK COMPLETE.  ELAPSED TIME = 00:02:06
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC, I/OS = 206, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51507I XBLKS = 22, XFERS = 129, EMPTY WAITS = 106, FULL WAITS =15
BMC51507I APBUFS = 15, XFERS = 215, EMPTY WAITS = 200, FULL WAITS =0
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51472I COMBINED PHASE STATISTICS:  100000 ROWS SELECTED FOR SPACE 'AMUEXCDB.EXCTS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I COMBINED PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC51475I LOAD STATISTICS:  100000 ROWS LOADED INTO TABLE SPACE 'AMUEXCDB.EXCTS'
BMC50004I COMBINED PHASE COMPLETE.  ELAPSED TIME = 00:02:08

BMC51577I                                    ROWS       DISCARDED
BMC51578I                                 INSERTED     SQLERRORS

BMC51579I SPACE AMUEXCDB.EXCTS               100000             0

BMC51579I PART  001                           20000             0
BMC51579I PART  002                           20000             0
BMC51579I PART  003                           20000             0
BMC51579I PART  004                           20000             0
BMC51579I PART  005                           20000             0

BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

Output from ORDER YES with Dynamic Allocation

The SYSPRINT for example 12B follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   15:39:14 ...
BMC50002I UTILITY ID = 'EXAMPLEC'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTE'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7856K,ABOVE 16M=1665220K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
```

(continued on following page)

(example 12B continued from preceding page)

```
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = YES                 YES                 YES                 YES                 YES
BMC50470I IFALLOC  = FREE                FREE                FREE                FREE                FREE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = YES                 YES                 YES                 YES
BMC50470I IFALLOC  = FREE                FREE                FREE                FREE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO
```

(continued on following page)

(example 12B continued from preceding page)

```
BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I  LOAD DATA RESUME YES
BMC50102I   SHRLEVEL CHANGE
BMC50102I   ENUMROWS(100000)
BMC50102I   ORDER YES
BMC50102I   DDTYPE LOAD     DSNPAT '&UID..RESUME&RESUME.&DDNAME'
BMC50102I   DDTYPE WORK     DSNPAT '&UID..RESUME&RESUME.&DDNAME'
BMC50102I   DDTYPE SORTWORK DSNPAT '&UID..RESUME&RESUME.&DDNAME'
BMC50102I   DDTYPE DISCARD  DSNPAT '&UID..RESUME&RESUME.&DDNAME'
BMC50102I   DDTYPE ERROR    DSNPAT '&UID..RESUME&RESUME.&DDNAME'
BMC50102I   DELETEFILES YES SYSDISC YES
BMC50102I   INTO TABLE EXAMPLEC.TBL1
BMC50102I   (RECNO        POS(1:4)   INTEGER,
BMC50102I    INPUT_A      POS(5:14)  CHAR(10),
BMC50102I    RECNO2       POS(15:18) INTEGER,
BMC50102I    INPUT_B      POS(19:28) CHAR(10),
BMC50102I    INPUT_C      POS(29:43) CHAR(15),
BMC50102I    INPUT_D      POS(44:58) CHAR(15)
BMC50102I    )

BMC51424I ANALYZE PHASE WILL BE PERFORMED DUE TO DYNAMIC WORK FILE ALLOCATION
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50425I &JOBNAME  = AMUEX12   &STEPNAME = LODPLUS2  &DB       = AMUEXCDB  &TS       = EXCTS     &UID      = RDAMZL
BMC50425I &RESUME   = Y         &REPLACE  = N         &DATE     = 090403    &TIME     = 153914    &SSID     = DEBA
BMC50425I &UTIL     = EXAMPLEC  &UTILPFX  = EXAMPLEC  &UTILSFX  =           &DATE8    = 09042003  &GRPNM    = DEBA
BMC50425I &VCAT     = DEBACAT   &TIME4    = 1539      &DATEJ    = 2003247

BMC50445I LOADPLUS DYNAMIC FILE ALLOCATION REPORT

BMC50446I                                                        UNIT OR                     KBYTES   KBYTES     ALOC      ALOC
BMC50447I DDNAME   DSNAME                                        DATACLAS MGMTCLAS STORCLAS     PRI      SEC      PRI       SEC

BMC50448I SORTWK01 RDAMZL.RESUMEY.SORTWK01                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK02 RDAMZL.RESUMEY.SORTWK02                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK03 RDAMZL.RESUMEY.SORTWK03                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK04 RDAMZL.RESUMEY.SORTWK04                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK05 RDAMZL.RESUMEY.SORTWK05                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK06 RDAMZL.RESUMEY.SORTWK06                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK07 RDAMZL.RESUMEY.SORTWK07                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK08 RDAMZL.RESUMEY.SORTWK08                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK09 RDAMZL.RESUMEY.SORTWK09                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK10 RDAMZL.RESUMEY.SORTWK10                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK11 RDAMZL.RESUMEY.SORTWK11                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SORTWK12 RDAMZL.RESUMEY.SORTWK12                    SYSALLDA                        717       72       14         2  TRK
BMC50448I SYSDISC  RDAMZL.RESUMEY.SYSDISC                     SYSALLDA                      10218     1277      165        24  TRK
BMC50448I SYSERR   RDAMZL.RESUMEY.SYSERR                      SYSALLDA                       3125     1250       58        24  TRK

BMC50474I BELOW 16M = 7328K, ABOVE 16M = 1663856K, CPUS = 3
BMC51495I COMBINED OPTIMIZATION, RC = 0, #SORTS = 5, #READERS = 1, INDEX TASKS = 0, TIME = 126208
BMC51495I COMBINED OPTIMIZATION, RC = 14, #SORTS = 4, #READERS = 1, INDEX TASKS = 0, TIME = 136832
BMC51496I COMBINED ANALYZE, #SORTS = 5, #READERS = 1, INDEX TASKS = 0
BMC51576I 15:39:21 000 APT0000I APPLY PLUS FOR DB2 FOR OS/390 V1.6.00
BMC51576I                      Copyright (c) 1999-2003 BMC Software, Inc. as an unpublished licensed work.  All rights reserved.
BMC51576I 15:39:21 000 APT0001I Date: Thu Sep  4, 2003
BMC51576I 15:39:22 000 APT0085I /STARTUP/
BMC51576I                      SSID=DEBA
BMC51576I                      INPUTTYPE={3895D640-0BF1-11D4-8027-00104B39B10F}
BMC51576I                      PLANNAME=AMUZDIST
BMC51576I                      DBMSTYPE=DB2
BMC51576I                      DISTRIBUTIONTYPE=OBJECT
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 12B continued from preceding page)

```
BMC51576I 15:39:22 000 APT0085I /CONFIGURATION/
BMC51576I                       FILENAME=NONE
BMC51576I 15:39:22 000 APT0085I /SQL/
BMC51576I                       FORMAT=PACKED72
BMC51576I 15:39:22 000 APT0085I /COMMITTRIGGERS/
BMC51576I                       STATEMENTCOUNT=2500
BMC51576I                       COMMITONDEMAND=NO
BMC51576I 15:39:22 000 APT0085I /OBJECTMAP/
BMC51576I                       SOURCETABLE=0X0386.0X0003
BMC51576I                       TARGETTABLE=EXAMPLEC.TBL1
BMC51576I 15:39:22 000 APT0085I /AGENT/
BMC51576I                       MAXAGENTS=10
BMC51576I                       INITIALAGENTS=5
BMC51576I                       MAXPREPARES=50
BMC51576I 15:39:22 000 APT0085I /TEMPORARYFILES/
BMC51576I                       UNIT=SYSALLDA
BMC51576I 15:39:22 000 APT0085I /DIAGNOSTICOUTPUT/
BMC51576I                       FILENAME=APTDIAG
BMC51576I 15:39:22 000 APT0085I /RESTART/
BMC51576I                       RESTARTID=EXAMPLEC
BMC51576I                       RESTARTTYPE=NEW
BMC51576I                       TABLENAME=BMCAPT_APTRSCL
BMC51576I                       RETAINTIME=7
BMC51576I                       RETRY=3
BMC51576I 15:39:22 000 APT0085I /DISTRIBUTIONTUNING/
BMC51576I                       SIMPLETSCLUSTERING=NO
BMC51576I                       RICLUSTERING=NO
BMC51576I                       PARTITIONCLUSTERING=NO
BMC51576I                       URCHECKRI=YES
BMC51576I 15:39:22 000 APT0085I /ANYCONFLICT/
BMC51576I                       CODE=-803
BMC51576I                       ACTION=CALLBACK
BMC51576I                       ACTION=CONTINUE
BMC51576I                       CODE=-530
BMC51576I                       ACTION=CALLBACK
BMC51576I                       ACTION=CONTINUE
BMC51576I                       CODE=-359
BMC51576I                       ACTION=CALLBACK
BMC51576I                       ACTION=CONTINUE
BMC51576I                       CODE=SQLWARN0
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN1
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN2
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN3
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN4
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN5
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN6
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN7
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN8
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARN9
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=SQLWARNA
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=NEGATIVE
BMC51576I                       ACTION=ABORT
BMC51576I                       CODE=POSITIVE
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=MULTIPLEROWS
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=NOROWS
BMC51576I                       ACTION=WARN
BMC51576I                       CODE=TIMEOUT
BMC51576I                       ACTION=RETRY
BMC51576I                       CODE=RICONFLICT
BMC51576I                       ACTION=ABORT
BMC51576I 15:39:22 000 APT0085I /CONFLICT/
BMC51576I                       RETRYLIMIT=COUNT
BMC51576I                       RETRYVALUE=5
BMC51576I                       DATATYPE=ID
BMC51576I                       DATATYPE=SQLCODE
BMC51576I                       DATATYPE=MESSAGE
BMC51576I                       RETRYFAIL=ABORT
BMC51576I                       RETRYDELAY=0
BMC51576I 15:39:22 000 APT0085I /CONFLICTFILE/
BMC51576I                       SPACE=TRACK
BMC51576I                       SPACEPRI=15
BMC51576I                       SPACESEC=5
BMC51576I                       FILENAMEMODEL=&SYSUID..BMCAPT.&REQUESTID..D&DATE..T&TIME.
BMC51576I                       SINGLEFILE=NO
```

(continued on following page)

(example 12B continued from preceding page)

```
BMC51576I                       DISP=NEW
BMC51576I                       MEDIATYPE=FILE
BMC51576I                       RELEASE=YES
BMC51576I 15:39:22 000 APT0085I /BIND/
BMC51576I                       FREEOPTION=ALL
BMC51576I                       PLANPREFIX=APT
BMC51576I 15:39:22 000 APT0085I /BINDTUNING/
BMC51576I                       STATEMENTCOUNT=1
BMC51576I                       MAXPACKAGES=25
BMC51576I                       MAXFAILEDBINDS=5
BMC51576I                       MAXRETRIES=5
BMC51576I                       PACKAGEUSAGE=500
BMC51576I                       STATEMENTUSAGE=200
BMC51576I                       SYNCHRONOUS=NO
BMC51576I 15:39:23 001 APT0097I Request ID: A826      Restart ID: EXAMPLEC
BMC51576I 15:39:24 000 APT0019I File DBRMLIB :SYS03247.T153923.RA000.AMUEX12.R0251500 allocated - elapsed time: 00:00:00
BMC51576I 15:39:24 001 APT0003I  DSN SYSTEM(DEBA)
BMC51576I 15:39:24 001 APT0003I  BIND PLAN(APTA826) +
BMC51576I 15:39:24 001 APT0003I       OWNER(RDAMZL) +
BMC51576I 15:39:24 001 APT0003I       ENCODING(EBCDIC) +
BMC51576I 15:39:24 001 APT0003I       KEEPDYNAMIC(YES) +
BMC51576I 15:39:24 001 APT0003I       PKLIST(AMUZDIST.APTREB2, APTA826.*) +
BMC51576I 15:39:24 001 APT0003I       RELEASE(COMMIT) +
BMC51576I 15:39:24 001 APT0003I       ACTION(REPLACE) +
BMC51576I 15:39:24 001 APT0003I      RETAIN +
BMC51576I 15:39:24 001 APT0003I       ISOLATION(CS) +
BMC51576I 15:39:24 001 APT0003I       VALIDATE(RUN)
BMC51576I 15:39:24 001 APT0003I  BIND PACKAGE(APTA826) +
BMC51576I 15:39:24 001 APT0003I       OWNER(RDAMZL) +
BMC51576I 15:39:24 001 APT0003I       ENCODING(EBCDIC) +
BMC51576I 15:39:24 001 APT0003I       KEEPDYNAMIC(YES) +
BMC51576I 15:39:24 001 APT0003I       MEMBER(A8260000) +
BMC51576I 15:39:24 001 APT0003I       RELEASE(COMMIT) +
BMC51576I 15:39:24 001 APT0003I       ACTION(REPLACE) +
BMC51576I 15:39:24 001 APT0003I       ISOLATION(CS) +
BMC51576I 15:39:24 001 APT0003I       VALIDATE(RUN)
BMC51576I 15:39:24 001 APT0003I  END
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 135, EMPTY WAITS = 7, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  100000 PHYSICAL (100000 LOGICAL) RECORDS READ FROM SYSREC
BMC51576I 15:39:26 001 APT0032I DSN output follows:
BMC51576I                       1READY
BMC51576I                         DSN SYSTEM(DEBA)
BMC51576I                        DSN
BMC51576I                         BIND PLAN(APTA826) OWNER(RDAMZL) ENCODING(EBCDIC) KEEPDYNAMIC(YES) PKLIST(AMUZ
BMC51576I                       DIST.APTREB2, APTA826.*) RELEASE(COMMIT) ACTION(REPLA
BMC51576I                        CE) RETAIN ISOLATION(CS) VALIDATE(RUN)
BMC51576I                        DSNT252I  *DEBA DSNTBCM1 BIND OPTIONS FOR PLAN APTA826
BMC51576I                                   ACTION        ADD
BMC51576I                                   OWNER         RDAMZL
BMC51576I                                   VALIDATE      RUN
BMC51576I                                   ISOLATION     CS
BMC51576I                                   ACQUIRE       USE
BMC51576I                                   RELEASE       COMMIT
BMC51576I                                   EXPLAIN       NO
BMC51576I                                   DYNAMICRULES  RUN
BMC51576I                        DSNT253I  *DEBA DSNTBCM1 BIND OPTIONS FOR PLAN APTA826
BMC51576I                                   NODEFER       PREPARE
BMC51576I                                   CACHESIZE     1024
BMC51576I                                   QUALIFIER     RDAMZL
BMC51576I                                   CURRENTSERVER
BMC51576I                                   CURRENTDATA   YES
BMC51576I                                   DEGREE        1
BMC51576I                                   SQLRULES      DB2
BMC51576I                                   DISCONNECT    EXPLICIT
BMC51576I                                   NOREOPT       VARS
BMC51576I                                   KEEPDYNAMIC   YES
BMC51576I                                   IMMEDWRITE    NO
BMC51576I                                   DBPROTOCOL    DRDA
BMC51576I                                   OPTHINT
BMC51576I                                   ENCODING      EBCDIC(00037)
BMC51576I                                   PATH
BMC51576I                        DSNT200I  *DEBA  BIND FOR PLAN APTA826  SUCCESSFUL
BMC51576I                        DSN
BMC51576I                         BIND PACKAGE(APTA826) OWNER(RDAMZL) ENCODING(EBCDIC) KEEPDYNAMIC(YES) MEMBER(A
BMC51576I                       8260000) RELEASE(COMMIT) ACTION(REPLACE) ISOLATION(CS
BMC51576I                        ) VALIDATE(RUN)
BMC51576I                        DSNT254I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                                   ACTION        ADD
BMC51576I                                   OWNER         RDAMZL
BMC51576I                                   QUALIFIER     RDAMZL
BMC51576I                                   VALIDATE      RUN
BMC51576I                                   EXPLAIN       NO
BMC51576I                                   ISOLATION     CS
BMC51576I                                   RELEASE       COMMIT
BMC51576I                                   COPY
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 12B continued from preceding page)

```
BMC51576I                        DSNT255I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                                   SQLERROR      NOPACKAGE
BMC51576I                                   CURRENTDATA   YES
BMC51576I                                   DEGREE        1
BMC51576I                                   DYNAMICRULES
BMC51576I                                   DEFER
BMC51576I                                   NOREOPT       VARS
BMC51576I                                   KEEPDYNAMIC   YES
BMC51576I                                   IMMEDWRITE    NO
BMC51576I                                   DBPROTOCOL    DRDA
BMC51576I                                   OPTHINT
BMC51576I                                  1            ENCODING      EBCDIC(00037)
BMC51576I                                   PATH
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL BIND FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                        DSN
BMC51576I                         END
BMC51576I                        READY
BMC51576I                        END
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:05
BMC50481I 3: SORT COMPLETE.  ELAPSED TIME = 00:00:05
BMC50481I 4: SORT COMPLETE.  ELAPSED TIME = 00:00:05
BMC50481I 5: SORT COMPLETE.  ELAPSED TIME = 00:00:05
BMC51576I 15:39:27 006 APT0079I Apply Processing Started - Agent: 1
BMC50481I 6: SORT COMPLETE.  ELAPSED TIME = 00:00:05
BMC51576I 15:39:27 007 APT0079I Apply Processing Started - Agent: 2
BMC51576I 15:39:27 008 APT0079I Apply Processing Started - Agent: 3
BMC51576I 15:39:27 009 APT0079I Apply Processing Started - Agent: 4
BMC51576I 15:39:27 010 APT0079I Apply Processing Started - Agent: 5
BMC51480I 6: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:06.  0 ROWS PASSED TO APPLY PLUS
BMC51576I 15:39:27 001 APT0148I Bind started for package A8260001 - Trigger statistics:
BMC51576I                          STATEMENTCOUNT.........................1
BMC51576I                          PACKAGEUSAGE...........................1
BMC51576I                          STATEMENTUSAGE.........................1
BMC51576I 15:39:27 001 APT0003I  DSN SYSTEM(DEBA)
BMC51576I 15:39:27 001 APT0003I  BIND PACKAGE(APTA826) +
BMC51576I 15:39:27 001 APT0003I       OWNER(RDAMZL) +
BMC51576I 15:39:27 001 APT0003I       ENCODING(EBCDIC) +
BMC51576I 15:39:27 001 APT0003I       KEEPDYNAMIC(YES) +
BMC51576I 15:39:27 001 APT0003I       MEMBER(A8260001) +
BMC51576I 15:39:27 001 APT0003I       RELEASE(COMMIT) +
BMC51576I 15:39:27 001 APT0003I       ACTION(REPLACE) +
BMC51576I 15:39:28 001 APT0003I       ISOLATION(CS) +
BMC51576I 15:39:28 001 APT0003I       VALIDATE(RUN)
BMC51576I 15:39:28 001 APT0003I  END
BMC51510I 6: SORT TASK, XBLK XFERS = 0, EMPTY WAITS = 0, FULL WAITS = 1
BMC51510I 6: SORT TASK, APBUF XFERS = 0, EMPTY WAITS = 0, FULL WAITS = 0
BMC51576I 15:39:30 001 APT0185I Statements committed: 10000
BMC51576I 15:39:31 001 APT0032I DSN output follows:
BMC51576I                       1READY
BMC51576I                         DSN SYSTEM(DEBA)
BMC51576I                        DSN
BMC51576I                         BIND PACKAGE(APTA826) OWNER(RDAMZL) ENCODING(EBCDIC) KEEPDYNAMIC(YES) MEMBER(A
BMC51576I                       8260001) RELEASE(COMMIT) ACTION(REPLACE) ISOLATION(CS
BMC51576I                        ) VALIDATE(RUN)
BMC51576I                        DSNT254I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                                   ACTION        ADD
BMC51576I                                   OWNER         RDAMZL
BMC51576I                                   QUALIFIER     RDAMZL
BMC51576I                                   VALIDATE      RUN
BMC51576I                                   EXPLAIN       NO
BMC51576I                                   ISOLATION     CS
BMC51576I                                   RELEASE       COMMIT
BMC51576I                                   COPY
BMC51576I                        DSNT255I  *DEBA DSNTBCM2 BIND OPTIONS FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                                   SQLERROR      NOPACKAGE
BMC51576I                                   CURRENTDATA   YES
BMC51576I                                   DEGREE        1
BMC51576I                                   DYNAMICRULES
BMC51576I                                   DEFER
BMC51576I                                   NOREOPT       VARS
BMC51576I                                   KEEPDYNAMIC   YES
BMC51576I                                   IMMEDWRITE    NO
BMC51576I                                   DBPROTOCOL    DRDA
BMC51576I                                   OPTHINT
BMC51576I                                   ENCODING      EBCDIC(00037)
BMC51576I                                   PATH
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL BIND FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                        DSN
BMC51576I                         END
BMC51576I                        READY
BMC51576I                        END
```

(continued on following page)

(example 12B continued from preceding page)

```
BMC51576I 15:39:33 001 APT0185I Statements committed: 15000
BMC51576I 15:39:34 001 APT0185I Statements committed: 20000
BMC51480I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:13.  20000 ROWS PASSED TO APPLY PLUS
BMC51510I 2: SORT TASK, XBLK XFERS = 27, EMPTY WAITS = 0, FULL WAITS = 6
BMC51510I 2: SORT TASK, APBUF XFERS = 43, EMPTY WAITS = 32, FULL WAITS = 0
BMC51480I 3: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:14.  20000 ROWS PASSED TO APPLY PLUS
BMC51480I 4: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:14.  20000 ROWS PASSED TO APPLY PLUS
BMC51510I 3: SORT TASK, XBLK XFERS = 27, EMPTY WAITS = 0, FULL WAITS = 6
BMC51510I 3: SORT TASK, APBUF XFERS = 43, EMPTY WAITS = 41, FULL WAITS = 0
BMC51576I 15:39:36 001 APT0185I Statements committed: 30000
BMC51510I 4: SORT TASK, XBLK XFERS = 27, EMPTY WAITS = 0, FULL WAITS = 4
BMC51510I 4: SORT TASK, APBUF XFERS = 43, EMPTY WAITS = 42, FULL WAITS = 0
BMC51576I 15:39:38 001 APT0118I Input statements processed: 87771
BMC51576I 15:39:39 001 APT0185I Statements committed: 37500
BMC51480I 5: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:22.  40000 ROWS PASSED TO APPLY PLUS
BMC51510I 5: SORT TASK, XBLK XFERS = 54, EMPTY WAITS = 0, FULL WAITS = 36
BMC51510I 5: SORT TASK, APBUF XFERS = 86, EMPTY WAITS = 85, FULL WAITS = 0
BMC51576I 15:39:45 001 APT0118I Input statements processed: 100000
BMC51576I 15:39:45 001 APT0066I Input processing complete
BMC51576I 15:39:46 001 APT0185I Statements committed: 42500
BMC51576I 15:39:48 001 APT0185I Statements committed: 52500
BMC51576I 15:39:53 001 APT0185I Statements committed: 57500
BMC51576I 15:39:54 001 APT0185I Statements committed: 62500
BMC51576I 15:39:58 001 APT0185I Statements committed: 67500
BMC51576I 15:40:03 001 APT0185I Statements committed: 72500
BMC51576I 15:40:13 001 APT0185I Statements committed: 77500
BMC51576I 15:40:18 006 APT0080I Apply Processing Ended - Agent: 1
BMC51576I 15:40:18 001 APT0185I Statements committed: 82500
BMC51576I 15:40:19 009 APT0080I Apply Processing Ended - Agent: 4
BMC51576I 15:40:19 007 APT0080I Apply Processing Ended - Agent: 2
BMC51576I 15:40:19 010 APT0080I Apply Processing Ended - Agent: 5
BMC51576I 15:40:19 001 APT0185I Statements committed: 92500
BMC51576I 15:40:22 001 APT0185I Statements committed: 97500
BMC51576I 15:40:23 008 APT0080I Apply Processing Ended - Agent: 3
BMC51576I 15:40:23 001 APT0047I TABLE STATISTICS - EXAMPLEC.TBL1
BMC51576I                               INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                               100000         0         0         0     100000
BMC51576I 15:40:23 001 APT0048I PARTITION STATISTICS - EXAMPLEC.TBL1
BMC51576I                        PART    INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                           1     20000         0         0         0      20000
BMC51576I                           2     20000         0         0         0      20000
BMC51576I                           3     20000         0         0         0      20000
BMC51576I                           4     20000         0         0         0      20000
BMC51576I                           5     20000         0         0         0      20000
BMC51576I 15:40:23 001 APT0003I  DSN SYSTEM(DEBA)
BMC51576I 15:40:23 001 APT0003I  FREE PACKAGE(APTA826.*)
BMC51576I 15:40:23 001 APT0003I  FREE PLAN(APTA826)
BMC51576I 15:40:23 001 APT0003I  END
BMC51576I 15:40:26 001 APT0032I DSN output follows:
BMC51576I                       1READY
BMC51576I                         DSN SYSTEM(DEBA)
BMC51576I                        DSN
BMC51576I                         FREE PACKAGE(APTA826.*)
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260000.()
BMC51576I                        DSNT232I  *DEBA SUCCESSFUL FREE FOR
BMC51576I                                   PACKAGE = DEBA.APTA826.A8260001.()
BMC51576I                        DSN
BMC51576I                         FREE PLAN(APTA826)
BMC51576I                        DSNT200I  *DEBA  FREE FOR PLAN APTA826  SUCCESSFUL
BMC51576I                        DSN
BMC51576I                         END
BMC51576I                        READY
BMC51576I                        END
BMC51576I 15:40:26 000 APT0020I File DBRMLIB :SYS03247.T153923.RA000.AMUEX12.R0251500 deallocated
BMC51576I 15:40:26 001 APT0111I INPUT PROCESSING STATISTICS
BMC51576I                               INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                               100000         0         0         0     100000
BMC51576I 15:40:26 001 APT0166I STATEMENTS COMMITTED STATISTICS
BMC51576I                       AGENT    INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                           1     20000         0         0         0      20000
BMC51576I                           2     20000         0         0         0      20000
BMC51576I                           3     20000         0         0         0      20000
BMC51576I                           4     20000         0         0         0      20000
BMC51576I                           5     20000         0         0         0      20000
BMC51576I                       ----- --------- --------- --------- --------- ----------
BMC51576I                       TOTAL    100000         0         0         0     100000
BMC51576I 15:40:26 001 APT0082I DYNAMIC SQL PROCESSING STATISTICS
BMC51576I                       AGENT    INSERT    UPDATE    DELETE    COMMIT  ROLLBACK      TOTAL
BMC51576I                           1     11750         0         0         0         0      11750
BMC51576I                           2      7050         0         0         0         0       7050
BMC51576I                           3         0         0         0         0         0          0
BMC51576I                           4      7520         0         0         0         0       7520
BMC51576I                           5      7483         0         0         0         0       7483
BMC51576I                       ----- --------- --------- --------- --------- --------- ----------
BMC51576I                       TOTAL     33803         0         0         0         0      33803
```

(continued on following page)

(example 12B continued from preceding page)

```
BMC51576I 15:40:26 001 APT0081I STATIC SQL PROCESSING STATISTICS
BMC51576I                       AGENT    INSERT    UPDATE    DELETE   MASSDEL    COMMIT  ROLLBACK   PREPARE   EXECUTE      TOTAL
BMC51576I                           1      8250         0         0         0         8         0         1     11750      20009
BMC51576I                           2     12950         0         0         0         8         0         1      7050      20009
BMC51576I                           3     20000         0         0         0         8         0         0         0      20008
BMC51576I                           4     12480         0         0         0         8         0         1      7520      20009
BMC51576I                           5     12517         0         0         0         8         0         1      7483      20009
BMC51576I                       ----- --------- --------- --------- --------- --------- --------- --------- --------- ----------
BMC51576I                       TOTAL     66197         0         0         0        40         0         4     33803     100044
BMC51576I 15:40:26 001 APT0200I EXECUTE IMMEDIATE PROCESSING STATISTICS
BMC51576I                       AGENT    INSERT    UPDATE    DELETE       DDL      TOTAL
BMC51576I                           1         0         0         0         0          0
BMC51576I                           2         0         0         0         0          0
BMC51576I                           3         0         0         0         0          0
BMC51576I                           4         0         0         0         0          0
BMC51576I                           5         0         0         0         0          0
BMC51576I                       ----- --------- --------- --------- --------- ----------
BMC51576I                       TOTAL         0         0         0         0          0
BMC51576I 15:40:26 001 APT0182I OTHER STATISTICS
BMC51576I                       AGENT FAILPREPS   RETRIES DBMS ELAPSED TIME
BMC51576I                           1         0         0   00:00:40.984435
BMC51576I                           2         0         0   00:00:41.805175
BMC51576I                           3         0         0   00:00:36.720688
BMC51576I                           4         0         0   00:00:40.771836
BMC51576I                           5         0         0   00:00:42.241417
BMC51576I                       ----- --------- --------- -----------------
BMC51576I                       TOTAL         0         0   00:03:22.523555
BMC51576I 15:40:26 001 APT0183I Request elapsed time: 00:01:04.031774
BMC51576I 15:40:26 000 APT0067I Request completed Return code: 0
BMC50481I 7: SQLAPPLY TASK COMPLETE.  ELAPSED TIME = 00:01:05
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC, I/OS = 206, I/O WAITS = 22, RDB LOCK WAITS = 0
BMC51507I XBLKS = 22, XFERS = 135, EMPTY WAITS = 7, FULL WAITS =53
BMC51507I APBUFS = 15, XFERS = 215, EMPTY WAITS = 200, FULL WAITS =0
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51472I COMBINED PHASE STATISTICS:  100000 ROWS SELECTED FOR SPACE 'AMUEXCDB.EXCTS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I COMBINED PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC51475I LOAD STATISTICS:  100000 ROWS LOADED INTO TABLE SPACE 'AMUEXCDB.EXCTS'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SYSERR  ', DSNAME = 'RDAMZL.RESUMEY.SYSERR'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK01', DSNAME = 'RDAMZL.RESUMEY.SORTWK01'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK02', DSNAME = 'RDAMZL.RESUMEY.SORTWK02'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK03', DSNAME = 'RDAMZL.RESUMEY.SORTWK03'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK04', DSNAME = 'RDAMZL.RESUMEY.SORTWK04'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK05', DSNAME = 'RDAMZL.RESUMEY.SORTWK05'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK06', DSNAME = 'RDAMZL.RESUMEY.SORTWK06'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK07', DSNAME = 'RDAMZL.RESUMEY.SORTWK07'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK08', DSNAME = 'RDAMZL.RESUMEY.SORTWK08'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK09', DSNAME = 'RDAMZL.RESUMEY.SORTWK09'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK10', DSNAME = 'RDAMZL.RESUMEY.SORTWK10'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK11', DSNAME = 'RDAMZL.RESUMEY.SORTWK11'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK12', DSNAME = 'RDAMZL.RESUMEY.SORTWK12'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SYSDISC ', DSNAME = 'RDAMZL.RESUMEY.SYSDISC'
BMC50004I COMBINED PHASE COMPLETE.  ELAPSED TIME = 00:01:07

BMC51577I                                  ROWS       DISCARDED
BMC51578I                                INSERTED     SQLERRORS

BMC51579I SPACE AMUEXCDB.EXCTS             100000             0

BMC51579I PART  001                         20000             0
BMC51579I PART  002                         20000             0
BMC51579I PART  003                         20000             0
BMC51579I PART  004                         20000             0
BMC51579I PART  005                         20000             0

BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

Example 13: LOAD REPLACE from CSV Input

This example shows a LOAD REPLACE of a segmented table space from CSV input. There is one unique clustering index on the table.

By specifying COPY NO COPYPEND NO, LOADPLUS does not make a copy and does not place the table space in COPY pending status following the load. Note that, because the value for the COPYPEND installation option is YES and not (YES,ENFORCE), the COPYPEND NO command option can override the YES value.

The field specification for the FLOAT_COLUMN column tells LOADPLUS to interpret any two consecutive delimiter characters in this column as null.

This example provides an image of the SYSREC data set that contains the CSV input. The input file uses the default delimiters (,) and enclosure characters ("), so the FORMAT CSV option does not need any additional parameters. The input file uses a fixed-length record format, so a delimiter was added to the end of each record to ensure that the load job does not encounter end-of-record errors.

For FORMAT CSV, LOADPLUS uses RULES=STANDARD. In this example, message BMC51414I, tells you that LOADPLUS has changed the value of the RULES installation option to STANDARD.

This example also provides the SPUFI output showing the loaded table.

The JCL for example 13 follows:

```
//        JOB
//EXAMPLED EXEC PGM=AMUUMAIN,
//              PARM='&SSID,EXAMPLED,NEW,,MSGLEVEL(1),AMU$OPTS'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB1
//         DD DISP=SHR,DSN=&AMULIB2
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SORTOUT  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,PASS),
//            DSN=AMU.EXAMPLED.SORTOUT
//SYSUT101 DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,PASS),
//            DSN=AMU.EXAMPLED.SYSUT1
//SYSERR   DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,PASS),
//            DSN=AMU.EXAMPLED.SYSERR
//SYSDISC  DD UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,PASS),
//            DSN=AMU.EXAMPLED.SYSDISC
//*
//SYSREC   DD DSN=AMU.QA.RGRTESTS.DTAEX13,DISP=SHR
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SYSIN    DD *
   LOAD REPLACE
      PRELOAD LOAD
      DELETEFILES YES
      COPY NO COPYPEND NO
      FORMAT CSV
   INTO TABLE EXAMPLED.TBL1
       (INTEGER_COLUMN     POSITION(*)
       ,SMALLINT_COLUMN    POSITION(*)
       ,DECIMAL_COLUMN     POSITION(*)
       ,CHAR_COLUMN        POSITION(*)
       ,VARCHAR_COLUMN     POSITION(*)
       ,DATE_COLUMN_D1     POSITION(*) DATE-D1       EXTERNAL
       ,TIME_COLUMN_D1     POSITION(*) TIME-D1       EXTERNAL
       ,TIMESTP_COLUMN_D1E POSITION(*) TIMESTAMP-1E  EXTERNAL
       ,FLOAT_COLUMN       POSITION(*)
            NULLIF FLOAT_COLUMN = ''
       )
```

An image of the SYSREC data set for example 13 follows:

```
"000001",524,"55521.15","KATHY","WISE","02/01/00","11:10:12","20021031211512","9.869217185085289E+09",
"000002","722","53251.78","JOHN","BROWN","10/07/00","01:01:45","20021210234211","9.231216185085289E+09",
"000003","755","45631.43","BERTHA","SMITH","11/08/00","14:55:42","20000110203423",,
"000004","572","34678.55","JEREMY","BROWN","05/03/00","16:39:39","20000503031639","9.169234115085689E+09",
"000005","289","93352.65","LINDA","KANDU","07/04/00","20:54:01","20000120110124","9.366213124064289E+09",
"000006","424","34553.48","CHRIS","CRUZ","02/11/00","22:23:37","20010322233711","9.267122386485689E+09",
"000007","522","72526.43","JULIE","CRIMSON","03/29/00","03:06:11","20000503061142",,
"000008","743","41781.25","ALEX","MCNALLA","05/30/00","15:45:03","20000311010312","9.859313455385283E+09",
"000009","455","31524.41","ANNA","STACK","12/15/00","02:24:47","20000302224332","9.469245685081539E+09",
"000010","667","91206.35","BOB","PRICE","10/19/00","04:29:29","20060204212112","9.569211235085349E+09",
"000011","724","41528.55","ALEXIA","ROBERTS","04/01/00","05:13:32","20021105133212","9.369237184125383E+09",
"000012","857","81520.55","CHUCK","JONES","08/02/00","08:06:53","20000201011112","3.321267332045211E+01",
"000013","999","36522.35","JANET","RYAN","07/14/00","10:13:29","20010210132011",,
"000014","370","41528.85","DOUG","FRAME","02/21/00","09:23:18","20000209231013","9.259237485085289E+09",
"000015","642","11521.95","RACHEL","POTTS","03/22/00","02:29:22","20000102212232","9.466414125082389E+09",
"000016","323","22527.05","RANDY","GRIFFIN","03/14/00","01:45:35","20000201031322","9.563717183085489E+09",
"000017","645","35524.72","WANDA","JENKINS","08/04/00","04:53:46","20031104134202","9.269287185425229E+09",
"000018","221","81471.65","PRIYA","LONG","04/29/00","05:28:30","20040105233032","9.869239225233423E+09",
"000019","255","51534.42","JOHN","MERCER","02/28/00","08:47:38","20001003012022","2.332235343235211E+02",
"000020","513","62561.15","MARIE","MORALEZ","09/20/00","07:18:43","20000901024302","9.769214184085289E+09",
```

The SYSPRINT for example 13 follows:

```
                         *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
               COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   15:43:50 ...
BMC50002I UTILITY ID = 'AMUEXDDB'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTS'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7872K,ABOVE 16M=1666916K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
```

(continued on following page)

(example 13 continued from preceding page)

```
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 13 continued from preceding page)

```
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I    LOAD REPLACE
BMC50102I       PRELOAD LOAD
BMC50102I       DELETEFILES YES
BMC50102I       COPY NO COPYPEND NO
BMC50102I       FORMAT CSV
BMC50102I    INTO TABLE EXAMPLED.TBL1
BMC50102I        (INTEGER_COLUMN     POSITION(*)
BMC50102I        ,SMALLINT_COLUMN    POSITION(*)
BMC50102I        ,DECIMAL_COLUMN     POSITION(*)
BMC50102I        ,CHAR_COLUMN        POSITION(*)
BMC50102I        ,VARCHAR_COLUMN     POSITION(*)
BMC50102I        ,DATE_COLUMN_D1     POSITION(*) DATE-D1       EXTERNAL
BMC50102I        ,TIME_COLUMN_D1     POSITION(*) TIME-D1       EXTERNAL
BMC50102I        ,TIMESTP_COLUMN_D1E POSITION(*) TIMESTAMP-1E  EXTERNAL
BMC50102I        ,FLOAT_COLUMN       POSITION(*)
BMC50102I             NULLIF FLOAT_COLUMN = ''
BMC50102I        )

BMC51414I 'FORMAT CSV' SPECIFIED.  STANDARD RULES IN EFFECT FOR WHEN COMPARISONS
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SYSERR'
BMC50466W TEMPORARY DATASET NOT RECOMMENDED FOR DDNAME = 'SYSDISC'
BMC50394I UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'
BMC50395I MAXSORTS SET TO 1
BMC50474I BELOW 16M = 7392K, ABOVE 16M = 1665140K, CPUS = 3
BMC51495I COMBINED OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 1, TIME = 135168
BMC51496I COMBINED ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEXDDB.EXDTS' DELETED
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  20 PHYSICAL (20 LOGICAL) RECORDS READ FROM SYSREC
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEXDDB.EXDTS.I0001.A001'
BMC50477I 2: PARTITION =   0, ROWS/KEYS = 20, I/O WAITS = 5, DDNAME = SYS00004
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:03
BMC51510I 2: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 1
BMC50481I 3: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:03
BMC50482I 3: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEXDDB.IX1.I0001.A001'
BMC50477I 3: PARTITION =   0, ROWS/KEYS = 20, I/O WAITS = 7, DDNAME = SYS00007
BMC51476I BUILD STATISTICS:  20 KEYS LOADED INTO INDEX 'EXAMPLED.IX1'
BMC50481I 3: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:06
BMC51510I 3: INDEX TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSREC, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 11, XFERS = 2, EMPTY WAITS = 0, FULL WAITS =2
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51472I COMBINED PHASE STATISTICS:  20 ROWS SELECTED FOR SPACE 'AMUEXDDB.EXDTS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I COMBINED PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC51475I LOAD STATISTICS:  20 ROWS LOADED INTO TABLE SPACE 'AMUEXDDB.EXDTS'
BMC50004I COMBINED PHASE COMPLETE.  ELAPSED TIME = 00:00:08

BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEXDDB.EXDTS
BMC50513I             PARTS   = 0         TABLES  = 1         SEGSIZE = 16
BMC50514I             NACTIVE = 3             SPACE   = 1
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   0         3         20          0          0        2        0   35    0       1    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLED.TBL1                         20          1         0      62       6       0
BMC50510I INDEX       EXAMPLED.IX1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLED.TBL1     COLUMN  INTEGER_COLUMN
BMC50514I             NACTIVE = 5             SPACE   = 1
BMC50542I             FIRSTKEY= 20            FULLKEY = 20            NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 5                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'0080000013404040'  LOW2KEY= X'0080000002404040'
BMC50547I             HIGH2KEY=          LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         5         20        0          0          1        0        7        1     2         1       1
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

The SPUFI output showing the loaded table in example 13 follows:

```
PAGE    1
***INPUT STATEMENT:
   SELECT *  FROM EXAMPLED.TBL1;
        +-----------------------------------------------------------------------------------------------------------------------
        | INTEGER_COLUMN | SMALLINT_COLUMN | DECIMAL_COLUMN | CHAR_COLUMN | VARCHAR_COLUMN | DATE_COLUMN_D1 | TIME_COLUMN_D1 |
        +-----------------------------------------------------------------------------------------------------------------------
      1_|              1 |             524 |       55521.15 | KATHY       | WISE           | 02/01/2000     | 11:10 AM       |
      2_|              2 |             722 |       53251.78 | JOHN        | BROWN          | 10/07/2000     | 01:01 AM       |
      3_|              3 |             755 |       45631.43 | BERTHA      | SMITH          | 11/08/2000     | 02:55 PM       |
      4_|              4 |             572 |       34678.55 | JEREMY      | BROWN          | 05/03/2000     | 04:39 PM       |
      5_|              5 |             289 |       93352.65 | LINDA       | KANDU          | 07/04/2000     | 08:54 PM       |
      6_|              6 |             424 |       34553.48 | CHRIS       | CRUZ           | 02/11/2000     | 10:23 PM       |
      7_|              7 |             522 |       72526.43 | JULIE       | CRIMSON        | 03/29/2000     | 03:06 AM       |
      8_|              8 |             743 |       41781.25 | ALEX        | MCNALLA        | 05/30/2000     | 03:45 PM       |
      9_|              9 |             455 |       31524.41 | ANNA        | STACK          | 12/15/2000     | 02:24 AM       |
     10_|             10 |             667 |       91206.35 | BOB         | PRICE          | 10/19/2000     | 04:29 AM       |
     11_|             11 |             724 |       41528.55 | ALEXIA      | ROBERTS        | 04/01/2000     | 05:13 AM       |
     12_|             12 |             857 |       81520.55 | CHUCK       | JONES          | 08/02/2000     | 08:06 AM       |
     13_|             13 |             999 |       36522.35 | JANET       | RYAN           | 07/14/2000     | 10:13 AM       |
     14_|             14 |             370 |       41528.85 | DOUG        | FRAME          | 02/21/2000     | 09:23 AM       |
     15_|             15 |             642 |       11521.95 | RACHEL      | POTTS          | 03/22/2000     | 02:29 AM       |
     16_|             16 |             323 |       22527.05 | RANDY       | GRIFFIN        | 03/14/2000     | 01:45 AM       |
     17_|             17 |             645 |       35524.72 | WANDA       | JENKINS        | 08/04/2000     | 04:53 AM       |
     18_|             18 |             221 |       81471.65 | PRIYA       | LONG           | 04/29/2000     | 05:28 AM       |
     19_|             19 |             255 |       51534.42 | JOHN        | MERCER         | 02/28/2000     | 08:47 AM       |
     20_|             20 |             513 |       62561.15 | MARIE       | MORALEZ        | 09/20/2000     | 07:18 AM       |
        +-----------------------------------------------------------------------------------------------------------------------
PAGE    2
                                    ------------------------------------------------------+
                                    |     TIMESTP_COLUMN_D1E     |      FLOAT_COLUMN      |
                                    ------------------------------------------------------+
                                  1_| 2002-10-31-21.15.12.000000 |  9.869217185085288E+09 |
                                  2_| 2002-12-10-23.42.11.000000 |  9.231216185085288E+09 |
                                  3_| 2000-01-10-20.34.23.000000 | ?                      |
                                  4_| 2000-05-03-03.16.39.000000 |  9.169234115085688E+09 |
                                  5_| 2000-01-20-11.01.24.000000 |  9.366213124064288E+09 |
                                  6_| 2001-03-22-23.37.11.000000 |  9.267122386485688E+09 |
                                  7_| 2000-05-03-06.11.42.000000 | ?                      |
                                  8_| 2000-03-11-01.03.12.000000 |  9.859313455385282E+09 |
                                  9_| 2000-03-02-22.43.32.000000 |  9.469245685081538E+09 |
                                 10_| 2006-02-04-21.21.12.000000 |  9.569211235085349E+09 |
                                 11_| 2002-11-05-13.32.12.000000 |  9.369237184125382E+09 |
                                 12_| 2000-02-01-01.11.12.000000 |  3.321267332045210E+01 |
                                 13_| 2001-02-10-13.20.11.000000 | ?                      |
                                 14_| 2000-02-09-23.10.13.000000 |  9.259237485085288E+09 |
                                 15_| 2000-01-02-21.22.32.000000 |  9.466414125082388E+09 |
                                 16_| 2000-02-01-03.13.22.000000 |  9.563717183085488E+09 |
                                 17_| 2003-11-04-13.42.02.000000 |  9.269287185425228E+09 |
                                 18_| 2004-01-05-23.30.32.000000 |  9.869239225233422E+09 |
                                 19_| 2000-10-03-01.20.22.000000 |  2.332235343235210E+02 |
                                 20_| 2000-09-01-02.43.02.000000 |  9.769214184085289E+09 |
                                    ------------------------------------------------------+
SUCCESSFUL RETRIEVAL OF       20 ROW(S)
```

BMC Software, Inc., Confidential and Proprietary Information

Example 14: LOAD REPLACE from UNLOAD PLUS Data in Internal Format

This example is a LOAD REPLACE of a partitioned table space with one unique clustering index and one nonunique nonclustering index. In this example, LOADPLUS loads data that has been unloaded by using UNLOAD PLUS. The UNLOAD PLUS command includes the FORMAT BMCLOAD option to unload the data in an internal format that only LOADPLUS can read. The LOADPLUS command includes the FORMAT BMCUNLOAD option to load this data.

To provide a complete picture of these features, this example includes the UNLOAD PLUS JCL and SYSPRINT, the LOADPLUS JCL and SYSPRINT, and output showing the loaded table.

In the UNLOAD PLUS JCL, the UNLOAD PLUS command includes the following options:

- The CNTLCARDS option passes LOADPLUS command options to the SYSCNTL data set. The control cards that are specified with this option will generate a LOAD REPLACE run with dynamic allocation.

  Note: When you specify FORMAT BMCLOAD in UNLOAD PLUS, UNLOAD PLUS forces CNTLCARDS BMCLOAD.

- The ENUMROWS control card tells UNLOAD PLUS to determine the value to include with the LOADPLUS ENUMROWS option from the number of rows that are unloaded.

- The ACTIVE (YES) option turns dynamic allocation on in UNLOAD PLUS for the SYSREC data set.

- The LIMIT 1 specification tells UNLOAD PLUS to select only the first qualifying row from each partition in this table space.

- The INTO parameter of the SELECT statement passes to LOADPLUS the name of the table into which LOADPLUS is to load the data. This table is identical in structure to the table from which UNLOAD PLUS is unloading data.

The UNLOAD PLUS SYSPRINT shows the LOADPLUS control cards that UNLOAD PLUS writes to the SYSCNTL data set. Note the following LOADPLUS options:

- The INDSN option shows the SYSREC data set that UNLOAD PLUS created. UNLOAD PLUS builds the data set name dynamically based on the DSNAME defaults in the UNLOAD PLUS installation options.

BMC Software, Inc., Confidential and Proprietary Information

- The ENUMROWS option shows that UNLOAD PLUS calculated the number of new rows that LOADPLUS will load as 13. This includes the header records that UNLOAD PLUS creates when using FORMAT BMCLOAD. LOADPLUS uses these header records to ensure that the table being loaded matches the table that was unloaded.

- The INTO statement contains the WHEN TABLE=*obid* option, which is required when using FORMAT BMCUNLOAD. The OBID is the identifier that UNLOAD PLUS generated for this table.

The SYSIN DD statement in the LOADPLUS JCL specifies the name of the SYSCNTL data set that contains the LOAD command options that UNLOAD PLUS created. Note that the JCL does not include a SYSREC DD statement because the input data set is specified with the INDSN option.

The UNLOAD PLUS JCL for example 14 follows:

```
//        JOB
//EXAMPLEF EXEC PGM=ADUUMAIN,
//         PARM='&SSID,AMUEX14,NEW,,MSGLEVEL(1)'
//STEPLIB  DD DISP=SHR,DSN=&ADULIB1
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSCNTL  DD DISP=(NEW,PASS),DSN=&UNLDCNTL,
//            SPACE=(TRK,1)
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//SYSIN    DD *
             UNLOAD CNTLCARDS
              'ENUMROWS' 'REPLACE' 'DELETEFILES YES SYSDISC YES'
             UNLOADDN (SYSREC) ACTIVE (YES) LIMIT 1
             ORDER YES
             FORMAT BMCLOAD
             SELECT * INTO NAME EXAMPLEF.TBLB
                    FROM EXAMPLEF.TBL1
```

The UNLOAD PLUS SYSPRINT for example 14 follows:

```
                        *****  B M C   U N L O A D   P L U S   F O R   D B 2    V7R1.00 *****
                COPYRIGHT (C) 1991 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   20:44:45 ...
BMC50002I UTILITY ID = 'AMUEX14'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'ADU$OPTS'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7820K,ABOVE 16M=1664460K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I SMAX=16
BMC50471I CMAX=16
BMC50471I MAXP=5
BMC50471I UBUFFS=25
BMC50471I IBUFFS=25
BMC50471I VOLSEP=NO
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
```

(continued on following page)

(example 14A continued from preceding page)

```
BMC50471I UNLOADDN=(SYSREC,SYSRED)
BMC50471I UNLOADDN_ACTIVE=(NO,NO)
BMC50471I UXSTATE=SUP
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I ACFORTSS=NO
BMC50471I TASKMAX=200%
BMC50471I UNLDMAX=200%
BMC50471I NULLCHAR=X'6F'
BMC50471I NULLTYPE=T1
BMC50471I HISTORY=YES
BMC50471I ANALYZE=(DB2STATS,NOLIMIT)
BMC50471I FILL=NO
BMC50471I CURRENTDEGREE=NONE
BMC50471I INLINE=NO
BMC50471I USELRECL=NO
BMC50471I CONSTRULES=BMC
BMC50471I ZONEDDECOVP=(C,D)
BMC50471I PLANAUTH=ADUA710D
BMC50471I PLANCAT =ADUC710D
BMC50471I PLANSER =ADUR710D
BMC50471I PLANLOCK=ADUL710D
BMC50471I PLANSYNC=ADUS710D
BMC50471I PLANDSQL=ADUQ710D
BMC50471I TAPES=NONE

BMC50470I OUTPUT   = SYSREC               SYSRED
BMC50470I UNIT     = SYSALLDA             SYSALLDA
BMC50470I VOLCNT   = 25                   25
BMC50470I GDGLIMIT = 5                    5
BMC50470I GDGEMPTY = NO                   NO
BMC50470I GDGSCRAT = NO                   NO
BMC50470I STORCLAS = NONE                 NONE
BMC50470I DATACLAS = NONE                 NONE
BMC50470I MGMTCLAS = NONE                 NONE
BMC50470I UNITCNT  = 0                    0
BMC50470I SPACE    = CYL                  CYL
BMC50470I PCTPRIM  = 0                    0
BMC50470I MAXPRIM  = 0                    0
BMC50470I NBRSECD  = 0                    0
BMC50470I DISKRETN = NONE                 NONE
BMC50470I DISKEXPD = NONE                 NONE
BMC50470I RETPD    = NONE                 NONE
BMC50470I EXPDT    = 99000                99000
BMC50470I TRTCH    = NONE                 NONE


BMC50483I SYSREC     VOLUMES=NONE
BMC50483I SYSRED     VOLUMES=NONE

BMC50483I SYSREC     DSNAME=&USERID.&TYPE.S&SELNUM
BMC50483I SYSRED     DSNAME=&USERID.&TYPE.S&SELNUM

BMC50471I BMCUTIL='AUSDB2UT.BMC_BMCUTIL'
BMC50471I BMCSYNC='AUSDB2UT.BMC_BMCSYNC'
BMC50471I BMCHIST='AUSDB2UT.BMC_BMCHIST'
BMC50471I BMCXCOPY='AUSDB2UT.BMC_BMCXCOPY'

BMC50102I              UNLOAD CNTLCARDS
BMC50102I               'ENUMROWS' 'REPLACE' 'DELETEFILES YES SYSDISC YES'
BMC50102I              UNLOADDN (SYSREC) ACTIVE (YES) LIMIT 1
BMC50102I              ORDER YES
BMC50102I              FORMAT BMCLOAD
BMC50102I              SELECT * INTO NAME EXAMPLEF.TBLB
BMC50102I                     FROM EXAMPLEF.TBL1

BMC51654I DIRECT YES IN EFFECT

BMC51687I ESTIMATED ROWS FOR TABLE SPACE AMUEXFDB.EXFTS PART 1 IS 53
BMC51687I ESTIMATED ROWS FOR TABLE SPACE AMUEXFDB.EXFTS PART 2 IS 53
BMC51687I ESTIMATED ROWS FOR TABLE SPACE AMUEXFDB.EXFTS PART 3 IS 53
BMC51687I ESTIMATED ROWS FOR TABLE SPACE AMUEXFDB.EXFTS PART 4 IS 53
BMC51687I ESTIMATED ROWS FOR TABLE SPACE AMUEXFDB.EXFTS PART 5 IS 53
BMC51688I ESTIMATED ROWS FOR TABLE     EXAMPLEF.TBL1 IS 265
BMC51689I ESTIMATED ROWS FOR SELECT 1 IS 5

BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:01


BMC50445I UNLOAD PLUS DYNAMIC FILE ALLOCATION REPORT

BMC50446I                                                  UNIT OR                    KBYTES   KBYTES     ALOC      ALOC
BMC50447I DDNAME   DSNAME                                  DATACLAS MGMTCLAS STORCLAS     PRI      SEC      PRI       SEC

BMC50448I SYS00035 RDAMZL.SYSREC.S001                      SYSALLDA                         1        0        1         1  CYL
```

(continued on following page)

(example 14A continued from preceding page)

```
BMC51639I FOR DDNAME 'SYS00035' DSN=RDAMZL.SYSREC.S001,DCB=(RECFM=VB,BLKSIZE=27993,LRECL=117)
BMC50394I UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'

BMC50474I BELOW 16M = 7464K, ABOVE 16M = 1663236K, CPUS = 3
BMC51701I MAX TASKS = 5, MAX PARTITIONS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51680I UNLOAD LIMIT OF 1 REACHED FOR PARTITION 5
BMC50477I 5: PARTITION =   5, ROWS/KEYS = 1, I/O WAITS = 1, DDNAME = SYS00039
BMC51680I UNLOAD LIMIT OF 1 REACHED FOR PARTITION 2
BMC50477I 2: PARTITION =   2, ROWS/KEYS = 1, I/O WAITS = 1, DDNAME = SYS00036
BMC51680I UNLOAD LIMIT OF 1 REACHED FOR PARTITION 3
BMC50477I 3: PARTITION =   3, ROWS/KEYS = 1, I/O WAITS = 1, DDNAME = SYS00037
BMC51680I UNLOAD LIMIT OF 1 REACHED FOR PARTITION 4
BMC50477I 4: PARTITION =   4, ROWS/KEYS = 1, I/O WAITS = 1, DDNAME = SYS00038
BMC51680I UNLOAD LIMIT OF 1 REACHED FOR PARTITION 1
BMC50477I 1: PARTITION =   1, ROWS/KEYS = 1, I/O WAITS = 1, DDNAME = SYS00040
BMC50478I 1: RDB LOCK WAITS = 0
BMC50478I 2: RDB LOCK WAITS = 0
BMC50478I 3: RDB LOCK WAITS = 0
BMC50478I 4: RDB LOCK WAITS = 0
BMC50478I 5: RDB LOCK WAITS = 0
BMC50476I DDNAME = SYS00035, I/OS = 6, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51671I UNLOAD STATISTICS:  1 ROWS PROCESSED FROM PARTITION 1
BMC51686I UNLOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTS.I0001.A001' READ 3 PAGES
BMC51671I UNLOAD STATISTICS:  1 ROWS PROCESSED FROM PARTITION 2
BMC51686I UNLOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTS.I0001.A002' READ 3 PAGES
BMC51671I UNLOAD STATISTICS:  1 ROWS PROCESSED FROM PARTITION 3
BMC51686I UNLOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTS.I0001.A003' READ 3 PAGES
BMC51671I UNLOAD STATISTICS:  1 ROWS PROCESSED FROM PARTITION 4
BMC51686I UNLOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTS.I0001.A004' READ 3 PAGES
BMC51671I UNLOAD STATISTICS:  1 ROWS PROCESSED FROM PARTITION 5
BMC51686I UNLOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTS.I0001.A005' READ 3 PAGES
BMC51672I UNLOAD STATISTICS:  5 ROWS PROCESSED FROM SPACE 'AMUEXFDB.EXFTS', 0 NOT SELECTED, 0 DISCARDED
BMC51674I UNLOAD STATISTICS:  13 RECORDS WRITTEN TO DDNAME 'SYS00035'
BMC51679I UNLOAD STATISTICS:  5 DATA RECORDS AND 8 CONTROL RECORDS WRITTEN TO DDNAME 'SYS00035'
BMC51675I UNLOAD STATISTICS:  0 RECORDS DISCARDED DUE TO ERRORS
BMC50004I UNLOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:04


BMC51639I FOR DDNAME 'SYSCNTL' DSN=SYS03247.T204405.RA000.AMUEX14.UNLDCNTL.H02,DCB=(RECFM=FB,BLKSIZE=3120,LRECL=80)
BMC51801I LOAD TABLE STATEMENTS WRITTEN TO DDNAME 'SYSCNTL'

BMC51936I LOAD DATA INDSN ('RDAMZL.SYSREC.S001'
BMC51809I )
BMC51800I ENUMROWS (,13)
BMC51809I REPLACE
BMC51809I DELETEFILES YES SYSDISC YES
BMC51955I      FORMAT BMCUNLOAD
BMC51811I      INTO TABLE EXAMPLEF.TBLB
BMC51956I        WHEN TABLE=1

BMC50476I DDNAME = SYSCNTL, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 0
```

The LOADPLUS JCL for example 14 follows:

```
//        JOB
//EXAMPLEF EXEC PGM=AMUUMAIN,
//             PARM='&SSID,AMUEXFDB,NEW,,MSGLEVEL(1),AMU$OPTE'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB1
//         DD DISP=SHR,DSN=&AMULIB2
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SYSIN    DD DISP=SHR,DSN=&UNLDCNTL
```

BMC Software, Inc., Confidential and Proprietary Information

The LOADPLUS SYSPRINT for example 14 follows:

```
                              *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                  COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/04/2003   20:44:51 ...
BMC50002I UTILITY ID = 'AMUEXFDB'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTE'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7820K,ABOVE 16M=1664456K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=100
BMC50471I SQLDELAY=3
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                 WORK                 SORTWORK             ERROR                DISCARD
BMC50470I ACTIVE   = YES                  YES                  YES                  YES                  YES
BMC50470I IFALLOC  = FREE                 FREE                 FREE                 FREE                 FREE
BMC50470I SMS      = NO                   NO                   NO                   NO                   NO
BMC50470I SMSUNIT  = NO                   NO                   NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)            (100,100)            (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                    0                    0                    0                    0
BMC50470I MAXEXTSZ = 0                    0                    N/A                  0                    0
```

(continued on following page)

(example 14B continued from preceding page)

```
BMC50470I DDTYPE   = LOCPFCPY             LOCBFCPY             REMPFCPY             REMBFCPY
BMC50470I ACTIVE   = YES                   YES                  YES                  YES
BMC50470I IFALLOC  = FREE                  FREE                 FREE                 FREE
BMC50470I SMS      = NO                    NO                   NO                   NO
BMC50470I SMSUNIT  = NO                    NO                   NO                   NO
BMC50470I SIZEPCT  = (100,100)             (100,100)            (100,100)            (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA)   (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)  (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)           (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)           (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)           (NONE,NONE)          (NONE,NONE)          (NONE,NONE)
BMC50470I THRESHLD = 0                     0                    0                    0
BMC50470I MAXEXTSZ = 0                     0                    0                    0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                     5                    5                    5
BMC50470I GDGEMPTY = NO                    NO                   NO                   NO
BMC50470I GDGSCRAT = NO                    NO                   NO                   NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I LOAD DATA INDSN ('RDAMZL.SYSREC.S001'
BMC50102I )
BMC50102I ENUMROWS (,13)
BMC50102I REPLACE
BMC50102I DELETEFILES YES SYSDISC YES
BMC50102I      FORMAT BMCUNLOAD
BMC50102I      INTO TABLE EXAMPLEF.TBLB
BMC50102I        WHEN TABLE=1

BMC51424I ANALYZE PHASE WILL BE PERFORMED DUE TO DYNAMIC WORK FILE ALLOCATION
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50425I &JOBNAME  = AMUEX14   &STEPNAME = LODPLUS2  &DB       = AMUEXFDB  &TS       = EXFTSB    &UID      = RDAMZL
BMC50425I &RESUME   = N         &REPLACE  = Y         &DATE     = 090403    &TIME     = 204451    &SSID     = DEBA
BMC50425I &UTIL     = AMUEXFDB  &UTILPFX  = AMUEXFDB  &UTILSFX  =           &DATE8    = 09042003  &GRPNM    = DEBA
BMC50425I &VCAT     = DEBACAT   &TIME4    = 2044      &DATEJ    = 2003247

BMC50445I LOADPLUS DYNAMIC FILE ALLOCATION REPORT

BMC50446I                                                    UNIT OR                    KBYTES   KBYTES     ALOC      ALOC
BMC50447I DDNAME   DSNAME                                    DATACLAS MGMTCLAS STORCLAS     PRI      SEC      PRI       SEC

BMC50448I SORTOUT1 RDAMZL.AMUEXFDB.SORTOUT1                  SYSALLDA                         2        1        1         1  TRK
BMC50448I SORTOUT2 RDAMZL.AMUEXFDB.SORTOUT2                  SYSALLDA                         2        1        1         1  TRK
BMC50448I SORTOUT3 RDAMZL.AMUEXFDB.SORTOUT3                  SYSALLDA                         2        1        1         1  TRK
BMC50448I SORTOUT4 RDAMZL.AMUEXFDB.SORTOUT4                  SYSALLDA                         2        1        1         1  TRK
BMC50448I SORTOUT5 RDAMZL.AMUEXFDB.SORTOUT5                  SYSALLDA                         2        1        1         1  TRK
BMC50448I SORTWK01 RDAMZL.AMUEXFDB.SORTWK01                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK02 RDAMZL.AMUEXFDB.SORTWK02                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK03 RDAMZL.AMUEXFDB.SORTWK03                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK04 RDAMZL.AMUEXFDB.SORTWK04                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK05 RDAMZL.AMUEXFDB.SORTWK05                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK06 RDAMZL.AMUEXFDB.SORTWK06                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK07 RDAMZL.AMUEXFDB.SORTWK07                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK08 RDAMZL.AMUEXFDB.SORTWK08                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK09 RDAMZL.AMUEXFDB.SORTWK09                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK10 RDAMZL.AMUEXFDB.SORTWK10                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK11 RDAMZL.AMUEXFDB.SORTWK11                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SORTWK12 RDAMZL.AMUEXFDB.SORTWK12                  SYSALLDA                         1        1        1         1  TRK
BMC50448I SYSDISC  RDAMZL.AMUEXFDB.SYSDISC                   SYSALLDA                         2        1        1         1  TRK
BMC50448I SYSERR   RDAMZL.AMUEXFDB.SYSERR                    SYSALLDA                         1        1        1         1  TRK
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(example 14B continued from preceding page)

```
BMC50448I SYSUT101 RDAMZL.AMUEXFDB.SYSUT101                  SYSALLDA                        1        1        1         1  TRK
BMC50448I SYSUT102 RDAMZL.AMUEXFDB.SYSUT102                  SYSALLDA                        1        1        1         1  TRK

BMC50474I BELOW 16M = 7260K, ABOVE 16M = 1650156K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 5, #READERS = 1, INDEX TASKS = 1, TIME = 53760
BMC51495I PRELOAD OPTIMIZATION, RC = 14, #SORTS = 4, #READERS = 1, INDEX TASKS = 1, TIME = 58688
BMC51496I PRELOAD ANALYZE, #SORTS = 5, #READERS = 1, INDEX TASKS = 1
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 5, EMPTY WAITS = 0, FULL WAITS = 0
BMC51592I PRELOAD STATISTICS:  8 BMCUNLOAD VERIFICATION RECORDS READ FROM SYSREC01
BMC51478I PRELOAD STATISTICS:  5 PHYSICAL (5 LOGICAL) RECORDS READ FROM SYSREC01
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 3: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 4: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 5: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 6: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 2: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 3: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 4: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 5: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC51510I 6: SORT TASK, XBLK XFERS = 2, EMPTY WAITS = 0, FULL WAITS = 2
BMC50481I 7: INDEX SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC50481I 7: INDEX TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 7: INDEX TASK, XBLK XFERS = 5, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSREC01, I/OS = 7, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT1, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT2, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT3, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT4, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT5, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT101, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSUT102, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51507I XBLKS = 34, XFERS = 10, EMPTY WAITS = 0, FULL WAITS =11
BMC50476I DDNAME = SYSDISC, I/OS = 0, I/O WAITS = 0, RDB LOCK WAITS = 0
BMC51471I PRELOAD STATISTICS:  1 ROWS SELECTED FOR PARTITION 1
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A001' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A001' WILL REQUIRE 7 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  1 ROWS SELECTED FOR PARTITION 2
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A002' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A002' WILL REQUIRE 7 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  1 ROWS SELECTED FOR PARTITION 3
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A003' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A003' WILL REQUIRE 7 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  1 ROWS SELECTED FOR PARTITION 4
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A004' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A004' WILL REQUIRE 7 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  1 ROWS SELECTED FOR PARTITION 5
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A005' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A005' WILL REQUIRE 7 PAGES (APPROX.)
BMC51488I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEXFDB.IXB2.I0001.A001' MAY REQUIRE 7 PAGES
BMC51472I PRELOAD PHASE STATISTICS:  5 ROWS SELECTED FOR SPACE 'AMUEXFDB.EXFTSB', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  0 PHYSICAL (0 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:01

BMC50474I BELOW 16M = 7336K, ABOVE 16M = 1662404K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 5, #COPY TASKS = 0, #INDEX TASKS = 2
BMC51508I MAX INDEX TASKS = 2, INDEXES PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51508I MAX DATA TASKS = 5, MAX PARTS PER TASK = 1, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEXFDB.EXFTSB' DELETED
BMC50476I DDNAME = SYSUT102, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:00
BMC50482I 1: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A005'
BMC50477I 1: PARTITION =   5, ROWS/KEYS = 1, I/O WAITS = 5, DDNAME = SYS00062
BMC50482I 4: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A002'
BMC50477I 4: PARTITION =   2, ROWS/KEYS = 1, I/O WAITS = 5, DDNAME = SYS00065
BMC50482I 3: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A001'
BMC50477I 3: PARTITION =   1, ROWS/KEYS = 1, I/O WAITS = 5, DDNAME = SYS00063
BMC50482I 2: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A003'
BMC50477I 2: PARTITION =   3, ROWS/KEYS = 1, I/O WAITS = 5, DDNAME = SYS00061
BMC51474I LOAD STATISTICS:  1 ROWS LOADED INTO PARTITION 5
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.EXFTSB.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 1, I/O WAITS = 5, DDNAME = SYS00064
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A001'
BMC50477I 1: PARTITION =   1, ROWS/KEYS = 1, I/O WAITS = 7, DDNAME = SYS00066
BMC51474I LOAD STATISTICS:  1 ROWS LOADED INTO PARTITION 2
BMC51474I BUILD STATISTICS:  1 KEYS LOADED INTO PARTITION 1
BMC51474I LOAD STATISTICS:  1 ROWS LOADED INTO PARTITION 1
BMC51474I LOAD STATISTICS:  1 ROWS LOADED INTO PARTITION 3
BMC50476I DDNAME = SORTOUT2, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51474I LOAD STATISTICS:  1 ROWS LOADED INTO PARTITION 4
BMC50476I DDNAME = SORTOUT5, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT4, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT1, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT3, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51475I LOAD STATISTICS:  5 ROWS LOADED INTO TABLE SPACE 'AMUEXFDB.EXFTSB'
BMC50482I 2: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.IXB2.I0001.A001'
```

(continued on following page)

(example 14B continued from preceding page)

```
BMC50477I 2: PARTITION =   0, ROWS/KEYS = 5, I/O WAITS = 7, DDNAME = SYS00068
BMC51476I BUILD STATISTICS:  5 KEYS LOADED INTO INDEX 'EXAMPLEF.IXB2'
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A002'
BMC50477I 1: PARTITION =   2, ROWS/KEYS = 1, I/O WAITS = 7, DDNAME = SYS00071
BMC51474I BUILD STATISTICS:  1 KEYS LOADED INTO PARTITION 2
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A003'
BMC50477I 1: PARTITION =   3, ROWS/KEYS = 1, I/O WAITS = 7, DDNAME = SYS00074
BMC51474I BUILD STATISTICS:  1 KEYS LOADED INTO PARTITION 3
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:00  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A004'
BMC50477I 1: PARTITION =   4, ROWS/KEYS = 1, I/O WAITS = 7, DDNAME = SYS00077
BMC51474I BUILD STATISTICS:  1 KEYS LOADED INTO PARTITION 4
BMC50482I 1: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEXFDB.IXB1.I0001.A005'
BMC50477I 1: PARTITION =   5, ROWS/KEYS = 1, I/O WAITS = 7, DDNAME = SYS00080
BMC51474I BUILD STATISTICS:  1 KEYS LOADED INTO PARTITION 5
BMC50476I DDNAME = SYSUT101, I/OS = 3, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51476I BUILD STATISTICS:  5 KEYS LOADED INTO INDEX 'EXAMPLEF.IXB1'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SYSERR  ', DSNAME = 'RDAMZL.AMUEXFDB.SYSERR'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK01', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK01'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK02', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK02'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK03', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK03'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK04', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK04'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK05', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK05'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK06', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK06'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK07', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK07'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK08', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK08'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK09', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK09'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK10', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK10'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK11', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK11'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTWK12', DSNAME = 'RDAMZL.AMUEXFDB.SORTWK12'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SYSUT101', DSNAME = 'RDAMZL.AMUEXFDB.SYSUT101'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTOUT3', DSNAME = 'RDAMZL.AMUEXFDB.SORTOUT3'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTOUT1', DSNAME = 'RDAMZL.AMUEXFDB.SORTOUT1'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTOUT4', DSNAME = 'RDAMZL.AMUEXFDB.SORTOUT4'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTOUT5', DSNAME = 'RDAMZL.AMUEXFDB.SORTOUT5'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SORTOUT2', DSNAME = 'RDAMZL.AMUEXFDB.SORTOUT2'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SYSUT102', DSNAME = 'RDAMZL.AMUEXFDB.SYSUT102'
BMC50318I DATASET SUCCESSFULLY DELETED, DDNAME = 'SYSDISC ', DSNAME = 'RDAMZL.AMUEXFDB.SYSDISC'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:07

BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEXFDB.EXFTSB
BMC50513I             PARTS   = 5         TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 15            SPACE   = 110
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL   DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED  PSAVED
BMC50512I   1         3          1          0          0        2        0    2    0      22    1           0    0         0     0
BMC50512I   2         3          1          0          0        2        0    2    0      22    1           0    0         0     0
BMC50512I   3         3          1          0          0        2        0    2    0      22    1           0    0         0     0
BMC50512I   4         3          1          0          0        2        0    2    0      22    1           0    0         0     0
BMC50512I   5         3          1          0          0        2        0    2    0      22    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPLEF.TBLB                          5          5         0      69      33       0
BMC50510I INDEX       EXAMPLEF.IXB1  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLEF.TBLB     COLUMN  RECNO
BMC50514I             NACTIVE = 30            SPACE   = 110
BMC50542I             FIRSTKEY= 5             FULLKEY = 5             NLEAF   = 5
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 5                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'0080014C08404040'  LOW2KEY= X'0080000001404040'
BMC50547I             HIGH2KEY=           LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   1         6          1        0          0          1        0        7        1     2        22       1
BMC50532I   2         6          1        0          1          0        0        7        1     2        22       1
BMC50532I   3         6          1        0          1          0        0        7        1     2        22       1
BMC50532I   4         6          1        0          1          0        0        7        1     2        22       1
BMC50532I   5         6          1        0          1          0        0        7        1     2        22       1
BMC50510I INDEX       EXAMPLEF.IXB2  (TYPE 2)
BMC50541I             ON TABLE  EXAMPLEF.TBLB     COLUMN  RECNO2
BMC50514I             NACTIVE = 6             SPACE   = 22
BMC50542I             FIRSTKEY= 5             FULLKEY = 5             NLEAF   = 1
BMC50543I             LEVELS  = 2                                     PCTCLUST= 60
BMC50544I             KEYLEN  = 4                                     COLCOUNT= 1
BMC50545I             SUBPAGE = 1             CLUSTER = N             UNIQUE  = D
BMC50546I             HIGH2KEY= X'8000000640404040'  LOW2KEY= X'8000000340404040'
BMC50547I             HIGH2KEY=           LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   0         6          5        0          4          1        0        7        1     2        22       1
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

The output that shows the loaded table for example 14 follows:

```
PAGE    1
***INPUT STATEMENT:
  SELECT * FROM EXAMPLEF.TBLB;
              +-------------------------------------------------------------------------------------------------------+
              |     RECNO      |     INPUT_A     |     RECNO2     |     INPUT_B     |     INPUT_C     |     INPUT_D     |
              +-------------------------------------------------------------------------------------------------------+
            1_|             1 | AAAAAAAAAA      |             1 |  ROW1 DATA      | AAAAAAAAAAAAAAA | BBBBBBBBBBBBBBB |
            2_|         35000 | CCCCCCCCCC      |             3 |  ROW3 DATA      | EEEEEEEEEEEEEEE | FFFFFFFFFFFFFFF |
            3_|         45000 | DDDDDDDDDD      |             4 |  ROW4 DATA      | GGGGGGGGGGGGGGG | HHHHHHHHHHHHHHH |
            4_|         65000 | FFFFFFFFFF      |             6 |  ROW6 DATA      | KKKKKKKKKKKKKKK | LLLLLLLLLLLLLLL |
            5_|         85000 | HHHHHHHHHH      |             8 |  ROW8 DATA      | NNNNNNNNNNNNNNN | PPPPPPPPPPPPPPP |
              +-------------------------------------------------------------------------------------------------------+
SUCCESSFUL RETRIEVAL OF        5 ROW(S)
```

Example 15: LOAD REPLACE Ignoring Indexes

This example is a LOAD REPLACE of a 10-part table with a unique clustering index and both a unique and a nonunique secondary index. In this example, LOADPLUS ignores both of the secondary indexes, which means that LOADPLUS does not check for uniqueness in the unique secondary index and does not build or update either secondary index. Message BMC51493W indicates that the two indexes were skipped and placed in REBUILD pending status.

The WHEN option illustrates using a predicate that compares a (*start:end*) position in the file to a hex constant.

Because the LOAD command includes the ORDER YES option, LOADPLUS sorts the data rows and clustering index keys. Because the LOAD command does not include the UNIQUECHECK NO option, LOADPLUS defaults to UNIQUECHECK YES, which tells LOADPLUS to check the clustering index for uniqueness.

The COPY NO option directs LOADPLUS not to make a copy of the table space. Because the COPYPEND installation default option is set to YES, LOADPLUS places the table space in COPY pending status.

The JCL for example 15 follows:

```
//        JOB
//EXAMPL15 EXEC PGM=AMUUMAIN,
//             PARM='&SSID,EXAMPL15,NEW/RESTART,,MSGLEVEL(1),AMU$OPTO'
//STEPLIB  DD DISP=SHR,DSN=&AMULIB
//         DD DISP=SHR,DSN=&DSNEXIT
//         DD DISP=SHR,DSN=&DSNLOAD
//         DD DISP=SHR,DSN=&PPLIB
//         DD DISP=SHR,DSN=&EDITPR
//SYSPRINT  DD SYSOUT=*
//SYSOUT   DD SYSOUT=*
//UTPRINT  DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//SORTOUT  DD DSN=AMU.EXAMPL15.SORTOUT,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSERR   DD DSN=AMU.EXAMPL15.SYSERR,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//SYSDISC  DD DSN=AMU.EXAMPL15.SYSDISC,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(NEW,CATLG,CATLG)
//*
//SYSUT101 DD DSN=AMU.EXAMPL15.SYSUT1,
//            UNIT=WORK,SPACE=(CYL,(1,1)),DISP=(MOD,CATLG,CATLG)
//*
//SORTWK01 DD UNIT=WORK,DISP=NEW,SPACE=(CYL,(1,1))
//SYSREC   DD DSN=AMU.QA.RGRTESTS(EX6DATA),
//            DISP=SHR
//*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
//SYSIN    DD *
  LOAD REPLACE
    ORDER YES SKIPIX SIX
  INTO TABLE EXAMPL15.TBL1
       WHEN (21:25) = X'000555555C'
      (DATE_OF_SALE        POSITION(67:72) DATE-2 EXTERNAL
      ,SALES_TRANS_ID      POSITION(47:56) INTEGER EXTERNAL
      ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
      ,SALES_QTY           POSITION(11:20) INTEGER EXTERNAL
      ,SALES_PR_PER_ITEM   POSITION(21:25) DECIMAL         (9,2)
      ,SALES_TOTAL_TX      POSITION(26:34) DECIMAL EXTERNAL(9,2)
      ,SALES_TOTAL         POSITION(37:45) DECIMAL EXTERNAL(9,2)
      ,SELLER_ID           POSITION(57:66) CHARACTER
      )
  LOG NO
  COPY NO
```

The SYSPRINT for example 15 follows:

```
                          *****  B M C   L O A D P L U S   F O R   D B 2    V7R1.00  *****
                COPYRIGHT (C) 1990 - 2003 BMC SOFTWARE, INC. AS AN UNPUBLISHED LICENSED WORK. ALL RIGHTS RESERVED.
BMC50001I UTILITY EXECUTION STARTING    9/10/2003   11:38:49 ...
BMC50002I UTILITY ID = 'EXAMPL15'.  DB2 SUBSYSTEM ID = 'DEBA'.  OPTION MODULE = 'AMU$OPTB'.
BMC50471I OS/390 2.10.0,PID=HBB7703,DFSMS FOR OS/390=2.10.0,DB2=7.1.0
BMC50471I REGION=0M,BELOW 16M=7872K,ABOVE 16M=1666976K,IEFUSI=NO,CPUS=3
BMC50471I DIAG1=00000000000000000000000000000000
BMC50471I DIAG2=00000000000000000000000000000000
BMC50471I DIAG3=F0000000000000000000000000000000
BMC50471I DIAG4=00000000000000000000000000000000
BMC50471I SMCORE=(0K,0K)
BMC50471I STOP@CMT=NO
BMC50471I SMAX=8
BMC50471I MAXP=5
BMC50471I XBLKS=3
BMC50471I CBUFFS=30
BMC50471I LBUFFS=20
BMC50471I WBUFFS=(20,10)
BMC50471I IBUFFS=20
BMC50471I OPNDB2ID=YES
BMC50471I SQLRETRY=3
BMC50471I SQLDELAY=60
BMC50471I COPYLVL=FULL
BMC50471I COPYDDN=(BMCCPY,BMCCPZ)
BMC50471I RCVYDDN=(BMCRCY,BMCRCZ)
BMC50471I UXSTATE=SUP
BMC50471I CHEKPEND=YES
BMC50471I FILECHK=WARN
BMC50471I RULES=BMC
BMC50471I IMAGECPY=YES
BMC50471I LOADCPY=YES
BMC50471I WORKUNIT=SYSALLDA
BMC50471I MAXTAPE=3
BMC50471I DSNUEXIT=(NONE,ASM)
BMC50471I TAPEDISP=DELETE
BMC50471I CENTURY=(1950,2049)
BMC50471I LOADDECP=NO
BMC50471I MSGLEVEL=1
BMC50471I LOCKROW=NO
BMC50471I DELFILES=(NO,NO)
BMC50471I ACFORTSS=NO
BMC50471I INLINECP=NO
BMC50471I INDDN=SYSREC
BMC50471I WORKDDN=SYSUT1
BMC50471I LOADDN=SORTOUT
BMC50471I ERRDDN=SYSERR
BMC50471I DISCDDN=SYSDISC
BMC50471I IDCDDN=SYSIDCIN
BMC50471I RENMMAX=30
BMC50471I ORIGDISP=DELETE
BMC50471I APMXAGNT=10
BMC50471I APCOMMIT=2500
BMC50471I APRETLIM=COUNT
BMC50471I APRETVAL=5
BMC50471I APCOLLECTION=
BMC50471I APOWNER=
BMC50471I IDERROR=DISCARD
BMC50471I IDCACHE=1000
BMC50471I UPDMAXA=NO
```

(continued on following page)

(example 15 continued from preceding page)

```
BMC50471I KEEPDICTIONARY=NO
BMC50471I PREFORMAT=NO
BMC50471I REDEFINE=YES
BMC50471I COPYPEND=YES
BMC50471I STOPRETRY=300
BMC50471I STOPDELAY=1
BMC50471I PLANAUTH=AMUADIST
BMC50471I PLANCAT =AMUCDIST
BMC50471I PLANSER =AMURDIST
BMC50471I PLANSYNC=AMUSDIST
BMC50471I PLANSTAT=AMUTDIST
BMC50471I PLANCOPY=AMUIDIST
BMC50471I PLANCHKP=AMUPDIST

BMC50470I DDTYPE   = LOAD                WORK                SORTWORK            ERROR               DISCARD
BMC50470I ACTIVE   = NO                  NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   N/A                 0                   0

BMC50470I DDTYPE   = LOCPFCPY            LOCBFCPY            REMPFCPY            REMBFCPY
BMC50470I ACTIVE   = NO                  NO                  NO                  NO
BMC50470I IFALLOC  = USE                 USE                 USE                 USE
BMC50470I SMS      = NO                  NO                  NO                  NO
BMC50470I SMSUNIT  = NO                  NO                  NO                  NO
BMC50470I SIZEPCT  = (100,100)           (100,100)           (100,100)           (100,100)
BMC50470I UNIT     = (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA) (SYSALLDA,SYSALLDA)
BMC50470I DATACLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I MGMTCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I STORCLAS = (NONE,NONE)         (NONE,NONE)         (NONE,NONE)         (NONE,NONE)
BMC50470I THRESHLD = 0                   0                   0                   0
BMC50470I MAXEXTSZ = 0                   0                   0                   0
BMC50470I EXPDT    =
BMC50470I RETPD    =
BMC50470I GDGLIMIT = 5                   5                   5                   5
BMC50470I GDGEMPTY = NO                  NO                  NO                  NO
BMC50470I GDGSCRAT = NO                  NO                  NO                  NO

BMC50483I LOAD       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I WORK       DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I SORTWORK   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I ERROR      DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I DISCARD    DSNPAT=&UID.&UTILPFX.&DDNAME

BMC50483I LOCPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I LOCBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMPFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50483I REMBFCPY   DSNPAT=&UID.&UTILPFX.&DDNAME
BMC50471I BMCUTIL   ='AUSDB2UT.BMCUTIL'
BMC50471I BMCSYNC   ='AUSDB2UT.BMCSYNC'
BMC50471I BMCHIST   ='AUSDB2UT.BMCHIST'
BMC50471I BMCDICT   ='AUSDB2UT.BMCDICT'
BMC50471I BMCCOPY   ='AUSDB2UT.BMCCOPY'
BMC50471I BMCTPART  ='AUSDB2UT.BMCTPART'
BMC50471I BMCSEQS   ='AUSDB2UT.BMCSEQUENCES'
BMC50471I DASD MANAGER PLUS TABLES:
BMC50471I ...TABLESPACES ='AUSDB2UT.V71_RS_TABLESPACE'
BMC50471I ...TABLEPARTS  ='AUSDB2UT.V71_RS_TABLEPART'
BMC50471I ...TABLES      ='AUSDB2UT.V71_RS_TABLES'
BMC50471I ...INDEXES     ='AUSDB2UT.V71_RS_INDEXES'
BMC50471I ...INDEXPARTS  ='AUSDB2UT.V71_RS_INDEXPART'

BMC50102I   LOAD REPLACE
BMC50102I     ORDER YES SKIPIX SIX
BMC50102I   INTO TABLE EXAMPL15.TBL1
BMC50102I        WHEN (21:25) = X'000555555C'
BMC50102I       (DATE_OF_SALE        POSITION(67:72) DATE-2 EXTERNAL
BMC50102I       ,SALES_TRANS_ID      POSITION(47:56) INTEGER EXTERNAL
BMC50102I       ,SALES_ITEM_ID       POSITION(1:10)  CHARACTER
BMC50102I       ,SALES_QTY           POSITION(11:20) INTEGER EXTERNAL
BMC50102I       ,SALES_PR_PER_ITEM   POSITION(21:25) DECIMAL         (9,2)
BMC50102I       ,SALES_TOTAL_TX      POSITION(26:34) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SALES_TOTAL         POSITION(37:45) DECIMAL EXTERNAL(9,2)
BMC50102I       ,SELLER_ID           POSITION(57:66) CHARACTER
BMC50102I       )
BMC50102I   LOG NO
BMC50102I   COPY NO
```

(continued on following page)

(example 15 continued from preceding page)

```
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'SALES_ITEM_DESC   '.  VALUE IS DEFAULTED
BMC51421I NO FIELD SPECIFICATION FOR COLUMN 'ENTRY_TIMESTAMP   '.  VALUE IS DEFAULTED
BMC50004I UTILINIT PHASE COMPLETE.  ELAPSED TIME = 00:00:03

BMC50004I ANALYZE PHASE COMPLETE.  ELAPSED TIME = 00:00:00

BMC50474I BELOW 16M = 7384K, ABOVE 16M = 1656340K, CPUS = 3
BMC51495I PRELOAD OPTIMIZATION, RC = 0, #SORTS = 1, #READERS = 1, INDEX TASKS = 0, TIME = 74240
BMC51496I PRELOAD ANALYZE, #SORTS = 1, #READERS = 1, INDEX TASKS = 0
BMC50481I 1: READ TASK COMPLETE.  ELAPSED TIME = 00:00:00
BMC51510I 1: READ TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 0
BMC51478I PRELOAD STATISTICS:  10 PHYSICAL (10 LOGICAL) RECORDS READ FROM SYSREC
BMC50481I 2: SORT COMPLETE.  ELAPSED TIME = 00:00:05
BMC50481I 2: DATA TASK COMPLETE.  ELAPSED TIME = 00:00:06
BMC51510I 2: SORT TASK, XBLK XFERS = 1, EMPTY WAITS = 0, FULL WAITS = 1
BMC50476I DDNAME = SYSREC, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SORTOUT, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSERR, I/OS = 2, I/O WAITS = 2, RDB LOCK WAITS = 0
BMC51507I XBLKS = 5, XFERS = 1, EMPTY WAITS = 0, FULL WAITS =1

BMC51521I LOADPLUS ERROR SUMMARY REPORT FOR SYSREC, ID# 1

BMC51522I  PHYSICAL   LOGICAL   DISCARD     DISCARD            RELATED       TABLE                        FIELD, INDEX, OR
BMC51523I  RECORD     RECORD    RECORD      TYPE              ID: RECORD     NAME                         CONSTRAINT NAME

BMC51524E         1         1         1  WHEN                  0:         0
BMC51524E         2         2         2  WHEN                  0:         0
BMC51524E         3         3         3  WHEN                  0:         0
BMC51524E         4         4         4  WHEN                  0:         0
BMC51524E         6         6         5  WHEN                  0:         0
BMC51524E         7         7         6  WHEN                  0:         0
BMC51524E         8         8         7  WHEN                  0:         0
BMC51524E         9         9         8  WHEN                  0:         0
BMC51524E        10        10         9  WHEN                  0:         0
BMC50476I DDNAME = SYSREC, I/OS = 2, I/O WAITS = 1, RDB LOCK WAITS = 0

BMC50476I DDNAME = SYSERR, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50476I DDNAME = SYSDISC, I/OS = 1, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 1
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A001' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A001' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 2
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A002' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A002' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 3
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A003' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A003' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 4
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A004' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A004' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 5
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A005' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A005' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 6
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A006' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A006' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 7
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A007' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A007' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 8
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A008' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A008' WILL REQUIRE 3 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  1 ROWS SELECTED FOR PARTITION 9
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A009' WILL REQUIRE 3 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A009' WILL REQUIRE 4 PAGES (APPROX.)
BMC51471I PRELOAD STATISTICS:  0 ROWS SELECTED FOR PARTITION 10
BMC51486I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A010' WILL REQUIRE 2 PAGES
BMC51490I LOADING OF DATASET 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A010' WILL REQUIRE 3 PAGES (APPROX.)
BMC51472I PRELOAD PHASE STATISTICS:  1 ROWS SELECTED FOR SPACE 'AMUEX15D.EX15TS', 0 ROWS SELECTED BUT DISCARDED DUE TO ERRORS
BMC51479I PRELOAD PHASE STATISTICS:  9 PHYSICAL (9 LOGICAL) RECORDS DISCARDED TO SYSDISC
BMC50004I PRELOAD PHASE COMPLETE.  ELAPSED TIME = 00:00:09

BMC50474I BELOW 16M = 7472K, ABOVE 16M = 1666080K, CPUS = 3
BMC51498I LOAD OPTIMIZATION, RC = 0, #LOAD TASKS = 1, #COPY TASKS = 0, #INDEX TASKS = 0
BMC51508I MAX DATA TASKS = 1, MAX PARTS PER TASK = 10, SORTWKS PER TASK = 0, MAX OPEN PARTITIONS PER TASK = 1
BMC51453I EXISTING ROWS IN TABLESPACE 'AMUEX15D.EX15TS' DELETED
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A009'
BMC50477I 0: PARTITION =   9, ROWS/KEYS = 1, I/O WAITS = 7, DDNAME = SYS00012
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:07  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A009'
BMC50477I 0: PARTITION =   9, ROWS/KEYS = 1, I/O WAITS = 5, DDNAME = SYS00009
BMC51474I LOAD STATISTICS:  1 ROWS LOADED INTO PARTITION 9
BMC50476I DDNAME = SORTOUT, I/OS = 3, I/O WAITS = 1, RDB LOCK WAITS = 0
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A001'
BMC50477I 0: PARTITION =   1, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00015
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A001'
BMC50477I 0: PARTITION =   1, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00018
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 1
```

(continued on following page)

(example 15 continued from preceding page)

```
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00021
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A002'
BMC50477I 0: PARTITION =   2, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00024
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 2
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A003'
BMC50477I 0: PARTITION =   3, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00027
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A003'
BMC50477I 0: PARTITION =   3, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00030
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 3
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00033
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A004'
BMC50477I 0: PARTITION =   4, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00036
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 4
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A005'
BMC50477I 0: PARTITION =   5, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00039
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:01  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A005'
BMC50477I 0: PARTITION =   5, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00042
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 5
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A006'
BMC50477I 0: PARTITION =   6, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00045
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A006'
BMC50477I 0: PARTITION =   6, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00048
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 6
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:02  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A007'
BMC50477I 0: PARTITION =   7, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00051
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A007'
BMC50477I 0: PARTITION =   7, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00054
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 7
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A008'
BMC50477I 0: PARTITION =   8, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00057
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:08  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A008'
BMC50477I 0: PARTITION =   8, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00060
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 8
BMC50482I 0: BUILD COMPLETE.  ELAPSED TIME = 00:00:04  DSN = 'DEBACAT.DSNDBD.AMUEX15D.IX0.I0001.A010'
BMC50477I 0: PARTITION =  10, ROWS/KEYS = 0, I/O WAITS = 7, DDNAME = SYS00063
BMC50482I 0: LOAD COMPLETE.  ELAPSED TIME = 00:00:03  DSN = 'DEBACAT.DSNDBD.AMUEX15D.EX15TS.I0001.A010'
BMC50477I 0: PARTITION =  10, ROWS/KEYS = 0, I/O WAITS = 5, DDNAME = SYS00066
BMC51474I LOAD STATISTICS:  0 ROWS LOADED INTO PARTITION 10
BMC51476I BUILD STATISTICS:  1 KEYS LOADED INTO INDEX 'EXAMPL15.IX0'
BMC51475I LOAD STATISTICS:  1 ROWS LOADED INTO TABLE SPACE 'AMUEX15D.EX15TS'
BMC50004I LOAD PHASE COMPLETE.  ELAPSED TIME = 00:01:08

BMC50387W IMAGE COPY REQUIRED.  TABLE SPACE STATE SET TO "COPY PENDING"
BMC50501I DB2 OBJECT STATISTICS
BMC50510I TABLESPACE  AMUEX15D.EX15TS
BMC50513I             PARTS   = 10        TABLES  = 1         SEGSIZE = 0
BMC50514I             NACTIVE = 21            SPACE   = 150
BMC50511I PART  NACTIVE       CARD     FARIND    NEARIND     FULL    DIRTY PACT PDRP   SPACE EXTENTS   DBCARD PCOMP   KSAVED PSAVED
BMC50512I   1         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   2         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   3         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   4         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   5         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   6         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   7         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   8         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50512I   9         3          1          0          0        2        0    2    0      15    1           0    0         0     0
BMC50512I  10         2          0          0          0        2        0    0    0      15    1           0    0         0     0
BMC50521I TABLE NAME                          CARD     NPAGES    INDREF  ROWAVG  PCTPAGES  PCOMP
BMC50522I EXAMPL15.TBL1                          1          1         0      68       4       0
BMC50510I INDEX       EXAMPL15.IX0  (TYPE 2)
BMC50541I             ON TABLE  EXAMPL15.TBL1     COLUMN  DATE_OF_SALE
BMC50514I             NACTIVE = 50            SPACE   = 10
BMC50542I             FIRSTKEY= 1             FULLKEY = 1             NLEAF   = 10
BMC50543I             LEVELS  = 2                                     PCTCLUST= 100
BMC50544I             KEYLEN  = 9                                     COLCOUNT= 2
BMC50545I             SUBPAGE = 1             CLUSTER = Y             UNIQUE  = U
BMC50546I             HIGH2KEY= X'1990012500800000'  LOW2KEY= X'1990012500800000'
BMC50547I             HIGH2KEY=        LOW2KEY=
BMC50531I PART  NACTIVE       CARD LEAFDIST     FAROFF    NEAROFF     FULL     FREE    NLEAF  LEVELS   SPACE  EXTENTS
BMC50532I   1         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   2         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   3         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   4         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   5         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   6         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   7         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   8         5          0        0          0          0        0        7        1     2        1       1
BMC50532I   9         5          1        0          1          0        0        7        1     2        1       1
BMC50532I  10         5          0        0          0          0        0        7        1     2        1       1
BMC51493W NON-UNIQUE INDEX 'EXAMPL15.IX1' WAS SKIPPED AND HAS BEEN PLACED IN REBUILD PENDING STATUS
BMC51493W UNIQUE INDEX 'EXAMPL15.IX2' WAS SKIPPED AND HAS BEEN PLACED IN REBUILD PENDING STATUS
BMC50006I UTILITY EXECUTION COMPLETE, RETURN CODE = 4
```

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Chapter 6 Performance Considerations

This chapter presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Overview

The LOADPLUS product provides several options that you can use to influence its performance. Some options affect the performance of the entire utility; others influence processing only for specific functions of LOADPLUS. This chapter explains the effect that these options have on the performance of LOADPLUS.

At installation, certain default processing options are specified for LOADPLUS by means of installation jobs that contain options macros that establish these processing options. If you modify any of these options, you must rerun the installation job for these modifications to take effect. For more information, see Appendix A, “LOADPLUS Installation Options.”

You can customize other performance-tuning options to meet the needs of individual executions of the LOADPLUS utility by specifying them on the LOAD command.

General Performance Considerations

The architecture of LOADPLUS differs from that of IBM’s DB2 LOAD utility. The IBM DB2 LOAD utility performs load processing in several phases—RELOAD, SORT, BUILD, INDEXVAL, ENFORCE, DISCARD, and REPORT—which run serially. However, LOADPLUS combines these phases into either a two-phase architecture or a single-phase architecture.

When you specify PRELOAD CONTINUE (the default), PRELOAD PAUSE or PRELOAD ANALYZE, LOADPLUS uses two phases—PRELOAD and LOAD. When you specify PRELOAD LOAD or LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY (also referred to as SQLAPPLY load), LOADPLUS combines the PRELOAD and LOAD phases into a single phase called COMBINED. The PRELOAD and LOAD phases or the COMBINED phase perform all the functions that are included in the multiple phases performed during execution of the IBM DB2 LOAD utility, except the referential integrity functions that occur during the ENFORCE phase.

With its two-phase architecture, LOADPLUS performs several processes concurrently, thus reducing the elapsed time for load processing over the IBM DB2 LOAD utility. For example, LOADPLUS checks for duplicate keys (IBM’s INDEXVAL phase) while it reads input data (IBM’s RELOAD phase) and sorts index keys (IBM’s SORT phase).

BMC Software, Inc., Confidential and Proprietary Information

The LOADPLUS single-phase architecture builds on the advantages of the two-phase processing and allows for even greater reductions in CPU usage and elapsed time due to the reduction of read and write operations. For most load jobs, single-phase processing eliminates the need for work data sets (SORTOUT and SYSUT1), because LOADPLUS writes the data directly to the table space and index.

LOADPLUS exploits the technology that large scale processors provide. During execution, LOADPLUS examines available resources and uses as much of these resources as possible to maximize performance.

Loading Data from UNLOAD PLUS

When used in conjunction with UNLOAD PLUS, LOADPLUS provides a high-speed option to move data from one table to another table that has a similar table structure. This feature is useful, for example, for migrating data from test to development databases. To unload the data, you specify the FORMAT BMCLOAD option in UNLOAD PLUS. UNLOAD PLUS unloads the data in an internal format that only LOADPLUS can use. To load the data, you specify the FORMAT BMCUNLOAD option on your LOAD command.

Because the data is never converted to an external format and because LOADPLUS does not perform data verification, using this option can help reduce CPU cycles and elapsed time for your load job as well as for the entire process of migrating data.

For additional information, see the *UNLOAD PLUS for DB2 Reference Manual* and “Data from UNLOAD PLUS in Internal Format” on page 2-44 in this book.

Using Dynamic Work File Allocation

Although dynamically allocating each data set does require some processing time, enabling dynamic work file allocation can improve performance in many cases. If you are loading very small quantities of data, the extra processing time might be a non-trivial percentage of the elapsed time. However, if you are loading large quantities of data, especially in a partitioned data set, using dynamic work file allocation might improve performance because it always uses the optimal number of data sets, enabling LOADPLUS to use DASD more efficiently.

BMC Software, Inc., Confidential and Proprietary Information

You can improve performance further with dynamic allocation by providing a value for ENUMROWS that is as accurate as possible. When you specify this option, BMC Software recommends that you overestimate rather than underestimate. Overestimating results in LOADPLUS allocating files that are larger than needed, but underestimating can cause LOADPLUS to terminate the job. In general, the best files to dynamically allocate are SORTOUT, SYSUT1, and SORTWK. For details, see “Dynamic Work File Allocation” on page 2-13 and “Dynamic Work File Allocation Options” on page 3-133.

Creating Inline Image Copies

LOADPLUS provides the ability to create inline image copies. LOADPLUS creates inline copies *as* it loads your table space, rather than *after* it loads your table space. This process can reduce the elapsed time of your load job. For details about creating inline image copies, refer to “Copy Options” on page 3-80.

Invoking MODESET for DB2 User Exits

LOADPLUS can invoke the following types of DB2 user exits during processing:

- EDITPROCs
- VALIDPROCs
- FIELDPROCs
- date exits
  - DSNXVDTX (EBCDIC date exit routine)
  - DSNXVDTA (ASCII date exit routine)
- time exits
  - DSNXVTMX (EBCDIC time exit routine)
  - DSNXVTMA (ASCII time exit routine)
- an authorization exit

By default, LOADPLUS invokes these exits in supervisor state and PSW key=7. LOADPLUS calls the MODESET SVC to perform this switch before invoking an exit, and again to reset the mode after returning from the exit.

BMC Software, Inc., Confidential and Proprietary Information

However, invoking MODESET is an expensive CPU process. If you can ensure that all exits that LOADPLUS calls can run in problem state, you can specify UXSTATE=PROB in the installation options. This option causes LOADPLUS to invoke user exits in problem state (PSW key=7) and achieves significant savings in CPU overhead.

Using Staging Data Sets

When running a load job that uses staging data sets (LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE), you can improve the performance of the job by specifying REDEFINE NO and preallocating the staging data sets.

Materializing DEFINE NO Data Sets

As required by DB2, if you are loading at least one partition of a table whose table space or index spaces are created with DEFINE NO, LOADPLUS materializes all partitions. For a table space with many partitions, this materialization might affect the performance of your load job. Subsequent jobs on the same table space, however, are not affected.

Enabling Multitasking

During execution, LOADPLUS determines the most effective arrangement of tasks when running in a multiprocessor environment. Although LOADPLUS runs very efficiently on single-processor computers, it performs most efficiently on large multiprocessor systems.

Loading data into DB2 tables requires several distinct tasks, including converting data, verifying indexes, loading data, building indexes, and creating image copies. LOADPLUS can perform these tasks concurrently, thus reducing the elapsed time of the load process.

You can specify multiple data sets for many of the data set types that LOADPLUS uses, including input data sets, work data sets, and copy data sets. For details about specifying multiple data sets when building your job, see “LOADPLUS DD Statements” on page 4-8. Additional performance considerations for each data set type are discussed in “Tuning I/O” on page 6-8.

BMC Software, Inc., Confidential and Proprietary Information

Loading Partitioned Table Spaces

LOADPLUS multitasks as much as possible in all cases. When loading partitioned table spaces, you can obtain additional multitasking benefits when multiple SYSREC and SORTOUT data sets are available. Multiple SYSREC and SORTOUT support provides the maximum throughput in the least amount of elapsed time when loading partitioned table spaces. While all types of load processes support multiple SYSREC data sets, LOADPLUS reads multiple SYSREC data sets *concurrently* only when loading a partitioned table space.

When using multiple SYSREC and SORTOUT data sets, you can gain the greatest performance improvement during large load jobs when there is a relatively even distribution of rows across partitions.

SORTOUT Data Sets

If a single partition is to receive 40% or more of the data, you should not use multiple SORTOUT data sets. When you do use multiple SORTOUT data sets, the number of SORTOUT data sets should be an even factor of the number of partitions. For example, a table space with 6 partitions should use either 2, 3, or 6 SORTOUT data sets. For additional information, see “Using Multiple SORTOUT Data Sets” on page 6-7.

SYSREC Data Sets

LOADPLUS allows up to 256 SYSREC data sets. LOADPLUS processes up to 16 SYSREC data sets concurrently when loading a partitioned table space.

The number of SYSREC data sets should be about the same as the number of SORTOUT data sets or should be an even multiple of the number of SORTOUT data sets. If the input data sets are of different sizes and there are more than 16, BMC Software recommends that you arrange them from the largest to the smallest.

These data sets are not partition dependent. Data for a particular partition can be in one or more SYSREC data sets. There is no advantage to putting all of the data for one partition into one SYSREC data set.

BMC Software, Inc., Confidential and Proprietary Information

Using Multiple SORTOUT Data Sets

LOADPLUS allows you to specify up to 16 SORTOUT data sets. However, the number of SORTOUT data sets that LOADPLUS processes concurrently might be smaller than the number that you specify.

LOADPLUS uses what it determines to be the optimal number of concurrent sort processes. The number of SORTOUT data sets that you specify limits the number of concurrent sort processes and can also affect how LOADPLUS determines the number of optimal sort processes. There is no advantage to specifying more data sets than the optimizer will select for use. See "Controlling the Number of Sort Processes" on page 6-15 for more information about how LOADPLUS determines the number of sort processes to use.

You can use either form of the data set name (SORTOUT or SORTOUT*n*). When specifying multiple data sets, you can use one or both forms. *n* is any valid single-digit national character or blank. If you plan to add more than one digit to the prefix or have LOADPLUS dynamically allocate more than nine SORTOUT data sets, you must override the default name to create a ddname prefix of six or fewer characters. To override the default ddname, use the LOADDN command option as described on page 3-37.

Using Multiple SORTWK Data Sets

To optimize performance, BMC Software recommends that you allocate 12 data sets, all of the same size.

Using Multiple SYSUT1 Data Sets

Using multiple SYSUT1 data sets reduces the amount of DASD that LOADPLUS requires for loading if the key lengths vary a great deal. Also, if enough sort work space and memory are available, using multiple SYSUT1 data sets allows LOADPLUS to build nonclustering indexes concurrently. For information about specifying multiple SYSUT1 data sets, see "Multiple Index Work Files (SYSUT1nn)" on page 6-27.

BMC Software, Inc., Confidential and Proprietary Information

Providing Maximum Virtual Storage

Because each task requires virtual storage for processing, LOADPLUS balances the multiprocessing that it performs with the amount of virtual storage that it finds available. The primary use of virtual storage is for I/O buffers by either LOADPLUS itself or by BMCSORT.

Because LOADPLUS uses as much virtual storage as needed, always specify as much virtual storage as you can. BMC Software recommends that you specify REGION=0M in the JOB or EXEC statement of the execution JCL to tell the system to allocate all available virtual storage to the LOADPLUS job. If your data center does not permit you to specify REGION=0M, specify the amount that allows the most virtual storage both above and below the 16-megabyte line.

Tuning I/O

Because LOADPLUS reads and writes large amounts of data during typical load processing, the single most important factor that affects LOADPLUS performance is I/O processing.

To maximize I/O performance, LOADPLUS handles all its own buffering, and performs I/O operations at the lowest level possible. This allows reading or writing of several blocks of data with each I/O operation and allows LOADPLUS to prefetch subsequent data.

In addition to the information in “Controlling Buffer Usage” on page 6-9 and “Controlling Sort Processing” on page 6-12, the following information can help you tune your I/O processing:

- Allocate input data sets with the largest block size possible to reduce the number of I/O operations that LOADPLUS requires to read the data set. LOADPLUS always calculates the optimum block size for output data sets.

- To avoid I/O queueing, allocate LOADPLUS data sets on separate channels and drives. If you do not have sufficient channels available, use separate drives and control units.

- Avoid DASD caching. Because LOADPLUS I/O processing is always sequential, DASD caching provides no benefit and the overhead can slightly increase I/O processing time.

BMC Software, Inc., Confidential and Proprietary Information

Controlling Buffer Usage

Installation options allow you to specify the number of buffers that the data sets use during LOADPLUS processing. The block size of these data sets, in combination with the number of buffers specified, determines the amount of virtual storage needed for the buffer pool. The default number of buffers can be set when LOADPLUS is installed, or you can modify the installation options later as you tune performance.

LOADPLUS allocates all VSAM buffers above the 16-megabyte line. LOADPLUS also allocates all sequential I/O buffers above the 16-megabyte line.

SYSREC Data Set

The SYSREC data set contains the input data that you are loading. LOADPLUS reads this data set at least once during the PRELOAD phase (if running a two-phase load) or during the COMBINED phase (if running a single-phase load) to build DB2 data rows for loading. LOADPLUS also reads the data set during discard processing if any records are discarded.

The IBUFFS installation option specifies the number of buffers that are allocated for reading each active SYSREC data set. BMC Software recommends a value of 20 buffers for this option.

SORTOUT Data Set

For a two-phase load job, the SORTOUT data set contains the DB2 row images for loading in the LOAD phase. LOADPLUS writes this data set in the PRELOAD phase and reads it in the LOAD phase.

For a single-phase LOAD RESUME YES SHRLEVEL NONE without PART REPLACE, but with ORDER YES specified, the SORTOUT data set contains the clustering index and the DB2 row images. LOADPLUS writes this data set in the COMBINED phase and reads it only if you restart your job.

If one or more of the tables that you are loading uses an EDITPROC routine for compression, the data is written in its compressed format to the SORTOUT data set. Using compressed data improves performance considerably because LOADPLUS requires fewer I/O operations to process the data.

BMC Software, Inc., Confidential and Proprietary Information

The LBUFFS installation option specifies the number of buffers that are allocated for reading and writing each active SORTOUT data set. BMC Software recommends a value of 20 buffers for this option. LOADPLUS determines the optimal block size of the SORTOUT data set, depending on the device type used. The block size of the SORTOUT data set, in turn, determines the buffer size.

See “SORTOUTn Data Sets” on page 4-12 for information about allocating SORTOUT data sets.

SYSUT1 Data Set

The SYSUT1 data set contains the information that LOADPLUS needs to build both the clustering and nonclustering indexes (when you specify ORDER NO) or only the nonclustering indexes (when you specify ORDER YES). For a two-phase load job, LOADPLUS writes to this data set during the PRELOAD phase and reads it during the LOAD phase. When used with single-phase LOAD RESUME YES SHRLEVEL NONE, LOADPLUS writes this data set in the COMBINED phase and reads it only if you restart your job.

When more than one nonclustering index is participating in the load, you can improve I/O performance by using multiple SYSUT1 data sets. Using multiple data sets reduces the number of data blocks that LOADPLUS writes and allows I/O operations to overlap. For more information about specifying multiple SYSUT1 data sets, see “Multiple Index Work Files (SYSUT1nn)” on page 6-27.

When you specify a single SYSUT1 data set for all indexes, the data set requires a record length that is long enough to hold information for the longest key. If shorter keys exist, LOADPLUS pads them so that they are as long as the longest key. With multiple SYSUT1 data sets, LOADPLUS writes information for each index to its own SYSUT1 data set, and does not pad the keys.

The WBUFFS installation option specifies the number of buffers that are allocated for reading and writing the SYSUT1 data set. The WBUFFS option has two parameter values. The first value specifies the number of buffers to use when using a single SYSUT1 data set. The second value specifies the number of buffers to use for each data set when using multiple SYSUT1 data sets. BMC Software recommends values of 20 buffers for a single SYSUT1 data set and 10 buffers for multiple SYSUT1 data sets.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS determines the optimal block size of the SYSUT1 data set, depending on the device type containing the data set. The block size of the SYSUT1 data set determines the buffer size.

Note: For a SQLAPPLY load or for a single-phase LOAD REPLACE job, LOADPLUS does not require a SYSUT1 data set and does not use it if you specify it.

Copy Data Sets

LOADPLUS writes the copy data sets (BMCCPY, BMCCPZ, BMCRCY and BMCRCZ) during the LOAD phase or the COMBINED phase. These data sets contain identical copies of the loaded table space or partitions of the table space after it is loaded. Registered copies are the DB2 local and remote copies that can provide input to a DB2 recover utility, such as RECOVER PLUS or the IBM RECOVER utility. (See page 4-9 for detailed information about the use of the copy data sets in LOADPLUS.)

The CBUFFS installation option specifies the number of buffers that are allocated for writing the copy data sets. BMC Software recommends a value of 30 buffers for this option. If multiple partitions for a copy of the table space are being copied to separate data sets, only one group of buffers is used.

LOADPLUS determines the optimal block size of the copy data sets based on the device type that contains the data set. Copy data sets can be on different device types. However, if additional copy data sets for the same object (for example, remote backup copy data sets) are on different device types, the block size for all copy data sets for that object is the block size that LOADPLUS determined was optimal for the local primary copy.

DB2 Data Sets

The DB2 data sets contain the loaded data and associated indexes in a format that is usable by DB2. LOADPLUS reads those data sets that contain the indexes during the PRELOAD or COMBINED phase if you specify the RESUME YES (with or without PART REPLACE) option on the LOAD command.

LOADPLUS assigns buffers for these data sets based on available virtual storage and the number of partitions that LOADPLUS is processing concurrently. Providing as much virtual storage as possible allows LOADPLUS to process several partitions and indexes concurrently. For more information, see “Providing Maximum Virtual Storage” on page 6-8.

BMC Software, Inc., Confidential and Proprietary Information

Controlling Sort Processing

The BMC Software BMCSORT technology provides LOADPLUS with more control of the sort process than external sort routines provide. This added control helps prevent memory-related problems during the sort process. LOADPLUS allocates the amount of resources to each sort process based on the amount of work that LOADPLUS determines that sort process will perform.

To enhance the performance of LOADPLUS and other applications running on your system, you can modify options that control the BMCSORT component. In addition, the SMAX or MAXSORTS option controls the number of sort processes. The following sections provide information that can help you improve the performance of your sort processing.

Invoking BMCSORT

Depending on the table that you are loading and its index characteristics, LOADPLUS invokes BMCSORT one or more times. The following sections describe the conditions under which LOADPLUS invokes BMCSORT during each processing phase.

PRELOAD Phase

LOADPLUS invokes BMCSORT during the PRELOAD phase under any of the following conditions:

- Clustering indexes exist and you specify the ORDER YES option on the LOADPLUS command.
- You specify UNIQUECHECK CLUSTER and unique clustering indexes exist.
- You specify UNIQUECHECK YES and unique indexes are participating in the load.
- The table space is segmented and contains multiple tables.
- LOADPLUS discards any records for duplicate key violations.

BMC Software, Inc., Confidential and Proprietary Information

LOAD Phase

LOADPLUS invokes BMCSORT during the LOAD phase under any of the following conditions:

- You specify UNIQUECHECK CLUSTER and unique nonclustering indexes are participating in the load.
- You specify UNIQUECHECK NO and unique indexes are participating in the load.
- The job builds nonunique nonclustering indexes.
- The job builds clustering indexes because you specified ORDER NO.
- You have discards.

COMBINED Phase of Single-Phase Load

LOADPLUS always invokes BMCSORT during the COMBINED phase of a single-phase load *except* when both of the following conditions occur:

- No records are discarded.
- No indexes are participating in the load, or you specify ORDER PRESORTED and no secondary indexes are participating in the load.

COMBINED Phase of SQLAPPLY Load

LOADPLUS invokes BMCSORT during the COMBINED phase of a SQLAPPLY load if you specify ORDER YES and you have discards.

Controlling Memory Usage

LOADPLUS provides the SMCORE installation option to give you control, when necessary, over the amount of memory that BMCSORT uses during a load job. This option contains two parameters, total memory and below-the-line memory, which the following sections describe.

BMC Software, Inc., Confidential and Proprietary Information

BMC Software strongly recommends that you use the values 0K and 0K for the SMCORE option. These values tell LOADPLUS to determine the appropriate amount of memory to use for each sort based on the following criteria:

- the value that you specify for REGION in either your JCL or system exits
- the amount of memory that is available during optimization
- the number of sorts to process

  For a discussion of how LOADPLUS determines the number of sorts to process, see “Controlling the Number of Sort Processes” on page 6-15.

You generally obtain the highest sort performance for your LOADPLUS job by using the values 0K and 0K for the SMCORE option.

Total Memory

The first parameter value of the SMCORE option tells LOADPLUS how much total memory, both above and below the 16-megabyte line, that you want BMCSORT to use during a single invocation. BMC Software strongly recommends that you specify a value of 0K, which allows LOADPLUS to determine the optimal amount of total memory to use. In addition to 0K, however, valid values are 4096K through 65536K. You can also specify this value in megabytes—0M or 4M through 64M.

Regardless of whether LOADPLUS determines the value for total memory or you specify a value, LOADPLUS multiplies this value by the number of required sort processes to determine a value for the total memory that the current job requires. Depending on the workload and system environment, LOADPLUS distributes this total memory among the sort processes for the job.

For example, if you specify 4096K and LOADPLUS determines that it needs 4 sort processes for this job, LOADPLUS calculates that it needs 16384 KB total memory for the job. However, if the workload for each sort process is different, LOADPLUS invokes BMCSORT for each sort process with varying amounts of memory. Some of these amounts will be lower and some of these amounts will be higher than the 4096 KB that you specified.

BMC Software, Inc., Confidential and Proprietary Information

Additional Considerations—The following additional considerations apply to the SMCORE option:

- The region size that is available for your load job in conjunction with the value that you specify for this parameter can constrain the number of sort processes that LOADPLUS starts. Because the region size must include space for buffers and other required structures, the entire region size is not available for sort processing. You can avoid this constraint by using values of 0K and 0K, allowing LOADPLUS to determine the optimal amount of total memory to use.

- When you allow LOADPLUS to optimize total memory, LOADPLUS never uses more than the value of your region parameter and never uses more than an average of 24 MB per sort process.

Below-the-Line Memory

The second parameter value of the SMCORE option tells LOADPLUS how much memory below the 16-megabyte line that you want BMCSORT to use during a single invocation. BMC Software recommends that you specify a value of 0K, which allows LOADPLUS to determine the optimal amount of below-the-line memory to use. In addition to 0K, valid values are 256K through 4096K. You can also specify this value in megabytes—0M or 1M through 4M.

BMCSORT never needs more than 256 KB of memory below the line. Specifying a value greater than this can limit the number of sort tasks that LOADPLUS can start concurrently.

Controlling the Number of Sort Processes

LOADPLUS determines the optimal number of sort processes to execute concurrently, depending on available resources. LOADPLUS calculates the optimal number of concurrent sort processes based on the following values:

- the number of SORTOUT data sets you specify
- the number of sort processes that will fit in available memory
- the number of partitions that are participating in the LOADPLUS job
- the value of the MAXP installation option
- the value of the SMAX installation option or MAXSORTS command option

For a two-phase load, you can specify the maximum number of concurrent sort processes of each type (index and data) with the SMAX installation option or the maximum number of concurrent index sort processes with the MAXSORTS command option.

BMC Software, Inc., Confidential and Proprietary Information

BMC Software recommends a value of 8 for the SMAX installation option. However, if system resources are constrained or other problems arise, you can change the SMAX installation option or MAXSORTS command option to limit the number of sort processes running concurrently.

Note: If you do not specify a value for SORTNUM on your LOAD command and only BMCSORT is allocating your sort work files, LOADPLUS uses a single sort process. For information about when BMCSORT dynamically allocates these data sets, see "Dynamically Allocating Sort Work Data Sets" on page 2-14.

Tuning Each of the LOADPLUS Processing Phases

LOADPLUS uses either a two-phase architecture or a single-phase architecture. In the two-phase architecture, the PRELOAD and LOAD phases perform all the functions that are included in the multiple phases performed during execution of the IBM DB2 LOAD utility, except the referential integrity functions that occur during the ENFORCE phase. In the single-phase architecture, the COMBINED phase performs in a single phase all the functions that the PRELOAD and LOAD phases perform, reducing the amount of read and write operations.

Performance is affected by the various LOADPLUS command options that you specify in the execution JCL, and by the values that were specified in the installation options module. The sections that follow present details about these options for each processing phase.

Tuning the ANALYZE Phase

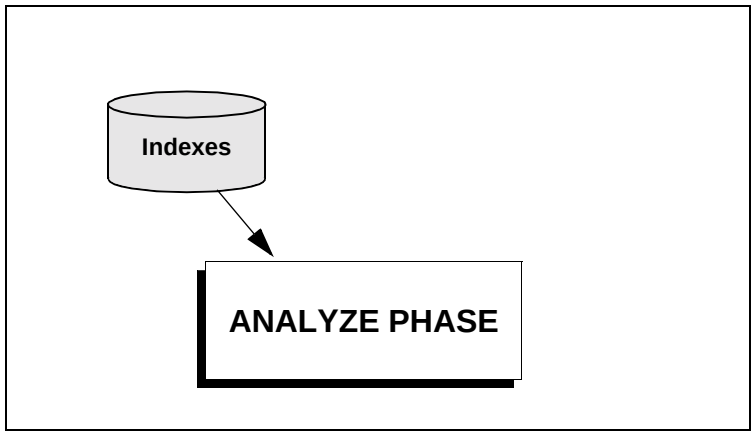
The ANALYZE phase provides information about the number of rows (cardinality) and the average row size of the data to be loaded. Two performance considerations arise from the ANALYZE phase and are controlled by the command options that you specify:

- the amount of time that the phase requires to run
- the accuracy of the information that LOADPLUS gathers during the phase

Figure 6-1 on page 6-17 shows the objects that the ANALYZE phase uses to determine the information that it provides.

BMC Software, Inc., Confidential and Proprietary Information

Figure 6-1 ANALYZE Phase



ANALYZE Options

The ANALYZE options limit abends that are caused by inadequate size allocations for data sets because they provide input to dynamic work file allocation processing or provide information that allows you to allocate space more accurately. LOADPLUS uses the results of the ANALYZE phase to verify work file sizes and optimize tasks.

ANALYZE

If you do not specify ANALYZE or if you specify ANALYZE without SAMPLE or SCAN, LOADPLUS determines whether to sample or scan for cardinality. LOADPLUS always obtains average row length information from the DB2 catalog.

ANALYZE SAMPLE and SCAN

If you specify ANALYZE SAMPLE, LOADPLUS samples the best index on each participating table to estimate the cardinality. If you specify ANALYZE SCAN, LOADPLUS scans the best index on each participating table to find the exact cardinality. For ANALYZE SAMPLE or SCAN, LOADPLUS always obtains the average row length information from the DB2 catalog.

Table 6-1 on page 6-18 shows how the SCAN and SAMPLE options affect the elapsed time of the reorganization and the accuracy of the data set sizings. Determine which option to select based on the needs of your organization.

BMC Software, Inc., Confidential and Proprietary Information

Table 6-1 Time vs. Accuracy for the ANALYZE Options

Option	Time Required	Accuracy
SCAN	more	more
SAMPLE	less	less

ANALYZE PAUSE and ONLY

The ANALYZE PAUSE and ANALYZE ONLY options provide estimates of the space that LOADPLUS needs for the following data sets:

- load (SORTOUT)
- work (SYSUT1)
- sort work (SORTWK)
- discard (SYSDISC),
- error (SYSERR)
- image copy (BMCCPY, BMCCPZ, BMCRCY, and BMCRCZ)

LOADPLUS writes these statistics to the SYSPRINT data set. You must specify PAUSE or ONLY to get this statistics report.

Messages from the ANALYZE Phase

When you specify ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS issues the following messages. LOADPLUS provides the data set size allocation information in a table format. As shown below, messages BMC51530I through BMC51532I provide the heading information. Multiple BMC51533I messages provide the estimated values. Because some types of load jobs allow you to specify single or multiple SORTOUT and SYSUT1 data sets, LOADPLUS provides values for both. Messages BMC50485I, BMC51535I, and BMC51536I provide estimates that you can use to gauge the elapsed time of the load.

For more information about using the report from the ANALYZE phase for allocating your data sets, see “ANALYZE Option for Estimating Data Set Allocation” on page 4-25. If you do not want to specify ANALYZE PAUSE or ANALYZE ONLY, you must calculate the data set sizes yourself or have LOADPLUS dynamically allocate your data sets. For detailed information and formulas you can use to calculate data set sizes, see Appendix D, “Calculating LOADPLUS Work Data Set Sizes.” For details about dynamic allocation, see “Dynamic Work File Allocation Options” on page 3-133.

BMC Software, Inc., Confidential and Proprietary Information

BMC51530I `LOADPLUS DASD REQUIREMENT ESTIMATES`

BMC51531I `3380 CYLS 3390 CYLS`

BMC51532I `DDNAME KBYTES PRI SEC PRI SEC INDEX`

BMC51533I LOADPLUS issues a separate BMC51533I message for each data set and provides the following information:

- data set name
- number of kilobytes
- primary and secondary 3380 cylinder quantities
- primary and secondary 3390 cylinder quantities
- index name, where applicable

BMC50485I This message provides either an estimate or the exact cardinality of each partition or table that is involved in the load. LOADPLUS issues this message for either an ANALYZE SAMPLE or an ANALYZE SCAN. You can use these values to gauge the elapsed time of the load.

BMC51535I This message provides an estimate of the cardinality and of the average row length of the table space. LOADPLUS issues this message when performing an ANALYZE SAMPLE. You can compare these values with estimates that other messages provide and with previous executions of the utility to validate the accuracy of these values and to gauge the elapsed time of the load.

BMC51536I This message provides the exact cardinality and an estimate of the average row length of the table space. LOADPLUS issues this message when it performs an ANALYZE SCAN. You can compare these values with estimates that other messages provide and with previous executions of the utility to validate the accuracy of these values and to gauge the elapsed time of the load.

Tuning the PRELOAD Phase

The PRELOAD phase of LOADPLUS prepares the input for loading into the specified tables by the LOAD phase. Depending on the data that you are loading and the characteristics of the table, the PRELOAD phase performs one or more of the following tasks:

- reads the input data from the SYSREC data set
- generates identity column values if needed
- verifies that the data is correct

BMC Software, Inc., Confidential and Proprietary Information

- sorts the data if needed
- sets up data to allow concurrent processing for loading table spaces and building indexes, and for concurrent loading of partitions
- optionally sorts input data by clustering key (when you specify ORDER YES)
- unloads existing indexes and merges them with the new index keys from the input data (when you specify RESUME YES and INDEX BUILD)

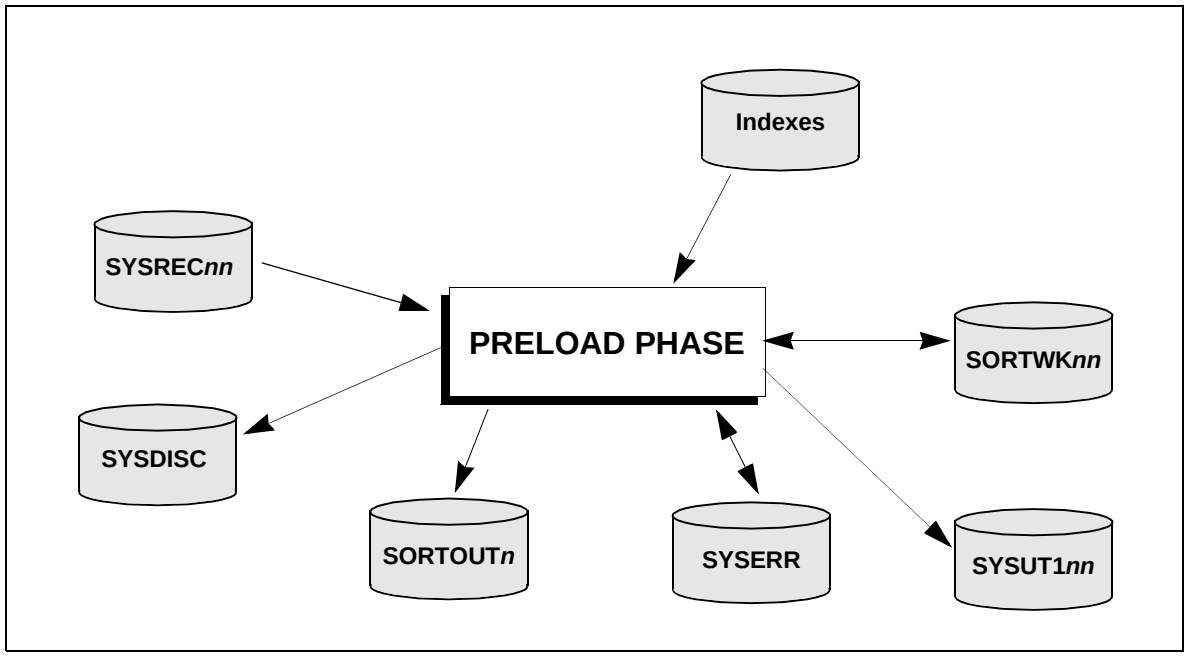
  LOADPLUS sorts merged index data to build organized indexes during the LOAD phase.
- produces a comprehensive discard report
- checks unique clustering keys for duplicates and discards them if you specify UNIQUECHECK YES or UNIQUECHECK CLUSTER
- checks unique nonclustering indexes for duplicates and discards them if you specify UNIQUECHECK YES
- builds or keeps the compression dictionary and compresses the data rows
- builds the DB2 data rows and writes them to the SORTOUT data set
- writes the index work records to the SYSUT1 data set
- at the end of the phase, frees the allocation for the SYSREC and SYSDISC data sets

Resource Allocation

Figure 6-2 on page 6-21 shows resources that are allocated during the PRELOAD phase of a two-phase load.

BMC Software, Inc., Confidential and Proprietary Information

Figure 6-2 Resource Allocation during the PRELOAD Phase



READ, SORT/DATA, and INDEX Tasks

During the PRELOAD phase, LOADPLUS performs the main tasks that Table 6-2 describes.

Table 6-2 PRELOAD Phase Tasks

Task	Description
READ	• reads input data • performs data conversion • if applicable, builds the clustering index key
SORT/DATA	• sorts data rows if you specify ORDER YES • creates secondary indexes, if any • writes data to the SORTOUT data set
INDEX	• builds index work records

LOADPLUS does not perform the INDEX task in the following cases:

- No unique indexes are participating in the load.
- You specify UNIQUECHECK NO.
- You specify UNIQUECHECK CLUSTER with ORDER YES.
- You specify INDEX UPDATE.
- Only the clustering index is unique and you specify ORDER YES or ORDER PRESORTED.

BMC Software, Inc., Confidential and Proprietary Information

PRELOAD ANALYZE Command Option

The PRELOAD ANALYZE command option directs LOADPLUS to stop after the optimization phase of PRELOAD. LOADPLUS displays a message that indicates the optimal number of concurrently processed SORT and READER tasks. You can use this information to adjust the number of SORTOUT data sets in your JCL.

RESUME and REPLACE Command Options

The RESUME and REPLACE command options control how LOADPLUS handles table spaces that contain existing data. Depending on the options that you choose, LOADPLUS might perform an initial analysis of the table space that you are loading, requiring additional processing overhead.

RESUME YES

If you specify the RESUME YES option, LOADPLUS must find the location where new data is to be loaded. LOADPLUS reads the table space, looking for the first available page after all existing data. Although this operation is not very time-consuming, it does require that LOADPLUS allocate, open, and perform some I/O operations on the table space data set.

LOADPLUS checks for active data by sequentially reading the table space and searching each page for active rows. If the table space is newly defined, this checking is very fast and requires only one I/O operation on the table space. However, if the table space contains many deleted rows or dropped tables, LOADPLUS must read each page, requiring many I/O operations and slowing processing.

The RESUME YES INDEX UPDATE option of the LOAD command tells LOADPLUS to update indexes rather than unload and reload them. LOADPLUS updates indexes during the LOAD or COMBINED phase. (See “INDEX” on page 3-15 for more information.) For best performance, when you are adding a small number of rows to an existing table that contains a large number of rows, use the INDEX UPDATE option instead of INDEX BUILD (which is the default).

RESUME NO

Specifying the RESUME NO option tells LOADPLUS to load new data at the beginning of the table space. Because this would destroy any existing data, LOADPLUS checks for existing data before performing the load operation. If LOADPLUS finds any active rows, the load operation terminates.

BMC Software, Inc., Confidential and Proprietary Information

REPLACE

The REPLACE option tells LOADPLUS to ignore any existing data in the table space, so LOADPLUS does not check for active rows. For best performance, when you refresh a table space, use the REPLACE option instead of deleting rows or dropping tables.

ORDER Command Option

The ORDER command option controls whether LOADPLUS sorts the input data in clustering key order. In determining how to optimize the performance of LOADPLUS, you must balance your need for fast execution and data availability with your need to have data sorted in clustering key order when it is loaded. The following discussion can help you determine how to specify the ORDER command option to meet your specific performance needs.

ORDER NO

ORDER NO tells LOADPLUS not to sort the data in clustering key order, thus decreasing processing time. However, if your tables have clustering indexes and the input data is not in clustering key order, you might need to reorganize the table space to obtain adequate DB2 performance.

ORDER YES

Specifying ORDER YES causes LOADPLUS to sort the input data during the data process of the PRELOAD phase or the COMBINED phase. Sorting at this point increases the elapsed time of the load process significantly. If you specify ORDER YES, LOADPLUS performs no other processing while the data is being sorted, because no index information can be passed to the index process until BMCSORT starts outputting ordered records. However, even though ORDER YES increases processing time, the total elapsed time that LOADPLUS requires to sort and load the data is still less than the time required to presort the data and then run LOADPLUS.

One advantage of specifying the ORDER YES option is that no clustering index work information is required in the SYSUT1 data set, thus reducing I/O operation and DASD space for that data set. Instead, LOADPLUS takes the clustering index information from the data rows during the LOAD phase.

BMC Software, Inc., Confidential and Proprietary Information

ORDER PRESORTED

ORDER PRESORTED tells LOADPLUS that the data is already in the same sequence as the clustering index. Therefore, LOADPLUS does not have to sort the data or the clustering index, saving a significant amount of time. LOADPLUS does verify that the data is in the correct order and terminates the job if it is not. In many cases, specifying LOAD REPLACE with ORDER PRESORTED is the fastest way to run a load.

Note: You must have only one SYSREC specified to use ORDER PRESORTED. If you have multiple SYSREC data sets, the job terminates.

ORDER Options With RESUME YES SHRLEVEL NONE

If you specify (or default to) LOAD RESUME YES SHRLEVEL NONE, LOADPLUS stops the table space and associated indexes at the beginning of the PRELOAD phase or COMBINED phase. If you must order your data, you need to maximize data availability, and you can afford to lengthen the overall elapsed time to complete the job, presort the data prior to running LOADPLUS. Specify ORDER NO if you have already sorted the data. When you specify ORDER NO, LOADPLUS processes clustering indexes like participating nonclustering indexes, avoiding the overhead of sorting the data.

RULES Installation Option

Because LOADPLUS does not convert data to internal format when the installation option RULES=STANDARD is in effect, WHEN, DEFAULTIF, and NULLIF comparisons that LOADPLUS performs under RULES=STANDARD always outperform those that LOADPLUS performs under RULES=BMC.

Note: When you specify FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD, LOADPLUS changes the value of the RULES option to BMC, regardless of the value that you specified at installation.

If you specify FORMAT CSV, LOADPLUS changes the value of the RULES option to STANDARD, regardless of the value that you specified at installation.

BMC Software, Inc., Confidential and Proprietary Information

When RULES=BMC and you specify a field name in your predicate where the field name matches a column name in the table that you are loading, LOADPLUS converts the value that the field defines to the data type and length of the column *before* it evaluates the predicate. (This is different than the IBM DB2 LOAD utility which only performs a byte-by-byte comparison based on the byte length of the field.) Depending on the conversion that is involved, converting the value can be time consuming and is always more so than a predicate that uses (*start*:*end*) or a field without a corresponding column. Therefore, to achieve the best performance, specify your predicates by using (*start*:*end*) or a field name that does not correspond to a column.

DISCARDS IGNORE Command Option

The DISCARDS IGNORE command option allows you to specify the discard types that you want to ignore. Except for duplicate key discards, LOADPLUS does not write ignored discards to the SYSERR file, does not report them in the SYSPRINT error summary report, and does not discard them to the discard (SYSDISC) data set. However, if you specify MSGLEVEL(1), LOADPLUS still produces other diagnostic messages associated with discards for debugging purposes.

Using the DISCARDS IGNORE option to ignore WHEN and PART discards can improve performance significantly when your SYSREC data set contains input records for more than one table or partition but you are not loading all of the tables or partitions in one load job.

INTO and UNIQUEINTO Command Options

The INTO command option allows you to select which input records to load into a specific table or partition. UNIQUEINTO YES tells LOADPLUS that as soon as an INTO specification selects an input record, LOADPLUS should not test the record against other INTO specifications.

Normally, LOADPLUS checks each input record to determine if it matches the WHEN or PART criteria of each INTO specification. Checking starts with the first INTO statement and ends with the last INTO statement. This can be extremely time consuming when you specify many INTO statements.

BMC Software, Inc., Confidential and Proprietary Information

INTO and PART

Use the following guidelines to determine how many INTO statements to use with your job and when to use the PART option:

- Under either of the following circumstances, specify only one INTO statement without the PART option. The resulting clustering key determines the partition into which the record is loaded, so the PART option is not necessary.

  — if you specify LOAD REPLACE
  — if you specify LOAD RESUME YES and are loading the majority of partitions in a partitioned table space

- If you specify LOAD RESUME YES and are loading only a few partitions, specify all partitions on one INTO statement (for example, INTO PART 1, 2, 3). This specification eliminates unloading and loading all partitions of the clustering index and can save considerable time.

- Avoid specifying a separate INTO statement with PART option for each partition.

UNIQUEINTO YES

Use UNIQUEINTO YES if both of the following conditions are true:

- You are loading multiple tables of a nonpartitioned table space or multiple partitions of a partitioned table space.

- Each input record matches at most one INTO specification.

As soon as an input record matches the WHEN selection criteria of an INTO statement, LOADPLUS does not check any more INTO specifications. For each input record, this reduces the number of INTO specifications that LOADPLUS checks to half of the INTO specifications made, on average, which can improve performance significantly. You can also improve performance by specifying the INTO statement that selects the most input records, followed by the INTO statement that selects the second most input records, and so on.

BMC Software, Inc., Confidential and Proprietary Information

Multiple Input Data Sets (SYSREC*nn*)

When loading partitioned table spaces, using multiple SYSREC data sets allows for greater concurrency, thus reducing elapsed time. These data sets are not partition dependent. Data for a particular partition can be in one or more SYSREC data sets. There is no performance advantage to specifying one SYSREC data set per partition.

Multiple Sort Data Sets (SORTOUT*n*)

Specifying multiple SORTOUT data sets might increase the number of concurrent sort processes, reducing elapsed time of the PRELOAD and LOAD phases. LOADPLUS uses the number of SORTOUT data sets that you specify as one of the criteria to determine the optimal number of concurrent sort processes. The number of sort processes that LOADPLUS selects and performs, however, can be smaller than the number that you specify and there is no advantage to specifying more data sets than the optimizer selects. For details, see the multitasking discussion on page 6-7.

Note: Specifying PRELOAD ANALYZE directs LOADPLUS to stop after optimization and display a message that provides the optimal number of concurrent SORT and READER tasks. You can use this information to adjust the number of SORTOUT data sets in your JCL.

Multiple Index Work Files (SYSUT1*nn*)

Using multiple index work files is one way that you can shorten the elapsed time for the PRELOAD and LOAD phases of LOADPLUS. Specifying a separate SYSUT1 data set for each participating nonclustering index (or for each participating index if you specify ORDER NO) provides several advantages:

- I/O processing to each SYSUT1 data set is overlapped with other I/O processing and with CPU processing.
- LOADPLUS writes any nonunique index information when the index process receives it and does not pass the information to BMCSORT, thus reducing the amount of data that is sorted. This process also reduces the amount of DASD space that is required for all index information.
- With a single SYSUT1 data set, LOADPLUS pads all keys to the length of the longest key that is being processed. Padded keys require more DASD space and more I/O operations are required to process the index information. With multiple SYSUT1 data sets, no padding of keys is needed.

BMC Software, Inc., Confidential and Proprietary Information

- If LOADPLUS finds that enough resources (that is, virtual storage and sort work space) are available, LOADPLUS starts multiple tasks to build the indexes concurrently in the LOAD phase.

If you are using multiple index work files, you must specify at least one SYSUT1 data set for each participating nonclustering index. If the number of SYSUT1 DD statements is greater than one and less than the number of participating nonclustering indexes, LOADPLUS terminates with an error. Also, the ORDER NO option processes clustering indexes as if they were participating nonclustering indexes. If you specify ORDER NO, you must specify one SYSUT1 data set for each clustering and participating nonclustering index.

When you specify multiple SYSUT1 data sets, the DD statement specification is SYSUT1*nn*, where *nn* is a unique suffix for each DD statement. The suffix is not used to identify which index is assigned to a specific SYSUT1 data set.

The JCL in Figure 6-3 illustrates a LOADPLUS job for a table space that contains three nonclustering indexes and one clustering index. In this example, you need only three SYSUT1 data sets, because the ORDER YES option is specified. Note the unique identifier that is used with the SYSUT1 prefix.

Figure 6-3 Multiple SYSUT1 Data Sets With ORDER YES Option

```
//LOADPLUS  JOB (1234),'YOUR NAME',REGION=0M,...
//EXECLP   EXEC PGM=AMUUMAIN,PARM='DB2,LOADUTIL'
//STEPLIB   DD  DSN=LOAD.PLUS.LOADLIB,DISP=SHR
//SYSREC    DD  DSN=LOAD.PLUS.INPUT,DISP=OLD
//SORTOUT   DD  DSN=LOAD.PLUS.SORTOUT,DISP=(MOD,CATLG),...
//****  Specify a SYSUT1 data set for each nonclustering index.
//SYSUT101  DD  DSN=LOAD.PLUS.SYSUT101,DISP=(MOD,CATLG),...
//SYSUT102  DD  DSN=LOAD.PLUS.SYSUT102,DISP=(MOD,CATLG),...
//SYSUT103  DD  DSN=LOAD.PLUS.SYSUT103,DISP=(MOD,CATLG),...
//SYSERR    DD  DSN=LOAD.PLUS.SYSERR,DISP=(MOD,CATLG),...
//SYSDISC   DD  DSN=LOAD.PLUS.SYSDISC,DISP=(NEW,CATLG),...
//SORTWK01  DD  DISP=(NEW,DELETE),...
//SORTWK02  DD  DISP=(NEW,DELETE),...
//SORTWK03  DD  DISP=(NEW,DELETE),...
   . . .
//SORTWK12  DD  DISP=(NEW,DELETE),...
//SYSPRINT  DD  SYSOUT=*
//SYSIN     DD  *
   LOAD ..... ORDER YES
//
```

BMC Software, Inc., Confidential and Proprietary Information

The JCL in Figure 6-4 shows the same job, but with the ORDER NO option specified. Note that you now need four SYSUT1 data sets, because LOADPLUS processes the clustering index as if it were a participating nonclustering index.

Figure 6-4 Multiple SYSUT1 Data Sets With ORDER NO Option

```
//LOADPLUS  JOB (1234),'YOUR NAME',REGION=0M,...
//EXECLP   EXEC PGM=AMUUMAIN,PARM='DB2,LOADUTIL'
//STEPLIB   DD  DSN=LOAD.PLUS.LOADLIB,DISP=SHR
//SYSREC    DD  DSN=LOAD.PLUS.INPUT,DISP=OLD
//SORTOUT   DD  DSN=LOAD.PLUS.SORTOUT,DISP=(MOD,CATLG),...
//****  Specify a SYSUT1 data set for each nonclustering index
//****  and the clustering index.
//SYSUT101  DD  DSN=LOAD.PLUS.SYSUT101,DISP=(MOD,CATLG),...
//SYSUT102  DD  DSN=LOAD.PLUS.SYSUT102,DISP=(MOD,CATLG),...
//SYSUT103  DD  DSN=LOAD.PLUS.SYSUT103,DISP=(MOD,CATLG),...
//SYSUT104  DD  DSN=LOAD.PLUS.SYSUT104,DISP=(MOD,CATLG),...
//SYSERR    DD  DSN=LOAD.PLUS.SYSERR,DISP=(MOD,CATLG),...
//SYSDISC   DD  DSN=LOAD.PLUS.SYSDISC,DISP=(NEW,CATLG),...
//SORTWK01  DD  DISP=(NEW,DELETE),...
//SORTWK02  DD  DISP=(NEW,DELETE),...
//SORTWK03  DD  DISP=(NEW,DELETE),...
   . . .
//SORTWK12  DD  DISP=(NEW,DELETE),...
//SYSPRINT  DD  SYSOUT=*
//SYSIN     DD  *
   LOAD ..... ORDER NO
//
```

Transfer Blocks (XBLKs)

XBLKS perform the following functions in the PRELOAD phase or the COMBINED phase of LOADPLUS:

- moving data from the READ task to the SORT/DATA task
- moving nonclustering index records from the SORT/DATA task to the INDEX task

The READ task reads data from SYSREC, completes all conditional processing, and moves the row image into an XBLK. If there are multiple SORT/DATA tasks, an XBLK is associated with each SORT/DATA task. Once an XBLK is full, it becomes available for processing by the SORT/DATA task.

The SORT/DATA process passes information to the INDEX task by using one or more XBLKs. The INDEX task is needed only when an additional sort is necessary to check for duplicate keys.

BMC Software, Inc., Confidential and Proprietary Information

When the row image has been processed by the SORT/DATA task and an INDEX task is active, the INDEX task builds an index work record for each index in the selected table. LOADPLUS then places this index work record in the currently available XBLK. When the XBLK becomes full, the index work records are passed to the index process for processing and the data process selects another available XBLK to fill. When the index process has finished processing all index work records in the XBLK, the XBLK becomes available again for use by the data process.

The number of XBLKS that LOADPLUS allocates is a function of the number of READ, SORT/DATA, and INDEX tasks that are active for a particular LOADPLUS job. The XBLK installation option specifies how many XBLKs to allocate per task. The minimum value for the XBLK option is 3. A larger number can provide some marginal performance improvement, but at a risk of increased paging. Each XBLK is 64 KB and is allocated from storage above the 16-megabyte line.

Tuning the LOAD Phase

The LOAD phase of LOADPLUS is basically an I/O driver set up by the PRELOAD phase. The LOAD phase performs the following basic functions:

- loads data into the specified tables and creates image copies if requested
- builds all associated participating indexes

The LOAD phase of LOADPLUS performs the following tasks:

- redefines the VSAM data sets (including the staging data sets for LOAD REPLACE SHRLEVEL CHANGE or LOAD REPLACE SHRLEVEL REFERENCE) when the value of the REDEFINE command or installation option is YES
- concurrently loads multiple partitions
- concurrently loads the table space and clustering indexes
- concurrently loads the table space and sorts and builds nonclustering indexes
- collects statistics while loading table spaces and building indexes
- produces image copies either as the data is loaded or after the data is loaded

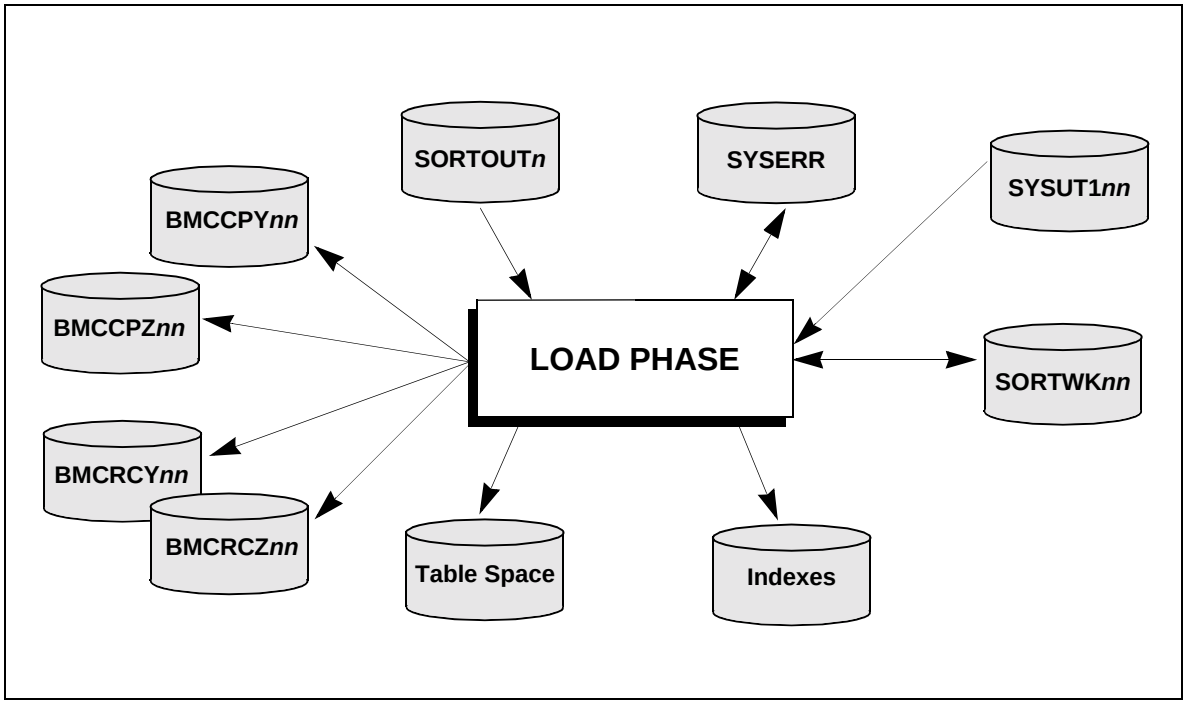
BMC Software, Inc., Confidential and Proprietary Information

- detects duplicates and writes error records out to SYSERR
- after loading table spaces and building indexes, resolves any duplicates that are detected during the LOAD phase

Resource Allocation

Figure 6-5 shows resources that are allocated during the LOAD phase of a two-phase load.

Figure 6-5 Resource Allocation during the LOAD Phase



Load Process and Index-Building Process

The LOAD phase is made up of one or more *load processes* and one or more *index-building processes*. LOADPLUS performs these processes concurrently. The execution strategy of the LOAD phase depends on the characteristics of the table space that you are loading and the options used during the PRELOAD phase.

The load process builds DB2 data pages by using the DB2 row information that was placed in the SORTOUT data set. After LOADPLUS builds an entire page, LOADPLUS writes the page to the table space data set. You can achieve parallel processing in either of the following two instances:

- if the table space has one or more clustering indexes defined
- if the table space is partitioned and your JCL specifies multiple SORTOUT data sets

If clustering indexes exist and you specify the ORDER YES option, the load process builds the clustering indexes while it is loading the data. The clustering index key values are taken from the rows as they are loaded into the table space. Index pages are built and written as they fill up. This process provides more overlapped I/O processing. As an example, a partitioned table space that is loaded with ORDER YES and a MAXP value of 5 will actually have 10 data sets open concurrently.

One or more index-building processes build the nonclustering indexes. If you specify ORDER NO, one or more index-building processes also build clustering indexes. If LOADPLUS uses only one SYSUT1 data set during the PRELOAD phase, LOADPLUS starts only one process to build all indexes. In this case, LOADPLUS sorts all of the index work records and then uses these work records to build indexes one at a time.

If LOADPLUS uses multiple SYSUT1 data sets during the PRELOAD phase, LOADPLUS starts multiple index-building processes. LOADPLUS first determines the amount of free virtual storage and the number of sort work data sets to determine an optimal number of tasks to start. LOADPLUS then tries to keep this number of tasks active until all indexes are built.

MAXP Installation Option

The MAXP installation option limits the number of partitions that LOADPLUS accesses concurrently. This option permits LOADPLUS to balance parallel processing and use of virtual storage. If the MAXP value is too small, many I/O waits can slow load processing. If the value is too large, excessive paging activity can slow load processing.

BMC Software, Inc., Confidential and Proprietary Information

Tuning the COMBINED Phase for a Single-Phase Load

Note: This book distinguishes between the COMBINED phase for a single-phase load (in which you specify PRELOAD LOAD) and the COMBINED phase for a SQLAPPLY load. Discussion of the COMBINED phase for the latter type of load starts on page 6-36.

During the COMBINED phase for a single-phase load, LOADPLUS performs almost all of the same processes as the PRELOAD and LOAD phases. Depending on the data that you are loading and the characteristics of the table, LOADPLUS performs one or more of the following tasks during the COMBINED phase:

- reads the input data from the SYSREC data set
- generates identity column values if needed
- verifies that the data is correct
- for certain types of single-phase jobs, writes index key entries to SYSUT1 and clustering key entries and the new DB2 row images to SORTOUT, if the data sets are allocated
- redefines the VSAM data sets (including the staging data sets for LOAD REPLACE SHRLEVEL CHANGE or LOAD REPLACE SHRLEVEL REFERENCE) when the value of the REDEFINE command or installation option is YES
- sets up data to allow concurrent processing for loading table spaces and building indexes, and for concurrent loading of partitions
- optionally sorts input data by clustering key (when you specify ORDER YES)
- unloads existing indexes and merges them with the new index keys from the input data (when you specify RESUME YES and INDEX BUILD)
- sorts merged indexes and writes them directly to the index data set
- produces a comprehensive discard report
- checks unique clustering keys for duplicates and deletes them from the table and index space
- checks participating unique nonclustering indexes for duplicates and deletes the duplicates from the table and index space
- builds or keeps the compression dictionary and compresses the data rows

BMC Software, Inc., Confidential and Proprietary Information

- builds the DB2 data rows and writes them directly to the table space
- concurrently loads multiple partitions
- concurrently loads the table space and clustering indexes
- collects statistics while loading table spaces and building indexes
- produces image copies either as the data loads or after the data is loaded
- at the end of the phase, frees the allocation for the SYSREC and SYSDISC data sets

Table 6-3 describes the main tasks that LOADPLUS performs during the COMBINED phase of a single-phase load.

Table 6-3 COMBINED Phase Tasks for Single-Phase Load

Task	Description
READ	• reads input data • performs data conversion • if applicable, builds the clustering index key
SORT/DATA	• sorts data rows when you specify ORDER YES • if applicable, writes clustering index records to the index space when you specify ORDER YES • builds secondary index keys, if any, and passes them to the index task • writes data to the table spaces
INDEX	• writes index records to the index space

LOADPLUS does not perform the INDEX task in the following cases:

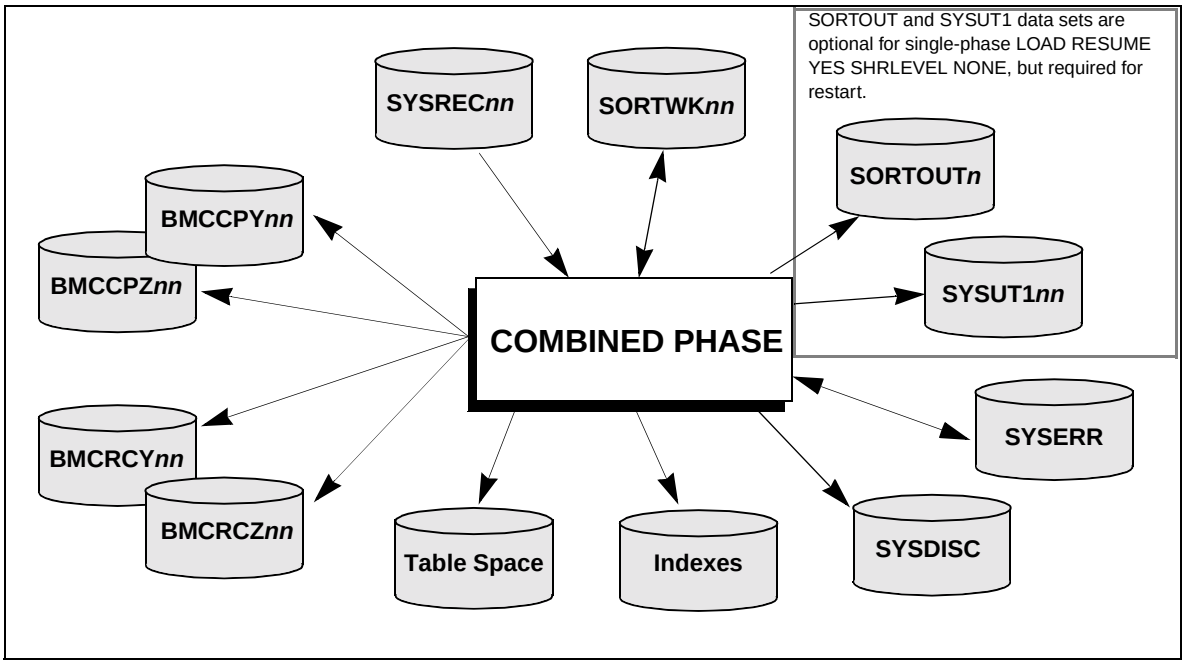
- No indexes are participating in the load.
- Only a clustering index exists and you specify ORDER YES or ORDER PRESORTED.

Resource Allocation

Figure 6-6 on page 6-35 shows resources that are allocated during the COMBINED phase of a single-phase load.

BMC Software, Inc., Confidential and Proprietary Information

Figure 6-6 Resource Allocation during the COMBINED Phase for Single-Phase Load



Single-Phase Load Performance Considerations

Because most single-phase load jobs do not use the SORTOUT and SYSUT1 data sets, these types of load jobs provide a significant resource savings when processing large amounts of data, including the elimination of all EXCP processing associated with these data sets. In environments that require the use of tape for the SORTOUT and SYSUT1 data sets, this type of load job can also provide significant CPU and elapsed time savings. Tape mount time becomes nonexistent and CPU consumption can be reduced because there is no read/write activity to these data sets.

Note: For certain types of single-phase jobs, you cannot restart your load job if you do not specify SORTOUT and SYSUT1 data sets. See “Single-Phase Load” on page 4-29 for information about restarting single-phase load jobs.

BMC Software, Inc., Confidential and Proprietary Information

The fastest load is a single-phase load with LOAD REPLACE ORDER PRESORTED and no participating nonclustering indexes. You can specify ORDER PRESORTED on a single-phase load with RESUME YES PART REPLACE if all of the following conditions apply to your load:

- You are replacing all partitions that are involved in the load.
- No nonclustering indexes are participating in the load.
- You use only one SYSREC data set.

Note: Two-phase load jobs can be faster if you expect to discard a significant number of duplicate keys. In the COMBINED phase, LOADPLUS loads the data into the table before checking for duplicate keys in the unique nonclustering indexes. At the end of the COMBINED phase, LOADPLUS deletes the rows causing the duplicates and the duplicate key entries for *all* indexes for those rows.

Keep in mind that the COMBINED phase of LOADPLUS is a combination of the PRELOAD and LOAD phases. If you specify SHRLEVEL NONE, the table space that you are loading is stopped at the beginning of the phase, which means that the amount of time that the objects are unavailable might be longer with a single-phase load.

The performance considerations for the COMBINED phase are the same as those given for the PRELOAD phase beginning on page 6-19 and the LOAD phase beginning on page 6-30.

Tuning the COMBINED Phase for SQLAPPLY

During the COMBINED phase for a SQLAPPLY load, LOADPLUS performs most of the same processes as in the PRELOAD and LOAD phases. Depending on the data loaded and the characteristics of the table, LOADPLUS performs one or more of the following tasks during the COMBINED phase:

- reads the input data from the SYSREC data set
- verifies that the data is correct, but does not perform checking for duplicates or table check constraints

  Note: DB2 handles checking for duplicates and table check constraints during apply processing.

- sets up data to allow concurrent processing for loading tables and partitions

BMC Software, Inc., Confidential and Proprietary Information

- optionally sorts input data:
  - — by clustering key if you are loading a single table and you specify ORDER YES
  - — by table and clustering key if you are loading multiple tables and you specify ORDER YES
  - — only by table if there are no clustering keys, you are loading multiple tables, and you specify ORDER YES
- passes the data to Apply Plus, for inserting into the table space and indexes
- produces a comprehensive discard report
- at the end of the phase, frees the allocation for the SYSREC and SYSDISC data sets

Table 6-4 shows the main tasks that LOADPLUS performs during the COMBINED phase of a SQLAPPLY load.

Table 6-4 COMBINED Phase Tasks for SQLAPPLY

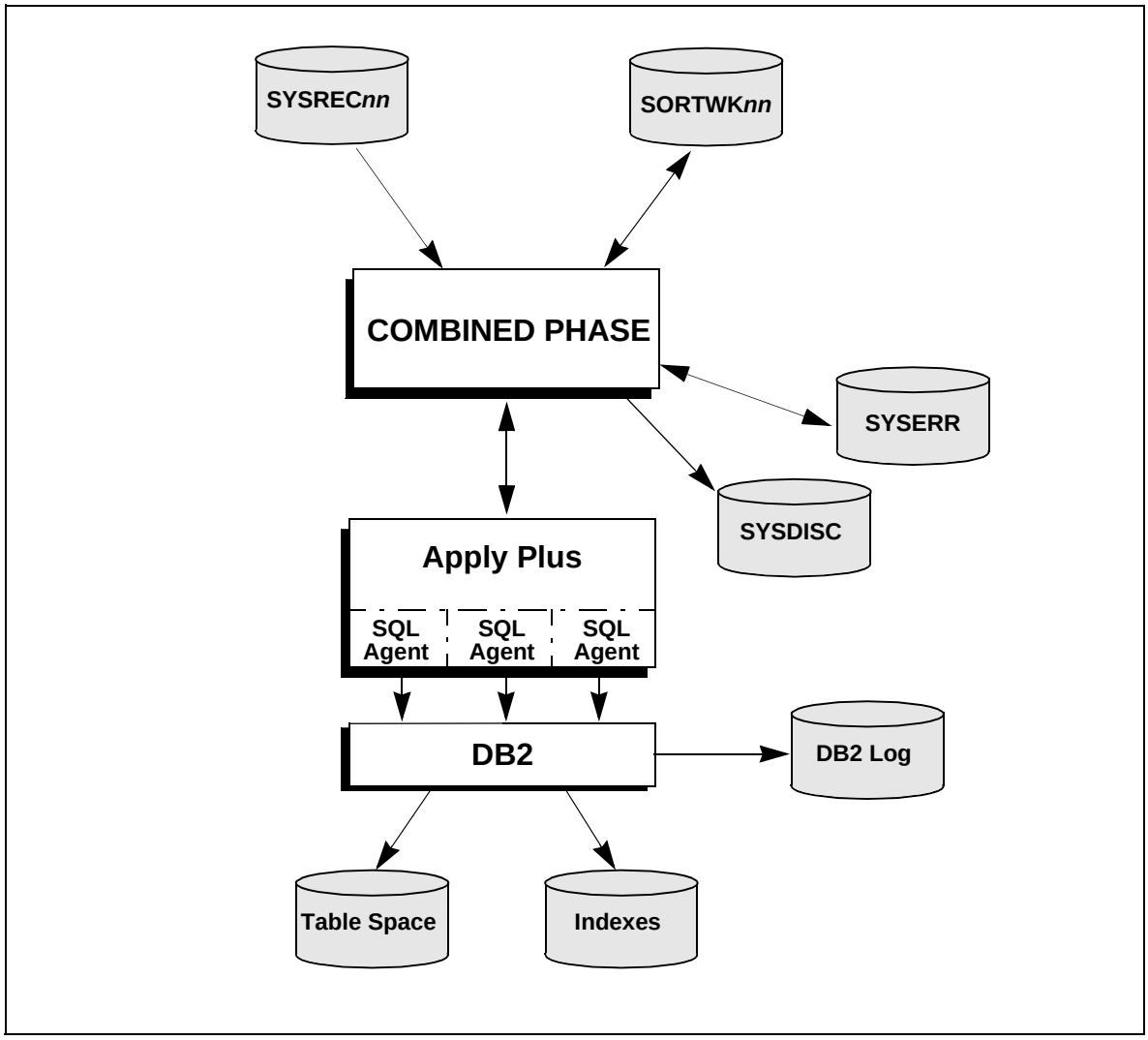
Task	Description
READ	• reads input data • performs data conversion
SORT	• sorts data rows if you specify ORDER YES
SQLAPPLY	• passes data to Apply Plus for loading

Resource Allocation

Figure 6-7 on page 6-38 shows resources that are allocated during the COMBINED phase for a SQLAPPLY load.

BMC Software, Inc., Confidential and Proprietary Information

Figure 6-7 Resource Allocation during the COMBINED Phase for SQLAPPLY



SQLAPPLY Performance Considerations

Although a SQLAPPLY load job uses the single-phase architecture, many of the performance considerations for a single-phase load do not apply. The following discussions provide information about how you can affect the performance of this type of load.

INTO and UNIQUEINTO Command Options

The INTO command option allows you to select which input records to load into a specific table or partition. UNIQUEINTO YES tells LOADPLUS that as soon as an INTO specification selects an input record, LOADPLUS should not test the record against other INTO specifications.

BMC Software, Inc., Confidential and Proprietary Information

Normally, LOADPLUS checks each input record to determine if the record matches the WHEN criteria of each INTO specification. Checking starts with the first INTO statement and ends with the last INTO statement. This process can be extremely time consuming if you have numerous INTO specifications.

For partitioned table spaces, LOADPLUS improves performance by automatically condensing multiple INTO table blocks into a single INTO statement if the field specifications for each INTO block are identical.

Use UNIQUEINTO YES if both of the following conditions are true:

- You are loading multiple tables of a nonpartitioned table space or multiple partitions of a partitioned table space.
- Each input record matches at most one INTO specification.

As soon as an input record matches the WHEN selection criteria of an INTO statement, LOADPLUS does not check any more INTO specifications. For each input record, this reduces the number of INTO specifications that LOADPLUS checks to half of the INTO specifications made, on average, which can improve performance significantly. You can also improve performance by specifying the INTO statement that selects the most input records, followed by the INTO statement that selects the second most input records, and so on.

Identity Columns

For SQLAPPLY load jobs, if you are loading an identity column, LOADPLUS does not generate values but allows DB2 to assign values during insert processing. Loading an identity column in a SQLAPPLY load job can reduce the performance of your load job.

Interpreting Performance-Related Messages

LOADPLUS issues performance-related messages if you specify MSGLEVEL(1) on your EXEC statement or in your installation options. See “Utility Parameters on the EXEC Statement” on page 4-3 for more details on specifying this utility parameter. Information that these messages provide can help you monitor the performance of LOADPLUS as well as fine-tune future runs.

BMC50398I This message indicates that the load job is constrained because the number of sort work files or the total amount of space for the sort work files is insufficient, but the job continues.

BMC Software, Inc., Confidential and Proprietary Information

BMC50400I This message indicates that the load job is constrained because there are not enough index work files, but the job continues.

BMC50471I This message displays current values for each option in the installation options module. Use this information to verify that LOADPLUS is using the option value that you want to use.

BMC50474I This message provides information about virtual storage and CPU usage. The message displays the following information:

- amount of virtual storage available below the 16-megabyte line
- amount of virtual storage available above the 16-megabyte line
- number of physical CPUs available in the processor

Use this information to ensure that adequate virtual storage is available for LOADPLUS to use. See “Providing Maximum Virtual Storage” on page 6-8. See message BMC50475I for additional information.

BMC50476I This message provides performance information about I/O operations to sequential data sets. The message displays the following information:

- ddname associated with the I/O operations
- number of I/Os (blocks) written to or read from the data set
- number of waits issued for I/O completion
- number of waits for serialization of the data set

A wait count that is greater than 10 percent of the block count might indicate degraded performance. Try allowing LOADPLUS more buffers for the associated data set (see “Controlling Buffer Usage” on page 6-9) or using multiple data sets. A high number of serialization waits might indicate the need to decrease the number of concurrent tasks or use multiple data sets.

BMC50477I This message provides performance information about I/O operations to VSAM data sets. The message displays the following information:

- processing task number
- table space partition number
- number of rows or keys in the partition
- number of waits issued for I/O completion
- ddname associated with the I/O operations

Use the ddname to find the actual number of I/Os that were issued to the associated data set. A wait count that is greater than 20 percent of the actual I/Os might indicate degraded performance. Try allowing LOADPLUS more virtual storage if LOADPLUS issues message BMC50397I. Provide more sort work file space if LOADPLUS issues message BMC50398I. Changing these values enables LOADPLUS to start more concurrent tasks.

BMC Software, Inc., Confidential and Proprietary Information

BMC51495I This message provides performance information about optimizing the number of sort processes during the PRELOAD phase or COMBINED phase. Each iteration of the optimizer produces a message and supplies the following information:

- return code
- number of sort processes that are being tested for this iteration of the optimizer
- number of input data sets (SYSRECs) that LOADPLUS will read concurrently if this is the optimal number of sort processes
- the number of index tasks that the load job will run concurrently
- *relative* amount of time that this configuration of read and sort processes will take to process the data

LOADPLUS uses the return code to monitor the success of the optimization. If at least one return code in the series of messages is 0, the PRELOAD optimization completed successfully. When optimization completes successfully, LOADPLUS follows this message with BMC51496I, which provides the optimal number of sort processes.

If no return code is 0, LOADPLUS follows this message with BMC51497E, indicating that optimization failed. You might be able to correct the problem by increasing the REGION parameter or by specifying REGION=0M and resubmitting the job.

For more information about how the optimizer determines the optimal number of sort processes, see “Controlling the Number of Sort Processes” on page 6-15.

BMC51507I This message provides PRELOAD phase or COMBINED phase performance information about data transfer between these tasks:

- the READ task and the SORT/DATA task
- the SORT/DATA task and the INDEX or SQLAPPLY task

This message provides the following information:

- number of XBLKs or Apply Plus buffers available for data transfer
- number of waits for an empty XBLK or Apply Plus buffer
- number of waits for a full XBLK

This message summarizes the information that BMC51510I provides.

BMC Software, Inc., Confidential and Proprietary Information

BMC51508I This message provides the results of task optimization during build processing. The message displays the following information:

- maximum number of concurrent tasks
- maximum number of partitions or indexes that LOADPLUS can process per task
- number of sort work files that are assigned to each task
- maximum number of open partitions per task

The maximum number of tasks depends on the system resources that are available, such as virtual storage, the number of physical CPUs, the number of SYSUT1 and SORTOUT data sets, and sort work file space. Increasing the amount of virtual storage, allocating more sort work files, or both might allow LOADPLUS to run more tasks concurrently.

Based on the size of the partition and available sort work file space, LOADPLUS determines the number of partitions to process per task that allows the sort to run in the fastest manner. The actual number of partitions that LOADPLUS processes per task can be less than the maximum that the message shows.

BMC51510I This message provides information about XBLK or Apply Plus buffer processing for a specific task. The message displays the following information:

- task number
- task type (READ, SORT, INDEX, SQLAPPLY)
- number of data transfers during task processing
- number of waits for an empty XBLK or Apply Plus buffer
- number of waits for a full XBLK

Have the XBLK or Apply Plus buffer information handy if you need to call BMC Software Customer Support.

However, for XBLKs, you can use the information that this message provides to refine the number of READ and SORT/DATA tasks that LOADPLUS will use.

Background

A READ task usually runs faster than a SORT/DATA task when a sort process is involved (ORDER YES). An INDEX task generally runs faster than a SORT/DATA task. When the job does not require a sort of the data (ORDER NO), the SORT/DATA task can be faster than the READ task.

BMC Software, Inc., Confidential and Proprietary Information

There are two pools of XBLKs. LOADPLUS uses one pool for transfers from the READ to the SORT/DATA tasks. LOADPLUS uses the other pool for transfers from the SORT/DATA tasks to the INDEX or SQLAPPLY tasks.

Steps That You Can Take

The situations that Table 6-5 describes indicate a need to refine the number of READ and SORT/DATA tasks when loading multiple partitions.

Table 6-5 Refining READ and SORT/DATA Tasks

Situation	Indicates	Action
EMPTY WAIT counts READ tasks quickly fill all available XBLKs. It is not unusual to see an EMPTY WAIT count that is almost equal to the XFER count.	An EMPTY WAIT count that is much greater than the XFER count might indicate an excessive number of READ tasks. This situation is usually caused by an overloaded processor complex, indicating some performance degradation.	Reduce the number of READ tasks by using fewer SYSREC*nn* DDs, if *all* of the following conditions are true: • Performance degradation is indicated. • It is the normal time to run the load. • The processor load is normal for the time frame.
excessive FULL WAIT counts from READ SORT/DATA tasks use both pools of XBLKs.	The FULL WAIT count indicates how often the SORT/DATA tasks waited for data from the READ tasks to process.	If the number exceeds 10% of the XFER counts from the READ tasks, LOADPLUS can effectively use additional READ tasks. If the job has concatenated DDs, provide separate SYSREC*nn* ddnames for each data set.
excessive EMPTY WAITS and FULL WAITS		If the SORT/DATA tasks report a high number of EMPTY WAITs *and* the INDEX task shows a high number of FULL WAITs, it might be beneficial to increase the number of XBLKs. Weigh the benefits of increasing the number of XBLKs against the potential problems that might occur with memory utilization and paging rates. Changing the XBLKS installation option affects all LOADPLUS executions. If you decide to alter the option, proceed slowly, in increments of one per day. Observe and evaluate the results before leaving the new setting or increasing it again.

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Appendix A LOADPLUS Installation Options

This appendix presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Overview

The LOADPLUS product is installed by using the BMC Software OS/390 and z/OS Installer. During this installation, the customization process generates a customized installation data set. This data set contains customized jobs that install LOADPLUS into your specific DB2 environment. Two of these jobs establish the default processing option values that LOADPLUS uses:

- $C30DOPT establishes the defaults for basic LOADPLUS processing options and the dynamic work file allocation options.
- $C32SOPT contains options for the BMC Software BMCSORT technology.

These jobs assemble options macros, which contain the LOADPLUS processing options and the values for those options that are shipped with LOADPLUS and BMCSORT. When the Installer-generated customization job is submitted, it links the AMU$OPTS default options module in the APF-authorized library that is designated by your site. If any values for these options are changed during customization, the new values override the values from the options macro.

You can tailor the installation of LOADPLUS, including changing plan names, by changing the values for the LOADPLUS installation options. However, if you modify any of the values in $C30DOPT or $C32SOPT after LOADPLUS has been installed, you must rerun the jobs for these changes to take effect.

You can also create additional options modules that allow you to use different values of these options for different executions of LOADPLUS. For example, you might use the default options module for most jobs but create another options module with customized values for certain options for special situations. For information about specifying an options module at execution time, see Chapter 4, “How to Build and Execute LOADPLUS Jobs.”

For additional information about customizing your installation of LOADPLUS, see the customization guide that is provided with the distribution tape.

To find a description of each options macro, use Table A-1 on page A-3. Each of the macro figures is followed by a description of each option within that macro. The options are listed in the order that they appear in the macro.

To find the description of a specific option, use Table A-2 on page A-3. This table lists the options in alphabetical order and provides the page number that corresponds to a description for each option.

BMC Software, Inc., Confidential and Proprietary Information

Table A-1 LOADPLUS Installation Macros

Macro Name	Description	Reference
$AMUOPTS	basic options	A-6
$AMUDYNA	options for dynamic work file allocation	A-23
$AUPSMAC	options for BMCSORT	*BMCSORT Reference Manual*

Table A-2 LOADPLUS Installation Options (Part 1 of 3)

Installation Option	See Page
ACFORTSS	A-15
ACTIVE	A-24
APCOLLECTION	A-22
APCOMMIT	A-19
APMXAGNT	A-19
APOWNER	A-22
APRETLIM	A-19
APRETVAL	A-19
CBUFFS	A-8
CENTURY	A-13
CHEKPEND	A-10
COPYDDN	A-9
COPYLVL	A-16
COPYPEND	A-11
DATACLAS	A-27
DDTYPE	A-23
DELFILES	A-16
DISCDDN	A-18
DSNPAT	A-25
DSNUEXIT	A-12
ERRDDN	A-18
EXPDT	A-28
FILECHK	A-11
GDGEMPTY	A-29
GDGLIMIT	A-29
GDGSCRATCH	A-29

BMC Software, Inc., Confidential and Proprietary Information

Table A-2 LOADPLUS Installation Options (Part 2 of 3)

Installation Option	See Page
IBUFFS	A-8
IDCACHE	A-21
IDCDDN	A-18
IDERROR	A-21
IFALLOC	A-24
IMAGECPY	A-12
INDDN	A-17
INLINECP	A-17
KEEPDICTIONARY	A-20
LBUFFS	A-8
LOADCPY	A-12
LOADDECP	A-13
LOADDN	A-18
LOCKROW	A-15
MAXEXTSZ	A-28
MAXP	A-8
MAXTAPE	A-12
MGMTCLAS	A-27
MSGLEVEL	A-14
OPNDB2ID	A-8
ORIGDISP	A-20
PLANAUTH	A-14
PLANCAT	A-14
PLANCHKP	A-15
PLANCOPY	A-15
PLANSER	A-14
PLANSTAT	A-14
PLANSYNC	A-14
PREFORMAT	A-20
RCVYDDN	A-9
REDEFINE	A-21
RENMMAX	A-20
RETPD	A-28
RULES	A-11
SIZEPCT	A-24

BMC Software, Inc., Confidential and Proprietary Information

Table A-2 LOADPLUS Installation Options (Part 3 of 3)

Installation Option	See Page
SMAX	A-8
SMCORE	A-7
SMS	A-27
SMSUNIT	A-28
SQLDELAY	A-9
SQLRETRY	A-9
STOP@CMT	A-8
STOPDELAY	A-9
STOPRETRY	A-9
STORCLAS	A-27
TAPEDISP	A-13
THRESHLD	A-25
UNIT	A-24
UPDMAXA	A-22
UXSTATE	A-10
WBUFFS	A-8
WORKDDN	A-18
WORKUNIT	A-15
XBLKS	A-8

Basic LOADPLUS Installation Options

Figure A-1 on page A-6 shows the contents of the $AMUOPTS macro in $C30DOPT. This figure shows the basic LOADPLUS installation options and the values that are shipped with this version of LOADPLUS.

Additional options macros provide installation options for dynamic work file allocation and BMCSORT. For more information about these additional macros, see the following references:

- “Dynamic Allocation Installation Options” on page A-23
- *BMCSORT Reference Manual*

BMC Software, Inc., Confidential and Proprietary Information

Figure A-1 LOADPLUS Installation Options Module—$AMUOPTS Macro (Part 1 of 2)

```
$AMUOPTS SMCORE=(0K,0K),          MAXIMUM TOTAL SORT MEMORY              X
         STOP@CMT=NO,             ENABLE STOP AT COMMIT?                 X
         SMAX=8,                  MAXIMUM SORTS INVOKED CONCURRENTLY     X
         MAXP=5,                  MAXIMUM PARTS UNLOADED CONCURRENTLY    X
         XBLKS=3,                 NO. OF BLOCKS FOR INDEX TASK INPUT     X
         CBUFFS=30,               COPY DATA SET BUFFERS                  X
         LBUFFS=20,               LOAD DATA SET BUFFERS                  X
         WBUFFS=(20,10),          WORK DATA SET BUFFERS                  X
         IBUFFS=20,               INPUT COPY DATA SET BUFFERS            X
         OPNDB2ID=YES,            USE ALTERNATE DB2 DATA SET OPEN        X
         SQLDELAY=3,              NUMBER OF SECS BETWEEN SQL-911 RETRIES X
         SQLRETRY=100,            NUMBER OF SQL ATTEMPTS                 X
         STOPRETRY=300,           NUMBER OF TIMES TO CHECK IF STOPPED    X
         STOPDELAY=1,             NUMBR SECS BETWEEN STOP CHECKS         X
         COPYDDN=(BMCCPY,BMCCPZ),   LOCAL IMAGE COPY DD PREFIX           X
         RCVYDDN=(BMCRCY,BMCRCZ),   REMOTE IMAGE COPY DD PREFIX          X
         UXSTATE=SUP,             CALL DB2 USER EXITS IN SUP STATE       X
         CHEKPEND=(YES,)          SET CHECK PENDING WHEN NEEDED          X
         COPYPEND=(YES,)          SET COPY PENDING WHEN NEEDED           X
         FILECHK=FAIL,            FAIL IF TEMPORARY WORK FILES           X
         RULES=STANDARD,          WHEN, NULLIF, DEFAULTIF COMPARISONS    X
         IMAGECPY=YES,            IMAGE COPY SUPPORT                     X
         LOADCPY=YES,             SYSCOPY LOAD ENTRY,COPY YES/NO?        X
         MAXTAPE=3,               MAX TAPE UNITS TO DYNAMICALLY ALLOC    X
         DSNUEXIT=(NONE,ASM),     DSN USER EXIT NAME AND LANGUAGE        X
         TAPEDISP=DELETE,         FINAL DISPOSITION OF TAPE WORK FILES   X
         CENTURY=,                CENTURY SPECIFICATION                  X
         LOADDECP=NO,             LOAD DSNHDECP                          X
         MSGLEVEL=1,              DEFAULT MESSAGE LEVEL PARAMETER        X
         PLANAUTH=AMUA71xx,       AUTHORIZATION PLAN NAME                X
         PLANCAT=AMUC71xx,        CATALOG LOOK-UP PLAN NAME              X
         PLANSER=AMUR71xx,        SERIALIZATION PLAN NAME                X
         PLANSYNC=AMUS71xx,       SYNC PLAN NAME                         X
         PLANSTAT=AMUT71xx,       BMCSTATS PLAN NAME                     X
         PLANCOPY=AMUI71xx,       IMAGE COPY SUPPORT PLAN                X
         PLANCHKP=AMUP71xx,       CHECK PENDING SUPPORT PLAN             X
         WORKUNIT=SYSALLDA,       UNIT FOR WORK DATA SETS                X
         LOCKROW=NO,              BMC TABLE SERIALIZATION                X
         ACFORTSS=NO,             CA-ACF2/DB2 OR CA-TOP SECRET/DB2?      X
         DELFILES=(NO,NO)         DELETE WORKFILES/SYSDISC FILES?        X
         COPYLVL=FULL,            FULL OR PARTITION LEVEL COPY DS        X
         INLINECP=NO,             CREATE INLINE IMAGE COPY?              X
         INDDN=SYSREC,            INPUT DDNAME OR DD PREFIX              X
         WORKDDN=SYSUT1,          INDEX WORK DDNAME OR DD PREFIX         X
         LOADDN=SORTOUT,          LOAD WORK DDNAME OR DD PREFIX          X
         ERRDDN=SYSERR,           ERROR DDNAME                           X
         DISCDDN=SYSDISC,         DISCARD DDNAME                         X
         IDCDDN=SYSIDCIN,         IDCAMS DDNAME                          X
```

BMC Software, Inc., Confidential and Proprietary Information

Figure A-1 LOADPLUS Installation Options Module (Part 2 of 2)

```
APMXAGNT=10,          MAX NUMBER AGENTS APPLY PLUS CAN START X
APCOMMIT=2500,        NUMBER RECORDS BETWN APPLY PLUS COMMIT X
APRETLIM=COUNT,       TYPE LIMIT APPLY PLUS RETRY ON ERROR   X
APRETVAL=5,           NUM LIMIT APPLY PLUS RETRY ON ERROR    X
ORIGDISP=DELETE,      ORIGINAL DATA SET DISPOSITION          X
RENMMAX=30,           MAX NUMBR RENAME/DEL TASKS TO START    X
KEEPDICTIONARY=NO,    KEEP COMPRESSION DICTIONARY?           X
PREFORMAT=NO,         PREFORMAT UNUSED DS PAGES?             X
REDEFINE=YES,         DELETE AND REDEFINE VSAM DATA SETS     X
IDERROR=DISCARD,      ACTION ON IDENTITY COLUMN RANGE ERROR  X
IDCACHE=1000,         CACHE SIZE OF IDENTITY COLUMN NUMBERS  X
UPDMAXA=NO,           UPDATE MAXASSIGNEDVAL ON GEN BY DEFAULTX
APCOLLECTION=,        COLLECTION ID FOR APPLY PLUS           X
APOWNER=              BIND OWNER ID FOR APPLY PLUS
```

This section describes each of the options and plan names.

SMCORE=(0K,0K) SMCORE specifies the amount of memory that you want each invocation of BMCSORT to use. BMC Software strongly recommends that you use the values 0K and 0K for this option. For additional information, see "Controlling Memory Usage" on page 6-13.

The values 0K and 0K tell LOADPLUS to determine the appropriate amount of memory to use for each sort based on the following criteria:

- the number of sorts to process
- the amount of memory that is available during optimization
- the value specified for REGION in either your JCL or system exits

The first value specifies the total amount of memory to use both above and below the 16-megabyte line for each sort. You can specify this value in either kilobytes or megabytes. The following values are valid for this parameter:

- 0K (or 0M) tells LOADPLUS to determine the appropriate amount.
- 4096K through 65536K (or 4M through 64M) tells LOADPLUS to use the specified amount.

The second value specifies the amount of memory that is required below the 16-megabyte line for each sort. You can specify this value in either kilobytes or megabytes. The following values are valid for this parameter:

- 0K (or 0M) tells LOADPLUS to determine the appropriate amount.
- 256K through 4096K (or 1M through 4M) tells LOADPLUS to use the specified amount.

BMC Software, Inc., Confidential and Proprietary Information

STOP@CMT=NO This option specifies whether to add the 'AT(COMMIT)' parameter to all DB2 STOP commands that LOADPLUS issues. STOP@CMT=YES adds the 'AT(COMMIT)' parameter. STOP@CMT=NO does not. See the IBM DB2 *Command Reference* manual for details and implications of the 'AT(COMMIT)' parameter.

SMAX=8 The SMAX option specifies the maximum number of sort tasks of a given type (data or index) that can run concurrently during a load. The maximum value that you can specify is 16. For more information, see “Controlling the Number of Sort Processes” on page 6-15.

MAXP=5 The MAXP option specifies the maximum number of partitions to load concurrently. For more information, see “MAXP Installation Option” on page 6-32.

XBLKS=3 The XBLKS option specifies the number of blocks to use for transferring data from the read task to the data process and index data from the data process to the index-building process during the PRELOAD phase or the COMBINED phase. The minimum value is 3. For more information, see “Transfer Blocks (XBLKs)” on page 6-29.

CBUFFS=30 The CBUFFS option specifies the number of buffers to use for each copy data set. If you are making multiple copies, the same set of buffers is used, not an additional set. For more information, see “Controlling Buffer Usage” on page 6-9.

LBUFFS=20 The LBUFFS option specifies the number of buffers to use for each active load (SORTOUT) data set. For more information, see “Controlling Buffer Usage” on page 6-9.

WBUFFS=(20,10) The WBUFFS option specifies the number of buffers to use for each work (SYSUT1) data set. LOADPLUS uses the first number if only one work data set is specified. LOADPLUS uses the second number if multiple work data sets are specified. For more information, see “Controlling Buffer Usage” on page 6-9.

IBUFFS=20 The IBUFFS option specifies the number of buffers to use for each active input (SYSREC) data set. For more information, see “Controlling Buffer Usage” on page 6-9.

OPNDB2ID=YES OPNDB2ID=YES tells LOADPLUS to use the DB2 RACF ID instead of the RACF ID of the user who is running LOADPLUS when opening or performing Access Method Services (AMS) functions on DB2 data sets.

Specifying OPNDB2ID=NO tells LOADPLUS to use the RACF ID of the user who is running LOADPLUS. If NO is specified, the user must have the appropriate RACF authority.

BMC Software, Inc., Confidential and Proprietary Information

For any security system other than RACF, specify OPNDB2ID=NO, and LOADPLUS uses the security authorization ID of the user who is running LOADPLUS.

SQLDELAY=3 The SQLDELAY option specifies the number of seconds that LOADPLUS should wait between retry attempts after receiving an SQL -911 return code. This interval is in addition to the time that elapses when DB2 waits for a timeout or deadlock. The number of seconds can range from 1 through 655.

SQLRETRY=100 The SQLRETRY option specifies the number of times that LOADPLUS should attempt an SQL statement. The number of attempts can range from 1 through 255.

STOPRETRY=300 The STOPRETRY option specifies the number of times that LOADPLUS checks to determine whether DB2 has changed the status of an object from stop pending (STOPP) to stopped (STOP). The number of checks can be any nonzero positive integer.

STOPDELAY=1 The STOPDELAY option specifies the number of seconds that LOADPLUS waits before it checks again to determine whether DB2 has stopped the object. The number of seconds can be any nonzero positive integer.

COPYDDN=(BMCCPY, BMCCPZ)

The COPYDDN option specifies default ddnames or ddname prefixes for the local copy data sets that receive a full image copy or a DSN1COPY either of the table space or of each partition in the table space that is being loaded. If you are registering the copies, the first name is the local primary copy, and the second name is the local backup copy.

When using this name as a prefix, ensure that you allow sufficient bytes for the partition number to be added and still have a valid ddname (eight bytes or less). In addition, if you change the default ddnames from BMCCPY and BMCCPZ, you must also change the names in your JCL. For more information, see “Copy Data Sets” on page 4-9.

You can override this value by using the COPYDDN command option. See page 3-85 for discussion of this option.

RCVYDDN=(BMCRCY, BMCRCZ)

The RCVYDDN option specifies default ddnames or ddname prefixes for the remote copy data sets that receive a full image copy or a DSN1COPY either of the table space or of each partition in the table space that is being loaded. If you are registering the copies, the first name is the remote primary copy, and the second name is the remote backup copy.

BMC Software, Inc., Confidential and Proprietary Information

When using this name as a prefix, ensure that you allow sufficient bytes for the partition number to be added and still have a valid ddname (eight bytes or less). In addition, if you change the default ddnames from BMCRCY and BMCRCZ, you must also change the names in your JCL. For more information, see “Copy Data Sets” on page 4-9.

You can override this value by using the RECOVERYDDN command option. See page 3-86 for discussion of this option.

UXSTATE=SUP UXSTATE=SUP specifies that LOADPLUS should call DB2 user exits (such as EDITPROCs or date/time exits) in supervisor state (and PSW key=7).

Specifying UXSTATE=PROB tells LOADPLUS to call DB2 user exits in problem state (and PSW key=7). The requirements of the exits dictate the UXSTATE setting. Check with the exit author (or vendor) before changing to UXSTATE=PROB.

CHEKPEND=(YES,) The CHEKPEND option specifies whether you want to set dependent table spaces to CHECK pending (CHKP) status. For more details about setting CHECK pending status, see “Referential Integrity and Check Constraints” on page 2-51 and “CHECK CONSTRAINTS” on page 3-66.

Unless you specify the ENFORCE keyword with this installation option, you can override the value for this option by using the CHECKPEND command option. See page 3-67 for discussion of this command option.

Table A-3 describes the action that LOADPLUS takes for each set of values that you can specify.

Table A-3 CHEKPEND Option Values

Values	Action
(YES,)	sets CHECK pending status unless overridden by CHECKPEND NO on the LOAD command
(NO,)	does not set CHECK pending status unless overridden by CHECKPEND YES on the LOAD command
(YES,ENFORCE)	sets CHECK pending status and allows only CHECKPEND YES on the LOAD command **Note**: If CHECKPEND NO is specified on the LOAD command, LOADPLUS issues message BMC50115E and terminates.
(NO,ENFORCE)	does not set CHECK pending status and allows only CHECKPEND NO on the LOAD command **Note**: If CHECKPEND YES is specified on the LOAD command, LOADPLUS issues message BMC50115E and terminates.

BMC Software, Inc., Confidential and Proprietary Information

COPYPEND=(YES,) The COPYPEND option specifies whether you want LOADPLUS to set the table space to COPY pending status if a copy was not made as part of the load job.

Unless you specify the ENFORCE keyword with this installation option, you can override the value for this option by using the COPYPEND command option. For discussion of this command option, see page 3-80.

Table A-4 describes the action that LOADPLUS takes for each set of values that you can specify.

Table A-4 COPYPEND Option Values

Values	Action
(YES,)	sets COPY pending status unless overridden by COPYPEND NO or NOCOPYPEND on the LOAD command
(NO,)	does not set COPY pending status unless overridden by COPYPEND YES on the LOAD command
(YES,ENFORCE)	sets COPY pending status and allows only COPYPEND YES on the LOAD command **Note**: If COPYPEND NO or NOCOPYPEND is specified on the LOAD command, LOADPLUS issues message BMC50115E and terminates.
(NO,ENFORCE)	does not set COPY pending status and allows only COPYPEND NO on the LOAD command **Note**: If COPYPEND YES is specified on the LOAD command, LOADPLUS issues message BMC50115E and terminates.

FILECHK=FAIL FILECHK=FAIL tells LOADPLUS to terminate when it encounters a work file that has been allocated as a temporary data set. If FILECHK=WARN is specified, LOADPLUS issues a warning message only and continues processing if it encounters work files that are temporary data sets. For a definition of temporary data sets within LOADPLUS, see “Check for Data Set Attributes” on page 2-22.

RULES=STANDARD

The RULES option specifies the set of comparison rules for LOADPLUS to use in WHEN, NULLIF, and DEFAULTIF processing. You can specify STANDARD or BMC. This specification determines the order in which WHEN testing and data conversion, if any, is performed. RULES also determines which set of comparison operators can be used in comparisons. For information about the differences between STANDARD and BMC, see Appendix E, “RULES Installation Option Examples.”

BMC Software, Inc., Confidential and Proprietary Information

When your LOAD command contains the FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD option, LOADPLUS changes the value of the RULES option to BMC, regardless of the value that you specify at installation. When your LOAD command contains the FORMAT CSV option, LOADPLUS changes the value of this option to STANDARD, regardless of the value that you specify at installation.

IMAGECPY=YES The IMAGECPY option specifies whether to provide image copy support. Image copy support inserts a row into SYSIBM.SYSCOPY if you request image copies as part of your load. You can specify YES or NO.

You must specify IMAGECPY=YES if you use the COPY YES command option and register any of the copies.

If you specify IMAGECPY=YES, LOADPLUS assumes a value of LOADCPY=YES when you create registered image copies by using LOADPLUS, regardless of the value that you specify for that option explicitly. See the COPY option in Chapter 3 for more information about making image copies as part of the load process.

LOADCPY=YES The LOADCPY option specifies whether you want rows to be inserted into SYSIBM.SYSCOPY that indicate that a load has been performed with LOG NO. You can specify YES or NO.

If you specify IMAGECPY=YES, LOADPLUS assumes a value of LOADCPY=YES, regardless of the value that you specify for this option explicitly.

MAXTAPE=3 The MAXTAPE option specifies the maximum number of tape drives to dynamically allocate at one time. The value must be greater than zero.

You can override this value by using the MAXTAPE command option. See page 3-134 for discussion of this option.

DSNUEXIT=(NONE,ASM)

The DSNUEXIT option supplies the name of a user-defined exit routine and the programming language in which the routine is written. This user exit allows you to provide user-defined variables for constructing data set name patterns with either the DSNPAT installation option (in the $AMUDYNA macro) or the DSNPAT command option. Page 3-145 provides details of the DSNPAT option. The valid values for the programming language, other than ASM, are COBOL2, LE_COBOL, C, and LE_C.

You can override the values for this option by using the DSNUEXIT command option. See page 3-135 for discussion of this option.

See Appendix F, “LOADPLUS User Exits,” for more information about user-defined exit routines.

BMC Software, Inc., Confidential and Proprietary Information

TAPEDISP=DELETE The TAPEDISP option specifies the final disposition of tape work files when the value of the DELETEFILES or SYSDISC option is YES.

If you specify TAPEDISP=DELETE, each tape file will have a disposition of OLD,DELETE,DELETE when the tape is deallocated.

If you specify TAPEDISP=UNCATLG, each tape file will have a disposition of OLD,UNCATLG,UNCATLG when the tape is deallocated. Depending on your tape management environment, using UNCATLG can prevent a tape remount.

CENTURY= The CENTURY option specifies the 100-year range that determines the century for the DATE and TIMESTAMP external formats that contain two-digit year values. The value for this option is two four-digit years in the format *(ccyy,ccyy)*. The first four-digit year value must be less than the second four-digit year. You must specify both values, and these values must span 100 years.

Warning! LOADPLUS does not supply a value for this option. If you do not supply a value, your installation options module will not assemble successfully.

Any two-digit year in your input data that lies between the first *yy* specification and 99 is prefixed with the first *cc* value to create a four-digit year. Any two-digit year in your input data that lies between 00 and the second *yy* specification is prefixed with the second *cc* value to create a four-digit year.

For example, if you specify CENTURY(1950,2049), LOADPLUS places 19 in front of each two-digit year with a value 50 through 99, and places 20 in front of each two-digit year with a value 00 through 49. The date 99/12/31 becomes 1999/12/31 and 00/12/31 becomes 2000/12/31.

You can override the values for this option with the CENTURY command option (see page 3-70) or the CENTURY field specification parameter (see page 3-115).

LOADDECP=NO A copy of DSNHDECP is always loaded from STEPLIB so that the utility can obtain the DB2 SSID value if it is not specified in the JCL.

If you specify YES, LOADPLUS obtains all DB2 installation default values from the STEPLIB copy. In addition, if the value of MSGLEVEL is 1, LOADPLUS displays these values in your SYSPRINT.

If you specify NO, LOADPLUS locates DSNHDECP in DB2 subsystem memory to obtain DB2 installation default values (other than the SSID) and does not display the DSNHDECP values in your SYSPRINT.

BMC Software, Inc., Confidential and Proprietary Information

MSGLEVEL=1 This option controls the messages that LOADPLUS returns to the user in the SYSPRINT data set. MSGLEVEL=0 returns minimal messages. MSGLEVEL=1 returns additional messages to help you diagnose problems and fine-tune performance.

You can override the value for this option with the MSGLEVEL parameter on the EXEC statement. See "Message Level (MSGLEVEL)" on page 4-7 for information about this parameter.

PLANAUTH=AMUA71*xx*

The PLANAUTH option specifies the plan that checks that the object to be loaded exists and that you have sufficient DB2 authorization to run LOADPLUS.

Note: The *xx* in the supplied plan name represents the maintenance number for the current release.

PLANCAT=AMUC71*xx*

The PLANCAT option specifies the plan that performs catalog look-up for information about the object to be loaded.

Note: The *xx* in the supplied plan name represents the maintenance number for the current release.

PLANSER=AMUR71*xx*

The PLANSER option specifies the plan that checks whether the object to be loaded is already in use by a DB2 utility or another LOADPLUS utility job.

Note: The *xx* in the supplied plan name represents the maintenance number for the current release.

PLANSYNC=AMUS71*xx*

The PLANSYNC option specifies the plan that the LOADPLUS SYNC subtask is to use. Note that if you rebind this plan at any time after installation, you must rerun the installation job AMUICOPY as well. If you do not run AMUICOPY, image copy support is lost and LOADPLUS issues message BMC50373E.

Note: The *xx* in the supplied plan name represents the maintenance number for the current release.

PLANSTAT=AMUT71*xx*

The PLANSTAT option specifies the plan to use to provide BMCSTATS to the DASD MANAGER PLUS statistics tables and UPDATEDB2STATS information to the DB2 catalog.

Note: The *xx* in the supplied plan name represents the maintenance number for the current release.

BMC Software, Inc., Confidential and Proprietary Information

PLANCOPY=AMUI71*xx*

The PLANCOPY option specifies the plan to use to install image copy support.

Note: The *xx* in the supplied plan name represents the maintenance number for the current release.

PLANCHKP=AMUP71*xx*

The PLANCHKP option specifies the plan to use to set table spaces in CHECK pending (CHKP) status.

Note: The *xx* in the supplied plan name represents the maintenance number for the current release.

WORKUNIT=SYSALLDA

The WORKUNIT option specifies the unit to use for a temporary, one-track work data set. VIO is an acceptable value for this option.

LOCKROW=NO

LOCKROW=NO tells LOADPLUS to use SQL LOCK TABLE statements for serialization when updating the BMCSYNC and BMCUTIL tables.

Specifying LOCKROW=YES tells LOADPLUS to use MVS enqueues instead of SQL LOCK TABLE statements for serialization when updating the BMCSYNC and BMCUTIL tables. Using LOCKROW=YES should prevent most SQL -911 return codes that occur when multiple BMC Software products concurrently update the BMCSYNC and BMCUTIL tables.

Note: Using LOCKROW=YES requires row-level locking. You must define the BMCSYNC and BMCUTIL tables with LOCKSIZE ROW to achieve row-level locking.

ACFORTSS=NO

The ACFORTSS option specifies whether LOADPLUS should look for the presence of the CA-ACF2 for DB2 or CA-Top Secret for DB2 security product from Computer Associates when verifying utility authorizations.

If you specify YES, LOADPLUS uses CA-ACF2 for DB2 or CA-Top Secret for DB2 if that security is enabled on the subsystem where LOADPLUS is running.

Warning! Specify YES only if you have applied the appropriate APAR to enable SAF support for CA-ACF2 for DB2 or CA-Top Secret for DB2. Obtain the APAR that applies to the current release of your security product from Computer Associates.

For more information, see “Authorization Verification Mechanisms” on page 2-3.

BMC Software, Inc., Confidential and Proprietary Information

DELFILES=(NO,NO) The DELFILES option tells LOADPLUS whether to delete the SORTOUT, SORTWK, SYSUT1, SYSERR, and SYSDISC files after the load completes successfully. The first value tells LOADPLUS whether to delete your work files after the load completes successfully, regardless of whether LOADPLUS uses the files. The second value tells LOADPLUS how to handle the discard file (SYSDISC) specifically. LOADPLUS deletes any SYSDISC files only if there were no discarded records during the load process.

Table A-5 describes the action that LOADPLUS takes for each set of values that you can specify.

Table A-5 DELFILES Option

Values	Action
NO	does not delete any work files
(NO,NO)	does not delete any work files
(NO,YES)	deletes only SYSDISC, if it is empty
YES	deletes all work files except SYSDISC (assumes NO for the second value) regardless of the JCL disposition of these files and whether the files were used
(YES,YES)	deletes all work files and SYSDISC (deletes SYSDISC only if it is empty) regardless of the JCL disposition of these files and whether the files were used
(YES,NO)	deletes all work files except SYSDISC

If your work files are defined with DISP=(*any*,DELETE,DELETE), LOADPLUS deletes these work files after the load completes even if you specify NO on this option.

If you are running the load job in a worklist environment, LOADPLUS ignores the value that you set in the installation option and, by default, does not delete the work files. LOADPLUS does this so that the work files exist for subsequent executions in the job. If you want to delete the work files, you must specify DELETEFILES YES on the LOAD command in the worklist. If you want to delete SYSDISC (if it is empty), you must specify SYSDISC YES on the LOAD command in the worklist.

You can override the installation value by using the DELETEFILES and SYSDISC command options. For discussion of these options, see page 3-57.

COPYLVL=FULL The COPYLVL option specifies how LOADPLUS is to assign dynamically allocated image copy data sets when loading all partitions of a partitioned table space. FULL allocates a single copy data set to contain all of the partitions that are being loaded. PART allocates individual copy data sets for each partition that is being loaded.

BMC Software, Inc., Confidential and Proprietary Information

You can override this value by using the COPYLVL command option. See page 3-84 for discussion of this option.

INLINECP=NO INLINECP=NO tells LOADPLUS to not make an inline image copy *as* it loads the table space, but to create an image copy *after* it loads the table space. If you specify YES, LOADPLUS creates an inline image copy as it loads the table space. A value of YES requires that COPY YES be specified on the LOAD command.

If any of the following conditions apply to a particular load job, LOADPLUS functions as if you specified NO, regardless of the value that is specified either here or with the LOAD command:

- You specify LOAD RESUME YES.

  Note: If you specify LOAD RESUME YES with PART *n* REPLACE for all participating partitions, LOADPLUS creates an inline image copy for the partitions that you are replacing.

- The copy data sets are on a stacked tape.

- You restart your load job, with the following exception: If you are running a two-phase load job and you specify the RESTART(PHASE) parameter on your EXEC statement, LOADPLUS generates an inline copy if the value of INLINE is YES.

- The size of the table space page is greater than 4 KB, you are loading multiple partitions, and you have one image copy data set.

You can override this value by using the INLINE command option. See page 3-82 for discussion of this option.

INDDN=SYSREC The INDDN option specifies a default ddname or ddname prefix for the input data set.

When using this name as a prefix, ensure that you allow sufficient bytes for LOADPLUS to add the number of input files and still have a valid ddname (eight bytes or less). In addition, if you change the default ddname from SYSREC and you are not using the INDSN command option, you must also change the name in your JCL. For more information, see “SYSRECnn Data Sets” on page 4-18.

If any DD statements in your JCL match the value for INDDN and you specify the INDSN command option, LOADPLUS fails and issues message BMC50460E.

You can override this value by using the INDDN command option. See page 3-36 for discussion of this option.

BMC Software, Inc., Confidential and Proprietary Information

WORKDDN=SYSUT1 The WORKDDN option specifies a default ddname or ddname prefix for the work data set.

When using this name as a prefix, ensure that you allow sufficient bytes for LOADPLUS to add the number of indexes and still have a valid ddname (eight bytes or less). In addition, if you change the default ddname from SYSUT1, you must also change the name in your JCL. For more information, see “SYSUT1nn Data Sets” on page 4-20.

You can override this value by using the WORKDDN command option. See page 3-38 for discussion of this option.

LOADDN=SORTOUT The LOADDN option specifies a default ddname or ddname prefix for the output data set from the PRELOAD phase.

When using this name as a prefix for a partitioned table space, ensure that you allow sufficient bytes for the partition number to be added and still have a valid ddname (eight bytes or less). In addition, if you change the default ddname from SORTOUT, you must also change the name in your JCL. For more information, see “SORTOUTn Data Sets” on page 4-12.

You can override this value by using the LOADDN command option. See page 3-37 for discussion of this option.

ERRDDN=SYSERR The ERRDDN option specifies a default ddname for the error data set (the data set that contains information about records that are discarded because of errors). If you change the default ddname from SYSERR, you must also change the name in your JCL. For more information, see “SYSERR Data Set” on page 4-17.

You can override this value by using the ERRDDN command option. See page 3-39 for discussion of this option.

DISCDDN=SYSDISC The DISCDDN option specifies a default ddname for the discard data set. LOADPLUS places a record in this data set when that record cannot be loaded for any of the reasons that are listed in “SYSDISC Data Set” on page 4-15. If you change the default ddname from SYSDISC, you must also change the name in your JCL.

You can override this value by using the DISCARDDN command option.

IDCDDN=SYSIDCIN The IDCDDN option specifies a default ddname for the input data set containing the IDCAMS command statements that LOADPLUS uses to redefine VSAM data sets. For more information, see “SYSIDCIN Data Set” on page 4-22.

You can override this value by using the IDCDDN command option. See page 3-57 for discussion of this option.

BMC Software, Inc., Confidential and Proprietary Information

APMXAGNT=10 (for a SQLAPPLY load only) The APMXAGNT option specifies the maximum number of agents that Apply Plus can start for this load. Valid values for this option are 1 through the number of batch threads that are available in your DB2 subsystem. APMXAGNT affects only partitioned or multiple-table table spaces. LOADPLUS uses at most one agent per partition and table.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the MaxAgents parameter in the *Apply Plus Reference Manual*.

You can override this value by using the APMAXAGENTS command option.

APCOMMIT=2500 (for a SQLAPPLY load only) The APCOMMIT specifies the maximum number of records that you want each Apply Plus agent to load before issuing a COMMIT statement. Valid values for this option are 1 through 32767.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the StatementCount parameter in the *Apply Plus Reference Manual*.

You can override this value by using the APCOMMIT command option.

APRETLIM=COUNT (for a SQLAPPLY load only) The APRETLIM option tells Apply Plus how to respond to an SQL -911, -913, or -904 error. A value of COUNT tells Apply Plus to retry the number of times specified in the APRETVAL option. A value of TIME tells Apply Plus to continue to retry until it reaches the number of seconds specified in the APRETVAL option.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the RetryLimit parameter in the *Apply Plus Reference Manual*.

You can override this value by using the APRETRYLIM command option.

APRETVAL=5 (for a SQLAPPLY load only) The APRETVAL option specifies the Apply Plus retry limit following an SQL -911, -913, or -904 error. Depending on the value for APRETLIM, this limit is either the number of retry attempts or the number of seconds after which Apply Plus should stop retrying. Valid values are 0 through 32767. A value of 0 tells Apply Plus not to retry.

LOADPLUS passes this parameter information to Apply Plus. For additional information, see the RetryValue parameter in the *Apply Plus Reference Manual*.

You can override this value by using the APRETRYVAL command option.

BMC Software, Inc., Confidential and Proprietary Information

ORIGDISP=DELETE ORIGDISP=DELETE tells LOADPLUS to delete the original data sets after successfully renaming the staging data sets and completing a load by using LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE.

If you specify RENAME, LOADPLUS renames the original data sets to the staging data set names (that is, LOADPLUS replaces OLD with BMC in the OLDDBC node of the cluster and in the OLDDBD node of the data component). ORIGDISP=RENAME preserves the space that was initially allocated for the original data sets. As a result, the staging data sets are ready to use in a subsequent load process.

For more information about the names assigned to the original data sets, see “Staging Data Sets” on page 2-31.

You can override this value by using the ORIGINALDISP command option (page 3-59).

RENMMAX=30 The RENMMAX option indicates the maximum number of tasks to start during the staging data set rename and delete process. You can specify any of the following values:

- 0 to have LOADPLUS determine the number of tasks to start
- *n* to specify that LOADPLUS can start a maximum of *n* tasks (where *n* is a positive integer from 1 through 32767)

KEEPDICTIONARY=NO

The KEEPDICTIONARY option tells LOADPLUS whether to keep the existing compression dictionary. KEEPDICTIONARY=NO tells LOADPLUS to build a new compression dictionary.

KEEPDICTIONARY=YES tells LOADPLUS to keep the existing compression dictionary. If a dictionary does not exist, LOADPLUS builds one.

You can override this value either at the table space level, by using the KEEPDICTIONARY option with LOAD REPLACE, or at the partition level, by using the KEEPDICTIONARY option with PART REPLACE.

PREFORMAT=NO PREFORMAT=NO tells LOADPLUS not to preformat unused pages in a data set.

If you specify PREFORMAT=YES, LOADPLUS writes pages initialized with zeros up to the high-allocated relative byte address (RBA) of the table space and index spaces that are participating in the load.

You can override this value, either at the table space level or at the partition level, by using the appropriate PREFORMAT option.

BMC Software, Inc., Confidential and Proprietary Information

REDEFINE=YES The REDEFINE option tells LOADPLUS whether to delete and redefine the VSAM data sets for the table space and index spaces that are participating in the load. LOADPLUS can delete and redefine both user-defined (VCAT-defined) data sets and data sets that are defined in DB2 storage groups (STOGROUP-defined).

For SHRLEVEL NONE, REDEFINE=YES tells LOADPLUS to delete and redefine the VSAM data sets for the space prior to loading it. For LOAD REPLACE SHRLEVEL REFERENCE or LOAD REPLACE SHRLEVEL CHANGE, LOADPLUS deletes and defines the staging data sets.

REDEFINE=NO tells LOADPLUS not to delete and redefine the existing VSAM data sets for the table space or indexes. Instead, LOADPLUS issues message BMC50391I, reuses the existing data sets, and resets the high-used RBA.

You can override this value by using the REDEFINE command option. For information about the command option as well as additional information about how LOADPLUS functions under each value of this option, see "REDEFINE" on page 3-54.

IDERROR=DISCARD

IDERROR=DISCARD tells LOADPLUS to discard any rows with generated identity column values that are outside the range that is defined on the identity column.

If you specify IDERROR=FAIL, LOADPLUS terminates as soon as it encounters a generated identity column value that is outside the range that is defined on the identity column. LOADPLUS ignores this option for a SQLAPPLY load job.

You can override this value by using the IDERROR command option.

IDCACHE=1000 The IDCACHE option allows you to specify the size of the cache that LOADPLUS uses when generating values for an identity column. LOADPLUS reserves this cache in SYSIBM.SYSSEQUENCES. LOADPLUS uses one cache for each READ task. Valid values for this option are 1 through 100000. For information about how the value of this option can affect the performance of your load jobs, see "IDCACHE" on page 3-71.

Note: This is a temporary cache that LOADPLUS uses and does not affect the cache that you specified when defining your identity column.

You can override this value by using the IDCACHE command option.

BMC Software, Inc., Confidential and Proprietary Information

UPDMAXA=NO UPDMAXA=NO tells LOADPLUS not to update the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES when loading identity column values from an input file. Specifying YES tells LOADPLUS to update this field when loading identity column values from an input file.

Note: When generating identity column values, LOADPLUS always updates MAXASSIGNEDVAL.

You can override this value by using the UPDATEMAXA command option.

APOWNER= (for a SQLAPPLY load only) The APOWNER option specifies the authorization ID that Apply Plus uses to bind the DB2 plan and packages for the apply request. If you change the value for this installation option, you can specify an owner ID with a length of up to eight bytes.

LOADPLUS passes this value to Apply Plus. However, you can override this value by using the APOWNER command option.

If you keep the null value that was shipped with LOADPLUS and no value is specified for the APOWNER command option, Apply Plus uses the value that is provided in its BindOwner parameter.

Note: You cannot specify a null value for the APOWNER command option.

APCOLLECTION= (for a SQLAPPLY load only) The APCOLLECTION option specifies the ID for the collection to which Apply Plus dynamically binds packages during execution. If you change the value for this installation option, you can specify a collection ID with a length of up to 18 bytes.

LOADPLUS passes this parameter information to Apply Plus. However, you can override this value by using the APCOLLECTION command option.

If you keep the null value that was shipped with LOADPLUS and no value is specified for the APCOLLECTION command option, Apply Plus uses the value that is provided in its CollectionID parameter.

Note: You cannot specify a null value for the APCOLLECTION command option.

BMC Software, Inc., Confidential and Proprietary Information

Dynamic Allocation Installation Options

Nine $AMUDYNA options macros produce the installation options for dynamic allocation—one macro for each data set type. Figure A-2 shows the contents of one of the compiled $AMUDYNA macros in $C30DOPT and the values that are shipped with this version of the product for one of the data set types. The other macros contain the same options but allow you to specify different values for different data set types.

Your $C30DOPT member must contain exactly nine $AMUDYNA macros. Otherwise, you will receive an assembly error. Each $AMUDYNA macro must specify a different DDTYPE.

Figure A-2 LOADPLUS Installation Options Module—$AMUDYNA Macro

```
$AMUDYNA DDTYPE=LOCPFCPY,              DYNAMIC ALLOCATION FOR BMCCPY      X
         ACTIVE=NO,                    DYNAMIC ALLOC ACTIVE FOR THIS TYPE?X
         IFALLOC=USE,                  USE DD'S IN JCL IF CODED           X
         SIZEPCT=(100,100),            USE 100% OF LOADS CALCULATED SPACE X
         UNIT=(SYSALLDA,SYSALLDA),     ALLOCATION UNITS                   X
         THRESHLD=0,                   DON'T USE SECONDARY UNIT/CLASS     X
         DSNPAT=&&UID.&&UTILPFX.&&DDNAME, DSN PATTERN                     X
         SMS=NO,                       DON'T DO SMS ALLOCATIONS           X
         DATACLAS=(NONE,NONE),         NO SMS DATACLAS                    X
         MGMTCLAS=(NONE,NONE),         NO SMS MGMTCLAS                    X
         STORCLAS=(NONE,NONE),         NO SMS STORCLAS                    X
         SMSUNIT=NO,                   DON’T PASS UNIT VALUE TO SMS       X
         MAXEXTSZ=0,                   MAXIMUM EXTENT SIZE                X
         EXPDT=,                       EXPIRATION DATE - COPY DS          X
         RETPD=,                       RETENTION PERIOD - COPY DS         X
         GDGLIMIT=5,                   NUMBER OF GDG GENERATIONS TO KEEP  X
         GDGEMPTY=NO,                  UNCATALOG ONLY OLDEST GENERATION   X
         GDGSCRATCH=NO                 DO NOT DELETE UNCATALOGED DS
```

This section describes each of the dynamic allocation options. All of these options can also be specified with the DDTYPE command option, allowing the user to override the defaults that were established at installation. See “Dynamic Work File Allocation Options” on page 3-133 for details.

DDTYPE=LOAD The DDTYPE option specifies the data set type for which you are establishing dynamic allocation options. The valid values for this option are listed in Table A-6 on page A-24. Each instance of the $AMUDYNA macro must contain a different value for this option. For details about each data set, see “LOADPLUS DD Statements” on page 4-8.

BMC Software, Inc., Confidential and Proprietary Information

Table A-6 Valid DDTYPE Values

DDTYPE Value	Corresponding Default ddname
LOAD	SORTOUT
WORK	SYSUT1
SORTWORK	SORTWK
DISCARD	SYSDISC
ERROR	SYSERR
LOCPFCPY	BMCCPY
LOCBFCPY	BMCCPZ
REMPFCPY	BMCRCY
REMBFCPY	BMCRCZ

ACTIVE=NO The ACTIVE option tells LOADPLUS whether to dynamically allocate the specified data sets. A value of YES activates dynamic allocation. A value of NO inactivates dynamic allocation.

Note: When LOADPLUS runs in a worklist environment, it ignores the ACTIVE option in your installation options module. LOADPLUS dynamically allocates your data sets only if the invoking product (DASD MANAGER PLUS, CATALOG MANAGER, or CHANGE MANAGER) supplies the ACTIVE YES syntax.

IFALLOC=USE The IFALLOC option tells LOADPLUS whether to use data sets that are specified in your JCL or free them. The two valid values are USE and FREE. Specifying USE allows you to merge dynamically allocated data sets with JCL-specified data sets.

SIZEPCT=(100,100) The SIZEPCT option specifies the percentages of the primary and secondary space calculated by LOADPLUS that will be used for allocation. The numbers that you specify must be greater than zero.

The first number indicates the percentage of the primary quantity calculated by LOADPLUS that you actually want allocated.

The second number indicates the percentage of the secondary quantity calculated by LOADPLUS that you actually want allocated.

UNIT=(SYSALLDA,SYSALLDA)
The UNIT option specifies the primary and secondary unit names that LOADPLUS will use during non-SMS dynamic allocation. These unit names may not exceed eight characters each.

BMC Software, Inc., Confidential and Proprietary Information

THRESHLD=0 The THRESHLD option specifies the threshold, in kilobytes, above which LOADPLUS will use the secondary unit or secondary SMS class for allocation. To prevent LOADPLUS from using secondary units or secondary classes that have been specified (or might be specified in the command options), use a value of 0 for this option.

DSNPAT=&&UID.&&UTILPFX.&&DDNAME

The DSNPAT option specifies the pattern that you want LOADPLUS to use to generate data set names during dynamic work file allocation. The keyword NONE is also valid, but *only* for SORTWK data sets, and it must not be enclosed by single quotation marks (as is required with the DSNPAT command option).

Warning! The pattern that you specify in your DSNPAT option must allow LOADPLUS to generate unique data set names. With multiple SORTOUT and SYSUT1 files, you must include the &DDNAME variable to generate unique names. For copy data sets, you might need to include additional variables, such as &VCAT, &DATEJ, or &TIME4 to generate unique names across multiple loads. If LOADPLUS encounters non-unique data set names, processing terminates.

You can use text or any of the symbolic variables from Table A-7 to construct your pattern. You can also provide user-defined variables from a user exit routine (specified with the DSNUEXIT option in the AMU$OPTS macro or with the DSNUEXIT command option). When specifying a pattern in your installation options, you must precede each variable with an additional ampersand (&) in your pattern.

Note: This pattern can also be specified with the DSNPAT command option and overrides any default that is established in the DSNPAT installation option. Additional ampersands are *not* required with the command option. However, if you use the keyword NONE with the DSNPAT command option, you must enclose it with single quotation marks.

Table A-7 Symbolic Variables for DSNPAT Installation Option (Part 1 of 2)

Symbolic Variable	Definition	Length of Result[a, b]
&DATE	current date (in the form MMDDYY)	6 bytes
&DATEJ	current Julian date (in the form YYYYDDD)	7 bytes
&DB	database containing the space being used for this data set allocation	8 bytes maximum
&DDNAME	ddname being used for this data set allocation	8 bytes maximum
&GRPNM	DB2 data-sharing group name[c]	4 bytes

BMC Software, Inc., Confidential and Proprietary Information

Table A-7 Symbolic Variables for DSNPAT Installation Option (Part 2 of 2)

Symbolic Variable	Definition	Length of Result[a, b]
&JOBNAME	JOB name used in the JCL	8 bytes maximum
&PART	partition being used for this data set allocation	3 bytes
&REPLACE	LOAD REPLACE being done (Y or N)	1 byte
&RESUME	LOAD RESUME being done (Y or N)	1 byte
&SSID	DB2 subsystem ID	4 bytes
&STEPNAME	STEP name used in the JCL[d]	8 bytes maximum
&TIME	current time (in the form HHMMSS)	6 bytes
&TIME4	current time (in the form HHMM)	4 bytes
&TS	table space containing the table specified in your LOAD command	8 bytes maximum
&USERID or &UID [e]	job user ID	8 bytes maximum
&UTIL	BMC Software utility ID	8 bytes maximum[f]
&UTILPFX	BMC Software utility ID prefix	8 bytes maximum
&UTILSFX	BMC Software utility ID suffix	8 bytes maximum
&VCAT	VCATNAME specified in the DB2 catalog for the table space that you are loading; or, if the table space is partitioned, the VCAT name from the first part that you are loading	8 bytes

[a] Any trailing blanks in the result are removed.
[b] The maximum total length allowed for a data set name is 44 bytes.
[c] In a non-data-sharing environment, &GRPNM will contain the DB2 SSID.
[d] PROC names are ignored.
[e] You cannot use this variable unless you have a security package.
[f] Longer utility IDs are truncated to 8 characters.

For your dynamically allocated copy data sets, you can also specify a pattern that contains a GDG name. Each DDTYPE must have a different GDG base.

The GDG format that you use to construct a data set name is the same as the format that you use in your JCL when you allocate copy data sets with DD statements. Simply append the generation number in parentheses. The open parenthesis tells LOADPLUS that the pattern is a GDG name. The generation number must be an integer from 1 through 255.

If the base does not exist, LOADPLUS creates it for you using everything in the pattern up to the open parenthesis as the base name. For more information about GDG names and options, see “Generating Data Set Names” on page 2-16.

BMC Software, Inc., Confidential and Proprietary Information

The following example shows a valid GDG name:

```
'&UTILPFX.&DDNAME..COPY(+1)'
```

If you are using a substitution variable as the last variable before the open parenthesis, you must include a period before the open parenthesis. The following example shows this usage:

```
'&UTILPFX.&DDNAME.(+1)'
```

If you specify COPYLVL PART on the LOAD command, each partition must have a different GDG base. To specify a pattern that includes a partition, the partition must not be in parentheses. The following example shows a valid name:

```
'&UTILPFX.&DDNAME..P&PART.(+1)'
```

You cannot specify a pattern that contains a PDS name. The following example shows an invalid name:

```
'&UTILPFX.&DDNAME..(P&PART)'
```

For additional information and guidelines for specifying data set name patterns, see “DSNPAT” on page 3-145. See Appendix F, “LOADPLUS User Exits,” for further discussion of user exits.

SMS=NO

The SMS option tells LOADPLUS whether to perform an SMS allocation or non-SMS allocation. Valid values are YES and NO.

DATACLAS=(NONE,NONE)

The DATACLAS option specifies the primary and secondary SMS data classes that LOADPLUS will use for an SMS allocation. The two class names must be valid SMS data class names, not exceeding eight characters each, or NONE.

MGMTCLAS=(NONE,NONE)

The MGMTCLAS option specifies the primary and secondary SMS management classes that LOADPLUS will use for an SMS allocation. The two class names must be valid SMS management class names, not exceeding eight characters each, or NONE.

STORCLAS=(NONE,NONE)

The STORCLAS option specifies the primary and secondary SMS storage classes that LOADPLUS will use for an SMS allocation. The two class names must be valid SMS storage class names, not exceeding eight characters each, or NONE.

BMC Software, Inc., Confidential and Proprietary Information

SMSUNIT=NO

The SMSUNIT option tells LOADPLUS whether to pass the UNIT value in the SMS allocation parameter list during dynamic allocation. LOADPLUS does not modify any other parameters based on this option. If the value of the SMS option is NO, LOADPLUS ignores the SMSUNIT option.

Valid values for SMSUNIT are YES and NO.

MAXEXTSZ=0

The MAXEXTSZ option specifies the maximum size, in kilobytes, for any extent that LOADPLUS allocates for the primary and secondary quantities. This option is not valid for a SORTWK file. LOADPLUS ignores MAXEXTSZ if you specify SMS YES.

If you do not want to set a limit for space allocation, specify 0 as the value for MAXEXTSZ. If you specify 0, or if you specify a value that is greater than the size limit of your physical device, LOADPLUS replaces this value at runtime with the size of your physical device.

If LOADPLUS determines that the amount of required space is greater than this value allows (for both primary and secondary quantities), LOADPLUS terminates the job with a return code of 8.

Note: LOADPLUS checks the value of MAXEXTSZ after it applies SIZEPCT to the allocation amount.

EXPDT=

Use EXPDT to specify the expiration date for the copy data set that you are dynamically allocating. The date must either be blank or in the format *yyddd* or *yyyyddd*. *yy* is the last two digits of the year, *yyyy* is the 4-digit year (1900 through 2155), and *ddd* is the 3-digit Julian day (000 through 366). For expiration dates of January 1, 2000 and later, you *must* use the form *yyyyddd*. A blank value means that LOADPLUS does not use an expiration date.

When you specify the EXPDT installation option, it takes precedence over the RETPD installation option. However, you can override the EXPDT installation option by using the EXPDT or RETPD command option. For information about the commands, see “EXPDT” on page 3-151 and “RETPD” on page 3-151.

LOADPLUS ignores this option if you specify it for any *non-copy* data set that you are dynamically allocating.

RETPD=

Use RETPD to specify the retention period in days for the copy data set that you are dynamically allocating. The number of days must be blank or in the range 0 through 9999. A blank value means that LOADPLUS does not use a retention period.

You can override this installation option by using the RETPD command option, which is described on page 3-151. If you specify the EXPDT installation or command option, EXPDT takes precedence over RETPD.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS ignores this option if you specify it for any *non-copy* data set that you are dynamically allocating.

GDGLIMIT=5 The GDGLIMIT option specifies the maximum number of GDG generations that the system should keep. The number must be an integer in the range 1 through 255.

LOADPLUS honors this option only when creating the GDG base. LOADPLUS ignores this option if you specify it for any *non-copy* data set that you are dynamically allocating.

You can override this installation option by using the GDGLIMIT command option, which is described on page 3-152.

GDGEMPTY=NO GDGEMPTY=NO indicates that the system uncatalogs only the oldest GDG data set when the GDGLIMIT is reached.

If you specify GDGEMPTY=YES, the system uncatalogs *all* preexisting generations of this data set when the limit is reached.

LOADPLUS honors this option only when creating the GDG base. LOADPLUS ignores this option if you specify it for any *non-copy* data set that you are dynamically allocating.

GDGSCRATCH=NO

GDGSCRATCH=NO specifies that the system should not delete an entry that is uncataloged as a result of the GDGEMPTY option.

If you specify GDGSCRATCH=YES, the system deletes the GDG entry from the volume's table of contents (VTOC) when uncataloging the data set. The space on the volume is then available to other users.

LOADPLUS honors this option only when creating the GDG base. LOADPLUS ignores this option if you specify it for any *non-copy* data set that you are dynamically allocating.

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Appendix B LOADPLUS Database

This appendix presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Overview

This appendix describes the DB2 tables that the LOADPLUS product uses:

- The BMCUTIL table provides information about currently running BMC Software utilities.
- The BMCSYNC table contains information about the DB2 objects on which the BMC Software utilities are operating.
- The BMCHIST table provides information about BMC Software utilities that have successfully completed execution.
- The BMCDICT table holds the compression dictionary. BMCDICT is a temporary work area.

You can access these tables through SQL statements or by using BMCDSN, the BMC Software command processor for the BMC Software utilities for DB2. See the *BMCDSN Command Processor Reference Manual* for information about using BMCDSN and the *Utility Products for DB2 Customization Guide* for information about installing BMCDSN.

With the exception of BMCHIST, the utility tables do not require maintenance. When LOADPLUS completes successfully, a row is inserted into the BMCHIST table. Periodically, you should review BMCHIST and delete old rows to control its expansion.

Warning! Note the following warnings:

— Do not run the BMC Software utilities against BMC Software tables or table spaces. Doing so can produce unpredictable results.

— Do not run the RUNSTATS utility against the BMC Software common utility tables. Doing so can negatively impact utility performance.

— BMC Software strongly recommends that you use isolation UR and issue SQL COMMIT statements when querying the objects in the BMC Software database. If objects in the BMC Software database are restricted for UPDATE, the BMC Software utilities that are executing might not be able to complete successfully.

BMC Software, Inc., Confidential and Proprietary Information

BMCUTIL Table

Table B-1 describes the BMCUTIL table, which contains information about utilities that are currently running or started. All BMC Software DB2 utility products use this table.

Table B-1 BMCUTIL Table

Column Name	Data Type	Description
UTILID	CHAR(16)	utility identifier
STATUS	CHAR(1)	execution status of the utility: **A** (active, not executing command) **I** (initializing) **P** (pausing or pause-stopped) **S** (stopped) **T** (terminating) **X** (executing command)
UTILNAME	CHAR(8)	name of the running utility (LOAD)
PHASE	CHAR(8)	current phase of the utility
USERID	CHAR(8)	user ID running the utility
SSID	CHAR(4)	DB2 subsystem where the utility is running
RESTART	CHAR(1)	restart option: **N** (not restart) **P** (RESTART(PHASE)) **Y** (RESTART)
NOTEID	CHAR(8)	TSO user ID to be notified
DBNAME	CHAR(8)	not used
SPNAME	CHAR(8)	not used
SPSTATUS	CHAR(5)	not used
COMMANDNO	SMALLINT	current command number (always 0)
COMMAND	VARCHAR(256)	first 256 characters of the utility command text
STATE	LONG VARCHAR	utility state and sync information
START_TIMESTAMP	TIMESTAMP	starting timestamp of the utility

BMC Software, Inc., Confidential and Proprietary Information

BMCSYNC Table

Table B-2 describes the BMCSYNC table, which contains information about the status of the objects that the currently running utilities are accessing. The BMCSYNC table synchronizes access to DB2 spaces by concurrently running BMC Software utility products. For more information, see “Serialization and Concurrency” on page 2-7.

Table B-2 BMCSYNC Table

Column Name	Data Type	Description
UTILID	CHAR(16)	utility identifier
NAME1	CHAR(8)	database name or creator name
NAME2	CHAR(18)	space, table, or index name
KIND	CHAR(2)	kind of object: • **IP** (index partition) • **IX** (index) • **TB** (table) • **TP** (table space partition) • **TS** (table space) • **DD, DW** (dynamic work file allocation) • **CI** (copy information) • **RD** (restart data set block)
PARTITION	SMALLINT	null or zero for nonpartitioned space or the partition number for rows that represent partitions
BMCID	SMALLINT	internal identifier of the object
UTILNAME	CHAR(8)	name of the running utility
SHRLEVEL[a]	CHAR(1)	degree to which this object can be shared with other BMC Software utility products • Blank means that no status is requested and any other utility can obtain any status. • *S* allows sharing among any number of *S* SHRLEVEL utilities. • *X* indicates that exclusive control is required. No other utility can run with *X* SHRLEVEL.
STATUS	CHAR(1)	status of the utility or object
XCOUNT	INTEGER	number of rows or keys processed in the current phase
DDNAME	CHAR(8)	load or work ddname
BLOCKS	INTEGER	number of blocks for the load or work data set
ORIG_STATUS	CHAR(8)	encoded representation of the original DB2 status of the space
EXTRBA	CHAR(6)	not used
STATE	LONG VARCHAR	restart information for the space

[a] For LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY, LOADPLUS registers objects with a SHRLEVEL of *S*. For all other load types, LOADPLUS registers objects with a SHRLEVEL of *X*.

BMC Software, Inc., Confidential and Proprietary Information

BMCHIST Table

Table B-3 describes the BMCHIST table, which contains information about completed executions of the BMC Software DB2 utilities. LOADPLUS and other BMC Software DB2 utility products share this table.

Table B-3 BMCHIST Table

Column Name	Data Type	Description
DBNAME	CHAR(8)	name of the database that contains the table or index space
SPNAME	CHAR(8)	name of the table or index space
UTILNAME	CHAR(8)	name of the utility (LOAD)
UTILID	CHAR(16)	utility identifier
AUTHID	CHAR(8)	user ID that ran the utility
DATE	DATE	date that the utility completed
TIME	TIME	time that the utility completed
ELAPSED	TIME	elapsed time of the utility
PARTITION	LONG VARCHAR	partition numbers as specified on the PART option or **ALL**
OBJNAME	VARCHAR(27)	fully qualified object name
PHASE_1	CHAR(8)	name of utility phase 1
ELAPSED_1	TIME	elapsed time of phase 1
PHASE_2	CHAR(8)	name of utility phase 2
ELAPSED_2	TIME	elapsed time of phase 2
PHASE_3	CHAR(8)	name of utility phase 3
ELAPSED_3	TIME	elapsed time of phase 3
PHASE_4	CHAR(8)	name of utility phase 4
ELAPSED_4	TIME	elapsed time of phase 4
PHASE_5	CHAR(8)	name of utility phase 5
ELAPSED_5	TIME	elapsed time of phase 5

The following SQL statement shows how to access (from the BMCHIST table) the database name, the table space name, the elapsed time of the utility, and the time in which the utility completed. You can use the format of this example to write other SQL statements for accessing the table.

```
SELECT DBNAME,SPNAME,CHAR(ELAPSED,ISO),CHAR(TIME,ISO)
FROM authid.BMCHIST WHERE
UTILID='utilid'
```

BMC Software, Inc., Confidential and Proprietary Information

The following SQL statement shows how to delete selected rows from the BMCHIST table based on the date that the utility completed. You can use this sample format to write your own SQL statements to delete BMCHIST rows.

```
DELETE
FROM authid.BMCHIST WHERE
DATE < 'yyyy-mm-dd'
```

BMCDICT Table

Table B-4 describes the BMCDICT table, which stores the compression dictionary during load processing. REORG PLUS also uses this table.

Table B-4 BMCDICT Table

Column Name	Data Type	Description
UTILID	CHAR(16)	utility identifier
DBNAME	CHAR(8)	database name or creator name
TSNAME	CHAR(8)	table space name
PARTITION	SMALLINT	partition number (0 if the table space is nonpartitioned)
SEQNO	SMALLINT	sequence number
DICTDATA	VARCHAR(4000)	dictionary data

BMC Software, Inc., Confidential and Proprietary Information

Appendix C LOADPLUS Messages and Codes

This appendix presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Online Documentation for Messages

The LOADPLUS product provides several options for viewing message information online. This section describes these options.

Chicago-Soft MVS/QuickRef

If your site has installed MVS/QuickRef from Chicago-Soft, you can use it to view messages for this product and other BMC Software products.

Note: For MVS/QuickRef version 5.7 or earlier, if you specify a message number with the WER prefix, MVS/QuickRef provides information about the standard SyncSort message, not the corresponding BMCSORT message.

BMC Software Online Message Processor

Installing the BMC Software online message processor is an option during installation of this product. If the online message processor was installed, you can access message information for this product directly from a TSO command line.

Note: If the BMCMSG CLIST was not copied from the *HLQ*.INSTALL library (where *HLQ* is the high-level qualifier specified during installation) to a library in the SYSPROC concatenation when the product was installed, BMC Software recommends that you do so.

To access message information, issue the following command:

```
TSO BMCMSG msg_ID
```

In this command, *msg_ID* is the message identifier (the BMC prefix followed by the unique message number). You can include or omit the severity code.

Alternatively, you can substitute an asterisk for the message identifier. If you supply an asterisk, the message processor displays syntax information and examples, and prompts you for a message identifier.

Note: You cannot use the online message processor to access information about BMCSORT messages.

BMC Software, Inc., Confidential and Proprietary Information

MVS Data Set

The installation process automatically downloads an MVS data set that contains the messages for all BMC Software utility products on the distribution tape. The messages for each product are in separate members. For the LOADPLUS product, the messages are in member AMU*vrmm* of the *HLQ*.MSGS installation data set.

To view the messages information, simply browse the data set.

Return Codes

Table C-1 describes the return codes that indicate the completion status of LOADPLUS.

Table C-1 LOADPLUS Return Codes

Return Code	Description
0	LOADPLUS executed successfully with no warnings or errors.
4	LOADPLUS executed successfully but warnings were issued, or LOADPLUS execution was incomplete. A pause after a specific phase causes incomplete execution.
8	LOADPLUS execution failed. A return code 8 usually indicates an error in the utility command or JCL.
12	LOADPLUS execution failed because it received an error from the system or DB2. These errors include DB2 Call Attachment Facility (CAF) errors, unexpected nonzero SQL codes, and data set open errors.
16	LOADPLUS execution failed while the utility was trying to load one of its modules. LOADPLUS might have been installed incorrectly.
other	LOADPLUS execution stopped, and you used a command option to override the default return code.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS Abend 3400 Reason Codes

LOADPLUS can abend or terminate with user abend code 3400 if certain errors or conditions occur that LOADPLUS does not expect. You can find the reason code in register 15 of the job log that contains the abend summary. Table C-2 lists examples of these unexpected errors or conditions.

Note: BMCSORT messages can produce additional abend codes. If you receive an abend code with a BMCSORT message (prefixed WER), contact BMC Software Customer Support.

Table C-2 LOADPLUS Abend 3400 Reason Codes

Reason Code	Description
1 (X'01')	The LOADPLUS SYNC subtask encountered an unrecoverable error. See accompanying messages.
3 (X'03')	LOADPLUS reached an end-of-file indication before the last segment of a spanned record. You might have used an incorrect file to restart LOADPLUS. If this is not the case, contact BMC Software Customer Support.
4 (X'04')	No sequential I/O buffers are available. Ensure that the LOADPLUS installation options specify the appropriate number of buffers for your data sets. If you specified the appropriate number of buffers, contact BMC Software Customer Support.
5 (X'05')	LOADPLUS encountered an unexpected error condition. This reason code usually indicates an error or mismatch in the table structure, data, or EDITPROC. Message BMC50498U or BMC50499U usually accompanies this abend. Verify your table structure and data. For further assistance, contact BMC Software Customer Support.
7 (X'07')	The number of SORTWK data sets available is fewer than the minimum required for this load job. To avoid this abend, specify a value for the SORTNUM command option that is greater than 1.
8 (X'08')	LOADPLUS encountered an unexpected error condition while trying to set or reset a DB2 pending status, such as COPY pending (COPY), CHECK pending (CHKP), or RECOVER pending (RECP). Message BMC50498U usually accompanies this abend. Check the system log and DB2MSTR address space for additional error messages that might help determine the cause of the problem. Restart the utility and allow it to complete. For further assistance, contact BMC Software Customer Support.
16 (X'10')	There is probably an error in BMCSORT. Check UTPRINT for additional messages.
100 (X'64')	The user canceled the job.
other	Contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

Message Severity Codes

Each BMC Software message number ends with a letter that indicates its severity. Different message severity codes require different responses. Table C-3 describes the standard message severity codes, listed by increasing severity.

Note: BMCSORT message numbers (numbers with the prefix WER) also end with a letter that indicates the severity and type of error; however, these codes differ from the standard BMC Software severity codes. See the *BMCSORT Reference Manual* for information about these messages.

Table C-3 Message Severity Codes

Severity Code	Meaning	Description
I	Informational	LOADPLUS issues this message for informational purposes only. Receiving this message does not mean that an error occurred.
W	Warning	LOADPLUS encountered a condition that allows it to proceed, but you might need to take additional action.
E	Error	An error occurred that is probably a command error, JCL specification error, or other error that requires your attention.
S	Severe	LOADPLUS encountered an unexpected processing error that does not preclude the utility from terminating in an orderly fashion.
U	Unrecoverable	LOADPLUS encountered an error that makes it impossible to continue processing (for example, an abnormal termination).

Message Groups

The remainder of this appendix lists and briefly explains each message that you can receive from LOADPLUS. The messages are listed in numeric order and are divided into the following groups:

- common utility messages (beginning on page C-6)
- LOADPLUS messages (beginning on page C-126)
- ICOPY installation messages (beginning on page C-162)

Note: For information about BMCSORT messages (messages with a WER prefix), see the *BMCSORT Reference Manual*.

BMC Software, Inc., Confidential and Proprietary Information

The explanation and user response information that is given for a LOADPLUS message applies only to the LOADPLUS product. In contrast, the explanation and user response information that is provided for a common utility message can also apply to any one or more of the following BMC Software products:

- REORG PLUS for DB2
- UNLOAD PLUS for DB2
- CHECK PLUS for DB2
- BMCDSN Command Processor

Common Utility Messages

When a message contains text that can vary, the type of variable is shown in lowercase letters, such as *nn* for *number*. The actual message contains specific text in place of the variables shown in this chapter.

BMC50001I UTILITY EXECUTION STARTING *date time*...

Explanation: This message is the start message for the utility.

User Response: No action is required.

BMC50002I UTILITY ID = '*utilid*'. DB2 SUBSYSTEM ID = '*ssid*'. OPTION MODULE = '*module*'.

Explanation: This message confirms the utility ID, DB2 subsystem ID, and installation options module that you specified. If you specified a group attachment name in your EXEC statement parameters, that name is the DB2 subsystem ID. For details about specifying these parameters with your EXEC statement, see Chapter 4 of the reference manual for the utility you are using.

User Response: No action is required.

BMC50003I RESTARTING UTILITY: COMMAND = '*command*', PHASE = '*phase*'

Explanation: The utility that the command names is restarting at the named phase.

User Response: No action is required.

BMC50004I *phase* PHASE COMPLETE. ELAPSED TIME = *time*

Explanation: The named phase completed with the elapsed time that the message states.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50005I ***utilname* COMMAND EXECUTION PAUSING BEFORE PHASE = '*phase*'**

Explanation: The named utility execution is pausing as the utility command requested.

User Response: After modifying your JCL, reallocating DB2 data sets, or changing the utility command, use the RESTART utility parameter and restart the job.

BMC50006I **UTILITY EXECUTION COMPLETE, RETURN CODE = *rc***

Explanation: The utility execution completed.

User Response: If the return code is not zero, check the prior messages to determine the problem. Otherwise, no action is required.

BMC50007I **OBJECT(S) RESET TO ORIGINAL STATUS; STOPPED OUTAGE ELAPSED TIME = *time***

Explanation: This message indicates the length of time that the objects were in STOP status for a SHRLEVEL CHANGE job. The objects are reset to their original status.

User Response: No action is required.

BMC50009S **UNABLE TO RESTART UTILITY IN '*phase*' PHASE DUE TO PREVIOUS WARNING OR ERROR**

Explanation: Due to a previously issued warning or error, the job cannot be restarted.

User Response: Check prior warning or error messages or the job log to determine the problem. If you understand the problem, correct it and resubmit the job with 'NEW'. Otherwise, you must recover the table space and any associated indexes and then resubmit the job.

BMC50011E **UTILITY WITH ID '*utilid*' ALREADY EXISTS**

Explanation: The utility ID that you specified is in use.

User Response: If you are restarting the utility, specify the RESTART utility parameter. Otherwise, choose a different utility ID.

BMC Software, Inc., Confidential and Proprietary Information

BMC50012E **UTILITY WITH ID '*utilid*' CURRENTLY EXECUTING**

Explanation: An executing utility is currently using the utility ID that you specified.

User Response: If you are restarting the utility or if you specified NEW and you know the utility is not currently running, another utility with the same ID is already running. Choose another ID if you are not trying to restart an existing utility.

If you know that a utility with the specified utility ID is not currently running and you want to restart the utility, perform the following steps:

1. Use BMCDSN to issue a +RESET command to set UTILID to a stopped status. Alternatively, you can use the following SQL statement to update the STATUS column to S for the row in the BMCUTIL table where the UTILID column equals the specified UTILID.

   UPDATE *authid*.BMCUTIL SET STATUS='S' WHERE UTILID='*utilid*'

2. Restart the utility, specifying the same utility ID.

If you want to terminate the utility, use the TERM keyword for the RESTART utility parameter. See the reference manual for the utility that you are running for information about terminating a utility.

BMC50013I **UTILITY EXECUTION TERMINATING, RETURN CODE = *rc***

Explanation: The execution of the utility ended with an error.

User Response: Check prior messages to determine the problem. Correct it the problem and restart the utility.

BMC50014U **UTILITY EXECUTION ABENDED, SYSTEM CODE = *abend_code*, REASON CODE = *reason***

Explanation: The execution of the utility abended.

User Response: Check prior messages or the job log to determine the problem. If you understand the problem (for example, your job was canceled because of insufficient time), correct it and restart the utility.

BMC50015U **UTILITY EXECUTION ABENDED, USER CODE = *abend_code*, REASON CODE = *reason***

Explanation: The execution of the utility abended.

User Response: Check prior messages or the job log to determine the problem. If you understand the problem (for example, an SQL time-out), correct it and restart the utility. For other errors, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50016E/I ***utility_name*** **UTILITY WITH ID '*utilid*' NOT FOUND IN BMCUTIL/BMCSYNC**

Explanation: You specified either the RESTART (severity code E) or the TERM (severity code I) parameter. Either the utility ID that you specified as the UTILID parameter cannot be found in the BMCUTIL or BMCSYNC table, or the utility ID belongs to another utility.

User Response: Ensure that you specified the correct utility ID and restart your job.

BMC50017E **RESTART OF '*utilname*' UTILITY ATTEMPTED WITH UTILITY COMMAND '*command*'**

Explanation: You restarted the utility with a command that is different from the command that you used for the original execution of the utility.

User Response: Submit the correct utility program with the correct utility command to restart the utility.

BMC50018I **UTILITY CANCELLED BY USER**

Explanation: Deleting the row for this utility ID in the BMCUTIL table canceled the utility.

User Response: UNLOAD PLUS users do not need to act because the DB2 table or table space that the utility was unloading was not affected. For all other utilities, see the associated reference manual for information about making the DB2 object usable after terminating or canceling the utility.

BMC50019I **UTILITY REQUESTED TO TERMINATE DUE TO FAILURE IN PHASE '*phase*'**

Explanation: The utility is abnormally terminating, and you specified the ON FAILURE phase TERMINATE option. All spaces were started (when appropriate), and the utility is removed from the BMCUTIL table.

User Response: No action is required.

BMC50020I **UTILITY TERMINATED DUE TO '*option*' SPECIFICATION**

Explanation: The specified option caused the utility job to end.

User Response: No action is required.

BMC50021E **UTILITY PARM '*parm*' IS LONGER THAN *n* CHARACTERS**

Explanation: The utility parameter is longer than the maximum number of allowed characters.

User Response: Correct the parameter and resubmit the utility job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50022I **SPECIFIED *parm* IS NULL. USING DEFAULT.**

Explanation: Because you did not specify the named parameter, the utility is using the default value. See the reference manual for the utility that you are running for more information about utility parameters.

User Response: No action is required. Refer also to message BMC50002I.

BMC50023E **INVALID UTILITY PARM: '*parm*'**

Explanation: You specified the named utility parameter incorrectly.

User Response: Correct the parameter and resubmit the job.

BMC50024I **CONNECTED TO SUBSYSTEM '*ssid*'**

Explanation: If you specified a group attachment name in your EXEC statement parameters and are running in a data-sharing environment, this message tells you the actual subsystem to which you are connected within the group. This message follows BMC50002.

User Response: No action is required.

BMC50031I ***utilname period_type* PERIOD WILL EXPIRE IN *n* DAYS REASON CODE=*reason***

Explanation: This message is displayed during the last 30 days before the expiration date that was set when the tape was created. The *period_type* is either GRACE or TRIAL.

User Response: Contact a BMC Software sales representative for assistance.

BMC50032E ***utilname period_type* PERIOD HAS EXPIRED**

Explanation: Either your GRACE or TRIAL period expired.

User Response: Contact a BMC Software sales representative for assistance.

BMC50033E/W **CPU NOT AUTHORIZED TO RUN *utilname* RETURN CODE=*rc* REASON CODE=*reason***

User Response: Contact a BMC Software sales representative for assistance.

BMC50034E ***utilname* NOT ACTIVATED**

User Response: Contact a BMC Software sales representative for assistance.

BMC50035E **INCONSISTENCY FOUND IN SECURITY CHECK RETURN CODE=*rc* REASON CODE=*reason***

User Response: Contact BMC Software Customer Support for assistance.

BMC Software, Inc., Confidential and Proprietary Information

BMC50036E NOT APF AUTHORIZED

Explanation: This warning message indicates that the utility is not running as an MVS-authorized program. The utility must run from a library that is an Authorized Program Facility (APF) library. During installation, the ICOPY job (image copy) can also produce this message.

User Response: Contact your system programming staff to copy the utility to an APF-authorized library.

BMC50037E CANNOT ESTABLISH SECURITY ENVIRONMENT

User Response: Contact BMC Software Customer Support for assistance.

BMC50038E THIS RELEASE OF *utility* DOES NOT SUPPORT RELEASES OF DB2 PRIOR TO V*n*

Explanation: You ran the utility against a version of DB2 that this release does not support.

User Response: Refer to the documentation for the version of the utility that you are running to determine the versions of DB2 that this release supports.

BMC50039E THIS RELEASE OF *utility* DOES NOT SUPPORT RELEASES OF DB2 AFTER V*n*

Explanation: You ran the utility against a version of DB2 that this release does not support.

User Response: Refer to the documentation for the version of the utility that you are running to determine the versions of DB2 that this release supports.

BMC50051E *utility* NOT ALLOWED ON SYSTEM OBJECT '*dbname.spname*'

Explanation: You specified a DB2 system table, table space, or index in the command statement.

User Response: You cannot use this utility to perform the specified action on a DB2 system object.

BMC50052E *user* DOES NOT HAVE SUFFICIENT PRIVILEGES TO EXECUTE '*utility*' ON DATABASE '*dbname*'

Explanation: You need sufficient authorization to run the utility on the database that the message names. Message BMC50054E, BMC50053E, or both may accompany this message.

User Response: See the reference manual for the utility that you are running for information about the authorization that you need to run the utility.

BMC Software, Inc., Confidential and Proprietary Information

BMC50053E **USER DOES NOT HAVE '*privilege*' PRIVILEGE ON TABLE '*creator.tbname*'**

Explanation: You (and all of your secondary authorization IDs) need the named privilege to perform the utility on the named table.

User Response: See the reference manual for the utility that you are running for information about the privileges that you need.

BMC50054E **USER DOES NOT HAVE LOAD PRIVILEGE ON TABLE '*creator.tbname*' FOR LOAD REPLACE OF TABLESPACE '*dbname.tsname*'**

Explanation: If you use LOAD REPLACE, you (or one of your secondary authorization IDs) must own all tables in the table space. See Chapter 2 of the *LOADPLUS for DB2 Reference Manual* for information about the privileges that you need to load a table.

User Response: Obtain the necessary authorization and resubmit the job, or delete all rows in your table and load with RESUME YES.

BMC50101E/W **EXPRESSION TOO COMPLEX**

Explanation: The conditional expression that you specified in the command statement contains too many parentheses or is too long.

User Response: If the message severity code is error (E), simplify the expression or make it shorter. If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50102I ***utility_command_line***

Explanation: This message echoes the command statement input lines that the utility read from SYSIN or the ALTER statements that are read from a DDLIN data set.

User Response: No action is required.

BMC50103E/W **CHARACTER '*c*' IS NOT ALLOWED IN COMMAND**

Explanation: The specified character is not valid for a utility command.

User Response: If the message severity code is error (E), correct the error and resubmit the job. If any part of the object name that is input to the utility contains a special character and you get this message, enclose that part of the name within double quotation marks (for example, "dbn!ame".tsname) and resubmit the job.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC Software, Inc., Confidential and Proprietary Information

BMC50104I/E **UNEXPECTED TOKEN '*token*' ENCOUNTERED IN COMMAND**

Explanation: If the message severity code is error (E), you may have misspelled a keyword or used invalid syntax for the utility command. The utility terminates the job.

If the message severity code is informational (I), you specified a DDLIN data set that is empty. The utility ignores the data set and continues processing the job.

User Response: For severity code E, ensure that you spelled the utility command keywords correctly and specified them in the order shown in the syntax diagram. Correct the error and resubmit the job.

For severity code I, add to the DDLIN data set the statements that you want processed and resubmit the job.

BMC50105E/W **PARAMETER '*parm*' IS NOT IN RANGE OF VALUES FOR OPTION '*option*'**

Explanation: The parameter value that you entered is not valid for this option.

User Response: If the message severity code is error (E), correct the error and resubmit the job.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50106E **PARTITION NUMBER '*p*' NOT IN RANGE 1 TO *n***

Explanation: The value for *n* is 64 for non-LARGE table spaces. It is 254 for LARGE table spaces.

User Response: Use a partition number from 1 through *n*, where *n* is the maximum number of partitions for your LARGE (254) or non-LARGE (64) table space, and resubmit the job.

BMC50107E **DUPLICATE PARTITION NUMBER '*p*' SPECIFIED**

Explanation: You specified the same partition number twice.

User Response: Remove the duplicate partition number and resubmit the job.

BMC50108I **PARAMETER '*parm*' IS IGNORED FOR OPTION '*option*'**

Explanation: You specified an option that the utility does not use. However, the option named in the message is allowed for compatibility with the corresponding IBM utility.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50109I **'*option_1*' OPTION SPECIFIED. OPTION '*option_2*' IGNORED**

Explanation: You specified *option_2*, which is meaningless because you also specified *option_1*.

User Response: No action is required.

BMC50110E **INVALID VALUE '*value*' SPECIFIED FOR '*option*' OPTION**

Explanation: You specified an incorrect value for the indicated option.

User Response: Correct the value for the option and resubmit the job.

BMC50111E/W **INVALID STRING *string***

Explanation: The specified string is either empty or has unpaired quotation marks.

User Response: If the message severity code is error (E), correct the string and resubmit the job. If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50112E **'*option*' OPTION NOT SPECIFIED**

Explanation: You must specify the identified option at least once in the command statement.

User Response: Specify the option and resubmit the job.

BMC50113E **'*option*' OPTION HAS CHANGED, BUT IS NOT ALLOWED TO CHANGE ON RESTART OF PHASE = '*phase*'**

Explanation: You changed an option before restarting the job, and you are not allowed to change it when restarting the specified phase. If the specified option is one that is associated with creating an index with REORG PLUS, see Chapter 4 for additional information.

User Response: Change the option back to its original value and restart the job.

BMC50114I **OPTION '*option*' IS NO LONGER SUPPORTED AND IS IGNORED**

Explanation: You specified an option that the utility no longer supports. See the reference manual for the utility that you are running for information about currently supported options.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50115E **'*option_1*' OPTION SPECIFIED. OPTION '*option_2*' CANNOT BE USED**

Explanation: You specified *option_1*, which is incompatible with *option_2*. See the reference manual for the utility that you are running for information about currently supported options.

User Response: Decide which action you want and resubmit the job.

BMC50116I **OPTION '*option_1*' IS IGNORED. OPTION '*option_2*' IS ASSUMED**

Explanation: You specified *option_1*, which the utility ignored and replaced with *option_2*.

User Response: No action is required.

BMC50117E/W **INVALID DBCS/DELIMITED IDENTIFIER *identifier***

Explanation: The DBCS/delimited identifier is invalid.

User Response: If the message severity code is error (E), correct the problem with the identifier and resubmit the job. If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50118E **DUPLICATE DDNAME '*ddname*' FOUND IN DDN OPTIONS**

Explanation: The utility found that two or more identical ddnames that you specified in the DDN default name override command option(s).

User Response: Ensure that each ddname that you specified in the DDN options is unique and has a matching ddname that you specified in your JCL.

BMC50119E **'*option_1*' SPECIFIED. '*option_2*' REQUIRED**

Explanation: You specified *option_1*, which has a requisite specification of *option_2*. See the reference manual for the utility you are running for information about option specification requirements.

User Response: Correct your command statement and resubmit the job.

BMC50120E **'*option*' OPTION HAS CHANGED, BUT IS NOT ALLOWED TO CHANGE ON 'RESTART' OF PHASE = '*phase*', USE 'RESTART(PHASE)'**

Explanation: Before restarting your job, you changed the value of the specified option. You cannot change this particular option before restarting this phase with the RESTART parameter.

User Response: If the change to the option is required, use RESTART(PHASE) to restart your job. Otherwise, revert to the original value of the option and RESTART.

BMC Software, Inc., Confidential and Proprietary Information

BMC50121E/W **LENGTH OF COLUMN '*colname*' IS TOO LONG FOR '*comparison*' COMPARISON**

Explanation: The column that you specified in the comparison is longer than the maximum length that the version of DB2 that you are using allows.

User Response: If the message severity code is error (E) and you specified the wrong column, correct the column and resubmit the job. If you did not specify the wrong column, no further action is required, because the utility does not support a comparison on a column whose length is greater than the length that the version of DB2 that you are using allows.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50122E/W **FOR '*option*' COLUMN '*colname*', CONSTANT IS WRONG TYPE OR COLUMN IS NOT NULLABLE: '*constant*'**

Explanation: The data type of the constant that you specified is not compatible with the data type of the column, or you specified NULL for a column that is defined as NOT NULL.

User Response: Refer to the description of the named option in the reference manual for the utility that you are running. If the message severity code is error (E), correct the constant and resubmit the job.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50123E/W **FOR '*option*' COLUMN '*colname*', CONSTANT IS NOT IN RANGE OF COLUMN DATA TYPE: '*constant*'**

Explanation: The constant that you specified is too small, too large, or too long for the data type of the column.

User Response: If the message severity code is error (E), correct the constant and resubmit the job. For LOADPLUS users, if you are using a field specification with a VALUE clause, the constants must match the columns' data types (numeric to numeric, string to string, and date/time to date/time). If the constants do not match, you must use a (*start:end*) for the predicate instead of a column or field name.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC Software, Inc., Confidential and Proprietary Information

BMC50124E/W **INVALID VALUE '*value*' SPECIFIED FOR '*option*' OPTION, COLUMN '*colname*'**

Explanation: The value of the option that you specified is incorrect. For example, the end position in POSITION(*start*:*end*) must be greater than or equal to the start position. Also, the length derived from (*start*:*end*) must match the explicit or implicit length of the input data type.

User Response: Refer to the description of the named option in the reference manual for the utility you are running. If the message severity code is error (E), correct the option value and resubmit the job.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50125E/W **TYPE OF COLUMN '*column*' IS NOT ALLOWED FOR '*option*' OPTION**

Explanation: The utility cannot use the type of column that you specified with the named option. See the reference manual for the utility that you are running for information about the column types that are allowed with this option.

User Response: If the message severity code is error (E), choose a different column, if possible, and resubmit the job. If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50127E **'FORMAT CSV' WAS SPECIFIED WITH THE 'TERMINATEDBY' CHARACTER EQUAL TO ONE OF THE 'ENCLOSEDBY' CHARACTERS**

Explanation: With the FORMAT CSV option, you specified the same character for your field delimiter (the TERMINATEDBY option) as you did for one or both of the characters enclosing your data (the ENCLOSEDBY and AND options).

User Response: Change the specifications in your utility command to ensure that your TERMINATEDBY value is different from both your ENCLOSEDBY and AND values. Resubmit your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50128E **LENGTH OF *type* STRING EXCEEDS MAXIMUM ALLOWED LENGTH OF *n_bytes***

Explanation: While processing DDL input, the utility encountered a string constant that exceeded the maximum length that the utility grants for the string type. The string could be too long, an apostrophe could be missing from the statement, or the statement might include an inadvertent typographical error. String constants in DDL input must conform to the following rules:

- A character string cannot exceed a length of 255 bytes.
- A hexadecimal string cannot exceed a length of 254 hexadecimal digits.
- A double-byte character set (DBCS) identifier string cannot exceed a length of 124 double-bytes.
- A delimited identifier string cannot exceed a length of 255 bytes.
- A graphic string cannot exceed a length of 124 double-bytes.

User Response: Ensure that the string that is named in the message is the correct length for its type, and ensure that you punctuated the input statement correctly. Resubmit the job.

BMC50129E **TOKEN THAT EXCEEDS MAXIMUM SIZE CAUSED PRE-PARSER BUFFER TO OVERFLOW PAST 32768 BYTES**

Explanation: The utility encountered invalid DDL syntax such as a keyword, user token, or character, graphic, or hexadecimal string constant that exceeds the maximum size and cannot fit into the pre-parser buffer.

User Response: Correct the DDL as follows and restart the job.

- Ensure that the DDL includes opening and closing apostrophes or opening and closing quotes that match.
- On a MIXED=YES system, ensure that a shift-out (X'0E') character is paired with a shift-in (X'0F') character in the first of a subsequent double-byte pair. (An X'0F' character in the second of a double-byte pair is not considered a shift-in character.)

BMC50131E **'ORDER YES' OPTION REQUIRED FOR A MULTI-TABLE SEGMENTED TABLESPACE**

Explanation: You cannot use ORDER NO when you are reorganizing or loading a multitable, segmented table space.

User Response: Specify ORDER YES and resubmit the job. For REORG PLUS users, you could alternatively remove the ORDER option from the command statement, because ORDER YES is the default.

BMC Software, Inc., Confidential and Proprietary Information

BMC50132E **'ASSOCIATE BYCLUSTERKEY' CANNOT BE USED FOR A MULTI-TABLE SEGMENTED TABLESPACE**

Explanation: You cannot specify ASSOCIATE BYCLUSTERKEY in the command statement when you are reorganizing or loading a multitable, segmented table space.

User Response: Specify ASSOCIATE BYTABLE or remove the ASSOCIATE option from the command statement, because ASSOCIATE BYTABLE is the default for ORDER YES, and resubmit the job. For REORG PLUS users, you could alternatively remove the ORDER and ASSOCIATE options from the command statement because ORDER YES ASSOCIATE BYTABLE is the default.

BMC50133I **'ORDER YES' IS MEANINGLESS FOR A SINGLE-TABLE TABLESPACE WITHOUT A CLUSTERING INDEX; CONVERTING TO ORDER NO**

Explanation: Because you have a single-table table space with no clustering index, the utility has no ordering to perform. Therefore, ORDER YES has no meaning.

Note: LOADPLUS issues this message for the table it is loading, not the table space.

User Response: No action is required.

BMC50134E **'ASSOCIATE BYCLUSTERKEY' IS MEANINGLESS WITHOUT ANY CLUSTERING INDEXES**

Explanation: Because you do not have any clustering indexes, ASSOCIATE BYCLUSTERKEY has no meaning.

User Response: Specify ASSOCIATE BYTABLE or remove the ASSOCIATE option from the command statement because ASSOCIATE BYTABLE is the default for ORDER YES, and resubmit the job. For REORG PLUS users, you could alternatively remove the ORDER and ASSOCIATE options from the command statement because ORDER YES ASSOCIATE BYTABLE is the default.

BMC50135E **'*option*' OPTION HAS CHANGED, BUT IS NOT ALLOWED TO CHANGE ON RESTART OF PHASE = '*phase*' FOR DDTYPE *ddtype***

Explanation: Before restarting your job, you changed the value of the specified option. This particular option cannot change before you restart this phase. The *ddtype* identifies which specification of the option that you need to revise.

User Response: Revert to the original value of the option and RESTART.

BMC Software, Inc., Confidential and Proprietary Information

BMC50136I **DEFAULT *option_1* INCOMPATIBLE WITH DEFAULT *option_2*; CHANGED TO *option_3***

Explanation: Two installation options were specified with incompatible values. The message displays the options and their values. The utility changed *option_1* to *option_3*.

User Response: No action is required.

BMC50137E **'CENTURY' PARAMETER VALUES '*ccyy*' AND '*ccyy*' ARE INVALID**

Explanation: The values you specified with the CENTURY option on the command or as part of the field specification are invalid. The first value is larger than the second value, the two values do not span 100 years, or at least one of the values specified is in an invalid format.

User Response: Correct the values for the CENTURY option and resubmit the job.

BMC50138I **'*keyword_string1*' CHANGED TO '*keyword_string2*' DUE TO '*reason*'**

Explanation: The first combination of keywords has been changed to the second string for the specified reason. For more information, see the description of the keywords in chapter 3 or in Appendix A of the reference manual for this utility. Processing continues.

User Response: No action is required.

BMC50139I **'*keyword1*' IS OBSOLETE AND HAS BEEN CHANGED TO '*keyword2*'**

Explanation: The first keyword is no longer supported by this release of the product. It has been changed to the second keyword. Processing continues.

User Response: No action is required.

BMC50140I **'*keyword_string1*' HAS BEEN CHANGED TO '*keyword_string2*'**

Explanation: The keywords in *keyword_string1* are not compatible. They have been changed to the keywords in *keyword_string2*. For more information about which keywords can be specified together, see chapter 3 of the reference manual for this utility.

User Response: No action is required.

BMC50141I **'*option*' IGNORED DUE TO *reason***

Explanation: The utility ignored the option for the reason given. For example, REORG PLUS ignores the KEEPDICTIONARY option if the table space is in REORG pending status (REORP).

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50142E **OUTPUT DESCRIPTOR '*descriptor*' HAS INVALID VALUE FOR *keyword***

Explanation: You specified an incorrect value for a keyword in the specified output descriptor.

User Response: Correct the value for the keyword and resubmit the job.

BMC50143E **OUTPUT DESCRIPTOR '*descriptor*' CANNOT SPECIFY REALDD WHEN STACK NO SPECIFIED**

Explanation: Because you specified NO for the STACK installation option, the utility cannot specify a ddname for allocating a tape unit for stacked output copies from multiple UNLOAD statements.

User Response: Specify STACK=YES during installation or before you submit the job.

BMC50144E **OUTPUT DESCRIPTOR '*descriptor*' DSNAME PATTERN '*pattern*' CAUSES INVALID NUMBER OF DATASETS FOR TYPE *type***

Explanation: The physical number of data sets that the utility created for the primary output descriptor does not match the physical number of data sets that the utility created for the secondary output descriptor.

User Response: Correct your OUTPUT statement and resubmit the job.

BMC50145E **OUTPUT DESCRIPTOR '*descriptor*' NOT DEFINED**

Explanation: The utility determined that the specified output descriptor name is not defined.

User Response: Use an appropriate DD statement in the JCL. If you want to use customized data set names instead of the current default names for the output descriptors, use the data set name options to change them.

BMC50146E **SYMBOLIC '*symbolic_variable*' IS INVALID IN OUTPUT DESCRIPTOR '*descriptor*'**

Explanation: The utility found a symbolic variable name that it does not support in the output descriptor. To obtain information about using symbolic variables to construct a valid data set name, see the reference manual for the utility that you are running.

User Response: Correct the data set name value in the OUTPUT statement and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50147E DSNAME TOO LONG IN OUTPUT DESCRIPTOR '*descriptor*'

Explanation: The data set name that you constructed for the specified output descriptor exceeds the maximum number of bytes that the utility allows.

If you used symbolic variables, the variables resolved to a data set name that is too long. See the reference manual for the utility that you are running for detailed information about using symbolic variables to construct a data set name for an output descriptor.

User Response: Specify a shorter data set name in the command. Resubmit or restart your job.

BMC50148E DSNAME CONFLICT BETWEEN OUTPUT DESCRIPTORS '*descriptor*' AND '*descriptor*' FOR DATASET TYPE *type*

Explanation: When the utility resolved the names of two different output descriptors, the names were identical. The real value of each data set name must be unique for different data set types.

User Response: Change the data set name of one of the output descriptors so that when the utility resolves them, each data set name is unique. Resubmit your job.

BMC50149E SYMBOL '*symbol*' MUST BE PREFIXED BY ALPHA OR NATIONAL CHARACTER IN OUTPUT DESCRIPTOR NAME '*name*'

Explanation: The symbol that you specified for a numeric variable did not contain a prefix with a form of a Latin alphabetic character.

User Response: Use the appropriate prefix in the symbol and resubmit the job. For information about using symbolic variables to construct the data set name, see the reference manual for the utility that you are running.

BMC50150E STACKED TAPE NOT SUPPORTED FOR NONPARTITIONED TABLE '*table_name*'

Explanation: The utility cannot support stacked output copies when it is using dynamic allocation to unload nonpartitioned tables.

User Response: Specify STACK=NO in your installation options, and resubmit the job.

BMC50151E UNABLE TO CONTINUE, DUPLICATE DATASET EXISTS FOR OUTPUT DESCRIPTOR '*descriptor*', DSN='*name*'

Explanation: The utility found a data set name in one output descriptor that is already cataloged.

User Response: Change the name of one of the data sets and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50153E **INVALID DSNAME '*dsname*' BUILT FROM OUTPUT DESCRIPTOR '*name*'**

Explanation: The utility determined that the data set name that you provided in the output descriptor is not valid.

User Response: Ensure that the name meets operating system requirements. To obtain information about using symbolic variables to construct the data set name, see the reference manual for the utility that you are running.

BMC50154E **INVALID UNIT '*unit*' SPECIFIED IN OUTPUT DESCRIPTOR NAME '*name*'**

Explanation: You specified a device type (disk or tape) that the utility does not support for dynamic allocation.

User Response: Correct the syntax and resubmit the job.

BMC50155E **NUMBER OF VOLUMES (*n*) EXCEEDS VOLCNT (*integer*) FOR OUTPUT DESCRIPTOR '*name*'**

Explanation: The number of volumes that the utility processed for the dynamically allocated data set exceeded the number of volumes that you specified for the integer in the VOLCNT option. The value for VOLCNT must be large enough to accommodate the number of volumes that the utility produces for the single largest data set.

User Response: Increase the number that you specified in the VOLCNT and resubmit the job.

BMC50156E **SMS ERROR ON DESCRIPTOR *descriptor* RETURN CODE *rc* REASON CODE *reason***

Explanation: An SMS error occurred when the utility attempted to retrieve SMS values from the SMS subsystem. The message shows the return code number and the SMS reason code. If the error is a known problem, the message contains a brief description of the problem.

User Response: The SMS return and reason codes are described in the *DFSMSdfp Diagnosis Reference* manual. Use the SMS return code and the reason code shown in the message to determine and fix the cause of the problem. Resubmit the job.

BMC50157I **FOR DESCRIPTOR *descriptor*, DATASET *dsname* WILL BE SMS MANAGED**

Explanation: The utility determined that SMS will manage the data set that will be dynamically allocated. Message BMC50158 follows this message.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50158I **DATACLAS *dataclas* MGMTCLAS *mgmtclas* STORCLAS *storclas***

Explanation: The utility determined that SMS will manage the data set that will be dynamically allocated. The message lists the SMS classes that the system ACS routines supplied.

User Response: No action is required.

BMC50159I **STORAGE GROUP *storgroup***

Explanation: This message shows the name of the SMS storage group that the utility is processing.

User Response: No action is required.

BMC50164W **NO STORAGE GROUP CONTAINS *n* MB, USING AVAILABLE SPACE FROM STORGROUP *storgroup***

Explanation: The utility determined that none of the storage groups that the system returned contain sufficient space for the file listed in the previous message (BMC50157). The utility will base its space calculations on the first eligible storage group.

User Response: No action is required.

BMC50165I **PRIMARY SPACE *n* RECORDS SECONDARY SPACE *n* RECORDS ON *n* UNITS SELECTED FOR STORGROUP *storgroup***

Explanation: The utility used the space available in storage group *storgroup* to calculate the number of primary and secondary records and the number of units that are required for the data set.

User Response: No action is required.

BMC50166W **NUMBER OF UNITS REQUIRED IS *n* BUT RESTRICTED BY UNIT COUNT OR VOLUMES OF *n***

Explanation: The utility calculated that it needs *n* units to accommodate the data set for SMS dynamic allocation. However, either you specified UNITCNT=*n* or you specified volume serial numbers, limiting the number of units that the utility can use. The utility will proceed with allocation and issue message BMC50165I.

User Response: Specify a higher value for UNITCNT or VOLUMES and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50167E **INSUFFICIENT PRIMARY SPACE AVAILABLE TO ALLOCATE PCTPRIM OF *percent* REQUIRED *n* MB AVAILABLE *n* MB**

Explanation: The utility calculated that it needs *n* MB for the primary extent for the PCTPRIM parameter, but the largest available extent is only *n* MB. The utility terminated the job.

Note: When you are running UNLOAD PLUS and the unload data set record length is greater than half a track, the value that the message shows for the required amount of space does not include unused space in a track.

User Response: You must either reduce the PCTPRIM value or remove it and resubmit the job.

BMC50170E **NO STORAGE GROUP CONTAINS ANY USABLE SPACE**

Explanation: After collecting all information for the storage group, the utility determined that none of the storage groups that the Automatic Class Selection (ACS) routines returned have any usable space.

User Response: Consult the technical support at your site.

BMC50171E **DATACLASS *data_class* SPECIFIES VSAM EXTENDED ADDRESSABILITY WHICH WILL CAUSE ALLOCATION TO FAIL**

Explanation: While attempting to dynamically allocate a sequential file, the utility determined that the data class that you implicitly or explicitly specified contains the VSAM extended addressability attribute. This attribute causes allocation of a sequential file to fail. The utility ended allocation and terminated the job.

User Response: Specify a different data class and resubmit the job.

BMC50172E ***space* EXCEEDS 2097152 MB**

Explanation: The utility determined that the space it requires for a data set exceeds 2 terabytes. The utility terminated the job.

User Response: Move the data set to tape and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50191W ERROR PARSING TABLE CHECK CONSTRAINT: *constraint*

Explanation: The utility cannot parse the identified table check constraint. The text of the table check constraint immediately follows this message. The utility might have encountered conditions that it does not allow when parsing table check constraints. An @ sign is displayed under the symbol that caused the parsing error. Processing continues as though ENFORCE NO was specified and CHECK pending (CHKP) is set. The following data types are invalid:

- floating-point constants
- VARCHAR columns exceeding a length of 255 bytes
- VARGRAPHIC or GRAPHIC data used with a LIKE predicate

User Response: If you received this message when running CHECK PLUS, run the IBM CHECK DATA utility to remove the CHKP status. If you received this message when running any other utility, run the CHECK PLUS or the IBM CHECK DATA utility to remove the CHKP status and resubmit the job.

BMC50192W PROCESSING WILL CONTINUE AS THOUGH 'ENFORCE NO' HAD BEEN SPECIFIED

Explanation: Processing continues as though ENFORCE NO was specified and CHECK pending (CHKP) is set.

User Response: Run the CHECK PLUS or the IBM CHECK DATA utility to remove the CHKP status and resubmit the job.

BMC50193W UNEXPECTED END-OF-LINE REACHED IN CHECK CONSTRAINT

Explanation: Two named columns have table check constraints defined but their column definitions are not identical. Processing continues as though ENFORCE NO was specified and CHECK pending (CHKP) is set.

User Response: Run the CHECK PLUS or the IBM CHECK DATA utility to remove the CHKP status and resubmit the job.

BMC50194W COLUMN '*colname*' AND COLUMN '*colname*' DO NOT HAVE IDENTICAL COLUMN DEFINITIONS

Explanation: The utility encountered an unexpected end-of-line while processing table check constraints. Processing continues as though ENFORCE NO was specified.

User Response: If you received this message when running CHECK PLUS, run the IBM CHECK DATA utility to remove the CHKP status. If you received this message when running any other utility, run the CHECK PLUS or the IBM CHECK DATA utility to remove the CHKP status and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50201S/W **CAF ERROR RC = *rc*, REASON = X'*reason*': '*explanation*'**

Explanation: This message usually accompanies an earlier message. It gives more specific information about a DB2 call attachment facility (CAF) error.

User Response: If you receive the severe version of this message, check the IBM DB2 messages manual for the reason code given. Correct the problem and resubmit or restart the job.

Some of the most likely explanations are:

- The plan is not bound on the specified DB2 subsystem.
- You are not authorized to run the plan.
- The maximum number of threads has been reached.
- The CAF release level does not match the release level of the DB2 subsystem.

For BMCDSN users, if you receive this message with RC=4, REASON=X'00C10823' and receive the explanation 'RELEASE LEVEL MISMATCH BETWEEN DB2 AND THE CAF CODE', this is just a warning and processing continues.

BMC50202S **DB2 IS NOT OPERATIONAL**

Explanation: The DB2 subsystem specified is not currently active. See message BMC50201S for more information.

User Response: Resubmit the job when the DB2 subsystem becomes active.

BMC50203S **CANNOT CONNECT TO DB2**

Explanation: The utility cannot connect to the DB2 subsystem specified. See message BMC50201S for more information.

User Response: Correct the problem and resubmit the job.

BMC50204S **CANNOT OPEN PLAN '*plan*'**

Explanation: The utility cannot open the specified plan. The following list describes some of the reasons that you might receive this message:

- The plan is not bound on the specified DB2 subsystem.
- You are not authorized to run the plan.
- The maximum number of threads has been reached.

See message BMC50201S for more information.

User Response: Correct the appropriate problem, based on this message and BMC50201S, and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50205S **UNEXPECTED SQL ERROR. SQLCODE = *n*, STMTID = '*stmtid*'**

Explanation: The utility encountered an unexpected SQL error. The SQL statement is uniquely identified by *stmtid*. See accompanying message BMC50207I for more information.

User Response: In many instances, the utility automatically retries -911 and related timeout-due-to-deadlock errors. If you understand the problem (for example, an SQL time-out error), correct the problem and restart the job.

BMC50206S **DSNTIAR RETURNED WITH RC = *rc***

Explanation: An SQL error occurred and DB2's message format routine (DSNTIAR) was invoked to format information from the SQL communication area (SQLCA). However, DSNTIAR returned a nonzero return code.

User Response: See message BMC50205S for the SQLCODE. If you cannot determine the error from this information, contact BMC Software Customer Support.

BMC50207I ***DSNTIAR_message***

Explanation: This message is returned from DB2's message format routine (DSNTIAR).

User Response: Refer to IBM's *DB2 Messages and Codes* manual for more information.

BMC50208S **DB2 COMMAND '*command*' RC = *rc*, REASON CODE = *reason***

Explanation: The specified DB2 command returned a nonzero return code.

User Response: Resubmit the job. If it fails a second time, contact BMC Software Customer Support.

BMC50209I ***message_text***

Explanation: The text of this message is generated by a non-BMC Software program invoked by the utility you are running.

User Response: Refer to the appropriate product's messages and codes for information.

BMC50210E **UNSUPPORTED COLUMN TYPE OF '*column_type*' FOR COLUMN '*colname*', TABLE '*creator.tbname*'**

Explanation: The column type for this column is not supported by this version of the utility. The utility terminates processing.

User Response: You cannot use this release of the utility to perform actions on a table containing the specified column type.

BMC Software, Inc., Confidential and Proprietary Information

BMC50211E **TABLE '*creator.tbname*' DOES NOT EXIST**

TABLESPACE '*dbname.tsname*' DOES NOT EXIST

INDEX '*creator.ixname*' DOES NOT EXIST

SYNONYM '*creator.synname*' DOES NOT EXIST

Explanation: The object you specified does not exist, or you renamed one of the utility's synonyms.

User Response: Ensure that you spelled the object correctly and specified the correct DB2 subsystem ID. You cannot change the utility's synonym names.

BMC50212E **TABLESPACE '*dbname.tsname*' IS NOT PARTITIONED**

INDEX '*creator.ixname*' IS NOT PARTITIONED

Explanation: You specified the PART command option, but the table space or index you specified is not partitioned.

User Response: Ensure that you specified the correct table space or index and the correct DB2 subsystem ID.

BMC50213E **TABLESPACE '*dbname.tsname*' DOES NOT HAVE A PARTITION NUMBER *p***

INDEX '*creator.ixname*' DOES NOT HAVE A PARTITION NUMBER *p*

Explanation: You specified the PART command option, but the table space or index you specified does not have partition *p*.

User Response: Ensure that you specified the correct table space or index, correct partition numbers, and DB2 subsystem ID.

BMC50214E **NO CLUSTER INDEX DEFINED FOR PARTITIONED TABLESPACE '*dbname.tsname*'. PROCESSING TERMINATED**

Explanation: You specified a partitioned table space that either does not have a table defined or does not have a clustering index defined.

User Response: Define the table or clustering index and resubmit the job.

BMC50215E **TABLE '*creator.tbname*' HAS A PRIMARY KEY, BUT NO UNIQUE INDEX HAS BEEN CREATED FOR THE PRIMARY KEY**

Explanation: You specified a table that has a primary key defined, but no unique index exists for the primary key.

User Response: Ensure that the correct unique index for the primary key is created and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50216E **TABLE '*creator1.tbname1*' and TABLE '*creator2.tbname2*' DO NOT RESIDE IN THE SAME TABLESPACE**

Explanation: The utility cannot perform the action on the tables you requested because they do not reside in the same table space.

User Response: Ensure that you specified the correct table names and that they are in the same table space. Resubmit the job.

BMC50217E **TABLESPACE '*dbname.tsname*' DOES NOT CONTAIN TABLE '*creator.tbname*'**

Explanation: The table that you specified in the SELECT/DELETE option is not contained in the table space to be used.

User Response: Ensure that you specified the correct table or table space names and resubmit the job.

BMC50218E/W **TABLE '*creator.tbname*' DOES NOT CONTAIN COLUMN '*colname*'**

Explanation: The column that you specified in the command is not contained in the named table.

User Response: If the message severity code is error (E), ensure that you specified the correct column or table. Then resubmit the job.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50219E **TABLE '*creator.tbname*' HAS ONE OR MORE UNIQUE CONSTRAINTS, BUT NOT ALL UNIQUE INDEXES HAVE BEEN CREATED**

Explanation: The specified table has one or more unique constraints, but not all unique indexes have been created.

User Response: You must create unique indexes for all unique constraints before the utility can process the table.

BMC50220E **STOGROUP '*stogroup*' DOES NOT CONTAIN ANY VOLUMES**

Explanation: No volumes are specified in the STOGROUP.

User Response: Ensure that the STOGROUP for the table space has volumes and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50221U **DB2 OBJECT DEFINITION(S) HAVE CHANGED PRIOR TO UTILITY RESTART. UNABLE TO PROCEED**

Explanation: The utility detected that DB2 internal identifiers have changed since the initial execution of the utility.

User Response: The utility is unable to proceed. Terminate the utility and take any other actions to make the affected DB2 objects usable.

BMC50227E **PRIQTY/SECQTY HAVE CHANGED FOR '*dsname*' (PRIQTY + 123 * SECQTY < MULTI-DS BOUNDARY)**

Explanation: When restarting, the utility detected a change in PRIQTY, SEQTY, or both for the indicated data set. When the utility was initially started, it determined that this data set needed to be a multi-data set object. The multi-data set requirement is either 2 gigabytes, the traditional size for a multi-data set object, or PIECESIZE, if this index is nonpartitioned.

User Response: If the object is a traditional multi-data set object, change the PRIQTY or SECQTY (or both) for the data set or corresponding DB2 object so that PRIQTY + 123*SECQTY >= 2 gigabytes and restart the utility.

If the object is an index and PIECESIZE was specified, the utility will not recognize a change in PIECESIZE between phases. Change the PRIQTY or SECQTY (or both) so that PRIQTY + 123*SECQTY is greater than or equal to the PIECESIZE value in effect at the initial start of the utility.

BMC50228E **'*creator.name*' IS NOT A LOCAL DB2 OBJECT**

Explanation: You have attempted to unload from an alias that defines an object on a foreign DB2 subsystem. UNLOAD PLUS can only process local objects.

User Response: Change the UNLOAD command and restart the utility.

BMC50229E **VIEW DEFINITION FOR VIEW '*creator.name*' NOT SUPPORTED, REASON CODE=*reason***

Explanation: The utility encountered an unsupported view definition for the reason code specified. The following list describes the reason codes that might display with this message:

1 DISTINCT keyword specified in view subselect
2 join specified in view subselect
3 WHERE clause specified in view subselect
4 GROUP BY clause specified in view subselect
5 HAVING clause specified in view subselect
6 view column derived from a function or expression

User Response: Use a view that does not violate the restrictions listed in the reason codes, or unload the data directly from the table referenced by the view.

BMC Software, Inc., Confidential and Proprietary Information

BMC50230E **UNSUPPORTED TABLESPACE TYPE '*type*' for '*dsname*.*tsname*'**

Explanation: The utility that you are running does not support the TYPE of the table space as defined in the DB2 catalog.

User Response: No action is required.

BMC50231S **CANNOT LOAD EDITPROC '*epname*', TABLE = '*creator*.*tbname*'**

Explanation: The utility attempted to load the specified EDITPROC but could not find it, or the utility did not have enough memory.

User Response: Check that the module is available in your LINKLIST, JOBLIB, or STEPLIB. It is usually found in the load library *db2*.DSNEXIT. Ensure that you specified a large enough region size.

BMC50232S **CANNOT LOAD VALIDPROC '*vpname*', TABLE = '*creator*.*tbname*'**

Explanation: The utility attempted to load the specified validation routine but could not find it, or the utility did not have enough memory.

User Response: Check that the module is available in your LINKLIST, JOBLIB, or STEPLIB. It is usually found in the load library *db2*.DSNEXIT. Ensure that you specified a large enough region size.

BMC50233S **CANNOT LOAD FIELDPROC '*fpname*', TABLE = '*creator*.*tbname*', COLUMN = '*colname*'**

Explanation: The utility attempted to load the specified FIELDPROC but could not find it, or the utility did not have enough memory.

User Response: Check that the module is available in your LINKLIST, JOBLIB, or STEPLIB. It is usually found in the load library *db2*.DSNEXIT. Ensure that you specified a large enough region size.

BMC50234S **CANNOT LOAD DATE/TIME ROUTINE '*routine_name*'**

Explanation: The utility attempted to load the specified date/time routine but could not find it, or the utility did not have enough memory.

User Response: Check that the module is available in your LINKLIST, JOBLIB, or STEPLIB. It is usually found in the load library *db2*.DSNEXIT. Ensure that you specified a large enough region size.

BMC50235E **EDITPROC '*epname*' RETURN CODE = *rc*, REASON CODE = *reason***

Explanation: The specified EDITPROC returned a nonzero return code.

User Response: Contact the author of the EDITPROC. (The EDITPROC is not part of the utility product.)

BMC Software, Inc., Confidential and Proprietary Information

BMC50236E **EDITPROC '*epname*' HAS MODIFIED STORAGE BEYOND THE END OF THE OUTPUT AREA**

Explanation: See DB2 reason code 00C90103.

User Response: Contact the author of the EDITPROC. (The EDITPROC is not part of the utility product.)

BMC50237E **EDITPROC '*epname*' RETURNED LENGTH = *l*; EXPECTING LENGTH <= THE MAXIMUM ROW LENGTH**

Explanation: See DB2 reason code 00C90103.

User Response: Contact the author of the EDITPROC. (The EDITPROC is not part of the utility product.)

BMC50238E **DATE/TIME ROUTINE '*routine_name*' RETURN CODE = *rc*, REASON CODE = *reason***

Explanation: The specified date/time routine returned a nonzero return code.

User Response: Contact the author of the date/time routine. (The date/time routine is not part of the utility product.)

BMC50239E **FIELDPROC '*fpname*' RETURN CODE = *rc*, REASON CODE = *reason*, MESSAGE TOKEN = '*text*'**

Explanation: The specified FIELDPROC returned a nonzero return code.

User Response: Contact the author of the FIELDPROC. (The FIELDPROC is not part of the utility product.)

BMC50240E **VALIDPROC '*vpname*' RETURN CODE = *rc*, REASON CODE = *reason***

Explanation: The specified VALIDPROC returned a nonzero return code.

User Response: Contact the author of the VALIDPROC. (The VALIDPROC is not part of the utility product.)

BMC50241I **CANNOT LOAD DB2 AUTHORIZATION EXIT 'DSNX@XAC', USING DB2 CATALOG FOR UTILITY AUTHORIZATION**

Explanation: The utility attempted to load the DB2 security exit but could not find it, or memory was insufficient. The utility should find this exit even if the utility is not actively using the exit for DB2 security authorization.

User Response: Ensure that this module, which is usually found in the load library *db2*.DSNEXIT, is included in your LINKLIST, JOBLIB, OR STEPLIB. Ensure that you specified an adequate region size.

BMC Software, Inc., Confidential and Proprietary Information

BMC50247E ***task_no*: PAGE *n* IN '*dsname*' HAS PGLOGRBA *rba/lrsn* WHICH IS GREATER THAN THE CONSISTENCY RBA, FAILURE IN SNAPSHOT**

Explanation: The software snapshot function of XBM or SUF failed to provide a consistent copy of the object. The utility terminated the job.

User Response: If you are in a data-sharing environment, ensure that each active DB2 member of the data-sharing group in which you are running also has an active XBM. Each active XBM must be joined to the same XBM group. After you have established the correct XBM environment, resubmit the job. If your XBM environment is correct and you still receive this message, contact BMC Software Customer Support.

If you are not running in a data-sharing environment, contact BMC Software Customer Support.

BMC50248I ***task_no*.: PAGE *n* IN '*dsname*' IS A ZERO PAGE**

Explanation: While attempting to read a data page from the named data set in a non-segmented table space, the utility encountered a page (*n*) numbered zero. The task number identifies the task associated with the error.

User Response: No action is required. The utility continues normal execution.

BMC50249E ***task_no*.: PAGE *n* PGCOMB INCONSISTENT WITH TRAILER BYTE**

Explanation: The utility found that the PGCOMB field is inconsistent with the page trailer byte for the specified page, indicating that the page might not have been successfully written. The page number displayed is the physical page for which the inconsistency occurred.

User Response: Recover the table space and resubmit your job.

BMC50250S THE HEADER PAGE INDICATES THE TABLESPACE IS *type* BUT THE CATALOG SAYS IT IS NOT

Explanation: The header page type indicator for the table space does not match the type specified in the DB2 catalog. The value of *type* can be one of the following table space types:

- PARTITIONED
- SEGMENTED
- SIMPLE

User Response: Verify the table space type that was specified when the table space was created. This error usually occurs when you use DSN1COPY to copy data into a table space that is not defined in the same way as the table space from which the data was copied.

BMC Software, Inc., Confidential and Proprietary Information

BMC50251S ***task_no*.: PAGE *n* IN '*dsname*' IS UNRECOGNIZABLE, DATA = X'*data*'**

Explanation: The utility encountered a page (*n*) that it did not expect, which might be due to an inline or incremental copy issue. The task number identifies the task associated with the error. The data that this message displays is nine bytes long, starting at offset X'08' in the page.

Note: The page number includes settings of high-order bits for individual partitions in a partitioned table space.

User Response: If you are running UNLOAD PLUS version 7.1.0 or later, specify the INFILE *ddname* FULL INLINE YES option for inline copies or the INFILE *ddname* INCREMENTAL option for incremental copies, and rerun the job. Otherwise, if you have determined that the DB2 data set does not contain any errors, contact BMC Software Customer Support.

BMC50252S ***task_no*.: PAGE *n* IN '*dsname*' IS BROKEN**

Explanation: The utility encountered a page that is marked broken. The task number identifies the task associated with the error.

User Response: Recover the table space and resubmit the job.

BMC50253S/I ***task_no*.: DBID '*dbid*' AND/OR PSID '*psid*' DO NOT MATCH THOSE FOUND IN DATASET '*dsname*'**

Explanation: The database ID and the page set ID don’t match the IDs found in the data set named in the message. The task number identifies the task associated with the error.

User Response: If you use DSN1COPY, ensure that you recovered the data set correctly and did not change the database ID and the object IDs. For REORG PLUS and UNLOAD PLUS users, if you receive this message and want to restart the job without making any changes, specify the ON MESSAGE option for this message and restart the job.

For REORG PLUS users, REORG PLUS resets the database and page set IDs to match the catalog IDs.

BMC Software, Inc., Confidential and Proprietary Information

BMC50254S ***task_no*.: UNEXPECTED EOF (PAGE=*n*) IN DATASET '*dsname*'**

Explanation: The utility encountered an end-of-file when it expected more data. The task number identifies the task associated with the error.

For REORG PLUS users, if you receive this message for a table space reorganization during ANALYZE processing with ANALYZE ONLY/PAUSE/YES specified, an index space does not contain valid DB2 pages. You can also receive this message during UNLOAD processing if the primary (PRIQTY) or secondary (SECQTY) quantity for the table space or index space was altered before the reorganization began. This alteration makes the multiple data set table space or index space a non-multiple data set table space or index space.

User Response: Possibly the table space or index is in error and needs to be recovered. For REORG PLUS and UNLOAD PLUS users, if you receive this message and want to restart the job without making any changes, specify the ON MESSAGE option for this message and restart the job. For an index space reorganization, consider an appropriate recovery scenario.

For REORG PLUS users who received this message because of changing PRIQTY and SECQTY, alter the PRIQTY and SECQTY values back to the original values to create a multiple data set table space or index space and restart the reorganization.

BMC50255S ***task_no*.: DATASET '*dsname*' DOES NOT CONTAIN AN INDEX**

Explanation: The data set named does not contain a DB2 index. The task number identifies the task associated with the error.

User Response: If you use DSN1COPY, ensure that you recovered the data set correctly. If you understand the problem, correct it and restart the job. For more help, contact BMC Software Customer Support.

BMC50256S ***task_no*: A KEY OR RID MISMATCH WAS FOUND FOR INDEX '*creator.ixname*', KEY = X'*key*'. 'RECOVER INDEX' IS NEEDED**

Explanation: You attempted to use one or more partitions of a partitioned table space that required the utility to update corresponding entries in the specified index. However, either too few or too many key/RID pairs were found in the index for the partitions you were using.

User Response: Recover the index after the partitions have been successfully reloaded.

BMC50257S ***task_no*.: DATASET '*dsname*' IS ROSHARE AND IS NOT MARKED CONSISTENT**

Explanation: You attempted to access a ROSHARE READ table space, but it is not in a consistent state.

User Response: The ROSHARE OWNER version of the table space must follow the DB2 procedures for making the data consistent.

BMC Software, Inc., Confidential and Proprietary Information

BMC50258E ***task_no*.: A DUPLICATE KEY VALUE HAS BEEN DETECTED IN UNIQUE INDEX '*creator.ixname*', RID = X'*rowid*',**

BMC50259E **KEY = X'*key*'**

Explanation: If this error results in a return code of 8, the utility detected a duplicate key during the index build process. The utility builds the index using only the first RID of the duplicate keys and places the index in recover pending (RECP) status. The rows corresponding to the duplicate key entries remain in the table space.

For LOADPLUS users, these messages indicate that LOADPLUS detected a duplicate key during the LOAD or COMBINED phase. If you receive a return code 4, LOADPLUS deleted the duplicate keys and their corresponding table space rows and wrote the duplicates to SYSERR.

User Response: For REORG PLUS users, consider an appropriate recovery scenario.

For LOADPLUS users, if you receive a return code 8, either reload the data correctly (use UNIQUECHECK YES or correct your input data) or recover the table space and indexes.

For LOADPLUS users, if you receive a return code 4, LOADPLUS deleted your duplicates; therefore, LOADPLUS did not load some of the data that you expected it to load. These messages provide information to help you identify the duplicate key and table space rows.

BMC50260E **SPACE '*dbname.spname*' IS NOT STARTED ACCESS(RO)**

Explanation: When you restarted the utility, it determined that the space was not started ACCESS(RO) when it should have been.

User Response: If you are sure that no changes have been made to the space or associated objects (such as dropping or creating an index or updating via SQL), you can start the space ACCESS(RO) and restart the job. If you are not sure whether changes have been made, terminate the utility and start again. If you terminate the utility, you might have to recover the table space and any associated indexes first and then resubmit the job. See the reference manual for the utility you are running for specific information about restarting the job.

BMC50261E **SPACE '*dbname.spname*' ALREADY IN USE BY A DB2 UTILITY**

Explanation: The table space or associated index spaces you requested to use are already marked UTRO, UTRW, or UTUT.

User Response: Wait for the DB2 utility to complete or terminate the DB2 job before restarting the BMC Software utility job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50262E **SPACE '*dbname.spname*' ALREADY IN USE BY A BMC UTILITY, UTILID = '*utilid*'**

Explanation: A BMC Software utility is currently using the table space or associated index spaces that you requested.

User Response: Wait for the other utility to complete or terminate the job. Review the RESTART parameter that is coded in the job and replace it if a more suitable value is appropriate. Resubmit the job.

BMC50263E **SPACE '*dbname.spname*' IS IN '*state*' STATE**

Explanation: The space is in the restrictive state that this message displays. The utility cannot process the object while it is in this state.

User Response: Follow the documented DB2 procedures to remove the object from this state before resubmitting the job. If you are running the utility with SHRLEVEL CHANGE and the object is in read-only (RO) status, contact your BMC Software technical support analyst for assistance. If you are running LOADPLUS and the specified *state* is LPL or WEPR, see Chapter 2 of the *LOADPLUS for DB2 Reference Manual* for information about the restrictions for loading an object that is in one of these states.

BMC50264E **SPACE '*dbname.spname*' IS IN 'CHECK PENDING' STATE**

Explanation: The space named is marked CHECK pending (CHKP).

User Response: Run the CHECK DATA utility to remove the CHKP status and resubmit the job.

BMC50265E **SPACE '*dbname.spname*' IS NOT STARTED**

Explanation: The table space or associated index space you requested to use is not started RW, RO, or UT.

User Response: Start the table space or index space named in the message and restart the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50266E **SPACE '*dbname.spname*' CANNOT BE STOPPED. IT MAY BE IN USE**

Explanation: The table space or index space (or both) that you requested to use cannot be stopped. The STOP command has been issued by the utility and is waiting for execution.

User Response: The space will be marked stop pending (STOPP) until no other jobs are using it. Wait for the status to become stopped and then restart the job. Other activity within the DB2 subsystem, such as active log dumps, can delay processing of the STOP command. Examine any BMC50209I messages (which might display outstanding locks on the space) or the DSNMSTR address space to identify the currently running jobs that are preventing the command from being processed.

For REORG PLUS users, if you specified ON FAILURE TERMINATE, REORG PLUS attempts to start the space in its original status. Because other activity within the DB2 subsystem can interfere with this attempt, you should verify that all spaces are in the appropriate status before resubmitting your job.

BMC50267E **SPACE '*dbname.spname*' IS NOT STOPPED**

Explanation: When you restarted the utility, it determined that the space was not stopped when it should have been.

User Response: If you are sure that no changes have been made to the space or associated objects (such as dropping or creating an index), you can stop the space and restart the job. If you are not sure whether changes have been made, terminate the utility and start again. If you terminate the utility, you might have to recover the table space and any associated indexes first and then resubmit the job. See the reference manual for the utility you are running for specific information about restarting the job.

BMC50268E **SPACE '*dbname.spname*' CANNOT BE STARTED ACCESS(RO). IT MAY BE IN USE**

Explanation: The utility attempted to START ACCESS(RO) a DB2 space. The table space could not be started ACCESS(RO).

User Response: The table space might have been in use. Because of the failure, the table space might now be in the RO state. Start the table space in the correct state and resubmit the job.

BMC50269W **SPACE '*dbname.spname*' PART *p* IS IN 'COPY PENDING' STATE AND WILL NOT BE QUIESCED**

Explanation: No quiesce on the specified space will be performed because the space is in copy pending (COPY) state. PART *p* is omitted from the message if the table space is not partitioned.

User Response: To ensure that all updated pages in the buffer pool are processed, perform a STOP/START on the space before the running the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50270I CHECK PENDING STATUS HAS BEEN *action* FOR *dbname.tsname* PART *p*

Explanation: The status for the named table space/table space partition has changed as indicated. PART *p* is omitted from the message if the table space is not partitioned. The *action* can be either SET (ON) or RESET (OFF).

User Response: If CHECK pending status (CHKP) is set, run CHECK PLUS with the CHECK DATA option or run the DB2 CHECK DATA utility with SCOPE ALL on the corresponding table space or table space partition to ensure that no referential integrity or DB2 table check constraints are violated.

BMC50271I ATTEMPTING TO START SPACE '*dbname.spname*'...

Explanation: You specified the ON FAILURE TERMINATE option and the utility is terminating due to an error. The utility is attempting to start the named space.

User Response: No action is required.

BMC50272I SPACE '*dbname.spname*' IS STARTED

Explanation: You specified the ON FAILURE TERMINATE option and the utility is terminating due to an error. The space has been successfully started. This message is always preceded by BMC50271I.

User Response: No action is required.

BMC50273I SPACE *dbname.tsname* PART *p* WILL REMAIN IN 'CHECK PENDING' STATUS

Explanation: The table space or table space partition was already in CHECK pending status (CHKP) and was bypassed by the utility. PART *p* is omitted from the message for non-partitioned table spaces.

User Response: No action is required.

BMC50274E SPACE '*dbname.spname*' IS IN '*pending*' STATE AND PREVENTS UTILITY FROM RUNNING

Explanation: The utility that you are running cannot complete the job because the table space or index space is in the indicated pending state.

User Response: Start the space in an allowable state and resubmit the job. Refer to the table and index space requirements section in the reference manual for the utility that you are running.

BMC Software, Inc., Confidential and Proprietary Information

BMC50275E **SPACE '*dbname.spname*' HAS AN ERROR RANGE. RECOVERY IS NEEDED**

Explanation: The table space or index has an error range indicated by DB2.

User Response: Recover the table space or index using the ERROR RANGE option of the DB2 RECOVER utility.

BMC50276E **XBM '*request_type*' ERROR. RC=*rc*, REASON=*reason*, DSN=*dsname***

Explanation: The indicated XBM error occurred during the snapshot process. Message BMC50278E might also be issued.

User Response: Correct the error and resubmit the job. For questions about the XBM error, contact BMC Software Customer Support.

BMC50277I **XBM STATISTICS: DSN='*dsname*', READS=*n*, HITS=*n*, WRITES=*n*, CACHE=*n***

Explanation: This message is issued after the snapshot function is complete and provides statistical information about the snapshot operation.

User Response: No action is required.

BMC50278E ***XBM_error_message_text***

Explanation: XBM experienced an error and has passed the message text to the utility product for output.

User Response: Take corrective action appropriate to the XBM error reported and resubmit the job. For questions about the reported error, contact BMC Software Customer Support.

BMC50279E **SPACE '*dbname.spname*' CANNOT BE STARTED ACCESS(UT). IT MAY BE IN USE**

Explanation: The utility attempted to start a DB2 space in ACCESS(UT) state. The space could not be started ACCESS(UT).

User Response: The space might be in use. Start the table space in the correct state and resubmit the job.

BMC50280E ***keywords* NOT SUPPORTED IN A DATA-SHARING DB2 SUBSYSTEM**

Explanation: You cannot use the keywords displayed in this message when running in a data-sharing DB2 subsystem.

User Response: Correct the command and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50281S THE BMCUTIL SUBTASK HAS TERMINATED OR TIMED-OUT

Explanation: The utility attempted a SYNC request but the SYNC subtask failed.

User Response: See prior messages to determine the problem. Also, the SYNC time-out value (SYNCTO) specified in the installation options may be too small.

BMC50282U THE BMCUTIL SUBTASK ABNORMALLY TERMINATED WITH SYSTEM ABEND CODE = *abend_code*

Explanation: The utility SYNC subtask abended.

User Response: Check prior messages or the job log to determine the problem. If you understand the problem (for example, an SQL time-out), correct it and restart the job. For other errors, contact BMC Software Customer Support.

BMC50283U THE BMCUTIL SUBTASK ABNORMALLY TERMINATED WITH RETURN CODE = *rc*

Explanation: The utility SYNC subtask has terminated abnormally.

User Response: Check prior messages or the job log to determine the problem. If you understand the problem (for example, an SQL time-out), correct it and restart the job. For other errors, contact BMC Software Customer Support.

BMC50284I RETRYING SQL -911 ON STMT *number_or_function*. RETRIES REMAINING=*n*. WAIT TIME = *number* SECS PLUS DB2 WAIT TIME

Explanation: The utility encountered an SQL -911 return code while processing the specified SQL statement number, performing the specified serialization function, or processing a drain.

The utility retries the SQL statement, serialization function, or drain until the RETRIES REMAINING number is reached. The interval between retries (WAIT TIME) is the amount of time specified on the SQLDELAY or the DRNDELAY option in the installation options module plus the time that DB2 waits for a time out or a deadlock, whichever caused this -911 to be issued.

The STMT value PREVENT *xx* indicates that a drain is being performed. A STMT with an actual statement ID value indicates a SELECT/DELETE/UPDATE statement.

User Response: No action is required.

BMC50285E SQL -911 RETRIES UNSUCCESSFUL. RETRY ATTEMPTS = *n*

Explanation: This message follows message BMC50284I if all the retry attempts were unsuccessful. The utility retried the statement the displayed number of times.

User Response: Try to restart the utility. If the unsuccessful retries were on the BMCUTIL or BMCSYNC tables, the restart might not be successful.

BMC Software, Inc., Confidential and Proprietary Information

BMC50286I **SQL -911 RETRY SUCCESSFUL**

Explanation: The utility encountered an SQL -911 return code and retried the statement. This message follows BMC50284I if one of the retries was successful.

User Response: No action is required.

BMC50287E **REXX ENVIRONMENT FAILURE AT '*description*' RETURN CODE = *rc*, REASON CODE = *reason***

Explanation: A problem occurred while the utility was establishing the environment for a REXX user exit. The utility terminated the job.

User Response: If the stated description and reason provide sufficient information for you to correct the problem, correct the problem and resubmit the job. Check your job log for any IRX messages that the REXX environment manager might have issued. If you cannot correct the problem, contact BMC Software Customer Support.

BMC50288S **CANNOT LOAD USER EXIT '*module*', TABLE = '*creator.tbname*', FIELD = '*fldname*'**

Explanation: You specified a user-written exit routine for conversion but the requested module cannot be loaded.

User Response: Check that the user-written exit routine module is available in your LINKLIST, JOBLIB, or STEPLIB. Ensure that you specify a large enough region size.

BMC50289E **USER EXIT '*module*' RETURN CODE = *rc*, REASON CODE = *reason*, MESSAGE TOKEN = '*text*'**

Explanation: The specified user-written exit routine produced a nonzero return code.

User Response: Contact the author of the user-written exit routine. (The user-written exit routine is not part of the utility product.)

BMC50291S **BLDL FOR MODULE '*module*' FAILED WITH RC = *rc*, REASON = *reason***

Explanation: The utility issued a BLDL request and the request failed.

User Response: Check the job log for more information. Ensure that you specified a large enough region size.

BMC50292S **ATTACH FOR '*module*' FAILED WITH RC = *rc***

Explanation: The utility tried to attach the specified module but either could not find it or the utility did not have enough memory.

User Response: Check that the module is available in your LINKLIST, JOBLIB or STEPLIB. Ensure that you specified a large enough region size.

BMC Software, Inc., Confidential and Proprietary Information

BMC50293S/I CANNOT LOAD MODULE '*module*'

Explanation: The utility tried to load the specified module but either could not find it or the utility did not have enough memory.

User Response: Check that the module is available in your LINKLIST, JOBLIB, or STEPLIB. Ensure that you specified a large enough region size.

BMC50294S SYSTEM CATALOG INFORMATION ROUTINE FAILED WITH RC = *rc*. DSN = '*dsname*'

Explanation: The utility tried to retrieve system catalog information for a data set, but the system routine returned a nonzero return code.

User Response: Ensure that the data set specified in the message has not been deleted, uncataloged, or archived. For example, if the value of the REDEFINE command or installation option is YES for a VCAT-defined object, and the define process failed in the RELOAD phase of REORG PLUS or the LOAD phase of LOADPLUS, you must manually define the object and restart your job. If none of these circumstances apply, contact BMC Software Customer Support for additional assistance.

BMC50295I IDCAMS SUCCEEDED DURING OBJECT REDEFINITION. SEE JOBLOG FOR IDC MESSAGES

BMC50295E IDCAMS FAILED DURING OBJECT REDEFINITION. SEE JOBLOG FOR IDC MESSAGES

Explanation: IDCAMS either succeeded or failed as indicated during VSAM data set redefinition. The IDCAMS input and output are displayed in the job log.

User Response: When you receive the successful version of this message, no action is required.

In all other cases, check the job log for the error. If you understand the problem, correct it and restart the job. If you need assistance, contact BMC Software Customer Support.

Note: If the job ends with a condition code of 4, the IDCAMS messages indicate that other volumes in the STOGROUP were used for data set placement.

BMC50296S UNABLE TO INVOKE SMS SERVICE ROUTINE '*name*'

Explanation: The utility cannot use the storage management subsystem (SMS) service routine named in the message.

User Response: Check the job log for additional messages and ensure that the SMS routines are running from an APF-authorized library. For other problems, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50297S **SMS SERVICE ROUTINE '*name*' FAILED WITH RC = *rc*, REASON = *reason***

Explanation: The storage management subsystem (SMS) service routine named in the message failed for the reason specified.

User Response: If you recognize the problem, correct it and resubmit the job. For other problems, contact BMC Software Customer Support.

BMC50298S **LINK TO MODULE '*module*' FAILED WITH RC = '*rc*', REASON CODE = *reason***

Explanation: A link to the specified module failed with the indicated return code and reason.

User Response: Check prior messages or the job log to determine the source of the problem. Review the explanation of the return code and reason reported in the appropriate manual. If you understand the problem, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50299S **DYNAMIC ALLOCATION FAILED IN '*module*' FOR DDNAME '*ddname*', S99ERROR = *error*, S99INFO = *error***

Explanation: A dynamic allocation failure occurred.

User Response: Check the job log for allocation errors and refer to the *MVS Programming Authorized Assembler Services Guide* for more information about S99ERROR and S99INFO reason codes. Also, check the job log for messages related to the error. “IKJ” type messages are described in IBM *TSO/E Messages*.

For example, a dynamic allocation error can occur because of an invalid unit specification on the WORKUNIT option in the utility installation options.

Correct the problem and restart the job.

BMC50300E **DYNAMIC ALLOCATION FAILED FOR DDTYPE '*ddtype*' UNIT '*unit_name*' CONTAINS NO VALID DEVICES**

Explanation: The UNIT specification in the installation options module or in the command for the specified DDTYPE does not contain a valid device. The utility cannot allocate the device it needs for the dynamically allocated data set.

User Response: Change the unit specification to one that contains at least one valid device and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50301I **REXX EXIT '*exit_name*' ALTERED *dbname.spname* PART *part* PRI/SEC PRI/SEC FROM *n/n* TO *n/n***

REXX EXIT '*exit_name*' ALTERED *dbname.spname* COMPONENT A*nnn* PRI/SEC FROM *n/n* TO *n/n*

Explanation: The specified REXX user exit successfully altered the primary or secondary quantity from the first displayed value to the second. For a partitioned table space or index space, the utility issued the request for the specified partition. For a nonpartitioned multiple data set table space or index space, the utility issued the request for the specified component. A*nnn* is the last node of the data set name, and *nnn* is a numeric value.

User Response: No action is required.

BMC50302I **REXX EXIT '*exit_name*' REQUESTED SQL ALTER OF DB2 CATALOG FOR *dbname.spname* PART *part***

Explanation: The specified REXX user exit requested that the utility issue an SQL ALTER command to the DB2 catalog. The utility queues the request, which DB2 will process before the utility ends. The utility issues message BMC50305I when DB2 successfully completes the SQL ALTER command or message BMC50304W if it fails.

If the part number in the message is zero, the table space or index space is nonpartitioned.

User Response: No action is required.

BMC50303I **SQL ALTER SUCCEEDED FOR *dbname.spname* PART *part* PRI/SEC FROM *n/n* TO *n/n***

Explanation: The utility successfully issued an SQL ALTER command to change the primary and secondary values of this partition for the specified table space or index space. If the part number is zero, the table space or index space is nonpartitioned. DB2 updated the values in the DB2 catalog.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50304W **SQL ALTER FAILED FOR *dbname.spname* PART *part* PRI/SEC FROM *n/n* TO *n/n***

Explanation: The utility issued an SQL ALTER command to change the primary and secondary values of this partition for the specified table space or index space, but the SQL ALTER command failed. If the part number is zero, the table space or index space is nonpartitioned. DB2 did not update the values in the DB2 catalog.

The utility continues to normal completion, but ends with a return code of 4. The most common reason for this failure is -911 SQLCODE, which indicates that DB2 cannot get a lock on the database to perform the SQL ALTER TABLESPACE or ALTER INDEX command.

User Response: You can use the values displayed in this message to perform your own SQL ALTER command. Refer to the appropriate IBM reference manual for the correct SQL syntax.

BMC50305I **REQUESTED SQL ALTER OF DB2 CATALOG IGNORED FOR *dbname.spname* COMPONENT A*nnn***

Explanation: A user exit requested that the utility issue an SQL ALTER command to update the DB2 catalog. However, the utility cannot issue the ALTER command for components of a nonpartitioned multiple data set table space or index space because they do not have entries in SYSIBM.SYSTABLEPART or SYSIBM.SYSINDEXPART. A*nnn* is the last node of the data set name, and *nnn* is a numeric value.

User Response: No action is required.

BMC50306I **REXX EXIT '*exit_name*' SELECTED 'REDEFINE NO' FOR DSN='*dsname*'**

Explanation: In the named user exit at the DSRSEXIT exit point, you told the utility to use REDEFINE NO for the named VSAM object. The utility will bypass the DELETE/DEFINE for the object.

User Response: No action is required.

BMC50307I **BMC_REDEFINE_OBJECT='NO' IGNORED FOR DSN='*dsname*' DUE TO *reason***

Explanation: In the DSRSEXIT user exit, you requested that the utility use REDEFINE NO for the named object. However, the utility cannot honor the request because of the stated reason. The utility will redefine the object.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50311S **UNABLE TO OPEN DATASET, DDNAME = '*ddname*'**

Explanation: The utility could not open the data set specified in the named DD statement.

User Response: Check the job log for errors and ensure that all the required input and output data sets are specified correctly. Correct the problem and restart the job.

BMC50312S **UNABLE TO CLOSE DATASET, DDNAME = '*ddname*'**

Explanation: The utility could not close the data set specified in the named DD statement.

User Response: Check the job log for errors. Correct the problem and restart the job.

BMC50313S **RDJFCB FAILED FOR DDNAME = '*ddname*'**

Explanation: The utility received a nonzero return code from an RDJFCB (Read Job File Control Block) request.

User Response: Check the job log. If you cannot identify and correct the problem, contact BMC Software Customer Support.

BMC50314S **I/O ERROR ON DATASET '*dsname*', CSW = '*channel_status_word*', SENSE = '*sense*'**

Explanation: The utility received an I/O error on a DB2 data set, with the indicated channel status word and sense data returned by MVS I/O support.

User Response: If you understand the error, correct the problem and restart the job. If you do not understand the error, contact BMC Software Customer Support.

BMC50315S **EXTEND REQUEST FOR DATASET '*dsname*' FAILED, DATASET IS OUT OF SPACE**

Explanation: The utility requested a secondary space allocation for the data set named, but the operating system denied the request. Either all secondary extents were allocated or all volumes allocated to the data set were out of space.

User Response: Correct the data set space allocation problem and resubmit the job.

BMC50316E **THE SECONDARY SPACE QUANTITY FOR DATASET '*dsname*' IS ZERO. IT CANNOT BE EXTENDED**

Explanation: The utility requested a secondary space allocation for the data set named, but the operating system denied the request because the secondary space quantity is set to zero and therefore cannot be extended.

User Response: Correct the data set space allocation problem and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50317I UNABLE TO *action* DATASET, DDNAME = '*ddname*', RC = *r15*, REASON = *r0*,*reason*

Explanation: The data set associated with the indicated ddname was not deleted or uncataloged. For REORG PLUS and LOADPLUS users, BMC50318I follows this message if you specified DELETEFILES YES and the data set name is known.

The value for *action* is either DELETE or UNCATLG.

User Response: The following list describes the reason codes that might be issued in this message:

- 00020001—The data set is not physical sequential and is not supported for DELETEFILES YES.
- 00020002—Dynamic allocation failed. See the dynamic allocation messages in the job log for the cause.
- 00020003—Dynamic deallocation failed. See the dynamic allocation messages in the job log for the cause.
- 00020004—Dynamic deallocation failed. See the dynamic allocation messages in the job log for the cause.
- other—Contact BMC Software Customer Support.

BMC50318I DATASET *status action*, DDNAME = '*ddname*', DSN = '*dsname*'

Explanation: This message is displayed only when you specify DELETEFILES YES and provides data concerning the status of DELETEFILES processing. If the file is a tape file and TAPEDISP=UNCATLG was specified in the installation options module, the data set is uncataloged rather than deleted.

The value of *status* can be one of the following statuses:

- SUCCESSFULLY—The utility deleted or uncataloged the specified data set.
- NOT—The utility did not delete or catalog the specified data set. When the data set name is known, this message follows BMC50317.

The value for *action* is either DELETED or UNCATLGD.

User Response: No action is required if the delete or uncatalog process was successful. If the specified data set was not deleted or uncataloged, see message BMC50317, issued before this one, for the reason code. Respond as specified in the description of message BMC50317.

BMC Software, Inc., Confidential and Proprietary Information

BMC50319E **DATASET '*dsname*' EXCEEDS THE MAXIMUM ALLOWABLE DATASETS OF '*maximum_number*'**

Explanation: The maximum number of data sets for the DB2 object was exceeded.

User Response: For table space objects, you can either process less data, reduce the PCTFREE value, or increase the FREEPAGE value. However, you must recover the table space and rerun the job with the NEW execution parameter after making the changes.

For index objects, you can increase the PIECESIZE value or specify NLPCTFREE on the command in addition to making the PCTFREE or FREEPAGE changes listed for a table space object. Rerun the job with the RESTART execution parameter.

BMC50320E **SPACE '*dbname.tsname*' PART *p* STATUS HAS CHANGED. IT IS NOW *status2*, BUT IT WAS *status1* WHEN THE UTILITY STARTED**

Explanation: When the utility started, the table space was marked *status1*. However, it was changed to *status2* while the utility was running, causing the utility to terminate.

User Response: Determine why the status changed. Rerun the job, ensuring that the status does not change while the utility is running.

BMC50321E **DB2 *utility_name* UTILITY RAN ON '*dbname.spname*' PART *number* AT *timestamp* WHILE *utility* WAS RUNNING**

Explanation: While REORG PLUS with SHRLEVEL CHANGE was running, an IBM DB2 utility was started and completed on the same DB2 object. The REORG PLUS job terminated.

User Response: After the IBM DB2 utility completes, rerun the SHRLEVEL CHANGE job. Do not start any IBM utilities on a DB2 object while REORG PLUS is using that object.

BMC50322E **DATABASE '*dbname*' IS NOT STARTED**

Explanation: The specified database, which contains the object on which you are running the utility, is not started RW, RO, or UT.

User Response: Start the database named in the message and restart the job.

BMC50323E **DATABASE '*dbname*' IS IN '*status*' STATE**

Explanation: The specified database is in the displayed state. The utility cannot continue and terminated the job.

User Response: Follow the documented DB2 procedures to remove the object from this state before resubmitting the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50324E SPACE '*dbname.spname*' IS NOT STARTED ACCESS(RW)

Explanation: The options that you specified for this utility require that the space is in read/write status. For example, LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY has this requirement.

User Response: If appropriate, start the space ACCESS(RW) and resubmit the job. If you do not want to allow read/write access during the utility operation, change your utility command and resubmit the job. See the reference manual for the utility that you are running for information about object status requirements.

BMC50325E SYSTABLEPART UPDATE SUPPORT HAS NOT BEEN INSTALLED

Explanation: If this message indicates a user abend 3200 (REORG PLUS) or 3400 (LOADPLUS), you can find the reason code in register 15, which is displayed by abend analysis in the job log.

User Response: Ensure that the utility was installed correctly and that the ICOPY installation job was run. You might need to modify default values in the installation options member and rerun the ICOPY installation job. Otherwise, rebind the plan specified in your PLANSYNC installation option and then run ICOPY.

If you ran these jobs before without getting this message, the most likely cause for getting the message now is that the product's SYNC plan was rebound but the ICOPY installation job was not rerun. The SYNC plan name is specified on the PLANSYNC installation option. Whenever the SYNC plan is rebound, you must rerun the ICOPY installation job to reinstate the SYSTABLEPART update support.

To find the SYNC plan name, see Appendix A of the reference manual for the utility that you are running.

BMC50326E SYSSEQUENCES UPDATE SUPPORT HAS NOT BEEN INSTALLED

Explanation: If this message indicates a user abend 3200 (REORG PLUS) or 3400 (LOADPLUS), you can find the reason code in register 15, which is displayed by abend analysis in the job log.

If you ran these jobs before without getting this message, the most likely cause for getting the message now is that the product's SYNC plan was rebound but the ICOPY installation job was not rerun. The SYNC plan name is specified on the PLANSYNC installation option. Whenever the SYNC plan is rebound, you must rerun the ICOPY installation job to reinstate the SYSSEQUENCES update support.

User Response: Ensure that the utility was installed correctly and that the ICOPY installation job was run. You might need to modify default values in the installation options member and rerun the ICOPY installation job. Otherwise, rebind the plan specified in your PLANSYNC installation option and then run ICOPY.

For the SYNC plan name, see Appendix A of the reference manual for the relevant utility.

BMC Software, Inc., Confidential and Proprietary Information

BMC50327E **SYSINDEXPART UPDATE SUPPORT HAS NOT BEEN INSTALLED**

Explanation: If this message indicates a user abend 3200 (REORG PLUS) or 3400 (LOADPLUS), you can find the reason code in register 15, which is displayed by abend analysis in the job log.

If you ran these jobs before without getting this message, the most likely cause for getting the message now is that the SYNC plan of the utility was rebound but the ICOPY installation job was not rerun. The SYNC plan name is specified on the PLANSYNC installation option. Whenever the SYNC plan is rebound, you must rerun the ICOPY installation job to reinstate the SYSINDEXPART update support.

User Response: Ensure that the utility was installed correctly and that the ICOPY installation job was run. You might need to modify default values in the installation options member and rerun the ICOPY installation job. Otherwise, rebind the plan specified in your PLANSYNC installation option and then run ICOPY.

For the SYNC plan name, see the reference manual for the relevant utility.

BMC50328I **THE '*BMCSTATS_table_type*' TABLE USED BY THIS UTILITY DOES NOT MATCH THE ONE USED BY BMCSTATS ('*creator.tbname*')**

Explanation: If you are running a utility that is part of a solution that contains the BMCSTATS component, the utility issues this message if the utility synonyms do not resolve to the same physical tables as the synonyms that are defined for DASD MANAGER PLUS or the BMCSTATS component.

If you specified ANALYZE BMCSTATS in your command syntax, the utility also issues message BMC50329E and terminates the job. Otherwise, the job continues to run. However, BMC Software recommends that you correct the problem in either case.

User Response: Change the synonyms to point to the same tables. Refer to the *Utility Products for DB2 Customization Guide* for the steps to complete this task. Resubmit the utility job.

BMC50329E **UTILITY SYNONYMS DO NOT RESOLVE TO SAME TABLES AS THOSE USED BY BMCSTATS. OPTION '*option_name*' CAN NOT BE USED**

Explanation: The synonym names that you defined for the utility do not resolve to the same tables as the synonyms that you defined for DASD MANAGER PLUS or the BMCSTATS component. You cannot use the command option ANALYZE BMCSTATS if the synonyms do not resolve to the same tables.

User Response: Remove ANALYZE BMCSTATS from the command statement or change the synonyms to point to the same tables. Refer to the *Utility Products for DB2 Customization Guide* for the steps to complete this task. Resubmit the utility job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50330I **UNABLE TO DETERMINE BASE TABLE USED FOR BMCSTATS SYNONYM '*owner.synonym_name*'**

Explanation: The named BMCSTATS synonym does not exist. The DASD MANAGER PLUS component of the solution was not installed completely.

User Response: Install the DASD MANAGER PLUS component of the solution completely, including creating the synonyms. For more information, see the *Administrative Products for DB2 Customization Guide.*

BMC50331S ***task_no*.: UNABLE TO ALLOCATE DATASET '*dsname*'**

Explanation: The utility cannot dynamically allocate the specified DB2 data set. The task number identifies the task associated with the error.

User Response: Check the job log for dynamic allocation error messages and check the STOGROUP for unlike device types. If you understand the problem, correct it and restart the job. Contact BMC Software Customer Support.

BMC50332S ***task_no*.: UNABLE TO OPEN DATASET '*dsname*'**

Explanation: The utility cannot open the specified DB2 data set. The task number identifies the task associated with the error.

User Response: Check the job log for more information. If this message occurs during REORG PLUS reload processing, ensure that the data set is defined with the REUSE option. If you understand the problem, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50333I ***task_no*.: CONNECT TO DATASET '*dsname*' FAILED WITH RC = *rc*, RETRYING**

BMC50333S ***task_no*.: CONNECT TO DATASET '*dsname*' FAILED WITH RC = *rc***

Explanation: The utility received a nonzero return code when it attempted to connect to the specified DB2 data set. The task number identifies the task associated with the error. If the utility issued the I version of the message, it retries the connection.

User Response: If the I version of the message was issued, no user action is required. Otherwise, check prior messages or the job log to determine the problem. If you need additional assistance, contact BMC Software Customer Support.

BMC50334S ***task_no*.: UNABLE TO CLOSE DATASET '*dsname*'**

Explanation: The utility cannot close the specified DB2 data set. The task number identifies the task associated with the error.

User Response: Check the job log for more information. If you understand the problem, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50335S ***task_no*.: DISCONNECT FROM DATASET '*dsname*' FAILED WITH RC = *rc***

Explanation: The utility received a nonzero return code when it attempted to disconnect from the specified DB2 data set. The task number identifies the task associated with the error.

User Response: Check prior messages or the job log to determine the problem. For more help, contact BMC Software Customer Support.

BMC50336S ***task_no*.: UNABLE TO DEALLOCATE DATASET '*dsname*'**

Explanation: The utility cannot deallocate the specified DB2 data set. The task number identifies the task associated with the error.

User Response: Check the job log for more information. If you understand the problem, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50337S ***task_no*.: UNABLE TO CALCULATE BUFFERS FOR DATASET '*dsname*'**

Explanation: The utility cannot obtain information to calculate the buffer requirements for the specified DB2 data set. The task number identifies the task associated with the error.

User Response: Check the job log for more information. Correct the problem and restart the job.

BMC50338S ***task_no*.: GENCB BLK = ACB FOR DATASET '*dsname*' FAILED WITH RC = *rc***

Explanation: The utility received a nonzero return code from a GENCB request. The task number identifies the task associated with the error.

User Response: Ensure that your region size is large enough. If region size is not the problem, contact BMC Software Customer Support.

BMC50339S ***task_no*.: GENCB BLK = RPL FOR DATASET '*dsname*' FAILED WITH RC = *rc***

Explanation: The utility received a nonzero return code from a GENCB request. The task number identifies the task associated with the error.

User Response: Ensure that your region size is large enough. If region size is not the problem, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50340S ***task_no*.: VSAM GET REQUEST FOR DATASET '*dsname*' REJECTED WITH R = *rc*, F = *data*, P = *page***

Explanation: The utility received a nonzero return code from a GET request. The task number identifies the task associated with the error. R is the return code, F is the VSAM feedback information, and P is the page number.

User Response: Check the job log, the return code, and the feedback-area information. Refer to IBM's documentation on VSAM feedback codes for an explanation of the feedback-area information. If you understand the error, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50341S ***task_no*.: VSAM PUT REQUEST FOR DATASET '*dsname*' REJECTED WITH R = *rc*, F = *data*, P = *page***

Explanation: The utility received a nonzero return code from a PUT request. The task number identifies the task associated with the error. R is the return code, F is the VSAM feedback information, and P is the page number.

User Response: Check the job log, the return code, and the feedback-area information. Refer to IBM's documentation on VSAM feedback codes for an explanation of the feedback-area information. If you understand the error, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50342S ***task_no*.: VSAM I/O ERROR ON DATASET '*dsname*', R = *rc*, F = *data*, P = *page***

Explanation: The utility encountered a VSAM I/O error. The task number identifies the task associated with the error. R is the return code, F is the VSAM feedback information, and P is the page number.

User Response: Check the job log, the return code, and the feedback-area information. Refer to IBM's documentation on VSAM feedback codes for an explanation of the feedback-area information. If you understand the error, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50343I ***SYNAD_error_message***

Explanation: The utility received the specified error message from an MVS data management access method services routine.

User Response: If you understand the problem, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50344S ***task_no*.: MODCB BLK = ACB FOR DATASET '*dsname*' FAILED WITH RC = *rc***

Explanation: The utility received a nonzero return code from a MODCB request. The task number identifies the task associated with the error.

User Response: Contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50345I ***task_no*.: WAITING ON I/O FOR DATASET '*dsname*'...**

Explanation: During task termination, this message indicates that the task is waiting for I/O to complete on the specified data set.

User Response: If multiple BMC50345I messages for the same data set appear and the job has not completed or continued, a RESERVE on the volume might contain the data set or some other system problem.

BMC50346E/I **THE VSAM DATASET DOES NOT EXIST FOR TABLESPACE *dbname.tsname* PART *part***

THE VSAM DATASET DOES NOT EXIST FOR TABLESPACE *dbname.tsname*

THE VSAM DATASET DOES NOT EXIST FOR INDEX *creator.ixname* PART *part*

THE VSAM DATASET DOES NOT EXIST FOR INDEX *creator.ixname*

Explanation: The utility attempted an operation on a table space or index that was created with DEFINE NO. The indicated VSAM data set does not exist.

If you received the I version of this message, the utility continued processing the job. If you received the E version of this message, the utility was unable to continue and terminated the job.

User Response: If you received the I version of this message, no action is required. If you received the E version of this message, refer to the reference manual for the utility that you are running for any restrictions regarding objects that were created with DEFINE NO.

BMC50347E **UTILITY EXECUTION TERMINATED BECAUSE TABLE '*creator.tbname*' CONTAINS A 'GENERATED ALWAYS AS IDENTITY' COLUMN '*column_name*' WITH A FIELD SPECIFICATION**

Explanation: The table that you specified contains an identity column that is defined as GENERATED-ALWAYS, but you also provided a field specification. The utility does not allow a field specification for this type of identity column.

User Response: Remove the field specification from your utility command and resubmit your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50348S **UTILITY EXECUTION TERMINATED BECAUSE COLUMN '*column_name*' IN TABLE '*creator.tablename*' HAS EXCEEDED ITS MAXVALUE OR MINVALUE**

Explanation: The utility computed the value to be assigned to the specified identity column and discovered that the value exceeds the range of allowed values listed in SYSIBM.SYSSEQUENCES. Most likely, the identity column is not large enough or the specified increment value is too large.

User Response: Consult the IBM DB2 manuals for information about identity columns and SYSIBM.SYSSEQUENCES.

BMC50349S **UNABLE TO RECALL MIGRATED DATASET, DSNAME = '*dsname*'**

Explanation: The utility cannot recall the specified migrated data set.

User Response: Attempt to manually recall the migrated data set. If this operation fails, check the job log for more information. If you understand the problem, correct it and restart the job. Otherwise, contact BMC Software Customer Support.

BMC50350E **TRANSLATION NOT SUPPORTED FROM CCSID *ccsid* TO CCSID *ccsid***

Explanation: The job requires data translation between CCSIDs, but translation did not occur because one of the following conditions is true:

- The translation is not defined in the SYSSTRINGS catalog table.
- The translation requires the use of a conversion procedure.
- The translation is for MIXED or double-byte character set (DBCS) data.

Message BMC50351E accompanies this message and describes the data to be translated.

User Response: Ensure that you are using the appropriate DB2 installation defaults (DSNHDECP) for the job. Eliminate the need for data translation by performing the following actions:

- Unload the data using the table’s encoding scheme.
- When constants cannot be translated for comparison, specify hexadecimal constants.

Resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50351E UNABLE TO PERFORM TRANSLATIONS(S) FOR *type name*

Explanation: The job requires data translation for the data that is described by *type* and *name*, but the translation could not be performed. BMC50350E indicates the translation that is not supported.

User Response: Eliminate the need for data translation before resubmitting the job by performing the following tasks:

- Unload the data in the table's encoding scheme.
- When constants cannot be translated for comparison, specify hexadecimal constants.

BMC50352E CANNOT LOAD TRANSPROC '*name*' FOR TRANSLATION FROM CCSID *ccsid* TO CCSID *ccsid*

Explanation: The utility cannot load the conversion procedure that is defined in the TRANSPROC column of the SYSIBM.SYSSTRINGS row that defines the translation for the listed CCSIDs.

User Response: Locate the named module and ensure that it is in the SYSLIB concatenation. Resubmit the job.

BMC50353I '*object_subtype*' '*object_type*' '*object_name*' WILL NOT BE TRANSLATED

Explanation: The utility will not translate the named object (field or column) of the listed subtype. The utility displays this message under one of the following circumstances:

- when MIXED DATA is not going to be translated

  MIXED DATA is translated only when both the source and target of the translation are mixed.

- for columns defined as FOR BIT DATA

  The utility does not translate these columns.

User Response: When your job requires translating MIXED DATA, ensure that both the field and the column are specified as MIXED. Resubmit the job.

BMC50354E UNABLE TO RESOLVE PAD CHARACTER: *field_or_column* = *name*, CCSID = *ccsid*

Explanation: The utility cannot determine the pad character to use for processing the data described in the message.

User Response: Contact BMC Software Customer Support for assistance.

BMC Software, Inc., Confidential and Proprietary Information

BMC50355W **TRANSLATION FAILED, ERRORBYTE/SUBBYTE FOUND FROM CCSID *ccsid* TO CCSID *ccsid***

Explanation: The utility attempted to translate a column and encountered one of the following situations:

- The string that the utility was translating contained an error byte.
- The string that the utility was translating contained a substitution character and you specified NOSUBS with the utility command.

User Response: Select a different CCSID to use for the translation or change the NOSUBS specification in your utility command. Resubmit the job.

BMC50360S **BMCSORT IS NOT AVAILABLE. BMCSORT MUST BE AVAILABLE IN STEPLIB, JOBLIB, OR LINKLIST.**

Explanation: BMCSORT could not be initialized because it was not installed or is not in a library that is part of a STEPLIB, JOBLIB, or LINKLIST concatenation that is active for this step.

User Response: Use the *Utilities for DB2 Customization Guide* to install BMCSORT. If BMCSORT is currently installed in a library, concatenate that library into your STEPLIB, JOBLIB, or system LINKLIST.

BMC50361S **UNABLE TO INVOKE SORT ROUTINE**

Explanation: The utility tried to invoke the sort routine but received a nonzero return code from the system LINK routine.

User Response: Check the job log for information about the failure.

BMC50362I **SORT ROUTINE TERMINATED WITH RETURN CODE = *rc***

Explanation: The sort routine failed with the specified return code.

User Response: Check the job log and prior messages for information about the failure. Check for earlier errors that caused the sort routine to fail. Correct the problem and restart the job.

BMC50363I **SORT ROUTINE TERMINATED WITH SYSTEM ABEND CODE = *abend_code*, REASON CODE = *reason***

Explanation: The sort routine failed with the specified system abend code and reason code.

User Response: Check the job log and prior messages for information about the failure. Check for earlier errors that caused the sort routine to fail. Correct the problem and restart the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50364S **SORT PROCESSES SEVERELY CONSTRAINED BY MEMORY RESOURCES. SORTING CONTINUES, BUT PERFORMANCE MAY BE IMPACTED.**

Explanation: BMCSORT found that the memory resources available are insufficient to provide the most efficient sorting. To allow the utility job to complete, BMCSORT might choose a less efficient sorting algorithm, which can elongate the elapsed time of the job.

User Response: No action is required. However, one of the following options can improve the sort efficiency for future jobs:

- Increase your region size, if possible.
- Limit the number of sort tasks by using the MAXSORTS command option.
- Limit the scope of the job you are running to reduce the amount of work that the utility needs to do.
- Message BMC50474 reports the memory that is available to the utility at the time of optimization. If the value for ABOVE as shown in BMC50474 is substantially less than the value for your REGION parameter, your system might have a memory-limiting exit active. In this case, contact your systems programmer to increase the limit, if possible.

BMC50365S ***task_no*.: LZ *process* ERROR. PARTITION = *p*, RC = *rc*, REASON = *reason***

Explanation: While processing a table space defined with the COMPRESS YES attribute, the utility encountered a problem in the specified LZ process. The task number identifies the task associated with the error.

User Response: Contact BMC Software Customer Support.

BMC50366I ***task_no*.: UNABLE TO BUILD THE COMPRESSION DICTIONARY. PARTITION = *p***

Explanation: While processing a table space defined with the COMPRESS YES attribute, the utility determined that the data was not sufficient to build a compression dictionary. The task number identifies the task associated with the error.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50369I **INDEX '*creator.ixname*' PARTITION *p* HAS BEEN PLACED IN INFORMATIONAL COPY PENDING STATUS**

Explanation: The utility has rebuilt the index. The message is a reminder that you must make an image copy of the index before you can recover it using an image copy and log records. PARTITION *p* is omitted from the message if the table space is not partitioned.

User Response: If you plan to use an image copy and log records to recover the index, first create an image copy of the index.

BMC50370I **STARTING *n* IMAGE COPY TASKS**

Explanation: The utility started the displayed number of tasks to create image copies.

User Response: No action is required.

BMC50372E **ICOPY JOB AUTHID PARAMETER '*authid1*' DOES NOT MATCH BIND AUTHID '*authid2*' OF ICOPY PLAN**

Explanation: The authorization ID that you specified in the ICOPY parameter is not the same as the owner of the ICOPY plan. It must be the same as the authorization ID used to bind the SYNC plan. To find the SYNC plan name, see *Appendix A* of the reference manual for the utility that you are running.

User Response: Correct the AUTHID parameter or correct the library that contains the installation options member. Rerun the ICOPY job.

BMC50373E **IMAGE COPY SUPPORT HAS NOT BEEN INSTALLED**

Explanation: If this message indicates a user abend 3200 (REORG PLUS) or 3400 (LOADPLUS), you can find the reason code in register 15, which is displayed by abend analysis in the job log.

Message BMC50373E indicates that the utility encountered one of the following conditions:

- You requested COPY YES and are registering the copies, but the utility is not installed with the IMAGECPY=YES option.
- You specified COPY YES and are registering the copies, but the ICOPY job to install image copy support was not run.

BMC Software, Inc., Confidential and Proprietary Information

- You specified COPY NO or YES. In addition, the utility is installed with either the LOADCPY=YES option for LOADPLUS or the REORGCPY=YES option for REORG PLUS, but the ICOPY installation job to install image copy support was not run.

User Response: Ensure that the utility was installed correctly and that the ICOPY installation job was run. You might need to modify default values in the installation options member and rerun the ICOPY installation job. Otherwise, rebind the plan specified in your PLANSYNC installation option and then run ICOPY.

If you ran these jobs before without getting this message, the most likely cause for getting the message now is that the product's SYNC plan was rebound but the ICOPY installation job was not rerun. The SYNC plan name is specified on the PLANSYNC installation option. Whenever the SYNC plan is rebound, you must rerun the ICOPY installation job to reinstate the SYSTABLEPART update support.

To find the SYNC plan name, see Appendix A of the reference manual for the utility that you are running.

BMC50374S **UNABLE TO OPEN COPY DATASET, DDNAME = '*ddname*'**

Explanation: The utility cannot open the copy data set that you specified in the named DD statement.

User Response: Check the job log for errors. Correct the problem and restart the job.

BMC50375I ***process* STARTED FOR DATASET = '*dsname*'**

Explanation: This message is for information only. The full copy, incremental copy, or inline copy process of the utility is making a copy of the data set named.

User Response: No action is required.

BMC50376I ***n* PAGES *action* DATASET = '*dsname*'**

Explanation: This message is for information only. The utility has copied to the named data set or updated in the named data set the specified number of pages. The value for *action* can be either COPIED TO or UPDATED IN.

User Response: No action is required.

BMC50377E **'*device_type*' IS AN INVALID DEVICE TYPE FOR DDNAME = '*ddname*'**

Explanation: The device type you specified is not valid or does not match the device type of the data set specified in the DD statement.

User Response: Correct the device type value and restart the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50378W IMAGE COPY REQUIRED. TABLE SPACE REMAINS STOPPED

Explanation: You did not specify COPY YES in the command statement. The utility continues processing, but completes with a return code of 4.

User Response: Take appropriate action to allow recovery of the space. For more information, see the reference manual for the utility that you are running.

BMC50379E DATASET '*dsname*' ALREADY USED FOR PREVIOUS IMAGE COPY

Explanation: You specified a data set that a previous image copy has already used.

User Response: You can use the specified data set name if you MODIFY the record from SYSCOPY to remove the entry that is causing the duplication. Then restart or rerun the job.

Keep in mind the following guidelines:

- If you are running LOADPLUS and LOADPLUS has completed creating the specified image copy, you must issue a RESTART(PHASE) to create a new image copy data set. In contrast, if LOADPLUS has *not* completed the image copy, you can change the name of the data set and restart the job.
- If you are running REORG PLUS for any type of reorganization except SHRLEVEL CHANGE and REORG PLUS has completed creating the specified image copy, you must issue a RESTART(PHASE) to create a new image copy data set. In contrast, if REORG PLUS has *not* completed the image copy, you can change the name of the data set and restart the job.
- If you are running REORG PLUS with SHRLEVEL CHANGE, you cannot restart the job. You must change the data set name and rerun the job from the beginning.

BMC50380I SPACE '*dbname.tsname*' IS IN 'CHECK PENDING' STATE. 'COPY PENDING' STATE IS NOT RESET

Explanation: The table space was not started with ACCESS(FORCE) to turn off the COPY pending flag because the CHECK pending flag was also on.

User Response: Either manually turn off COPY pending by starting the table space with ACCESS(FORCE), or by running the REPAIR utility.

BMC50381I CORRESPONDING RECOVERY SITE IMAGE COPIES SHOULD BE MADE FOR TABLESPACE '*dbname.tsname*'

Explanation: The utility found recovery site image copies (remote primary, remote backup, or both) in the SYSIBM.SYSCOPY table for the named table space.

User Response: You should make recovery copies for the table space.

BMC Software, Inc., Confidential and Proprietary Information

BMC50382I ***copy_technique* OF SECONDARY INDEX '*creator.ixname*' COMPLETED IN PRIOR RUN**

Explanation: This message offers information only. The utility copied the specified index using the software-based copy method or snapshot technology in a prior run; the copy will not be redone in the current run.

User Response: No action is required.

BMC50383E **UNABLE TO FIND COPY OR RECOVERY DDNAME '*ddname*' USED IN REGISTER OPTION**

Explanation: The utility cannot find the default ddname or the ddname that you specified in the COPYDDN or RECOVERYDDN option in your JCL.

User Response: Ensure that the ddname you specified in the REGISTER option has a matching ddname specified in your JCL.

BMC50384E **DUPLICATE DDNAME '*ddname*' FOUND IN REGISTER OPTION**

Explanation: The utility found two or more identical ddnames specified in the REGISTER option.

User Response: Ensure that each ddname that you specified in the REGISTER option is unique and has a matching ddname specified in your JCL.

BMC50385E **DUPLICATE DDNAME '*ddname*' FOUND IN COPYDDN/RECOVERYDDN OPTION**

Explanation: The utility found two or more identical ddnames specified in the COPYDDN or RECOVERYDDN option (or both).

User Response: Ensure that each ddname that you specified in the COPYDDN and RECOVERYDDN options is unique and has a matching ddname specified in your JCL.

BMC50386W **IMAGE COPY REQUIRED. TABLE SPACE REMAINS IN 'COPY PENDING' STATE**

Explanation: No image copy was made, and the table space is already in COPY pending (COPY) status.

User Response: Take appropriate action to allow recovery of the space. For more information, see the reference manual for the utility that you are running.

BMC Software, Inc., Confidential and Proprietary Information

BMC50387W **IMAGE COPY REQUIRED. TABLE SPACE SET TO 'COPY PENDING'**

Explanation: No image copy was made, and the table space is now in COPY pending (COPY) status.

User Response: Take appropriate action to allow recovery of the space. For more information, see the reference manual for the utility that you are running.

BMC50388I ***task_no*.: *copy_technique* OF SECONDARY INDEX: '*creator.ixname*' STARTED**

Explanation: This message indicates that a copy process (using standard or snapshot technology) has started for the specified secondary index data set.

User Response: No action is required.

BMC50389I ***task_no*.: *copy_technique* OF SECONDARY INDEX: '*creator.ixname*' COMPLETE**

Explanation: This message indicates that a copy process (using standard or snapshot technology) completed for the specified secondary index data set.

User Response: No action is required.

BMC50390E ***task_no*.: *copy_technique* OF SECONDARY INDEX: '*creator.ixname*' FAILED**

Explanation: This message indicates that a copy process (using standard or snapshot technology) failed for the specified secondary index. The failing copy task terminates. The remaining copy tasks, if any, continue copying other index data sets. If all index data sets are not successfully copied, the utility terminates the job.

User Response: Check the job log for errors. Correct the errors and restart the job.

BMC50391I/E **UNABLE TO LOCATE *type* DATASET, DDNAME = '*ddname*'**

Explanation: The utility cannot find the default ddname or the ddname that you specified in the command statement in your JCL.

User Response: If you receive the I (informational) version of this message, no action is required and processing continues.

If you receive the E (error) version of this message, ensure that you specify all required DD statements and that they match the ddnames that you specified for the *xxxx*DDN option of the command. If you run multiple loads or reorganizations in one job step using the BMC Software DASD MANAGER PLUS product, each SYSREC DD must have a unique DSN.

BMC Software, Inc., Confidential and Proprietary Information

BMC50392E **UNABLE TO LOCATE SUFFICIENT WORK DATASETS, DDNAME = '*ddname*'**

Explanation: The utility cannot find the work data set ddnames in your JCL.

User Response: Ensure that you specify the correct SYSUT1 or SYSUT1*nn* DD statement or the same ddnames you indicated in the WORKDDN option. You can specify one ddname for all indexes or one for each index. For more detail, see the WORKDDN command option description and the SYSUT1 allocation section of the utility reference manual.

BMC50393W **UTILITY MAY NOT BE RESTARTABLE IN THE '*phase*' PHASE BECAUSE THERE ARE INSUFFICIENT OR NO *type* DATASET(S)**

Explanation: To restart the utility in the specified phase, you must specify sufficient work data sets of the specified type. If you elect to not specify them (as indicated by this message) and the job abnormally terminates, you must recover the table space, the index, or both.

User Response: Take appropriate action to allow recovery of the table space or index space or both. For more information, see the reference manual for the utility that you are running.

BMC50394I **UNABLE TO LOCATE SORT WORK DATASETS, DDNAME = 'SORTWKNN'**

Explanation: The utility cannot locate any SORTWK ddnames or cannot obtain any allocation information about any of the SORTWK ddnames that you specified. If you did not specify the SORTNUM option or you did not use the dynamic allocation option, you might not obtain optimal processing.

User Response: No action is required. See message BMC50395I for more information.

BMC50395I **MAXSORTS SET TO 1**

Explanation: The utility cannot obtain enough information to run multiple tasks or sorts concurrently, so the utility will invoke only one task or sort. This restriction might prevent the utility from running as efficiently as possible.

User Response: No action is required. However, you might want to adjust the installation options to help you get the best performance.

BMC50396S **UNABLE TO CALCULATE BLOCKSIZE FOR DDNAME = '*ddname*'**

Explanation: The utility tried to determine the block size for the data set in the DD statement that you specified, but received a nonzero return code from a system routine.

User Response: Check the job log for errors. Ensure that you did not code FREE=CLOSE for the SYSREC ddname. Correct the problem and restart the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50397I ***phase*** **PROCESSING CONSTRAINED BY REGION SIZE**

Explanation: If more memory were available, the utility could make use of the memory to speed up the named utility phase.

User Response: No action is required. However, you might want to adjust the installation options to get the best performance. For information about performance considerations, see the reference manual for the utility that you are running.

BMC50398I ***phase*** **PROCESSING CONSTRAINED BY SORT WORK FILES**

Explanation: If more sort work file space were available, the utility could make use of the space to speed up the named phase and subsequent phase processing. The utility uses only the primary quantity when determining the number of concurrent tasks to start.

If the utility issued this message because different device types are defined for the unit being used for the SORT WORK files, the utility also issues message BMC50405I.

User Response: No action is required. However, you might want to specify larger sort work files or more sort work files to get the best performance. For information about performance considerations, see the reference manual for the utility that you are running.

If message BMC50405I accompanies this message, refer to the user response for BMC50405I for other actions you can take.

BMC50399E ***phase*** **PROCESSING UNABLE TO CONTINUE DUE TO CONSTRAINED RESOURCES**

Explanation: The utility cannot get enough memory or sort work space to process the named utility phase. Previous messages BMC50397I or BMC50398I explain the problem in more detail.

User Response: Make one or more of the following adjustments:

- Specify a larger region size.
- Specify larger sort work files or more sort work files.
- Adjust the installation options to get the best performance.

If you are using REORG PLUS to perform an index reorganization, see Chapter 4 for SORTWK calculation formulas. For information about performance considerations, see the reference manual for the utility that you are running.

BMC Software, Inc., Confidential and Proprietary Information

BMC50400I ***phase*** **PROCESSING CONSTRAINED BY INDEX WORK FILES**

Explanation: You specified only one index work file (SYSUT1). Because more than one nonclustering index is participating in the job (or clustering index, if you specified ORDER NO), you can improve performance by specifying one SYSUT1 per participating index.

User Response: Specify additional index work files (SYSUT1*nn*), one for each participating nonclustering index (plus one for each clustering index if you specified ORDER NO). For information about performance considerations, see the reference manual for the utility that you are running.

BMC50401E **INCONSISTENT NUMBER OF** ***file_type*** **DD NAMES FOR RESTART**

Explanation: The job restarted but the number of work (SYSUT1*nn*) or unload (SYSREC) files is different from the number specified in the original job.

User Response: Restore the original number of files, or end this job and submit a new job. For more information, see the description of the WORKDDN or UNLDDN option in the reference manual for the utility that you are running.

BMC50402E **DDNAME = '*ddname*' NOT FOUND FOR RESTART**

Explanation: The job restarted but one of the previous ddnames is now missing and is required.

User Response: Restore the original ddnames, or terminate the utility program and submit a new job.

BMC50403E **INPUT AND DISCARD DATASETS HAVE CONFLICTING ATTRIBUTES**

Explanation: You requested discard processing, but the RECFM, logical record length (LRECL), or DSORG attributes of the input and discard data set are not compatible.

User Response: Change the RECFM, LRECL, or DSORG values of the discard data set and resubmit the job.

BMC50404E **DUPLICATE DDNAME '*ddname*' FOUND**

Explanation: You specified the same ddname for the indicated file.

User Response: Change the ddname so that it is unique.

BMC Software, Inc., Confidential and Proprietary Information

BMC50405I **SORT WORK FILES ARE NOT THE SAME SIZE. AVERAGE SIZE *n*K ASSUMED**

Explanation: The utility determined that the sort work files are not the same size. The utility will now use average size to determine subsequent processing. Using average size can result in an abnormal termination of the utility or reduced utility performance.

User Response: No action is required. However, if other messages indicate performance constraints or abnormal termination, you probably need to ensure that all the sort work files have the same allocation quantities and are on the same device type. For REORG PLUS users, consider using the 3380 device type SORTWK space provided in the ANALYZE output.

BMC50406I **UNABLE TO OBTAIN VOLUME INFORMATION FOR DDNAME = '*ddname*', SORT WORK FILE IGNORED**

Explanation: The utility attempted, but was unable, to obtain the volume information for the specified ddname.

User Response: The probable cause for this error is assigning the sort work file to VIO. Correct the problem and rerun the utility.

BMC50407E **FOR DDNAME '*ddname*', UNLIKE DCB ATTRIBUTES FOUND FOR CONCATENATED DATA SETS**

Explanation: The utility determined that the concatenated data sets have unlike DCB attributes.

User Response: Ensure that all data sets have the same DCB attributes.

BMC50408E **DUPLICATE DATASET NAME '*dsname*' FOUND FOR DDNAMES '*ddname*' AND '*ddname*'**

Explanation: You used the same data set name in the specified DD statements.

User Response: Modify the data set name in your JCL to make it unique. Resubmit the job.

BMC50410I **NUMBER OF SORTWKNN(*ww*) FILES SHOULD BE EVENLY DIVISIBLE BY THE NUMBER OF TASKS(*tt*)**

Explanation: The number of SORTWK*nn* statements (*ww*) coded in the JCL should be evenly divisible by the number of tasks (*tt*), with a minimum of two sort work files per task.

User Response: To optimize multitasking, change the number of SORTWK*nn* statements in the JCL and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50411U ***phase* TASK NUMBER *task_no* ABNORMALLY TERMINATED WITH SYSTEM ABEND CODE = *abend_code*, REASON CODE = *reason***

Explanation: The specified subtask abended with the indicated system abend code and reason. The task number identifies the phase and specific task of the phase associated with the error.

User Response: Check prior messages or the job log to determine the problem. Review the explanation in the appropriate manual. If you understand the problem, correct it and restart the job. For other errors, contact BMC Software Customer Support.

BMC50412U ***phase* TASK NUMBER *task_no* ABNORMALLY TERMINATED WITH RETURN CODE = *rc***

Explanation: The specified subtask terminated with a nonzero return code. If this message indicates a user abend 3200 (REORG PLUS), 3400 (LOADPLUS), 3600 (UNLOAD PLUS), or 3800 (CHECK PLUS), you can find the reason code in register 15 of the job log that contains the abend summary. The task number identifies the specific task of the phase associated with the error.

User Response: Check prior messages or the job log to determine the problem. Convert the decimal-formatted reason code to hexadecimal and review the explanation in the appropriate manual. If you understand the problem, correct it and restart the job. For other errors, contact BMC Software Customer Support.

BMC50413S ***task_no*: *task* TASK STARTED WITH NO WORK ASSIGNED**

Explanation: This message indicates that an internal error has occurred. The task number identifies the task associated with the error.

User Response: Contact BMC Software Customer Support.

BMC50415S ***task_no*: UNABLE TO ESTABLISH RECOVERY ENVIRONMENT**

Explanation: The utility received a nonzero return code from an ESTAE request. The task number identifies the task associated with the error.

User Response: Ensure that your region size is large enough. If region size is not the problem, contact BMC Software Customer Support.

BMC50416W **SYSTEM DUMP CREATION FAILED WITH RETURN CODE = *rc*, REASON CODE = *reason,* TITLE = *"title"***

Explanation: After the utility abended, it attempted to generate a system dump. The return code and reason code tell why the system could not generate the dump. *Title* provides the title that the dump would have had if the dump had been generated.

User Response: For a description of the return code and reason code, and a possible course of action, see the IBM reference manual for the SDUMPX option.

BMC Software, Inc., Confidential and Proprietary Information

BMC50417S **INSUFFICIENT STORAGE FOR SORTING. MINIMUM SMCORE REQUIRED = *nn*K. RECOMMENDED SMCORE = *nn*K.**

Explanation: The sort routine determined that you do not have enough memory for the sort that your job requires. The values specified in the SMCORE installation option are incompatible with your job and system requirements.

Note: BMC Software recommends that you use the default values for the SMCORE option. The default value for this option is (0K,0K).

nn is the amount of memory that you can specify in the first subparameter of the SMCORE option. This message displays

- the minimum value that you can specify for this subparameter
- the value that BMC Software recommends that you specify, if you must use something other than the default of 0K

User Response: Edit your installation options macro or create a new one. Ensure that the SMCORE option uses either the default values (0K,0K) or the recommended value supplied in this message. If you edit an existing options macro, you must recompile and link it. If you create a new options module, edit your JCL to include the new options module name. Resubmit your job.

BMC50418U **NO SDWA, CLEANUP SUPRESSED**

Explanation: The utility abended and attempted to perform cleanup processing as part of termination processing. However, the system was not able to create the system diagnostic work area (SDWA), probably because it had insufficient memory. The utility terminated without performing cleanup processing. This message is displayed on the console only. It does not appear in the SYSPRINT listing.

User Response: Check the messages in the SYSPRINT listing to determine why the utility failed initially. If you can correct the problem and resubmit the job. If you cannot correct the problem, contact BMC Software Customer Support.

BMC50419W **INVALID UCA ADDRESS, NO DUMP PRODUCED**

Explanation: The utility had an internal error and cannot produce a dump. This message is displayed on the console only. It does not appear in the SYSPRINT listing.

User Response: Contact BMC Software Customer Support.

BMC50420I ***utility* EXIT '*exit_name*' WILL BE INVOKED AS LANGUAGE TYPE '*language*'**

Explanation: The utility will invoke the user exit named in the DSNUEXIT option as the language type specified in the message. This message is informational only. (The user-written exit routine is not part of the utility.)

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50421I ***utility*** **EXIT '*exit_name*' IS TERMINATING THIS RUN WITH RC=*rc***

Explanation: The named user-written exit routine has requested run termination via the specified return code.

User Response: Contact the author of the exit routine for the reason for termination of the exit and correct the problem. (The user-written exit routine is not part of the utility.) Resubmit or restart your job.

BMC50422E **USER VARIABLE '*varname*' HAS INVALID *name_or_data*, C'*xx*', X'*xx*'**

Explanation: The specified variable that your user-written exit routine returned has either an invalid NAME or contains invalid DATA. This messages displays the NAME or DATA in question in character (C) and hexadecimal (X) forms in this message. For rules governing user-defined variables, see the appendix on User Exits in the reference manual of the utility that you are running.

User Response: Modify the user-written exit routine to comply with the rules for variable naming and variable data. (The user-written exit routine is not part of the utility.) Resubmit or restart your job.

BMC50423E **COULD NOT LOAD THE *language* RUN TIME ROUTINE NAMED '*routine_name*'**

Explanation: The utility attempted to load the specified COBOL II or C runtime routine but could not find it.

User Response: Check that the routine is available in your STEPLIB and resubmit or restart your job.

BMC50424E ***language*** **ROUTINE '*routine_name*' RETURNED RC=*rc***

Explanation: The specified COBOL II or C runtime routine produced a nonzero return code.

User Response: Refer to the appropriate programming language documentation to determine the problem and correct it. Resubmit or restart your job.

BMC50425I **'*varname*' = '*value*' ...**

Explanation: This message displays at the beginning of the dynamic work file allocation process and is a listing of the utility's variables and their values. You can use these variables with the DSNPAT keyword. For additional information about the DSNPAT command option, see chapter 3 in the reference manual for the utility that you are running.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50426I **'*varname*' = '*value*' ...**

Explanation: This message lists the variables for the user-written exit routine and their values. You can use these variables with the DSNPAT keyword. For additional information about the DSNPAT command option, see Chapter 3 in the reference manual for the utility that you are running.

This message displays upon returning from your user-written exit routine *if* dynamic work file allocation is active. (The user-written exit routine is not part of the utility.)

User Response: No action is required.

BMC50428E **REXX EXIT '*exit_name*' RETURNED AN INVALID STOGROUP VOLUME LIST AND SET BMC_REORDER_STOGROUP_VOLUMES='YES'**

Explanation: In the named user exit at the DSRSEXIT exit point, you requested that the utility reorder the STOGROUP volumes. However, you modified the exit to return an invalid STOGROUP volume list to the utility. The most likely reason for this message is that a volume name in the list did not appear in the original STOGROUP volume list. The utility terminated the job.

User Response: Use the REXX SAY command to compare the volumes in the list that the exit returned to the utility with the original STOGROUP volume list. Delete all volumes that do not appear in the original list and resubmit the job.

BMC50430E/I ***ddtype* DDNAME PREFIX '*ddname_prefix*' TOO LONG FOR *description***

Explanation: The specified ddname prefix is too long to allow the addition of unique partition numbers or work data set numbers. The resulting ddnames will be greater than eight bytes. The description provides the purpose of the multiple data sets. If you receive the E (error) version of the message, LOADPLUS or REORG PLUS terminated the job. If you receive the I (informational) version of the message, the utility continued processing but might not be multitasking as much as it could. For more information about specifying ddnames, see the reference manual for the utility that you are running.

User Response: Specify a shorter ddname prefix in the command or in the installation options module. This prefix must allow the utility to append the maximum partition number or work data set number without exceeding eight bytes. Resubmit or restart your job, as appropriate.

If this prefix is for dynamically allocated data sets and you attempt to restart your job after specifying a shorter ddname prefix, you will continue to receive this message. On restart, the utility retains the prefix for dynamically allocated data sets that you specified originally.

BMC Software, Inc., Confidential and Proprietary Information

BMC50431E **DSNPAT MAY NOT BE SPECIFIED AS 'NONE' OR ' ' FOR DDTYPE '*ddtype*'**

Explanation: When dynamically allocating any work file or copy data set other than SORTWK, you must specify a data-set name pattern; otherwise, the utility cannot allocate your data sets. The 'NONE' or ' ' option is valid only for SORTWK data sets. The *ddtype* identifies which DSNPAT specification needs to be revised.

For additional information about the DSNPAT command option, see chapter 3 in the reference manual for the utility that you are running.

User Response: Specify a valid data set name pattern with the DSNPAT command option and resubmit or restart your job.

BMC50432E **DSNPAT FOR DDTYPE '*ddtype*' IS INVALID, DSN WOULD EXCEED 44 BYTES IN LENGTH**

Explanation: A pattern that you specified with the DSNPAT command option or specified in your installation option will result in a data set name greater than 44 bytes. The *ddtype* identifies which DSNPAT specification needs to be revised.

For additional information about the DSNPAT command option, see chapter 3 in the reference manual for the utility that you are running.

User Response: Change your DSNPAT specification and resubmit or restart your job.

BMC50433E **DSNPAT FOR DDTYPE '*ddtype*' IS INVALID, SUBSTITUTION VARIABLE NAME IS LONGER THAN 9 BYTES '*varname*'**

Explanation: A user-defined variable name that you specified in your DSNPAT command or installation option is invalid because its length is greater than 9 bytes, including the ampersand or underscore (X'6D'). The *ddtype* identifies which DSNPAT specification needs to be revised.

For additional information about the DSNPAT command option, see chapter 3 in the reference manual for the utility that you are running. For more information about rules for user-defined variable names, see the appendix about user exits in the reference manual.

User Response: Change your DSNPAT specification and resubmit or restart your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50434E **DSNPAT FOR DDTYPE '*ddtype*' IS INVALID, VARIABLE '*varname*' IS NOT DEFINED**

Explanation: The specified variable was not defined during user exit processing. The *ddtype* identifies which DSNPAT specification is affected.

For additional information about the DSNPAT command option, see chapter 3 in the reference manual for the utility that you are running. For more information about user-defined variables, see the appendix about user exits in the reference manual.

User Response: Use a different variable name in your DSNPAT specification or contact the author of your user-written exit routine. (The user-written exit routine is not part of the utility.) Resubmit or restart your job.

BMC50435E **DATA SET NAME CONTAINS INVALID CHARACTER AT POSITION *n*, DD='*ddname*', DSN='*dsname*'**

Explanation: The data set name *dsname* for DDNAME *ddname* has an invalid character at position *n*. This problem can occur when you use the DSNPAT command to specify a pattern or, in LOADPLUS, when you use the INDSN command to specify input data sets.

For additional information about these command options, see the reference manual for the utility that you are running. For more information about user-defined variables, see the information about user exits in the reference manual.

User Response: Change the data set specified in INDSN, or change your DSNPAT specification. If a user-defined variable is causing the problem, contact the author of your user-written exit routine. (The user-written exit routine is not part of the utility.) Resubmit or restart your job.

BMC50436E **DATASET NAME CONTAINS INVALID NODE AT POSITION *n*, DD='*ddname*', DSN='*dsname*'**

Explanation: The pattern that you specified with the DSNPAT command option or specified in your installation option resulted in a data set name that contains an invalid node. The invalid node is identified by its position *n*.

For additional information about the DSNPAT command option, see chapter 3 in the reference manual for the utility that you are running. For more information about user-defined variables, see the appendix about user exits in the reference manual.

User Response: Change your DSNPAT specification or, if the problem is with a user-defined variable, contact the author of your user-written exit routine. (The user-written exit routine is not part of the utility.) Resubmit or restart your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50437E SORTWORK DATASETS MUST BE ALLOCATED ON DASD UNITS, UNIT(*unit_name*) IS INVALID

Explanation: SORTWK data sets must be allocated on DASD units. The unit that you specified with your UNIT command option or installation option for SORTWK data set dynamic allocation is not a valid DASD unit. For details about the UNIT command option, see chapter 3 in the reference manual for the utility that you are running.

User Response: Change your UNIT command option or installation option for SORTWK to specify a valid DASD unit. Resubmit or restart your job.

BMC50438E DYNAMIC WORK FILE ALLOCATION SUPPORTS DASD AND TAPE UNITS ONLY, UNIT(*unit_name*) FOR DDTYPE '*ddtype*' IS INVALID

Explanation: Dynamically allocated data sets must be allocated on DASD or tape units. The unit that you specified with your UNIT command option or installation option is not a valid DASD or tape unit. The *ddtype* identifies which UNIT specification needs to be revised. For details about the UNIT command option, see chapter 3 in the reference manual for the utility that you are running.

User Response: Change your UNIT command option or installation option to specify a valid DASD or tape unit. Resubmit or restart your job.

BMC50439E UNABLE TO CONTINUE. NUMBER OF REQUIRED TAPE DEVICES (*nn*) EXCEEDS MAXTAPE SPECIFICATION (*mm*)

Explanation: The utility requires *nn* number of tape devices to complete dynamic work file allocation. The number that you specified in your MAXTAPE command option or specified in your MAXTAPE installation option (*mm*) limits the number of available tape devices to less than the number required.

For information about the MAXTAPE option and the related THRESHLD and UNIT options, see chapter 3 in the reference manual for the utility that you are running.

User Response: If additional tape devices are available on your system, use the MAXTAPE command option to increase the number of tape devices that the utility can use for dynamic work file allocation. Alternatively, you can change one of the following options:

- your THRESHLD command option or installation option to specify a different threshold
- your UNIT command option or installation option to specify a different or secondary unit name

Resubmit or restart your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50440E **UNABLE TO CONTINUE. SMS SUBSYSTEM NOT ACTIVE, BUT SOME DYNAMIC WORK FILE ALLOCATIONS SPECIFIED AS SMS**

Explanation: The SMS subsystem is not active on your system, but you specified SMS in your dynamic allocation command options or the value of one or more of the SMS installation options is YES.

You can find information about installation options in Appendix A in the reference manual for the utility that you are running. For details about the SMS command option, see chapter 3 in the reference manual.

User Response: Change your SMS command options or installation options to a value of NO and resubmit or restart your job.

BMC50441E **'UNIT' SPECIFICATION FOR DDTYPE '*ddtype*' CANNOT BE BLANK**

Explanation: If you request that the utility perform a non-SMS dynamic allocation, you must specify a unit for each data set to be allocated. You receive this message if dynamic allocation is active for your job, you have set SMS NO, but have not defined a UNIT for the specified *ddtype*.

You can find information about installation options in Appendix A in the reference manual for the utility that you are running. For details about the SMS and UNIT command options, see chapter 3 in the reference manual.

User Response: Provide a valid unit name with your UNIT command option or installation option. If your intent is to have the utility perform an SMS allocation, do not change your UNIT option, but change your SMS command option or installation option to a value of YES. Resubmit or restart your job.

BMC50442E ***nn* 'SIZEPCT' VALUE FOR DDTYPE '*ddtype*' IS NOT GREATER THAN ZERO**

Explanation: You specified a value for the SIZEPCT command or installation option that is not greater than zero. The *ddtype* identifies which SIZEPCT specification needs to be revised. For details about the SIZEPCT command option, see chapter 3 in the reference manual for the utility that you are running.

User Response: Change the appropriate SIZEPCT command option and resubmit or restart your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50443E **DSNPAT FOR DDTYPE '*ddtype*' IS INVALID BECAUSE IT CONTAINS EMBEDDED BLANKS**

Explanation: A pattern that you specified with the DSNPAT command option or specified in your installation option contains embedded blanks. The *ddtype* identifies which DSNPAT specification needs to be revised.

For additional information about the DSNPAT command option, see chapter 3 in the reference manual for the utility that you are running. For more information about user-defined variables, see the appendix on User Exits in the reference manual.

User Response: Change the appropriate DSNPAT command option or installation option, omitting any embedded blanks, and resubmit or restart your job.

BMC50444E **UNABLE TO CONTINUE, DUPLICATE DATASET EXISTS FOR DD='*ddname*', DSN='*dsname*'**

Explanation: During dynamic allocation, a pattern that you specified with the DSNPAT command option or specified in your installation option has resulted in a data set name that is already cataloged on your system. This message provides the affected ddname and DSNAME.

User Response: Change the appropriate DSNPAT command option or installation option, ensuring that it will generate unique names for your data set. Resubmit or restart your job.

BMC50445I **utility DYNAMIC FILE ALLOCATION REPORT**

BMC50446I **UNIT OR KBYTES KBYTES ALOC ALOC**

BMC50447I **DDNAME DSNAME DATACLAS MGMTCLAS STORCLAS PRI SEC PRI SEC**

BMC50448I ***ddname dsname unit/class pk sk pa sa tlr***

Explanation: These messages provide a report of the results of dynamic work file allocation on your job. Messages BMC50445I, BMC50446I, and BMC50447I provide the heading information and multiple BMC50448I messages provide the allocation data. A separate BMC50448I message is issued for each data set and provides the following information:

- *ddname*—ddname
- *dsname*—data set name
- *unit/class*—unit name (for non-SMS allocation) or data class, management class, or storage class name (for SMS allocation)

  DATACLAS is preceded by an asterisk (*) if you specify SMSUNIT=YES.
- *pk*—primary allocation in kilobytes

BMC Software, Inc., Confidential and Proprietary Information

- *sk*—secondary allocation in kilobytes
- *pa*—primary allocation in tracks or records
- *sa*—secondary allocation in tracks or records
- *t/r*—whether allocation is in tracks or records

You see this report when the utility completes dynamic work file allocation processing. You also see this report if you received message BMC50439E or message BMC50449E, indicating that the utility could not complete dynamic allocation processing. In those cases, the report is displayed to help you determine the cause of the failure.

If you specify SMSUNIT YES, the utility puts an asterisk (*) before the value in the DATACLAS position. If the report shows *DATACLAS 1, the utility passed unit 1 to SMS. If the report shows *DATACLAS 2, the utility passed unit 2 to SMS.

The allocation value may include a 'T' or 'M' , which signify as multipliers. For example, if the prefix before the value includes a T, the record value is a multiple of 1000. If the prefix includes an M, the record value is a multiple of 1,000,000.

User Response: No action is required.

BMC50449E DYNAMIC ALLOCATION FAILED, SEE JOBLOG FOR MESSAGES

Explanation: Dynamic allocation processing failed and your job terminated.

User Response: Check prior messages or the job log to determine the problem. If the job log contains messages BMC50445I through BMC50448I, the listed work file is the one that caused the allocation process failure. If you understand the problem, correct it and resubmit or restart your job. For other errors, contact BMC Software Customer Support.

BMC50450E UNIT '*unit*' SPECIFIED FOR DDTYPE '*ddtype*' NOT FOUND IN ELIGIBLE DEVICE TABLE (EDT)

Explanation: The unit that you specified with your UNIT command option or in your installation option is not valid for your system. The variable *ddtype* identifies which UNIT specification is incorrect.

User Response: Contact your system programmer to determine which units are valid for your system. Change the UNIT command option or installation option appropriately. Then, resubmit or restart your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50451E **FOR DDNAME '*ddname*' UNABLE TO ALLOCATE TOTAL REQUIRED *n* TRACKS DUE TO MAXEXTSZ *m* TRACKS**

Explanation: The utility requires *n* tracks for the specified ddname. However, the value that you specified for MAXEXTSZ (translated to *m* tracks) is insufficient to size the primary and secondary quantities. The utility terminated the job.

User Response: Increase the value that you specified for MAXEXTSZ, or decrease the value that you specified for SIZEPCT so that the utility can allocate at least *n* tracks. Resubmit the job.

BMC50452E **FOR DDNAME '*ddname*' CALCULATED SECONDARY QUANTITY *n* TRACKS EXCEEDS MAXEXTSZ *m* TRACKS**

Explanation: After the utility applied the SIZEPCT value, the amount of space that it calculated for the primary quantity was within the MAXEXTSZ limit. However, the amount of space it needed for the secondary quantity exceeded the MAXEXTSZ limit. The utility terminated the job.

Note: This problem occurs only when you specify a large number for the second SIZEPCT value.

User Response: Decrease the second value for the SIZEPCT option. Resubmit the job.

BMC50453I **FOR DDNAME '*ddname*' CALCULATED PRI/SEC OF *pri/sec* ALTERED TO *pri/sec* FOR MAXEXTSZ *n* TRACKS**

Explanation: The utility changed the values that it calculated for the primary and secondary quantities so that they are within your specified limit for the MAXEXTSZ option.

User Response: No action is required.

BMC50454I **FOR DDTYPE '*ddname*' MULTIPLE DATASET ANALYZE BYPASSED BECAUSE SPECIFIED DDN PREFIX TOO LONG**

Explanation: The ANALYZE phase completed, but it was not able to generate all of the information for the report because the specified ddname was too long.

User Response: Specify a shorter ddname prefix in the command or the installation options module. This prefix must allow the utility to append the maximum partition number or work data set number without exceeding eight bytes.

BMC Software, Inc., Confidential and Proprietary Information

BMC50455E **SPECIFIED DDN PREFIXES '*prefix1*' AND '*prefix2*' ARE TOO SIMILAR**

Explanation: For dynamic allocation of work files, the utility must be able to distinguish the ddnames specified in the JCL for one type of work file from those specified for another type of work file. The prefixes that this message names do not create file names that enable the utility to make the distinction.

If the prefixes are different only because one prefix has additional trailing characters (for example, BMC and BMCWK), the trailing bytes of the longer prefix must contain at least one non-numeric byte.

User Response: Change your prefix specification and resubmit your job.

BMC50456S ***task_no*: INVALID DATA FOUND IN FILE, DDNAME = '*ddname*'**

Explanation: The utility encountered invalid data in the specified work data set file. The task number identifies the task associated with the error.

User Response: Ensure that you did not change any of the work data set files after preloading or unloading is complete or before a restart. If you did not change the data sets in any way, contact BMC Software Customer Support. Depending on which phase was executing when the error was detected, you might have to recover the associated DB2 data sets.

BMC50457E **GDG BASE > 35 CHARACTERS, DDNAME '*ddname*' DSN '*dsname*'**

Explanation: The utility attempted to create a base generation data group (GDG) for the specified ddname. However, when the utility expanded the data set name by using the pattern from the DSNPAT installation or command option, the resulting base name was too long. (The utility makes the variable substitutions and uses everything up to the open parenthesis to create the base name.)

User Response: Shorten the pattern in the DSNPAT installation or command option so that the expanded data set name is less than or equal to 35 characters. Then, resubmit the job.

BMC50458E **GDG BASE COULD NOT BE DEFINED, DDNAME '*ddname*' DSN '*dsname*'**

Explanation: The utility cannot create a base generation data group (GDG) for the specified ddname.

User Response: Take the appropriate action based on the additional information that is displayed in the job log. Then, resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50459E **UNABLE TO CONTINUE - 'DEFINE NO' INDEX *creator.ixname* PART *partition_number* MATERIALIZED DURING PROCESSING**

Explanation: The utility started SHRLEVEL CHANGE processing of a table space that contained at least one nonmaterialized index with the DEFINE NO attribute. During processing, the index was materialized, causing the utility to terminate.

User Response: The table space remains in its original status. You cannot restart the utility from this point. You must initiate the utility from the beginning.

BMC50460E **'*jcl_statement*'' NOT VALID IN JCL WITH '*command_option*' OPTION**

Explanation: One of the statements in your JCL is not valid with the specified command option. For example, this error occurs in LOADPLUS when your JCL contains a SYSREC DD statement and your LOAD command includes the INDSN option.

User Response: Correct your JCL or utility command and rerun the job.

BMC50466E **TEMPORARY DATA SET NOT *allowed_or_recommended* FOR DDNAME = '*ddname*'**

Explanation: The utility determined that the file allocated to the specified ddname is not allocated properly. You should not use the utility's work data sets that are allocated as temporary data sets. A temporary data set is defined as one whose normal or abnormal DISP is one of the following values:

- DELETE
- NEW,PASS
- OLD,PASS and the original status is not OLD.

(For additional information about the treatment of temporary data sets by the product, see Chapter 2 of the reference manual for the utility that you are running.) If your installation options specify FILECHK=FAIL, the utility issues the NOT ALLOWED version of this message and terminates processing. If your installation options specify FILECHK=WARN, the utility issues the NOT RECOMMENDED version of this message and continues processing. If processing continues after this warning, unpredictable errors can occur due to invalid data in the work data set.

If you specified SHRLEVEL CHANGE, you get the NOT ALLOWED form of this message if any full or incremental copy data sets are allocated as temporary data sets.

User Response: Correct the allocation error specified in the message and restart the utility with the same utility ID.

BMC Software, Inc., Confidential and Proprietary Information

BMC50467S **DATASET VERIFICATION *n* FAILED FOR DDNAME = '*ddname*'**

Explanation: The utility opened a work data set for input processing and determined that it is not the same work data set that the utility created during output processing. The utility might issue message BMC50466E with this message.

User Response: Correct the error and restart the utility with the same utility ID.

BMC50470I ***install_option=value value value***

Explanation: This message displays each dynamic work file allocation installation parameter and its current value. The utility issues one message for each parameter, but shows the parameter value for all DDTYPEs in a single message. For example, you might see the following messages:

```
BMC50470I   DDTYPE=   WORK   UNLOAD   SORTWORK
BMC50470I   ACTIVE=   YES    YES      YES
```

User Response: No action is required.

BMC50471I ***install_option=value***

Explanation: This message displays each parameter and its current value from the installation options. The utility issues one message for each parameter.

The message also displays the names of the BMC Software tables that were specified at installation time. If the value of the LOADDECP installation option is YES, the utility also includes the values from the DSNHDECP module that the product is using.

User Response: No action is required. For information about performance considerations, see the reference manual for the utility that you are running.

BMC50472I **UNABLE TO LOCATE SMS INTERFACE ROUTINES, SMS SUPPORT UNAVAILABLE**

Explanation: The utility cannot find either the SMS support interface module IGWAQSMS, or modules IGWASYS and IGWASMS. Therefore, it determined that IBM's Storage Management Subsystem is not installed on your system. These modules normally reside in SYS1.CSSLIB.

User Response: If you do not require SMS support, no further action is required. If you need SMS support, ensure that SYS1.CSSLIB is in your system LINKLIST or allocated as part of your JOBLIB or STEPLIB.

BMC Software, Inc., Confidential and Proprietary Information

BMC50473I SMS SUBSYSTEM NOT ACTIVE, SMS SUPPORT UNAVAILABLE

Explanation: LOADPLUS found either the SMS support interface module IGWAQSMS, or modules IGWASYS and IGWASMS. However, it determined that IBM's Storage Management Subsystem is not active on your system.

User Response: If you do not require SMS support, no further action is required. If you do need SMS support, ensure that SMS is active before you run the utility.

BMC50474I BELOW 16M = *n*K, ABOVE 16M = *m*K, CPUS = *c*

Explanation: This message tells you the amount of memory and number of CPUs that the utility found available for processing.

User Response: No action is required. However, you can use this information to help improve performance. For more information about performance considerations, see the reference manual for the utility that you are running.

BMC50475I MAX TASKS = *t*, PARTITIONS PER TASK = *p*, SORTWKS PER TASK = *s*

Explanation: This message tells you how the utility has determined that it will perform processing. The value *t* indicates the maximum number of concurrent tasks in which multitasking can be utilized. The number of tasks does not include other processes that always use a single task. The value *p* indicates the maximum number of partitions that the utility can process per task based on the available resources. The value *s* indicates the number of sort work files assigned to each task.

User Response: No action is required. However, you can use this information to help improve performance. For more information about performance considerations and performance-related messages, see the reference manual for the utility that you are running.

BMC50476I DDNAME = '*ddname*', I/OS = *i*, I/O WAITS = *w*, RDB LOCK WAITS = *r*

Explanation: The utility gathered these statistics during processing for the specified file. The *w* value indicates the number of I/O waits and *r* is the data set contention count.

User Response: No action is required. However, you can use this information to help improve performance. For more information about performance considerations, see the reference manual for the utility that you are running.

BMC50477I *task_no*: PARTITION = *p*, ROWS/KEYS = *n*, I/O WAITS = *w*, DDNAME = *ddname*

Explanation: The utility gathered these statistics during processing for the specified partition. If *p* is zero, it indicates a nonpartitioned table space. The task number identifies the task from which the statistics were generated.

User Response: No action is required. However, you can use this information to help improve performance. For more information about performance considerations, see the reference manual for the utility that you are running.

BMC Software, Inc., Confidential and Proprietary Information

BMC50478I ***task_no*: RDB LOCK WAITS = *r***

Explanation: The utility gathered this statistic during unload processing. The variable *r* indicates the total data set contention count.

User Response: No action is required. However, you can use this information to help improve performance. For more information about performance considerations, see the reference manual for the utility that you are running.

BMC50481I ***task_no*: *task_name* TASK COMPLETE. ELAPSED TIME = *time***

Explanation: The named task completed with the given elapsed time.

User Response: No action is required.

BMC50482I ***task_no*: *task_name* COMPLETE. ELAPSED TIME = *time***
DSN = '*dsname*'

Explanation: The named task completed with the given elapsed time.

User Response: No action is required.

BMC50483I ***ddtype* DSNPAT=*pattern***

User Response: This message displays the data set name pattern for the data set that this utility will dynamically allocate. The *ddtype* identifies the data set type to which the pattern will apply.

User Response: No action is required.

BMC50484I **ESTIMATED CARDINALITY OF *object object_name* = *n* AVG SORTWK ROW LENGTH = *s* AVG UNLOAD ROW LENGTH = *u***

Explanation: The ANALYZE phase estimated or exactly calculated the cardinality of the specified table in the nonpartitioned table space or the specified partition of the partitioned table space. It also estimated the average row length. ESTIMATED is omitted from this message when the calculation is exact.

User Response: No action is required.

BMC50485I **ESTIMATED CARDINALITY OF PART *part* = *number***

ESTIMATED CARDINALITY OF TABLE *tbname* = *number*

Explanation: The ANALYZE phase estimated or exactly calculated the cardinality of the specified partition or table. ESTIMATED is omitted from this message when the calculation is exact.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50489I **-DISPLAY COMMAND ISSUE AT - *mm*/*dd*/*yyyy* *hh*:*mm*:*ss***

Explanation: This message displays the date and time that the utility issued a DB2 DISPLAY LOCKS command. The DISPLAY LOCKS command, which the utility issues immediately after a drain timeout, shows the locks that are held at that time.

User Response: No action is required.

BMC50490E **'*module*' MODULE INCOMPATIBLE WITH THIS RELEASE**

Explanation: The specified options module was assembled with a macro from a different version of the utility than the one that you are using.

User Response: To use this release, you must install the corresponding installation options macro and reassemble. See the *Utilities for DB2 Customization Guide*.

BMC50491I **RECALL IN PROGRESS FOR DATASET '*dsname*'**

Explanation: The utility detected that one of the data sets that is participating in the utility job has been migrated. The utility is recalling the data set. The utility issues BMC50492I when the recall completes successfully.

User Response: No action is required.

BMC50492I **RECALL SUCCESSFUL FOR DATASET '*dsname*'**

Explanation: This message follows message BMC50491I. The utility successfully recalled the migrated data set.

User Response: No action is required.

BMC50497I **HEXDUMP GENERATED BY *routine_name* OF *area*, LENGTH=*length***

Explanation: The utility issues this message when needing a block of storage for diagnostic purposes. The named routine requested a dump of the specified area for the indicated length.

Print lines that show the contents of the area follow the header line of this message. Each line can display up to 32 bytes of memory, and each line is in the following format:

ADDRESS OFFSET DATA IN HEX FORMAT DATA IN PRINT FORMAT

- ADDRESS—8-byte hexadecimal address of the first byte that is displayed on this line
- OFFSET—hexadecimal offset of the first byte of this line from the first byte in the area

BMC Software, Inc., Confidential and Proprietary Information

- DATA IN HEX FORMAT—data bytes displayed in hexadecimal format that was two characters per byte
- DATA IN PRINT FORMAT—the data bytes in print format

User Response: No action is required.

BMC50498U **UNEXPECTED ERROR *n* IN MODULE '*module*', *info1=value*, *info2=value***

Explanation: This message indicates that an internal error occurred. This message displays information that assists BMC Software Customer Support. If the information in *info2* is "LASP INSTRUCTION FAILED", an enabled PER slip is preventing the utility from running.

User Response: If the information in *info2* is "LASP INSTRUCTION FAILED", disable the PER slip and rerun the utility. Otherwise,
Contact BMC Software Customer Support.

BMC50499U/W **UNEXPECTED ERROR *n* IN MODULE '*module*'**

Explanation: This message indicates that an internal error has occurred and usually precedes a utility abend.

User Response: If the message severity code is unrecoverable (U), check the abend reason code. You can find a listing of the abend reason codes at the beginning of Appendix C of the reference manual for the utility you are running. If you are unable to resolve the problem, contact BMC Software Customer Support.

If the message severity code is warning (W), see any additional associated messages for the correct response.

BMC50500I **DRQFUNC(*function_code*) = *function_description*, DRQEC(*error_code*) = *error_description*, DB2 REASON CODE (IF AVAILABLE) = *reason_code***

Explanation: The utility encountered an unrecoverable error and terminated the job. The *function_description* displays the function that was running at the time that the error (displayed in *error_description*) occurred. If the *error_description* displays "LASP INSTRUCTION FAILED", an enabled PER slip is preventing the utility from running.

User Response: If a DB2 reason code is displayed, you can find information about it in the IBM DB2 messages and codes manual. If you can determine the cause of the error from the information in the message, correct the problem and resubmit the job. If the *error_description* displays "LASP INSTRUCTION FAILED", disable the PER slip and rerun the utility. Otherwise, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50501I DB2 OBJECT STATISTICS

Explanation: This is the heading for the statistics reported in subsequent messages associated with the named DB2 object.

User Response: No action is required.

BMC50502I THE BMCSTATS FOR OBJECT '*object_name*' ARE MISSING OR INCOMPLETE. 'ANALYZE SAMPLE' WILL BE SUBSTITUTED.

Explanation: The utility was unable to use the statistics from BMCSTATS for the named table space or index because the statistics were missing or incomplete. The utility will run the job using ANALYZE SAMPLE.

User Response: Run BMCSTATS to populate the statistics tables before rerunning the job.

BMC50503I STATISTICS FOR *object* '*object_name*' NOT AVAILABLE

Explanation: The utility cannot gather and report statistics for this table space, table, or index because of the type of load (LOADPLUS) or reorganization (REORG PLUS) that you are performing. The utility cannot produce object statistics when only some of the partitions of a partitioned table space are loaded or reorganized. In addition, LOADPLUS cannot produce statistics when you specify RESUME YES or PRELOAD LOAD and duplicates exist in unique secondary indexes.

User Response: No action is required.

BMC50504I INDEX STATISTICS ARE FOR THE PROCESSED PARTITIONS ONLY

Explanation: Because all partitions were not loaded (LOADPLUS) or reorganized (REORG PLUS), the utility gathered index statistics only for the loaded or reorganized partitions. The utility does not use the statistics to update the BMCSTATS tables or the DB2 catalog.

User Response: No action is required.

BMC50505I BMCSTATS UPDATED IN THE DASD MANAGER PLUS DATABASE

Explanation: This message verifies that the current statistics were placed in the DASD MANAGER PLUS statistics table. The utility takes this action when you specify BMCSTATS YES and when partial or complete statistics for the object are available.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50506I STATISTICS COLUMNS UPDATED IN THE DB2 CATALOG

Explanation: This message verifies that the current statistics were placed in the DB2 catalog. The utility takes this action when you specify UPDATEDB2STATS YES and when complete statistics for the object are available.

User Response: No action is required.

BMC50507I BMC STATISTICS COULD NOT BE WRITTEN TO THE STATISTICS DATABASE

Explanation: The utility could not locate the DASD MANAGER PLUS statistics database.

User Response: If the DASD MANAGER PLUS product is installed, ensure that the *xxx*SYNM member specifies the correct names for the DASD MANAGER PLUS statistics database. Also, ensure that the BIND process for the BMCSTATS plan completed successfully. To find the BMCSTATS plan name, see Appendix A of the reference manual for the utility that you are running for the BMCSTATS plan name.

BMC50508I DB2 AGGREGATE *object* STATISTICS NOT UPDATED. SOME PARTS HAVE NO ROW IN SYSIBM.*tablename*

Explanation: For a partitioned table space, you specified UPDATEDB2STATS YES, but at least one part of the table space did not have a row existing on the specified DB2 catalog table. The following list show the values for *object* and the corresponding *tablename*:

- TABLESPACE SYSTABSTATS
- TABLE SYSTABSTATS
- INDEX SYSINDEXSTATS

DB2 does not allow the insertion of rows into SYSIBM.SYSTABSTATS or SYSIBM.SYSINDEXSTATS. The utility updates existing rows on these two tables with statistics gathered, but cannot calculate aggregate statistics for the table space or index without information from existing rows. The utility does not update aggregate statistics.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50510I **TABLESPACE *dbname.tsname* (HARDWARE COMPRESSION IN EFFECT)**

TABLESPACE *dbname.tsname* (SOFTWARE COMPRESSION IN EFFECT)

TABLESPACE *dbname.tsname*

TABLE *creator.tbname*

INDEX *creator.ixname* (TYPE *n*)

INDEX *creator.ixname* (TYPE *n*) (TYPE ALTERED IN THIS EXECUTION)

Explanation: This is the heading for the statistics reported in subsequent messages associated with the named DB2 object. If one or more partitions are defined with COMPRESS YES, the TABLESPACE message indicates whether hardware is installed to support compression, or whether software emulation is being used. If the indexes were to be altered, the message indicates whether the alter was run.

User Response: No action is required.

BMC50511I **PART NACTIVE CARD FARIND NEARIND FULL DIRTY PACT PDRP SPACE EXTENTS DBCARD PCOMP KSAVED PSAVED**

Explanation: These are the report headings for the information given in messages BMC50510I and BMC50512I. The information is the statistics associated with a table space partition. A brief description of each heading follows.

- PART—partition number
- NACTIVE—number of active pages in the partition
- CARD—number of rows in the partition
- FARIND—number of referenced rows greater than or equal to 64 pages from the original page
- NEARIND—number of referenced rows fewer than 64 pages from the original page
- FULL—number of full pages in the partition
- DIRTY—number of modified pages since the last copy was made (should always be zero)
- PACT—percentage of space occupied by the rows of active tables
- PDRP—percentage of space occupied by the rows of dropped tables (should always be zero)

BMC Software, Inc., Confidential and Proprietary Information

- SPACE—tracks of DASD allocated to the partition
- EXTENTS—number of extents used for space allocation
- DBCARD—number of rows in the partition used to build the compression dictionary
- PCOMP—percentage of rows in the partition that are compressed (does not include rows compressed by EDITPROC processing)
- KSAVED—number of kilobytes saved by compression
- PSAVED—percentage of bytes saved by compression

User Response: No action is required.

BMC50512I ***statistics_data***

Explanation: This is the statistical data for the table space partitions related to the report and heading information given in messages BMC50510I and BMC50511I

User Response: No action is required.

BMC50513I **PARTS = *n* TABLES = *n* SEGSIZE = *n***

Explanation: This message provides statistical information about the table space.

- PARTS—number of partitions in the table space
- TABLES—number of tables in the table space
- SEGSIZE—number of pages in each segment

User Response: No action is required.

BMC50514I **NACTIVE = *n* SPACE = *n***

Explanation: This message provides statistical information about the table space or index spaces or both.

- NACTIVE—number of active pages in the table space or index spaces or both
- SPACE—tracks of DASD allocated to the table space or index spaces or both

User Response: No action is required.

BMC50521I **TABLE NAME CARD NPAGES INDREF ROWAVG PCTPAGES PCOMP**

Explanation: These are the report headings for the table information given in message number BMC50522I. A brief description of each heading follows.

- TABLE NAME—name of the table
- CARD—number of rows in the table

BMC Software, Inc., Confidential and Proprietary Information

- NPAGES—number of pages containing rows of the table
- INDREF—number of indirectly referenced rows (should always be zero)
- ROWAVG—average row length
- PCTPAGES—percentage of table space pages containing rows from this table.

  If the table space is segmented, this is the percentage of total pages in the set of segments assigned to this table.
- PCOMP—percentage of rows in the table space that are compressed (does not include rows compressed by EDITPROC processing)

User Response: No action is required.

BMC50522I ***statistics_data***

Explanation: This message provides statistical data for the tables related to the report and heading information given in message BMC50521I.

User Response: No action is required.

BMC50531I **PART NACTIVE CARD LEAFDIST FAROFF NEAROFF FULL FREE NLEAF LEVELS SPACE EXTENTS**

Explanation: These are the report headings for the index partition information given in message number BMC50532I. A brief description of each heading follows.

- PART—partition number
- NACTIVE—number of active pages in the partition
- CARD—number of rows referenced by the partition
- LEAFDIST—100 times the average number of pages between successive leaf pages of the index
- FAROFF—number of referenced rows greater than or equal to 64 pages from the original page
- NEAROFF—number of referenced rows fewer than 64 pages from the original page
- FULL—number of full pages in the partition
- FREE—number of kilobytes of free space in active pages in the partition
- NLEAF—number of active leaf pages
- LEVELS—number of levels in the index tree

BMC Software, Inc., Confidential and Proprietary Information

- SPACE—tracks of DASD allocated to the partition
- EXTENTS—number of extents used for space allocation

User Response: No action is required.

BMC50532I *statistics_data*

Explanation: This message provides statistical data for the index partition(s) related to the report and heading information given in message BMC50531I.

User Response: No action is required.

BMC50541I ON TABLE *creator.tbname* COLUMN *colname*

Explanation: This is the heading for the statistical report for the index.

- TABLE—name of table on which the index is defined
- COLUMN—first key column in index

User Response: No action is required.

BMC50542I FIRSTKEY = *key* FULLKEY = *key* NLEAF = *n*

Explanation: This message provides statistical data for the index.

- FIRSTKEY—number of distinct values in the first 8 bytes of the first key column
- FULLKEY—number of distinct values of the entire key
- NLEAF—number of leaf pages in the index

User Response: No action is required.

BMC50543I LEVELS = *n* PCTCLUST = *n*

Explanation: This message provides statistical data for the index.

- LEVELS—number of levels of the index
- PCTCLUST—percentage of clustering of the index

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50544I KEYLEN = *n* COLCOUNT = *n*

Explanation: This message provides statistical data for the index.

- KEYLEN—length of the key
- COLCOUNT—number of columns used for the key

User Response: No action is required.

BMC50545I SUBPAGE = *n* CLUSTER = *n* UNIQUE = *n*

Explanation: This message provides statistical data for the index.

- SUBPAGE—number of subpages in index
- CLUSTER—(Y=yes or N=no) the value you specified for CLUSTER when the index was created
- UNIQUE—(U=unique, P=primary, or D=duplicate) whether the index is unique, primary, or allows duplicate keys

User Response: No action is required.

BMC50546I HIGH2KEY = X '*hex_value*' LOW2KEY = X '*hex_value*'

Explanation: This message provides statistical data for the index.

- HIGH2KEY—second highest key value for the column; written in hexadecimal format
- LOW2KEY—second lowest key value for the column; written in hexadecimal format

User Response: No action is required.

BMC50547I HIGH2KEY = EBCDIC *value* LOW2KEY = EBCDIC *value*

Explanation: This message provides statistical data for the index.

- HIGH2KEY—second highest key value for the column; written in character format
- LOW2KEY—second lowest key value for the column; written in character format

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50601E **UNSUPPORTED/INVALID IDCAMS COMMAND '*command*' FOUND, DSN = *n* LINE = *line***

Explanation: You specified a command other than DEFINE, DELETE, SET, or IF-THEN-ELSE in the SYSIDCIN data set. *n* is the number of the concatenation. The first data set in the concatenation is 1, the second data set is 2, and so on.

User Response: Correct your IDCAMS command statement in the SYSIDCIN data set and resubmit the job. Refer to "SYSIDCIN Data Set" in Chapter 4 for specification guidelines.

BMC50602E **DATASET NAME NOT FOUND OR INCOMPLETE '*dsname*', COMMAND '*command*', DSN = *n* LINE = *line***

Explanation: The utility cannot find a valid data set name for the specified command. The utility considers data sets incomplete if the data set name continues to another line using the IDCAMS symbol "+" or if the NAME parameter and following left parenthesis are separated by one or more blanks. *n* is the number of the concatenation. The first data set in the concatenation is 1, the second data set is 2, and so on.

User Response: Correct your IDCAMS command statement in the SYSIDCIN data set and resubmit the job. Refer to "SYSIDCIN Data Set" in Chapter 4 for specification guidelines.

BMC50603E **DATASET NAME '*dsname*' DIFFERS FROM PREVIOUS NAME, COMMAND '*command*', DSN = *n* LINE = *line***

Explanation: For an IF command, the previous DEFINE or DELETE data set name is not the same as the *dsname* specified in the message. All data sets referenced by the DEFINE and DELETE commands in a single IF command must be the same. *n* is the number of the concatenation. The first data set in the concatenation is 1, the second data set is 2, and so on.

User Response: Correct your IDCAMS command statement in the SYSIDCIN data set and resubmit the job. Refer to "SYSIDCIN Data Set" in Chapter 4 for specification guidelines.

BMC50604I **NAME '*dsname*' DOES NOT MATCH ANY DB2 OBJECT USED, COMMAND '*command*', DSN = *n* LINE = *line***

Explanation: This is an informational message only. The specified IDCAMS command references a data set name that is not contained in the object(s) being reorganized or loaded and will be ignored. *n* is the number of the concatenation. The first data set in the concatenation is 1, the second data set is 2, and so on.

User Response: No action is required. However, if you determine that your command statement was in error, correct the problem and resubmit the job. Otherwise, the command will be ignored at processing time.

BMC Software, Inc., Confidential and Proprietary Information

BMC50605I **UNABLE TO MATCH COMMAND '*command*' WITH ANY PREVIOUS DATASET NAME, DSN = *n* LINE = *line***

Explanation: Commands that do not reference a data set name are matched with the immediate preceding command with a data set name reference. No previous command with a data set name exists for the specified command. *n* is the number of the concatenation. The first data set in the concatenation is 1, the second data set is 2, and so on.

User Response: No action is required. Refer to “SYSIDCIN Data Set” in Chapter 4 for specification guidelines.

BMC50606E **INCOMPLETE IDCAMS COMMAND '*command*', DSN = *n* LINE = *line***

Explanation: The end of the specified command or the end of file was reached before the command was complete. *n* is the number of the concatenation. The first data set in the concatenation is 1, the second data set is 2, and so on.

User Response: Correct your IDCAMS command statement in the SYSIDCIN data set and resubmit the job. Refer to “SYSIDCIN Data Set” in Chapter 4 for specification guidelines.

BMC50620E **THE OBJECT '*creator.tbname*' IS AN AUXILIARY TABLE FOR WHICH THE REQUESTED OPERATION IS NOT PERMITTED**

Explanation: You attempted an operation that is not allowed. The utility cannot execute on an auxiliary table.

User Response: No action is required.

BMC50700S **INTERNAL ERROR: FILE *file_name*, LINE *line***

Explanation: An internal error has occurred during processing.

User Response: Contact BMC Software Customer Support with the file and line information. Additional diagnostic messages and trace information can also appear. If a DD statement is present for SYSERR, a snap dump is produced that can be useful in diagnosing the problem.

BMC50701I **DIAGNOSTICS: *text***

Explanation: This message displays diagnostic information for analyzing internal errors.

User Response: See message BMC50701.

BMC50702I **LOG PROCESSING STARTED AT *start_time***

Explanation: The log control task started.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50703I **LOG PROCESSING COMPLETED, RC = *rc*, AT *start_time*, ELAPSED TIME = *elapsed_time***

Explanation: The log processing for the reorganization completed.

User Response: If the return code is non-zero, check the preceding messages.

BMC50704E **ATTACH FAILURE, RC = *rc*, TASK = *task_type*, AT *timestamp***

Explanation: The log control task cannot attach a subtask. The return code value was received from MVS attach.

User Response: Check the return code value in the MVS documentation.

BMC50705I ***text***

Explanation: The text shown is output from a facility that this product uses (for example, DB2 SQL, DB2 Call Attachment Facility, DB2 Instrumentation Facility, or IDCAMS).

User Response: Examine the context of the message to see if action is required.

BMC50706I **MONITOR TRACE CLASS (1) STARTED, TRACE NUMBER = *number***

Explanation: A trace with the specified number has been started to capture DB2 log records.

User Response: No action is required.

BMC50707E **ERROR STARTING MONITOR TRACE CLASS (1), RC = *rc***

Explanation: An error occurred while attempting to start the DB2 log capture trace.

User Response: Review the diagnostic messages that follow this message to see whether you can correct the error. If you need further assistance, contact BMC Software Customer Support.

BMC50708I **DB2CA RC=*rc* REASON CODE=*reason***

Explanation: This message displays the return code and reason code from the DB2 communications area for a failed DB2 command.

User Response: Use this data in conjunction with the other messages that precede and follow this message to determine the nature of the error.

BMC Software, Inc., Confidential and Proprietary Information

BMC50709I **MONITOR TRACE CLASS (1) TRACE NUMBER = *number* STOPPED**

Explanation: The DB2 log capture trace with the specified number has been stopped.

User Response: No action is required.

BMC50710W **ERROR STOPPING MONITOR TRACE CLASS (1), RC = *rc***

Explanation: An error occurred while attempting to stop the DB2 log capture trace.

User Response: Review the diagnostic messages that follow this message to see whether you can correct the error. You can manually terminate the trace by using the following command, where *num* is the trace number:

```
-STOP TRACE(MON) CLASS(1) TNO(num)
```

BMC50711W **UNABLE TO STOP MONITOR TRACE CLASS (1) DUE TO END-OF-TASK CONDITION**

Explanation: The utility cannot stop the DB2 log capture trace due to an end-of-task condition (for example, the job was canceled).

User Response: You can manually terminate the trace by using the following command, where *num* is the trace number.

```
-STOP TRACE(MON) CLASS(1) TNO(num)
```

BMC50712E **ERROR SETTING UP LOGSCAN ESTAE, RC = *rc***

Explanation: The initialization of an ESTAE failed with the return code indicated.

User Response: Contact BMC Software Customer Support.

BMC50713E **USER LACKS SUFFICIENT MONITORING AUTHORITY**

Explanation: The primary authorization ID of the process that runs the program must have MONITOR1 or MONITOR2 privilege.

User Response: Obtain the proper privilege or run the program with a different authorization ID.

BMC Software, Inc., Confidential and Proprietary Information

BMC50714E **ERROR PROCESSING LOG RECORDS, IFI RC = *rc***

Explanation: An error occurred during processing of a DB2 instrumentation facility interface (IFI) call.

User Response: Review the diagnostic messages that follow this message to see whether you can correct the error. If you need further assistance, contact BMC Software Customer Support.

BMC50715S **UNSUCCESSFUL CONNECT TO DB2 SUBSYSTEM/GROUP *ssid* USING PLAN *plan_name* IN LOGSCAN**

Explanation: The program cannot connect to the specified subsystem or group attach name using the plan name indicated and using the DB2 call attachment facility (CAF).

User Response: Review the accompanying DB2 CAF diagnostic messages. The plan might not be bound on the specified subsystem or you are not authorized to execute the plan. If the plan name is incorrect, check the assembled options module to determine the plan name.

BMC50716I ***task_no*: BUFFER CREATED FOR PAGE SIZE *n*K, REFRESH = *n***

Explanation: This message indicates that a buffer was created for holding pages for the specified task.

User Response: No action is required.

BMC50717I ***task_no*: RANDOM BUFFER: NUMBER OF PAGES = *n*, MINIMUM CLEAN = *n*, NUMBER TO WRITE PER I/O = *n***

Explanation: This message reports specific values that are associated with the page buffer for the specified task.

User Response: No action is required

BMC50718I ***task_no*: RANDOM BUFFER: MAXIMUM OUTSTANDING WRITES = *n*, MAXIMUM OUTSTANDING READS = *n*, HASH HEADS = *n***

Explanation: This message reports specific values that are associated with the page buffer for the specified task.

User Response: No action is required.

BMC50719I ***task_no*: SEQUENTIAL BUFFER: NUMBER OF CYLINDERS = *n*, IDLE CYLINDERS = *n*, CI'S PER CYLINDER = *n***

Explanation: This message reports specific values that are associated with the page buffer for the specified task.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50720I ***task_no*: BUFFER MANAGER STATISTICS, GET PAGE COUNT = *n*, RANDOM SEARCH COUNT = *n***

Explanation: This message reports performance counters that are associated with the specified task number.

User Response: No action is required.

BMC50721I ***task_no*: CYLINDERS WRITTEN = *n***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50722I ***task_no*: TOTAL WAITS IN BUFFER MANAGER = *n***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50723I ***task_no*: RANDOM WRITE WAITS = *n*, RANDOM READ WAITS = *n*, RANDOM SYNC WAITS = *n***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50724I ***task_no*: CYLINDER WAITS = *n*, SEQUENTIAL SYNC WAITS = *n***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50725I ***task_no*: RANDOM READS = *n*, HITS = *n***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50726I ***task_no*: PREFETCH COUNT = *n*, HITS CAUSED BY PREFETCH = *n*, HITS ON PREFETCH = *n***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50727I ***task_no*: RANDOM WRITES = *n* IN *n* GROUPS**

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50728I ***task_no*: TOTAL WAIT TIME = *time*, RANDOM WRITE TIME = *time*, RANDOM READ TIME = *time***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50729I ***task_no*: CYLINDER WAIT TIME = *time***

Explanation: This message reports performance counters associated with the specified task number.

User Response: No action is required.

BMC50730S ***task_no*: WRITE ECB ABNORMAL COMPLETION CODE = X'*rc*'**

Explanation: A write operation on a data set completed with the abnormal return code shown. The data set name is displayed in an accompanying message.

User Response: Contact BMC Software Customer Support.

BMC50731S ***task_no*: ECB ABNORMAL COMPLETION CODE = X'*rc*'**

Explanation: A write operation on a data set completed with the abnormal return code shown. The data set name is displayed in an accompanying message.

User Response: Contact BMC Software Customer Support.

BMC50732I ***task_no*: I/O REQUEST ELEMENT FLAG = X'*flag*'**

Explanation: This message shows additional information for an I/O error.

User Response: Review accompanying messages and contact BMC Software Customer Support.

BMC50733S ***task_no*: WRITE REQUEST ERROR, RC = *rc***

Explanation: A write error occurred with the return code listed.

User Response: Contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50734S ***task_no*: READ REQUEST ERROR, RC = *rc***

Explanation: A read error occurred with the return code listed.

User Response: Contact BMC Software Customer Support.

BMC50736S ***task_no*: ERROR DURING AUTHORIZATION SWITCH, RC = *rc*, FUNCTION = *function***

Explanation: The options for this utility indicate that this utility is to use DB2's RACF security authority to open DB2 data sets. A failure occurred in attempting to switch the RACF authorization ID.

User Response: If your security system is not RACF, change the option in the options module for the BMC Software utility that you are running. Ensure that the user has authority on all of the necessary data sets. If you are using RACF, contact BMC Software Customer Support.

BMC50737I ***task_no*: EXTEND SUCCESSFUL FOR DATASET *dsname***

Explanation: A VSAM EXTEND was successful for the named data set.

User Response: No action is required.

BMC50738I ***task_no*: ATTEMPTING ALLOCATION FOR NEW MULTI DATASET *dsname***

Explanation: A VSAM CREATE is being attempted for the named data set in a multi-data set table space.

User Response: No action is required.

BMC50739I ***task_no*: ALLOCATION SUCCESSFUL FOR NEW MULTI DATASET *dsname***

Explanation: The VSAM CREATE for the named data set was successful.

User Response: No action is required.

BMC50740E **FAILURE IN REORG/LOG APPLY INTERFACE, RC = *rc***

Explanation: The interface received invalid data.

User Response: Contact BMC Software Customer Support.

BMC50741I ***task_no*: LOGAPPLY STATISTICS**

Explanation: This message is the header message for statistics from a log apply task.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50742I ***task_no*: PROCESSED TRANSACTIONS**

Explanation: This heading message indicates that the counts shown in following messages refer to log records received at the log exit.

User Response: No action is required.

BMC50743I ***task_no*: CONSOLIDATED TRANSACTIONS**

Explanation: This heading message indicates that the counts shown in following messages refer to log records after multiple records for the same row have been combined.

User Response: No action is required.

BMC50744I ***task_no*: INSERTS: *count* INSERTS (COMPENSATION): *count***

Explanation: This message displays the number of insert records found in both the regular and rollback portions of the transactions.

User Response: No action is required.

BMC50745I ***task_no*: UPDATES: *count* UPDATES (COMPENSATION): *count***

Explanation: This message displays the number of update records found in both the regular and rollback portions of the transactions.

User Response: No action is required.

BMC50746I ***task_no*: DELETES: *count* DELETES (COMPENSATION): *count***

Explanation: This message displays the number of delete records found in both the regular and rollback portions of the transactions.

User Response: No action is required.

BMC50747I ***task_no*: PAGES UPDATED COUNTERS FOR *dbname.tsname***

Explanation: This message is the heading for information reported in subsequent messages (BMC50748I and BMC50749I) associated with the named table space.

User Response: See subsequent messages for the data.

BMC Software, Inc., Confidential and Proprietary Information

BMC50748I ***task_no*: PART TOTAL PAGES DATA PAGES SPACE MAPS HEADER PAGES**

Explanation: This message accompanies message BMC50747I and provides the heading for the information that follows in message BMC50749I. The information shows the number of pages updated by partition. If the table space is nonpartitioned, the part number is zero.

User Response: No action is required.

BMC50749I ***text***

Explanation: This is the page count data for the table space partitions related to the heading information given in messages BMC50747I and BMC50748I.

User Response: No action is required.

BMC50750I ***task_no*: IGNORED TRANSACTIONS DUE TO SELECT/DELETE**

Explanation: If you specified SELECT or DELETE on the utility command, the utility might display this message as each LOGAPPLY task ends. Messages BMC50744I, BMC50745I, and BMC50746I, which display counts of the number of ignored log records, follows this message.

User Response: No action is required.

BMC50751E ***task_no*: ERROR EXTENDING OUTPUT DATASET *dsname*, RC = *rc***

Explanation: A request to extend the named data set failed with the return code shown.

User Response: Check the target DASD volumes for the named data set to ensure that enough space is available for extension. If you need assistance to determine the cause of this error, contact BMC Software Customer Support.

BMC50752S **ERROR IN CALL OF EDITPROC *epname*, RC = *rc*, REASON = *reason***

Explanation: The named editproc returned the displayed error code.

User Response: Consult the editproc documentation.

BMC50753S **COMPRESSION SYSTEM *function* FAILED - RC=*rc*, REASON=*reason***

Explanation: The specified compression system function failed with the displayed return code.

User Response: Contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50756I **XBM RESPONSE BUFFER OVERFLOW**

Explanation: A message was too large for the buffer XBM response provided.

User Response: Contact BMC Software Customer Support.

BMC50770S **LOG APPLY ABNORMAL TERMINATION, SYSTEM COMPLETION CODE = *code*, RETURN CODE = *rc***

Explanation: During a SHRLEVEL CHANGE operation, the LOG APPLY component encountered an unrecoverable error. The utility terminated the job.

User Response: Contact BMC Software Customer Support.

BMC50771S **UNABLE TO ALLOCATE LOG APPLY MEMORY**

Explanation: Insufficient memory is available to allocate even a minimum set of log apply tasks.

User Response: Increase the region above the line and resubmit the job.

BMC50772S **MEMORY AVAILABLE: *size*K, *type* MEMORY REQUIRED: *size*K**

Explanation: This message follows message BMC50771S and indicates the memory available, the type of memory required, and the minimum amount required.

User Response: Subtract the amount of available memory from the amount of required memory. Multiply the result by two and increase the region size above the line by this amount. Resubmit the job.

BMC50773I ***task_no*: *task_type* TASK STARTED AT *timestamp***

Explanation: The specified log apply task started at the indicated time.

User Response: No action is required.

BMC50774S ***task_no*: UNEXPECTED TASK TERMINATION, SYSTEM COMPLETION CODE=*code*, RETURN CODE=*rc***

Explanation: A log apply task ended unexpectedly.

User Response: Check the preceding messages.

BMC50775E ***task_no*: NEW MULTI DATASET CANNOT BE ALLOCATED FOR *dbname.spacename*, MAXIMUM NUMBER OF DATASETS REACHED**

Explanation: The utility required a new multi-data set for the named space, but the space was already at its maximum number of data sets.

User Response: If the maximum number of data sets has not been reached or more space is available, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50776E ***task_no*: VSAM CREATE FOR '*dsname*' FAILED, RC=*rc***

Explanation: An error occurred during a VSAM DEFINE for the named data set.

User Response: Look for accompanying messages that indicate the nature of the error. Correct the problem and resubmit the job.

BMC50777E ***task_no*: UPDATES TO DATASET '*dsname*' WILL EXCEED THE MAXIMUM SIZE OF *n* PAGES**

Explanation: Update activity to a partitioned table space during the LOGAPPLY phase caused the utility to expand the partition beyond the physical size of the object. This might have occurred because changes were made to the definition of the object before running the utility. Examples are changes in compression or altering the MAXROW value. The utility terminated the job.

User Response: Examine the changes made to the object definition to determine the probable cause of the problem.

BMC50778I LOG RECORD QUEUE SIZE AT OR BELOW THRESHOLD

Explanation: The number of records in the log record queue is at or below the user-specified threshold (LOGTHRESHLD). Therefore, the LOGFINAL phase has started.

User Response: No action required.

BMC50779I LOG APPLY CAN BE COMPLETED WITHIN MAXRO TIME SPECIFIED

Explanation: The product estimates that the LOGAPPLY phase can complete within the user-specified maximum read-only time (MAXRO value). Therefore, the LOGFINAL phase has started.

User Response: No action is required.

BMC50780I LONG LOG DETECTED AT *timestamp*, WILL DELAY UNTIL *timestamp*

Explanation: Log records are arriving faster than they can be applied, causing a long log condition. The product delays until the time specified before taking the action specified on the LONGLOG option. It continues to apply log records.

User Response: No action is required.

BMC50781E LONG LOG DELAY REACHED AT *timestamp*, TERMINATING

Explanation: The long log condition persisted until the specified delay time. Therefore, the utility takes the action specified by the user, which was to terminate the job.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50782I **LONG LOG DELAY REACHED AT *timestamp***

Explanation: The long log condition persisted until the specified delay time. Therefore, the utility takes the action specified by the user, which is to begin the LOGFINAL phase.

User Response: No action is required.

BMC50783I **LONG LOG NO LONGER DETECTED AT *timestamp***

Explanation: The log record arrival rate no longer exceeds the log record apply rate. The delay time has been reset.

User Response: No action is required.

BMC50784E **UNABLE TO COMPLETE PROCESSING BY DEADLINE, TERMINATING**

Explanation: The utility calculated that it cannot complete the LOGAPPLY phase by the deadline that the user specified (DEADLINE option). The utility terminates.

User Response: No action is required.

BMC50785E **UNABLE TO QUIESCE SPACE, RC = *rc*, LOGFINAL PHASE BYPASSED**

Explanation: The quiesce of the space failed and the LOGFINAL phase processing will not be performed.

User Response: See the preceding messages for more information.

BMC50786I ***task_no*: *task_type* TASK ENDED AT *timestamp*, ELAPSED TIME = *time***

Explanation: This message shows when the listed task ended and how long it took to process.

User Response: No action is required.

BMC50787I ***task_no*: DSN = *dsname***

Explanation: This message reports the data set name for accompanying messages.

User Response: See the accompanying messages for appropriate action.

BMC50788S ***task_no*: INDEX MAINTENANCE ON *index_name* CANNOT BE PERFORMED BECAUSE THE INDEX IS EMPTY**

Explanation: The index cannot be updated because it is not initialized.

User Response: Contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50789W ***task_no*: THERE ARE NO KEY VALUES FOR INDEX *index_name***

Explanation: The index contains no keys.

User Response: No action is required.

BMC50790E **XBM *ssid* ERROR, RC=*rc***

Explanation: An error code was received from the XBM identified by the subsystem ID (*ssid*).

User Response: Check for XBM error messages.

BMC50791I ***task_no*: *type count type count***

Explanation: This message provides index update statistics.

User Response: No action is required.

BMC50792S ***task_no*: FLAG BYTE/LAST BYTE DISCREPANCY ON PAGE *page_number* OF INDEX *index_name***

Explanation: The utility encountered an error in an index page.

User Response: Contact BMC Software Customer Support.

BMC50793S ***task_no*: PAGE X '*page_number*' OF INDEX *index_name* IS MARKED 'READ WITH I/O ERROR'**

Explanation: The page in the listed index is marked as having an I/O error.

User Response: Correct the I/O error range and resubmit the job.

BMC50794I ***task_no*: INDEX MAINTENANCE STATISTICS FOR INDEX *index_name***

Explanation: This message is the heading message for the index update statistics information that follows.

User Response: No action is required.

BMC50795I **XBM COMMAND RECEIVED AT *timestamp* FROM *userid***

Explanation: This message is the heading message for the display of the XBM command and response that follows.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50796I ***xbm_command/response***

Explanation: This messages displays an XBM command and response, and follows message BMC50795I.

User Response: No action is required.

BMC50797W **UNABLE TO POST RESPONSE TO XBM SUBSYSTEM, RC X'*rc*'**

Explanation: The utility cannot send a response to an XBM command.

User Response: Check the status of the XBM subsystem.

BMC50798I ***task_no*: CATALOG RETURN CODE *rc* (FEEDBACK CODE = X'*fdbk*') LISTING *dsname***

Explanation: An error occurred while accessing catalog information for the named data set.

User Response: Note the values of the return code and feedback code and then contact BMC Software Customer Support.

BMC50799S ***task_no*: UNABLE TO RETRIEVE DEVICE TYPE, DDNAME = *ddname*, RC = *rc***

Explanation: An internal error occurred. The utility cannot read the device type information for the named ddname.

User Response: Contact BMC Software Customer Support.

BMC50800E **EXPECTING *token* FOUND *token***

Explanation: An unexpected parameter or keyword was found when processing a command from the XBM Utility Monitor.

User Response: Correct the command and resubmit the job.

BMC50801I ***parameter* VALUE SET TO *value***

Explanation: The value of the displayed parameter was set via the XBM Utility Monitor.

User Response: No action is required.

BMC50802W **TOO LATE TO STOP AFTER INITIALIZATION**

Explanation: You cannot change the log apply control options after the LOGFINAL phase has started.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50803I LOG APPLY DISPLAY AT *timestamp*

Explanation: This message is the heading for the DISPLAY command output from the XBM Utility Monitor.

User Response: No action is required.

BMC50804I LOG APPLY TASKS: *total_tasks* TASKS, *started_tasks* STARTED, *completed_tasks* FINISHED

Explanation: This message is part of the XBM Utility Monitor DISPLAY command output. It shows the number of log record apply tasks allocated, started, and completed.

User Response: No action is required.

BMC50805I *queued_count* RECORDS QUEUED, *applied_count* RECORDS APPLIED

Explanation: This message is part of the XBM Utility Monitor DISPLAY command output. It displays the number of records queued (either table space or index) and the total number applied.

User Response: No action is required.

BMC50806I INDEX APPLY TASKS: *total_tasks* TASKS, *started_tasks* STARTED, *completed_tasks* FINISHED

Explanation: This message is part of the XBM Utility Monitor DISPLAY command output. It shows the number of index record apply tasks allocated, started, and completed.

User Response: No action is required.

BMC50807I TASK *task_number*: *task_type*

Explanation: This message is part of the XBM Utility Monitor DISPLAY command output. It shows the task number and the task type.

User Response: No action is required.

BMC50808I PARTITION(S): *partition_number*

Explanation: This message is part of the XBM Utility Monitor DISPLAY command output. It shows the set of partitions assigned to a task.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50809I ***partition_number***

Explanation: This message is part of the XBM Utility Monitor DISPLAY command output. It follows message BMC50808 if more partitions will display.

User Response: No action is required.

BMC50810I **INDEX *creator.index name***

Explanation: This message is part of the XBM Utility Monitor DISPLAY command output. It shows the name of an index.

User Response: No action is required.

BMC50811I ***task_no*: ALTER TABLE *table_name* DATA CAPTURE *status***

Explanation: The named table had its DATA CAPTURE flag status changed to ON, CHANGES, or NONE.

User Response: No action is required.

BMC50821I **LONG LOG DELAY REACHED, CONTINUING TO *timestamp***

Explanation: The long log delay time has been reached and processing continues to the next time indicated per the user's request.

User Response: No action is required.

BMC50822I **CURRENT PHASE: *phase_name*, STARTED: *timestamp***

Explanation: This message displays the currently executing phase and the time it started.

User Response: No action is required.

BMC50823I **CURRENT PHASE: *phase_name***

Explanation: This message displays the phase that is currently executing.

User Response: No action is required.

BMC50824I **LONG LOG DETECTED AT *timestamp***

Explanation: This message displays when the utility detects a long log condition.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50825I LONG LOG ACTION WILL OCCUR AT *timestamp*

Explanation: This message displays the time when the utility will take the action specified with the LONGLOG option.

User Response: No action is required.

BMC50826I *timestamp*: IDCAMS: *idcams_output*

Explanation: This message follows message BMC50814 and displays IDCAMs output.

User Response: No action is required.

BMC50827I COMMAND ISSUED FOR MEMBER *member_name* AT *timestamp*, '*command*'

Explanation: This message displays a DB2 command (within the quotation marks) that the utility issued against the specified DB2 member and the time that it was issued. The utility issues this message only if MSGLEVEL is 1 and you are running in a data-sharing environment. The message can be helpful to BMC Software Customer Support when diagnosing a problem.

User Response: No action is required.

BMC50828I LOGAPPLY TERMINATED DUE TO REORG REQUEST, ELAPSED TIME = *time*

Explanation: The utility encountered an exceptional condition and terminated the log apply processing.

User Response: Determine the cause of the original utility error, fix the error, and resubmit the job.

BMC50829I LOGAPPLY PHASE STARTING AT *start_time*

Explanation: The LOGAPPLY execution phase started at the specified time.

User Response: No action is required.

BMC50830I LOGFINAL PHASE STARTING AT *start_time*

Explanation: The LOGFINAL execution phase started at the specified time.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50831E LOG RECORDS REQUIRED ARE NOT AVAILABLE ON ACTIVE LOG(S)

Explanation: The log processing exit cannot keep pace with the DB2 logging activity. This resulted in the exit requiring records that are no longer available on the active log files.

User Response: Run the program during a period of lesser activity or expand the size of the active log data sets.

BMC50832E MONITOR TRACE CLASS (1) IS NOT ACTIVE

Explanation: The DB2 monitor trace is not active. This usually occurs when someone issues a STOP TRACE command while running with SHRLEVEL CHANGE.

User Response: Rerun the job and ensure that nothing stops the monitor trace that the utility started.

BMC50833E XBM INTERFACE FORCED TERMINATION

Explanation: XBM detected a potentially severe problem and signalled the utility to terminate. This message applies only to a utility command with SHRLEVEL CHANGE specified.

User Response: See additional messages for more information.

BMC50834I COMMAND RESPONSE FOR MEMBER *member*, '*response_text*'

Explanation: The utility with SHRLEVEL CHANGE issued a DB2 command. This message displays the response received from the specified DB2 member of the data-sharing group.

User Response: No action is required.

BMC50835E COMMAND FAILED FOR MEMBER *member*, '*command*'

Explanation: The displayed DB2 command failed for the specified data-sharing group member. Message BMC50834I precedes this message.

User Response: See message BMC50834I for the response to this command to determine the appropriate action. If the problem persists, contact BMC Software Customer Support.

BMC50836E CONSOLE INTERFACE ERROR, FUNCTION=*name* RC=*rc*

Explanation: An error occurred during a SHRLEVEL CHANGE job while the utility was processing a command through the extended console interface.

User Response: Refer to the system log for additional information. If the problem persists, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC50837E **NO RESPONSE RECEIVED FROM MEMBER *member***

Explanation: Using the extended console interface, the utility issued a DB2 command to the specified member of a data-sharing group. The member did not respond, so the utility terminated.

User Response: Refer to the system log for console errors. Check to see if the DB2 member received the command. If you cannot resolve the problem, contact BMC Software Customer Support.

BMC50838E **REQUIRED MONITOR TRACE CLASS (1) FOR MEMBER *member* NO LONGER ACTIVE**

Explanation: A SHRLEVEL CHANGE job requires an active monitor trace. The trace for the member became inactive while the job was still running. The utility ended all monitor traces and terminated. The monitor trace must be active on each member of a DB2 data-sharing group to ensure that all log records are collected.

User Response: Rerun the utility job, ensuring that none of the monitor traces are stopped while the utility is running.

BMC50839E **REQUIRED MONITOR TRACES NOT ACTIVE WITH OTHER SHRLEVEL CHANGE REORGS ACTIVE. STOPPING ALL TRACES**

Explanation: During a SHRLEVEL CHANGE job, the utility discovered that one or more required monitor traces was not active, issued message BMC50838E, and stopped any active monitor traces for other SHRLEVEL CHANGE jobs that were running in the same data-sharing group. The utility issues this message to inform the user that it is stopping the monitor traces and terminating the other SHRLEVEL CHANGE jobs running in that data-sharing group. The monitor trace must be active on each member of a DB2 data-sharing group to ensure that all log records are collected.

User Response: Rerun the utility job, ensuring that none of the monitor traces are stopped while the utility is running.

BMC50840I **MONITOR TRACE CLASS (1) STARTED FOR ALL ACTIVE MEMBERS**

Explanation: The utility started a monitor trace for each active DB2 member of a data-sharing group.

User Response: No action is required.

BMC50841I **MONITOR TRACE CLASS (1) ALREADY ACTIVE ON ALL ACTIVE MEMBERS**

Explanation: A utility job with SHRLEVEL CHANGE discovered that the monitor traces for all of the members of a data-sharing group were already active. This situation occurs when another SHRLEVEL CHANGE job is already running in this data-sharing group.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50842I MONITOR TRACE CLASS (1) STOPPED FOR ALL ACTIVE MEMBERS

Explanation: The utility job with SHRLEVEL CHANGE completed. Because no other SHRLEVEL CHANGE jobs are running in this data-sharing group, the utility ended all the monitor traces.

User Response: No action is required.

BMC50843I MONITOR TRACE CLASS (1) WILL REMAIN ACTIVE ON ALL ACTIVE MEMBERS TO SERVICE OTHER SHRLEVEL CHANGE REORGS

Explanation: More than one SHRLEVEL CHANGE job is running in this data-sharing group. The job that finished did not end any monitor traces.

User Response: No action is required.

BMC50844I MONITOR TRACE CLASS (1) ALREADY ACTIVE, TRACE NUMBER=*n*

Explanation: The utility issued a START TRACE command, but the trace is already active.

User Response: No action is required.

BMC50845I MONITOR TRACE CLASS (1) STOPPED FOR MEMBER *member*

Explanation: A utility job with SHRLEVEL CHANGE stopped the monitor trace for the specified member of a data-sharing group. No other SHRLEVEL CHANGE jobs were running.

User Response: No action is required.

BMC50846I MEMBER *member* ACTIVATED AFTER SHRLEVEL CHANGE REORG BEGAN, REORG TERMINATING

Explanation: The specified DB2 member was started after a SHRLEVEL CHANGE job started. Because the utility cannot guarantee the integrity of the job, it terminated.

User Response: Rerun the job, ensuring that no members are started or stopped while the SHRLEVEL CHANGE job is running.

BMC50847I *task_no*: BUFFER ALLOCATIONS: CHANGED PAGE BUFFERS = *c*, I/O BUFFERS = *i*, LARGEST BLOCK SIZE = *size*

Explanation: This message displays the number of changed page buffers, I/O buffers, and the largest block size that the utility initially allocated. The utility issues this message when the utility updates full copy data sets as part of a SHRLEVEL CHANGE job.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50848E ***task_no*: I/O ERROR: *function*, *module*, *line_number*, *reason_code***

Explanation: The utility issues this message if an I/O routine fails while the utility is updating a full copy data set for a SHRLEVEL CHANGE job. The message contains information that is useful to BMC Software Support.

User Response: Contact BMC Software Customer Support.

BMC50849E ***task_no*: I/O ERROR: *text***

Explanation: This message displays information that was returned by an I/O routine that failed while the utility was updating a full copy data set for a SHRLEVEL CHANGE job. The message contains information that is useful to BMC Software Support.

User Response: Contact BMC Software Customer Support.

BMC50850I ***task_no*: PARTITION = *p*, PART TOTALS UPDATED PAGES = *u*, UPDATED BLOCKS = *b*, APPENDED PAGES = *a*, APPENDED BLOCKS = *n***

Explanation: This message displays statistical information for the partition. The utility gathers this information when it updates a full copy data set for a SHRLEVEL CHANGE job. Message BMC50851I always follows this message.

User Response: No action is required.

BMC50851I ***task_no*: PARTITION = *p*, PART TOTALS DEFERRED WRITES = *d*, READ IOS = *r*, WRITE IOS =** w**, DEFERRED IOS = *i*, DEFERRED BLOCKS = *b***

Explanation: This message always follows message BMC50851I and provides additional statistics for the partition. The utility gathers this information when it updates a full copy data set for a SHRLEVEL CHANGE job.

User Response: No action is required.

BMC50852I ***task_no*: TASK TOTALS, UPDATED PAGES = *u*, UPDATED BLOCKS = *b*, APPENDED PAGES = *a*, APPENDED BLOCKS = *n***

Explanation: This message displays statistical information for the specified task. The utility gathers this information when it updates a full copy data set for a SHRLEVEL CHANGE job. Message BMC50853I always follows this message.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50853I ***task_no*: TASK TOTALS, DEFERRED WRITES = *d*, READ IOS = *r*, WRITE IOS = *w*, DEFERRED IOS = *i*, DEFERRED BLOCKS = *b***

Explanation: This message always follows message BMC50852I and provides additional statistics for the specified task. The utility gathers this information when it updates a full copy data set for a SHRLEVEL CHANGE job.

User Response: No action is required.

BMC50854I ***task_no*: FINAL STATISTICS, COLLECTOR (CALLS = *c*, TIME = seconds), PAGES COLLECTED = *p*, WAIT FOR BUFFERS = *b*, DUPLICATES = *d***

Explanation: This message displays statistical information for a specific partition. The utility gathers this information when it updates a full copy data set for a SHRLEVEL CHANGE job. Message BMC50855I always follows this message.

User Response: No action is required.

BMC50855I ***task_no*: FINAL STATISTICS, TOTAL PAGE SLOTS = *s*, MAXIMUM USED = *m*, I/O BUFFERS = *b*, MAXIMUM USED = *u***

Explanation: This message always follows message BMC50854I and provides additional statistics for a particular partition. The utility gathers this information when it updates a full copy data set for a SHRLEVEL CHANGE job.

User Response: No action is required.

BMC50856E **CATALOG ACTIVITY DETECTED ON OBJECT, PROCESSING TERMINATED AT LOG RBA *number***

Explanation: A data definition language (DDL) statement such as CREATE, DROP, or ALTER, was issued against the object being operated on. The utility terminated the utility job. The message contains the relative byte address (RBA) or the log record sequence number (LRSN), if in a data-sharing environment, of the associated DB2 log record.

User Response: Resubmit the job and ensure that no DDL statements are issued for the object while the utility job is running.

BMC50857E **OBJECT STATUS CHANGE DETECTED ON OBJECT, PROCESSING TERMINATED AT LOG RBA *number***

Explanation: The status of the object was changed while the utility job was running. The utility terminated the utility job. The message contains the relative byte address (RBA) or the log record sequence number (LRSN), if in a data-sharing environment, of the associated DB2 log record.

User Response: Resubmit the job and ensure that the status of the object does not change while the utility job is running.

BMC Software, Inc., Confidential and Proprietary Information

BMC50858E **DELETE WITHOUT WHERE CLAUSE DETECTED FOR TABLE *tbname* AT LOG RBA *rba/lrsn***

Explanation: While applying log records during the LOGAPPLY phase, the utility detected a mass DELETE operation (caused by an SQL DELETE without a WHERE clause). The utility terminated the job. This failure occurs only if the object on which the utility is operating is a segmented table space.

User Response: Resubmit the job, ensuring that no mass DELETE operation occurs while the utility is running.

BMC50859I **LOG RECORD STORE STATISTICS: MEMORY AVAILABLE = *amount_in_K*, MEMORY USED = *amount_in_K***

Explanation: This message displays (in kilobytes) the amount of data space memory available for log records and the amount the utility used to store them. The amount of memory available is the amount that was specified on the LOGMEM option. The amount of memory used might be slightly greater than the amount specified on the LOGMEM option because of how the utility allocates the memory.

User Response: No action is required. See message BMC50860I for additional information.

BMC50860I **TOTAL WAIT TIME FOR LOG RECORD STORE MEMORY = *seconds* DATA = *seconds* INDEX = *seconds***

Explanation: This message displays the total time in seconds that the utility spent waiting for available memory in order to store log records. The total time is the sum of the second and third times displayed.

User Response: To reduce the total wait time, increase the amount of available memory specified on the LOGMEM option.

BMC50861E ***task_no*: *operation* COULD NOT BE PERFORMED ON ROW FILTERED BY SELECT/DELETE RID: *record* ID RBA: *number* KEY: *record_key***

Explanation: The utility displays this message if an application issues an SQL UPDATE or DELETE statement on a filtered row. If the key value is not available, 'N/A' is displayed as the variable. A dump of any available log images follows this message.

User Response: Review the filter criteria specified for this job. Resubmit the job to avoid a conflict between the filter criteria and the application issuing the UPDATE or DELETE SQL statement.

BMC Software, Inc., Confidential and Proprietary Information

BMC50862E ***task_no*: DELETE OPERATIONS AGAINST ROWS FILTERED BY SELECT/DELETE EXCEEDED LIMIT OF *count***

Explanation: The utility displays this message if the application issues multiple DELETE statements for filtered rows and the number of DELETE statements to be ignored exceeds the limit shown. The job terminates.

User Response: Review the filter criteria specified for this job. Resubmit the job to avoid a conflict between the filter criteria and the application issuing the SQL DELETE statements.

If the limit shown is inappropriate for your environment, contact BMC Software Customer Support.

BMC50863I **LOGMEM OF *amount_in_K* SPECIFIED, FOR IMPROVED PERFORMANCE *amount_in_K* OR MORE IS RECOMMENDED**

Explanation: The amount of data space memory that the LOGMEM option specifies is not enough to achieve optimal performance.

User Response: To improve the performance of the SHRLEVEL CHANGE operation, change the amount of data space memory specified on the LOGMEM option to the second number displayed in the message. Then, resubmit the job.

BMC50864I **INSUFFICIENT LOGMEM OF *amount_in* K SPECIFIED, CHANGED TO *amount_in* K**

LOGMEM OF *amount_in* K SPECIFIED, CHANGED TO *amount_in* K

Explanation: If you receive the INSUFFICIENT LOGMEM version of this message, the first number is the amount of data space memory that you specified on the LOGMEM option. This amount is insufficient for the SHRLEVEL CHANGE operation. The utility changed the value to the second amount so that it could continue processing the job.

If you receive the other version of the message, you specified zero as the LOGMEM value, thus telling the utility to calculate the amount of memory it needed. The message displays the amount that the utility calculated.

User Response: If you receive the INSUFFICIENT LOGMEM version of this message, increase the amount on the LOGMEM option so that it is greater than or equal to the second value (the amount the utility needed); or, specify zero as the LOGMEM value to allow the utility to allocate the amount it needs.

If you receive the other version of the message, no action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50865I **TOTAL RECORDS INSERTED INTO LOG RECORD STORE, DATA = *n*, INDEX = *n***

Explanation: This message displays the total number of log records stored in the log record store for the table space updates (data) and index updates (index).

User Response: No action is required.

BMC50866I **TOTAL *type* STRINGS PROCESSED = *n*, WHEN NOT FULL = *n*, FROM SPILL = *n***

Explanation: This message is for internal use only. It displays the total number of strings processed, the number of incomplete strings, and the number of strings processed from the spill data sets.

User Response: No action is required.

BMC50867I **LOG RECORD STORE SPILL REQUESTS = *n*, HIGH SPILL PAGE = *n*, SPILL DATASETS CREATED = *n***

Explanation: This message displays the number of requests to store log records in the spill data sets, the largest amount of data that was stored in the spill data sets, and the number of spill data sets that the utility created.

User Response: No action is required.

BMC50868I ***component* WRITE REQUESTS = *n*, WRITE WAIT TIME = *time*, READ REQUESTS = *n*, READ WAIT TIME = *time***

Explanation: This message displays the number of read and write requests to the spill data sets and the time that the log record store or RID map component spent waiting for the read and write requests to complete.

User Response: No action is required.

BMC50869I ***component*: SPILLING TO DATASET '*dsname*'**

Explanation: The log records used all of the data space memory and spilled to the named data set, or the existing spill data set could not be extended further. The utility issues this message each time that it creates a new spill data set.

User Response: No action is required.

BMC50870E ***component* COULD NOT CREATE DATASET '*dsname*'**

Explanation: The log record store or RID map component was not able to create the named data set, which it needed in order to store the spilled log records.

User Response: See the preceding BMC50826I messages for more information about why the component could not create the data set. Correct the problem if possible, and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50871E ***component*** **OUT OF SPACE OR DATASET LIMIT REACHED**

Explanation: The log record store or RID map component could not allocate the spill data set, or it reached the maximum number of spill data sets.

User Response: See the preceding BMC50826I messages and message BMC50870I for more information about why the data set could not be created. Correct the problem if possible, and resubmit the job.

BMC50872I ***component pri\|sec*** **ALLOCATION CHANGED FROM** ***amount_in_K*** **TO** ***amount_in_K***

Explanation: The log record store or RID map component found the primary or secondary DASD allocation that you specified for the spill data sets to be insufficient. The component increased the allocation to the minimum amount that it needs to allocate the data space.

User Response: To improve performance, change the amount you specify on the LOGSPILL option to at least as much as the component allocated.

BMC50873I **LOGMEM REDUCED FROM** ***amount_in_K*** **TO** ***amount_in_K*** **DUE TO DATASPACE ALLOCATION LIMITS**

Explanation: The amount of data space that you requested on the LOGMEM option exceeded the system-allowed limits for maximum data space count or maximum data set size. The log record store component decreased the amount of data space that it used to remain within the system limits.

User Response: Contact BMC Software Customer Support.

BMC50874S ***component*** **SPILL TERMINATION IN PROGRESS**

Explanation: An abend occurred during log processing, causing the component to invoke its termination routine. The termination routine performs cleanup activities before the utility terminates the job.

User Response: Contact BMC Software Customer Support.

BMC50875E **MINIMUM REQUIRED LOGMEM** ***n*****K CANNOT BE OBTAINED DUE TO DATASPACE ALLOCATION LIMITS**

Explanation: The utility calculated that it needs *n*K of storage to hold the log records. However, the maximum amount of available memory is set by multiplying the value of the LOGRMAXD installation option by the value of the LOGRDSSZ installation option. If the result is less than the minimum amount of storage that the utility needs, the utility terminates the job.

User Response: Increase the values of the LOGRMAXD and LOGRDSSZ options so that when they are multiplied, the resulting value is greater than the number given in this message.

BMC Software, Inc., Confidential and Proprietary Information

BMC50876S **RIDMAP INITIALIZATION FAILED RC=*rc***

Explanation: The utility failed while initializing the RIDMAP component and terminated with the specified return code.

User Response: Refer to previous messages to determine if you can take some action to remedy the situation. If you cannot remedy the situation, contact BMC Software Customer Support.

BMC50877I **POINT OF CONSISTENCY ESTABLISHED AT RBA/LRSN = *rba/lrsn***

Explanation: The utility externalized page set buffers for all of the required objects, establishing a point of consistency at the displayed relative byte address (RBA) or log record sequence number (LRSN).

User Response: No action is required.

BMC50880I **RIDMAP: SUMMARY: ROWS=*r*, PAIRS=*p*, STORAGE=*s*K, MEM WAITS=*mw*, WAIT TIME=*wt***

RIDMAP: PART *part*: ROWS=*r*, PAIRS=*p*, STORAGE=*s*K, MEM WAITS=*mw*, WAIT TIME=*wt*

Explanation: This message displays statistics for a full table space or all partitions in a partitioned table space (*SUMMARY*), or for an individual part (*PART*). The statistics displayed show the maximum amounts for the following resources:

- ROWS—number of rows represented in the RID map
- PAIRS—number of RID pairs (old RID/new RID) stored
- STORAGE—total amount in K of storage used (DASD and data-space memory)
- MEM WAITS—number of times a wait was issued for memory
- WAIT TIME—total amount of time (in seconds) spent waiting for memory

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC50881I **RIDMAP: SUMMARY: TRANS=*t*, ADDS=*a*, DELS=*d*, SPILL READS=*r*, WAIT TIME=*wt***

RIDMAP: PART part: TRANS=*t*, ADDS=*a*, DELS=*d*, SPILL READS=*r*, WAIT TIME=*wt*

Explanation: This message displays statistics for a full table space or all partitions in a partitioned table space (*SUMMARY*), or for an individual part (*PART*). The statistics displayed show the amount of activity in the RID map for the following operations:

- TRANS—number of RID translations performed
- ADD—number of RID pairs inserted
- DELS—number of RID pairs deleted
- SPILL READS—number of read requests against a spill data set
- WAIT TIME—total amount of time (in seconds) spent waiting for I/O operations to complete

User Response: No action is required.

BMC50882I **RIDMAP: SUMMARY: MAX DATASPACES=*ds*, USED DATASPACES=*us*, RIDMAPMEM=*rm*K, USED MEM=*um*K**

Explanation: This message displays information about data spaces.

- MAX DATASPACES—maximum number of data space that the utility was allowed to use
- USED DATA SPACES—number of data spaces that the utility actually used
- RIDMAPMEM—value that the user specified for RIDMAPMEM in kilobytes
- USED MEM—total amount of data space memory used in kilobytes

User Response: No action is required.

BMC50883I **RIDMAP: SUMMARY: MAX PIPES=*mp*, PIPES USED=*pu*, PIPES BUFFER SIZE=*pb*, PIPE WAITS=*pw*, WAIT TIME=*wt***

Explanation: This message provides information for BMC Software Customer Support when they are resolving problems with the RID map.

User Response: No action is required

BMC50884E **RIDMAP: MINIMUM VALUE FOR RIDMAPMEM SHOULD BE *n*K**

Explanation: The value specified on RIDMAPMEM is insufficient for the utility to continue processing.

User Response: Change the value on RIDMAPMEM to a value equal to or greater than the value in the message, and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC50885I **RIDMAP: SPILLING OCCURRED BECAUSE MAX MEMORY EXCEEDED**

RIDMAP: SPILLING OCCURRED BECAUSE NUMBER OF DATASPACES EXCEEDED

Explanation: The utility used the spill data sets because it either exceeded the amount of memory it could use (specified on RIDMAPMEM) or exceeded the number of data spaces that it was allowed to allocate.

User Response: If the utility exceeded the amount of memory, increase the amount you specified on RIDMAPMEM, and resubmit the job. If the utility exceeded the number of data spaces, increase the value for RIDMDSSZ and resubmit the job. If you get the message a second time because the number of data spaces was exceeded, increase the value of RIDMMAXD and resubmit the job.

BMC50886E **RIDMAP: INTERNAL ERROR, *diagnostic_text***

Explanation: The utility failed with an internal error.

User Response: Contact BMC Software Customer Support.

BMC50887I **RIDMAP: RIDMAPMEM CHANGED TO *amount in* K**

Explanation: The amount of memory that you specified for RIDMAPMEM is insufficient. The utility changed the value to the amount in kilobytes shown so that it can continue processing the job.

User Response: No action is required.

BMC50890I ***task_no*: ATTEMPTING TO *action* DATASET '*dsname*'...**

Explanation: The utility issues this message during the rename and delete operations associated with the staging data sets when you specify one of the following options:

- For REORG PLUS, SHRLEVEL REFERENCE or SHRLEVEL CHANGE
- For LOADPLUS, LOAD REPLACE SHRLEVEL CHANGE or LOAD REPLACE SHRLEVEL REFERENCE.

The task number identifies the task associated with the operation. The value for *action* can be either RENAME or DELETE.

User Response: No action is required. However, you should ensure that the utility subsequently issues a corresponding message BMC50891I, which indicates that the operation for that data set completed successfully.

BMC Software, Inc., Confidential and Proprietary Information

BMC50891I ***task_no*: DATASET '*dsname*' IS *status***

Explanation: The utility issues this message during the rename and delete operations associated with the staging data sets when you specify SHRLEVEL REFERENCE or, for LOADPLUS, LOAD REPLACE SHRLEVEL CHANGE. The utility issues this message in conjunction with BMC50890I, indicating that the operation for the data set completed successfully.

The task number identifies the task associated with the operation. The value for *status* can be either RENAMED or DELETED.

User Response: No action is required.

BMC50892E **OBD UPDATE HAS FAILED FOR OBJECT '*dbname.tsname*'**

OBD UPDATE HAS FAILED FOR OBJECT '*creator.ixname*'

Explanation: An error occurred during FASTSWITCH processing. The utility cannot update the object descriptor (OBD) for the named object and terminates the job.

User Response: Attempt to restart the job. If you cannot restart the job, contact BMC Software Customer Support.

BMC50894I ***process_name* PROCESS STARTING AT *date timestamp***

Explanation: This message marks the beginning of the named process. If this process completes successfully, the utility issues message BMC50895I.

User Response: No action is required.

BMC50895I ***process_name* COMPLETE. ELAPSED TIME = *time***

Explanation: The named process completed successfully in the amount of time indicated.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS Messages

Messages that LOADPLUS issues are shown here in bold. When a message contains text that can vary, the type of variable is shown in lowercase and is italicized. The actual message contains the specific text of the variable type shown.

BMC51401E **DUPLICATE FIELD SPECIFICATIONS FOR COLUMN '*colname*'**

Explanation: You used the same field name more than once in your field specification.

User Response: Correct the field specification and resubmit the job.

BMC51402E **FOR FIELD '*fldname*', POSITION START AND END MUST BOTH BE ABSOLUTE VALUES OR BOTH RELATIVE VALUES**

Explanation: If you use an integer for the start position, you must either specify an integer end position or not specify an end position. If you use * or *+*n* for the start position, you must either specify a *+*m* end position or not specify an end position. *m* must be greater than or equal to *n*.

User Response: Correct the position specification and resubmit the job.

BMC51403I **FOR FIELD '*fldname*', END POSITION IGNORED**

Explanation: You specified an end position for a VARCHAR or VARGRAPHIC field or for a field with an explicit length. LOADPLUS ignored the specification.

User Response: No action is required.

BMC51404E **FOR FIELD '*fldname*', END POSITION OR LENGTH MUST BE SPECIFIED**

Explanation: For the field type named, you must specify POSITION(*start*:*end*) or a length. For more information, see the POSITION keyword description on page 3-104.

User Response: Correct the field specification and resubmit the job.

BMC51405E **FOR FIELD '*fldname*', DATA TYPE '*data_type1*' IS INCOMPATIBLE WITH COLUMN DATA TYPE '*data_type2*'**

Explanation: LOADPLUS does not support the data conversion that is implied by the data type of the named field and the column data type. See Table 3-23 on page 3-121 for the allowed conversions.

User Response: Correct the field specification and resubmit the job. You can also write your own conversion routine to perform the conversion.

BMC Software, Inc., Confidential and Proprietary Information

BMC51406E **NO FIELD SPECIFICATION FOR COLUMN '*colname*', BUT COLUMN IS DEFINED NOT NULL**

Explanation: You did not include a field specification for the named column and the column is not nullable or defaultable.

User Response: Add the field specification and resubmit the job.

BMC51407E **NULLIF IS SPECIFIED FOR COLUMN '*colname*', BUT THE COLUMN IS NOT NULLABLE**

Explanation: You specified NULLIF but the column is not nullable.

User Response: Correct the field specification and resubmit the job.

BMC51408E **UNKNOWN DATE/TIME EXTERNAL FORMAT '*format*' FOR COLUMN '*colname*'**

Explanation: You specified a date, time, or timestamp external format that LOADPLUS does not recognize.

User Response: Correct the field specification and resubmit the job.

BMC51409E **FIELD SPECIFICATION '*fldname*' HAS NO CORRESPONDING COLUMN AND DATA TYPE IS NOT CHARACTER**

Explanation: For a field that has no corresponding column, the data type can only be CHARACTER (CHAR).

User Response: Correct the field specification and resubmit the job.

BMC51410E **DATA TYPE '*data_type*' FOR COLUMN '*colname*' IS INVALID *reason***

Explanation: The data type that you provided in your field specification for this column is valid only under certain circumstances, as described by *reason*. The most likely scenario for this message is that you specified a delimited timestamp data type, but you did not specify FORMAT CSV.

User Response: Change the data type in your field specification or change other appropriate options on your LOAD command. Resubmit the job.

BMC51411E **THE COLUMN OR FIELD SPECIFICATION '*name*' USED IN A COMPARISON CANNOT BE FOUND**

Explanation: You specified a column in a WHEN, NULLIF, or DEFAULTIF comparison that is not a column in the table named, or you specified a field in a comparison that is not a field listed in the INTO TABLE option.

User Response: Correct the name of the column or field and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51412E ***option* NOT ALLOWED WITH FORMAT UNLOAD, FORMAT BMC, FORMAT CSV, OR FORMAT BMCUNLOAD**

Explanation: You specified an option on your LOAD command that is not valid for one of the FORMAT options. The specified option indicates one of the following limitations with the FORMAT option:

- You cannot specify CONTINUEIF with FORMAT UNLOAD, FORMAT BMC, FORMAT BMCUNLOAD, or FORMAT CSV.
- You cannot include any field specifications with FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD.
- If you include the WHEN option with FORMAT UNLOAD, FORMAT BMC, or FORMAT CSV, your comparison must use a table column name, not a *start:end* field.

User Response: Correct the LOAD command and resubmit the job.

BMC51413E **'WHEN TABLE = ' ALLOWED ONLY WITH FORMAT UNLOAD, FORMAT BMC, OR FORMAT BMCUNLOAD**

Explanation: You can specify WHEN TABLE=*obid* only with FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD.

User Response: Correct the LOAD command and resubmit the job.

BMC51414I **'FORMAT *option*' SPECIFIED. *rules_option* RULES IN EFFECT FOR WHEN COMPARISONS**

Explanation: You specified a LOAD command option that is not valid with the value of the RULES option in your installation options module. For this execution, LOADPLUS has overridden this value with the value that displays in the message.

User Response: No action is required.

BMC51415E **FOR '*option*' FIELD '*fldname*', ONLY '=', '¬ =','<>', 'IN', & 'NOT IN' COMPARISONS ARE ALLOWED**

Explanation: RULES=STANDARD was specified at installation. You used operators in your LOAD command that are only valid if LOADPLUS is installed with RULES=BMC.

User Response: Correct the LOAD command or your RULES option and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51416E **FOR '*option*' FIELD '*fldname*', CONSTANT MUST BE A CHARACTER STRING OR A HEX STRING**

Explanation: You specified a comparison constant that is not valid when LOADPLUS is installed with RULES=STANDARD.

User Response: Correct the LOAD command or your RULES option and resubmit the job.

BMC51417E **FOR FIELD '*fldname*', ONLY ONE NULLIF/DEFAULTIF CAN SPECIFY THE 'ERROR' OPTION**

Explanation: You cannot specify more than one error condition per column.

User Response: Correct the LOAD command and resubmit the job.

BMC51418E **FOR FIELD '*fldname*', NULLIF CANNOT SPECIFY A 'VALUE' OPTION**

Explanation: You cannot specify a VALUE clause on a NULLIF statement.

User Response: Correct the LOAD command and resubmit the job.

BMC51419E **'INDEX UPDATE' REQUIRES 'UNIQUECHECK NO'**

Explanation: If you specify INDEX UPDATE for a two-phase load, you must specify UNIQUECHECK NO.

User Response: Correct the LOAD command and resubmit the job.

BMC51420E **FOR INPUT RECORD NUMBER *recno*, FIELD '*fldname*' STARTS IN POSITION *n* BUT HAS NO END STRING DELIMITER**

Explanation: While loading your CSV data, LOADPLUS found an input record that contains a delimiter that indicates the beginning of a string. However, LOADPLUS reached the end of the record without finding an end-of-string delimiter. LOADPLUS discards the record.

User Response: Correct your input data and resubmit your job.

BMC51421I **NO FIELD SPECIFICATION FOR COLUMN '*colname*'. VALUE IS DEFAULTED**

Explanation: You did not provide a field specification for a nullable or defaultable column of the table that you are loading.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51422I **FOR TABLE '*creator.tbname*' THE FOLLOWING INPUT FIELDS HAVE KNOWN START:END POSITIONS THAT WERE NOT SPECIFIED**

Explanation: One or more instances of message BMC51423I follows this message. BMC51423I lists the fields with complete field *start*:*end* information.

User Response: No action is required.

BMC51423I **FIELD '*fldname*' HAS POSITION(*start*:*end*)**

Explanation: LOADPLUS issues this message if you did not specify POSITION on the field specification and LOADPLUS can determine the start position, or if you specified only POSITION(*start*) and LOADPLUS can determine the end position.

User Response: No action is required.

BMC51424I **ANALYZE PHASE WILL BE PERFORMED DUE TO DYNAMIC WORK FILE ALLOCATION**

Explanation: LOADPLUS performs the ANALYZE phase because you enabled dynamic work file allocation for this utility execution.

User Response: No action is required.

BMC51425I **ANALYZE PHASE BYPASSED AND DYNAMIC ALLOCATION DISABLED BECAUSE ENUMROWS WAS NOT SPECIFIED**

Explanation: LOADPLUS did not perform the ANALYZE phase and did not dynamically allocate any data sets because you did not specify ENUMROWS with a value for *total* or *new.*

User Response: Specify ENUMROWS (*total*) or ENUMROWS (,*new*) and resubmit the job. See “ENUMROWS” on page 3-46 for information about the values to use for *total* or *new.*

BMC51426I **'ORDER *option*' IS MEANINGLESS FOR A TABLE WITHOUT A CLUSTERING INDEX; CONVERTING TO 'ORDER NO'**

Explanation: ORDER PRESORTED or ORDER YES only applies to a table with a clustering index. LOADPLUS changed ORDER PRESORTED or ORDER YES to ORDER NO.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51427E **MULTIPLE PART NUMBERS ARE NOT ALLOWED WITH MORE THAN ONE INTO TABLE BLOCK**

Explanation: LOADPLUS does not allow you to specify multiple partition numbers on a single INTO TABLE option if you specified multiple INTO TABLE blocks on the LOAD command.

User Response: Change the LOAD command to specify either multiple INTO table blocks, each with only one partition listed, or one INTO TABLE block with multiple partitions listed. Resubmit the job.

BMC51428E **COLUMN '*colname*' CANNOT BE A PREDICATE IN A NULLIF, DEFAULTIF, WHEN CLAUSE, OR A DB2 CHECK CONSTRAINT. IF IT IS A LOB COLUMN, IT CANNOT BE USED IN A FIELD SPECIFICATION**

Explanation: Your LOAD command included one of the following options:

- You referenced an identity column or a column that is defined with a data type of ROWID, BLOB, CLOB, or DBCLOB in one of the following clauses:
  - — the predicate of a NULLIF, DEFAULTIF, or WHEN clause
  - — the check condition of a DB2 check constraint
- You included a column that is defined with a data type of BLOB, CLOB, or DBCLOB in your INTO field specification.

LOADPLUS does not support these options.

User Response: Change the LOAD command and resubmit the job. If this error occurred because the identity column is referenced in the check condition of a DB2 check constraint, you can specify ENFORCE NO on your LOAD command and resubmit the job.

BMC51429E ***clause* CLAUSE NOT VALID FOR COLUMN '*colname*'**

Explanation: You specified one of the following options:

- NULLIF on a column with data type ROWID
- DEFAULTIF ... VALUE on a column with data type ROWID

LOADPLUS does not support these clauses for ROWID columns.

User Response: Change the LOAD command and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51430E **'*option*' ONLY VALID IF 'PART REPLACE' SPECIFIED AND NO NON-PARTITIONING INDEXES**

Explanation: The displayed option is valid only if you specify PART REPLACE for all partitions that are involved in the load and no nonpartitioning indexes exist. This condition is true, for example, when performing a LOAD RESUME YES SHRLEVEL REFERENCE job.

User Response: Correct the LOAD command and resubmit the job.

BMC51431E **LOB COLUMN '*colname*' CANNOT BE PROCESSED IF THE TABLESPACE OR ANY TABLESPACE PARTITION IS BEING REPLACED**

Explanation: You specified LOAD REPLACE or LOAD RESUME YES with the PART REPLACE option, and your table includes a column that is defined with a LOB data type (BLOB, CLOB, and DBCLOB). LOADPLUS cannot process this job.

LOADPLUS can load data into base tables that are defined with LOB columns (BLOB, CLOB, and DBCLOB) when you specify a LOAD RESUME YES (without the PART REPLACE option). However, LOADPLUS does not load data into the LOB auxiliary tables. LOADPLUS puts an empty LOB indicator, or a null LOB indicator if the column allows nulls, into the base table to indicate an empty or null LOB column.

User Response: Revise your LOAD command to perform a LOAD RESUME YES without the PART REPLACE option, or, if you are not using the defined LOB columns, alter your table to remove them. Resubmit the job.

BMC51432W **ROWID COLUMN '*colname*' IS 'GENERATED ALWAYS'. INPUT DATA WILL BE IGNORED**

Explanation: You are loading a table that contains a ROWID column that is defined as GENERATED ALWAYS. LOADPLUS ignores the field specification and any input data from this column, and generates its own unique row identifiers.

User Response: No action is required. However, if you want LOADPLUS to load data from a ROWID column, you must define it as GENERATED BY DEFAULT.

BMC51433E **SPACE *dbname.spname* WOULD REQUIRE *state* STATE AFTER LOAD**

Explanation: This message is displayed after message BMC50263E, which indicates that the object that you are trying to load is in either LPL or WEPR state. LOADPLUS cannot load this object because the object must be in a restricted state after the load.

User Response: Follow the documented DB2 procedures to remove the object from LPL or WEPR state (for example, by using RECOVER). See Chapter 2, “Operational Considerations,” for information about the restrictions for loading an object that is in LPL or WEPR state. Resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51434E **FIELD SPECIFICATIONS REQUIRED FOR *data_type* COLUMN '*colname*' WHEN 'FORMAT CSV' IS SPECIFIED**

Explanation: You specified FORMAT CSV on the LOAD command but did not provide field specifications for the named column. With the exception noted below, LOADPLUS requires field specifications for all input fields when you are loading CSV data.

Note: You do not have to supply a field specification for the last input field or fields in your CSV file if you do not want to load these fields.

User Response: Add a field specification for the named column. If you do not want to load the input field from your CSV file, include a dummy field specification. Refer to Chapter 2, "Operational Considerations," for additional information about field specifications when loading CSV data.

BMC51435S **RESTART IS NOT POSSIBLE BECAUSE INDEX UPDATE WAS IN PROGRESS AT TIME OF FAILURE; RECOVERY REQUIRED**

Explanation: In a LOAD RESUME YES INDEX UPDATE job, you specified RESTART after a failure. However, because LOADPLUS started but did not complete the index update process, RESTART is not possible.

User Response: Determine the reason for the original failure and recover the table space and indexes. Resubmit the job.

BMC51436E **RESTART(PHASE) IS NOT POSSIBLE BECAUSE INDEXES ARE ALREADY UPDATED; USE RESTART**

Explanation: In a LOAD RESUME YES INDEX UPDATE job, you specified RESTART(PHASE) after a failure. However, because LOADPLUS completed the index update process, RESTART(PHASE) is not possible.

User Response: Specify RESTART without the PHASE option and resubmit the job.

BMC51437E **RESTART IS NOT POSSIBLE BECAUSE OF NO WORK DATASETS**

Explanation: You ran a single-phase LOAD RESUME YES SHRLEVEL NONE job without allocating SYSUT1 and SORTOUT data sets, and then you attempted to restart the job. LOADPLUS requires these data sets for restart under the following circumstances:

- SYSUT1 is always required to restart this type of load job.
- SORTOUT is required to restart if you also specify ORDER YES and you do not specify PART REPLACE.

User Response: Recover the table space and indexes, and resubmit the original job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51438E ***task_no*.: INPUT RECORD *nn* NOT IN CLUSTERING INDEX SEQUENCE**

Explanation: You specified ORDER PRESORTED but the input data is not in clustering index sequence. Record number *nn* is not in correct sequence after all FIELDPROCs, EDITPROCs, and VALIDPROCs were processed. LOADPLUS checks sequence as the row is about to be loaded, and the internal key might not reflect the external data fields.

Two BMC50497I messages follow this message. The first BMC50497I message displays the previous key. The second message displays the current key. The length of each of these keys includes the NULL indicator byte for any field that is nullable.

User Response: Sort the data in clustering index order and resubmit this job; or change the ORDER option to ORDER NO or ORDER YES and resubmit the job.

BMC51440E **PAGE X '*an*' IS MARKED 'READ WITH I/O ERROR'**

Explanation: LOADPLUS found a page (*an*) that showed an I/O error during an I/O operation for that page.

User Response: Contact BMC Software Customer Support.

BMC51441E **FLAG BYTE/LAST BYTE DISCREPANCY ON PAGE X'*nn*'**

Explanation: LOADPLUS detected a discrepancy between the flag byte and last byte on page *nn.*

User Response: Contact BMC Software Customer Support.

BMC51442E **FOUND PAGE FLAGS X'*flags*' INDICATING HEADER PAGE AT PAGE NUMBER X'*nn*'**

Explanation: LOADPLUS found an inappropriate page number or page flag. The page flags indicate a header page.

User Response: Contact BMC Software Customer Support.

BMC51443E **FOUND PAGE FLAGS X'*flags*' INDICATING SPACE MAP AT PAGE NUMBER X'*nn*'**

Explanation: LOADPLUS found an inappropriate page number or page flag. The page flags indicate a space map.

User Response: Contact BMC Software Customer Support.

BMC51444E **FOUND UNIDENTIFIED PAGE WITH FLAGS X'*flags*' AT PAGE NUMBER X'*nn*'**

Explanation: The page at *nn* has an unrecognized page type indicated by *flags.*

User Response: Contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC51445E **INTERNAL ERROR: *routine*, CODE = *code***

Explanation: An internal error occurred.

User Response: Contact BMC Software Customer Support with the routine and code that this message displays.

BMC51446E **INVALID PAGE X'*nn*' REASON: *reason_text***

Explanation: During INDEX BUILD processing, LOADPLUS read a page from the index and the page is invalid for the reason shown.

User Response: Contact BMC Software Customer Support.

BMC51447E **INDEX MAINTENANCE UNAVAILABLE - *reason***

Explanation: LOADPLUS cannot perform logical index processing for the reason shown.

User Response: Contact BMC Software Customer Support.

BMC51448E **INDEX ERROR IN '*dsname*'**

Explanation: A severe error occurred while processing a page within this data set. See accompanying messages for more information about the error.

User Response: Contact BMC Software Customer Support.

BMC51449W **INDEX WILL BE LEFT IN RECOVER PENDING. PROCESSING CONTINUES WITH THE NEXT INDEX**

Explanation: An error that a previous message describes occurred while LOADPLUS was deleting duplicate index entries. LOADPLUS leaves the index in RECOVER pending status and continues processing the next index, if one exists.

User Response: Run RECOVER PLUS or IBM's recovery utility on the index named in message BMC51448E.

BMC51450E **CONTINUEIF LOGICAL RECORD FROM *ddname* GREATER THAN 32K**

Explanation: A logical record that is greater than 32,768 bytes long was being created during CONTINUEIF processing. This is not permitted.

User Response: Correct the input and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51451E **'REPLACE' NOT SPECIFIED AND SPACE '*dbname.tsname*', PARTITION *p* IS NOT EMPTY**

Explanation: You did not specify LOAD REPLACE or PART REPLACE when you specified RESUME NO and the specified partition of the named table space contains active DB2 data.

User Response: Specify LOAD REPLACE, PART REPLACE, or RESUME YES and restart the job.

BMC51452E **'REPLACE' NOT SPECIFIED AND SPACE '*dbname.tsname*' IS NOT EMPTY**

Explanation: You specified RESUME NO and did not specify LOAD REPLACE and the specified table space contains active DB2 data.

User Response: Specify LOAD REPLACE or RESUME YES and restart the job.

BMC51453I **EXISTING ROWS IN TABLESPACE '*dbname.tsname*' DELETED**

Explanation: This message is for information only. LOADPLUS issues this message when you specify LOAD REPLACE or RESUME NO REPLACE.

User Response: No action is required.

BMC51454I **EXISTING ROWS IN TABLESPACE '*dbname.tsname*' PARTITION *p* DELETED**

Explanation: This message is for information only. LOADPLUS issues this message when you specify PART REPLACE.

User Response: No action is required.

BMC51455E **FIELD '*fldname*' EXTENDS BEYOND END-OF-RECORD FOR ALL RECORDS**

Explanation: LOADPLUS determined that the named field is not entirely contained in the input record for all records. Processing stops.

User Response: Correct the field specification and resubmit the job.

BMC51456E **CONTINUEIF SPECIFICATION EXTENDS BEYOND END-OF-RECORD FOR ALL RECORDS**

Explanation: LOADPLUS determined that the CONTINUEIF field is not entirely contained in the input record for all records. Processing stops.

User Response: Correct the CONTINUEIF option and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51457E **WHEN SPECIFICATION FOR TABLE '*creator.tbname*' EXTENDS BEYOND END-OF-RECORD FOR ALL RECORDS**

Explanation: LOADPLUS determined that the WHEN field is not entirely contained in the input record for all records. Processing stops.

User Response: Correct the WHEN option and resubmit the job.

BMC51458E **NULLIF/DEFAULTIF SPECIFICATION FOR FIELD '*fldname*' EXTENDS BEYOND END-OF-RECORD FOR ALL RECORDS**

Explanation: You specified a NULLIF or DEFAULTIF comparison that is not entirely contained in the input record for all records. Processing stops.

User Response: Correct the NULLIF/DEFAULTIF comparison or correct your input data. Resubmit the job.

BMC51459E **END-OF-FILE ON *ddname* DATASET WHEN CONTINUATION RECORD EXPECTED**

Explanation: You specified CONTINUEIF and the current record indicates that the next record should be concatenated, but LOADPLUS encountered an end-of-file indicator on the named input data set.

User Response: Check and correct the continuation or input data and resubmit the job.

BMC51460S **UNEXPECTED END-OF-FILE ON *ddname* DATA SET DURING SYSERR PROCESSING, RECORD *recno*.**

Explanation: This message indicates a LOADPLUS internal error.

User Response: Contact BMC Software Customer Support.

BMC51461E **DISCARD LIMIT *limit* REACHED. PROCESSING TERMINATED**

Explanation: The number of discards exceeds the discard limit that you specified in the LOAD command. LOADPLUS did not complete and the table space remains in use by LOADPLUS.

User Response: Correct the errors that are causing records to be discarded or increase the discard limit. Restart LOADPLUS with the same utility ID.

BMC51462I **ONE OR MORE DISCARDS IGNORED DUE TO DISCARD IGNORE OPTION**

Explanation: This message is for information only.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51463I **ONE OR MORE DISCARDS NOT REPORTED DUE TO DISCARD REPORT OPTION**

Explanation: This message is for information only.

User Response: No action is required.

BMC51464I **LOAD PHASE EXECUTION IS BYPASSED DUE TO NO ROWS ADDED TO TABLESPACE**

Explanation: LOADPLUS bypasses the LOAD phase and does not rebuild the indexes because you specified RESUME YES and no rows were added to the table space.

User Response: No action is required, but you should verify that the options in your LOAD command are correct.

BMC51465I **BUILD BYPASSED FOR INDEX '*creator.ixname*' DUE TO NO ROWS ADDED TO TABLE '*creator.tbname*'**

Explanation: LOADPLUS bypasses the build process for the specified index because no rows were added to the table.

User Response: No action is required.

BMC51466I **BUILD BYPASSED FOR INDEX '*creator.ixname*' PART *p* DUE TO NO ROWS ADDED TO PART**

Explanation: LOADPLUS bypasses the build process for the specified index partition because no rows were added to the table.

User Response: No action is required.

BMC51467E **TOO MANY *ddtype* DD STATEMENTS PRESENT**

Explanation: LOADPLUS detected more than 256 SYSREC DD statements or 16 SORTOUT DD statements.

User Response: Reduce the number of SYSREC and/or SORTOUT DD statements and resubmit the job.

BMC51468E **DCB RECFM FOR *ddname* IS NOT COMPATIBLE WITH PREVIOUS RECFM**

Explanation: The record formats for your SYSREC*nn* data sets are inconsistent. The SYSREC*nn* data sets you specify must be all fixed (F or FB) or all variable (V, VB, or VBS).

User Response: Revise the record formats for your SYSREC data sets or revise your SYSREC DD statements and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51469E **DCB LRECL FOR *ddname* IS NOT THE SAME AS PREVIOUS LRECL**

Explanation: The LRECL (logical record length) for your data sets is inconsistent. All specified data sets must have the same LRECL.

User Response: Revise your data sets or your DD statements and resubmit the job.

BMC51470I **PRELOAD STATISTICS: *n* ROWS SELECTED FOR TABLE '*creator.tbname*', *d* ROWS SELECTED BUT DISCARDED DUE TO ERRORS**

Explanation: This message is for information only. PRELOAD processing completed for the named table. *n* is the number of rows that LOADPLUS selected and *d* is the number of rows that LOADPLUS discarded for the table named.

User Response: No action is required.

BMC51471I **PRELOAD STATISTICS: *n* ROWS SELECTED FOR PARTITION *p***

Explanation: PRELOAD processing completed for the named partition. *n* is the number of rows that LOADPLUS selected for the data set partition identified by *p*. This message is for information only.

User Response: No action is required.

BMC51472I ***phase* STATISTICS: *n* ROWS SELECTED FOR SPACE '*dbname.tsname*', *d* ROWS SELECTED BUT DISCARDED DUE TO ERRORS**

Explanation: PRELOAD or COMBINED phase processing completed for the named table space. *n* is the number of rows that LOADPLUS selected and *d* is the number of rows that LOADPLUS discarded for the table space named. This message is for information only.

Note: Records that LOADPLUS does not select because they do not meet the WHEN criteria on any INTO clause are not considered errors.

User Response: No action is required.

BMC51473I **[ADJUSTED] LOAD STATISTICS: *n* ROWS LOADED INTO TABLE '*creator.tbname*'**

Explanation: Load processing completed for the named table. With the following exception, *n* is the number of rows that LOADPLUS actually loaded into the named table.

If you are running a two-phase load and you specified UNIQUECHECK NO, LOADPLUS issues this message twice. The first instance of this message displays the number of rows that LOADPLUS loaded into the table before processing any discards. The second version of this message displays adjusted statistics after discard processing completes, reflecting the actual number of loaded rows after discards.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51474I **LOAD STATISTICS: *n* ROWS LOADED INTO PARTITION *p***

BUILD STATISTICS: *n* KEYS LOADED INTO PARTITION *p*

Explanation: Load or build processing completed for the named partition. *n* is the number of rows or keys that LOADPLUS actually loaded into the data set partition identified as *p*. This message is for information only.

User Response: No action is required.

BMC51475I **LOAD STATISTICS: *n* ROWS LOADED INTO TABLE SPACE '*dbname.tsname*'**

Explanation: Load processing completed for the named table space. *n* is the number of rows that LOADPLUS actually loaded into the table space named. This message is for information only.

User Response: No action is required.

BMC51476I **[ADJUSTED] BUILD STATISTICS: *n* KEYS LOADED INTO INDEX '*creator.ixname*'**

Explanation: Load processing completed for the named index. With the following exception, *n* is the number of keys that LOADPLUS built for the named index.

If you are running a two-phase load and you specified UNIQUECHECK NO, LOADPLUS issues this message twice. The first instance of this message displays the number of keys that LOADPLUS built before processing any discards. The second version of this message displays adjusted statistics after discard processing completes, reflecting the actual number of built keys after discards.

User Response: No action is required.

BMC51477I **LOAD PHASE DISCARD STATISTICS: *n* ROWS/KEYS DELETED FROM DSN *dsname***

Explanation: One of the following situations occurred:

- You specified UNIQUECHECK CLUSTER and LOADPLUS detected one or more duplicate keys in a nonclustering index.
- You specified UNIQUECHECK NO and LOADPLUS detected one or more duplicate keys.

BMC Software, Inc., Confidential and Proprietary Information

- You specified PRELOAD LOAD and LOADPLUS detected one or more duplicate keys.

At the end of the LOAD or COMBINED phase, LOADPLUS deleted the duplicate keys from your index and rows from your table space. *n* is the number of keys or rows that LOADPLUS deleted and *dsname* identifies the data set that is affected. This message is for information only.

User Response: No action is required.

BMC51478I PRELOAD STATISTICS: *n* PHYSICAL (l LOGICAL) RECORDS READ FROM *ddname*

Explanation: PRELOAD processing completed for the input data set. This message indicates the number of physical (and logical) records (after CONTINUEIF processing) that LOADPLUS read from the input data set in the specified SYSREC DD statement. This message is for information only.

User Response: No action is required.

BMC51479I *phase* STATISTICS: *n* PHYSICAL (l LOGICAL) RECORDS DISCARDED TO SYSDISC

Explanation: This message accompanies message BMC51478I, and indicates the number of physical (and logical) records (after CONTINUEIF processing) that LOADPLUS wrote to the discard data set specified in the SYSDISC DD statement. This message is for information only.

User Response: No action is required.

BMC51480I *task_no*.: *task_name* COMPLETE. ELAPSED TIME = *t*. *n* ROWS PASSED TO APPLY PLUS

Explanation: The named task completed in the given elapsed time. LOADPLUS issues this message when using Apply Plus to perform an online load (LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY). The message indicates the number of rows that LOADPLUS passed to Apply Plus.

User Response: No action is required.

BMC51481E INDEX STATISTICS ARE FOR THE REPLACED PARTITIONS ONLY

Explanation: Because all partitions were not replaced, index statistics are gathered for the replaced partitions. LOADPLUS does not use the statistics to update the BMCSTATS tables or the DB2 catalog.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51483S **FOR TABLESPACE '*dbname.tsname*', PRELOAD COUNT *m* DOES NOT MATCH LOAD COUNT *n*, PARTITION = *p***

Explanation: The number of rows that LOADPLUS wrote to the SORTOUT file during the PRELOAD phase does not match the number of rows that LOADPLUS loaded into the specified table space partition during the LOAD phase. (If *p* is zero, the table space is nonpartitioned.)

User Response: Verify that the SORTOUT file that LOADPLUS created during preload processing is the same file that LOADPLUS used for load processing. If the same file was used, contact BMC Software Customer Support.

BMC51484S **FOR INDEX '*creator.ixname*', PRELOAD COUNT *m* DOES NOT MATCH LOAD COUNT *n*, PARTITION = *p***

Explanation: The number of keys that LOADPLUS wrote to the SYSUT1 file during the PRELOAD phase does not match the number of keys that LOADPLUS loaded into the specified index partition during the LOAD phase. (If *p* is zero, the table space is nonpartitioned.)

User Response: Verify that the SYSUT1 file that LOADPLUS created during preload processing is the same file that LOADPLUS used for load processing. If LOADPLUS used the same file, contact BMC Software Customer Support.

BMC51485S **FOR TABLE '*creator.tbname*', PRELOAD COUNT *m* DOES NOT MATCH LOAD COUNT *n***

Explanation: The number of rows that LOADPLUS wrote to the SORTOUT file during the PRELOAD phase does not match the number of rows that LOADPLUS loaded into the specified table during the LOAD phase.

User Response: Verify that the SORTOUT file that LOADPLUS created during preload processing is the same file that LOADPLUS used for load processing. If LOADPLUS used the same file, contact BMC Software Customer Support.

BMC51486I **LOADING OF DATASET '*dsname*' WILL REQUIRE *n* PAGES**

Explanation: During PRELOAD processing, LOADPLUS determined that loading the specified data set requires *n* pages.

User Response: No action is required. However, you can use this information to reallocate your DB2 data sets before loading.

BMC Software, Inc., Confidential and Proprietary Information

BMC51487E **LOADING OF DATASET '*dsname*' WILL EXCEED THE MAXIMUM SIZE OF *n* PAGES**

Explanation: Based on the primary and secondary quantities for this data set, the number of extents allowed, or DB2 restrictions, LOADPLUS determined that it is not possible for *n* pages to fit in this data set.

User Response: For nonpartitioned objects only, you can restart the job if you increase the primary and secondary quantities of the data set. For nonpartitioned and partitioned objects, you can reduce the number of rows loaded, reduce the PCTFREE value, increase the FREEPAGE value, or increase the PIECESIZE value. However, this change requires that you resubmit the job with the NEW execution parameter.

BMC51488I **LOADING OF DATASET '*dsname*' MAY REQUIRE *n* PAGES**

Explanation: During PRELOAD processing, LOADPLUS determined that loading the specified data set might require *n* pages. LOADPLUS issues this message instead of BMC51486I for nonunique indexes. If no duplicates exist, *n* pages is exact. Otherwise, *n* can be significantly higher than the actual number of pages required.

User Response: No action is required. However, you can use this information to reallocate your DB2 data sets before loading.

BMC51489I **LOADING OF DATASET '*dsname*' MAY EXCEED THE MAXIMUM SIZE OF *n* PAGES**

Explanation: LOADPLUS determined that more pages might be required to load the data than is possible for the named data set. LOADPLUS issues this message instead of BMC51487E for nonunique indexes. If no duplicates exist, the loading of the data set will exceed the maximum size. Otherwise, the loading of the data set might or might not exceed the maximum size.

User Response: Reduce the number of rows loaded, reduce the PCTFREE value, increase the FREEPAGE value, or increase the PIECESIZE value, and resubmit the job. If the loading of the data set fails due to insufficient space in the data set, increase the PRIQTY/SECQTY values, reduce the PCTFREE value, increase the FREEPAGE value, or increase the PIECESIZE value and restart the utility.

BMC51490I **LOADING OF DATASET '*dsname*' WILL REQUIRE *n* PAGES (APPROX.)**

Explanation: During preload processing, LOADPLUS determined that loading the specified data set will require approximately *n* pages.

User Response: No action is required. However, if you specified PRELOAD PAUSE, you can use this information to reallocate your DB2 data sets before restarting the job at the LOAD phase.

BMC Software, Inc., Confidential and Proprietary Information

BMC51491I/E **MATERIALIZATION OF MISSING 'DEFINE NO' DATASETS FOR TABLE *creator.tbname status***

Explanation: For the specified table, LOADPLUS attempted to materialize the data sets that are defined with DEFINE NO by performing a DB2 INSERT and ROLLBACK on the table. This message is either informational (I) or an error message (E), depending on the value for *status*:

- IS BEGINNING is an informational (I) message.
- WAS SUCCESSFUL is an informational (I) message.
- WAS UNSUCCESSFUL is an error (E) message.

User Response: If the materialization was successful, no action is required.

If the materialization was unsuccessful, the load job terminates and LOADPLUS provides you with additional information about the reason that it could not materialize the data set. If the table has a check constraint that LOADPLUS could not resolve or a VALIDPROC, you must materialize the data set manually by performing an INSERT on the table, and then restart the load job.

BMC51492W **LOADPLUS WAS UNABLE TO RESOLVE ALL VALUES FOR TABLE *creator.tbname check_constraint_or_validproc* '*name*'. INSERT/ROLLBACK TO MATERIALIZE 'DEFINE NO' DATASETS MAY FAIL**

Explanation: Data sets for the specified table are defined with DEFINE NO, and you have a check constraint or VALIDPROC on the table.

If you have a check constraint on the table, LOADPLUS attempted to build a row that satisfies the check constraint (attempted to *resolve* the values that are associated with the constraint), but could not resolve all of the values. In this case, LOADPLUS might not be able to materialize the data sets.

If you have a VALIDPROC on the table, LOADPLUS might not be able to materialize the data sets.

User Response: If message BMC51491I with a *status* of WAS SUCCESSFUL follows this message, no action is required. If message BMC51491E with a *status* of WAS UNSUCCESSFUL follows this message, materialize the data sets manually by performing an INSERT on the table, and then restart the load job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51493W ***unique_type* INDEX '*creator.ixname*' WAS SKIPPED AND HAS BEEN PLACED IN REBUILD PENDING STATUS**

Explanation: You specified SKIPIX SIX or SKIPIX NUSIX on the LOAD command for a table that has an associated secondary index. Because LOADPLUS did not build or update the secondary index, LOADPLUS placed the index in a REBUILD pending status (RBDP, RBDP*, or PSRBD). LOADPLUS completes with a return code 4.

This message also indicates whether the index is unique or nonunique.

User Response: After the load job completes, you must rebuild the specified index.

BMC51494I ***unique_type* INDEX '*creator.ixname*' WAS SKIPPED BUT HAS BEEN LEFT IN ITS ORIGINAL STATUS DUE TO 0 ROWS LOADED**

Explanation: On a LOAD RESUME YES load job, you specified SKIPIX SIX or SKIPIX NUSIX on the LOAD command for a table that has an associated secondary index, but no rows were loaded in the table. LOADPLUS did not build or update the secondary index, but left the index in its original status.

This message also indicates whether the index is unique or nonunique.

User Response: No action is required.

BMC51495I ***phase* OPTIMIZATION, RC = *rc*, #SORTS = *n*, #READERS = *r*, INDEX TASKS = *i*, TIME = *t***

Explanation: LOADPLUS is in the process of determining the optimal number of sort, reader, and index tasks. Each iteration of the optimizer produces a message.

- *phase*—indicates the LOADPLUS processing phase in which the optimization is occurring
- *rc*—indicates the return code for the particular iteration
- *n*—indicates the number of sort processes being tested for this iteration of the optimizer
- *r*—indicates the number of input data sets (SYSRECs) that LOADPLUS will read concurrently
- *i*—indicates the number of index tasks that the load job will perform concurrently
- *t*—indicates the relative amount of time that the load job will take, based on the ability to process data concurrently. Note that *t* is a relative number, not a measure of any unit of time

At least one return code in the series of messages should be 0. Table C-4 on page C-146 describes the possible return codes.

BMC Software, Inc., Confidential and Proprietary Information

Table C-4 Sort, Read, and Index Optimization Return Codes

Return Code	Tasks Affected	Meaning
0	any	valid solution found
1	SORT	need more than 2 GB of storage above the line
2		insufficient storage above the line
3		need more than 16 MB of storage below the line
4		insufficient storage below the line
6	READ	insufficient memory for the specified number of XBLKS
7		insufficient storage above the line
8		insufficient storage below the line
10	INDEX	insufficient memory for the specified number of XBLKS
11		insufficient storage above the line
12		insufficient storage below the line
14	any	not the optimal time (must be preceded by a return code of 0)
15	SORT	cannot optimize SORTWK data sets
16	INDEX	cannot optimize SORTWK data sets
94	any	insufficient storage below the line for one sort task
95		insufficient storage above the line for one sort task
96		insufficient total storage for one sort task
97		insufficient total storage for work buffers
98		insufficient storage above the line for work buffers
99		insufficient memory for the number of SYSUT1 files
100	N/A	no memory available after subtracting global overhead

If at least one return code in the series is 0, BMC51496I follows this message. BMC51496I indicates the results of the optimization process.

If no return code is 0, BMC51497E follows this message.

For more information about how the optimizer determines the optimal number of sort processes, see “Controlling the Number of Sort Processes” on page 6-15. See “Interpreting Performance-Related Messages” on page 6-39 for more information.

User Response: No action is required for this message, but see the user response for any message that follows.

BMC Software, Inc., Confidential and Proprietary Information

BMC51496I ***phase* ANALYZE, #SORTS = *n*, #READERS = *r*, INDEX TASKS = *i***

Explanation: During the *phase* processing phase, LOADPLUS determined that *n* is the optimal number of sort processes for this job. The variable *r* indicates the number of input data sets (SYSRECs) that LOADPLUS will read concurrently (the total number of SYSRECs read can be greater than *r*). *i* indicates the number of index tasks that LOADPLUS will perform concurrently.

LOADPLUS issues this message following BMC51495I. For more information about how the optimizer determines the optimal number of sort processes, see "Controlling the Number of Sort Processes" on page 6-15.

User Response: No action is required. However, if you specified PRELOAD ANALYZE, you can use this information to reallocate your SORTOUT and SYSREC data sets before restarting your job.

BMC51497E ***phase* OPTIMIZATION FAILED. LOAD TERMINATED**

Explanation: This message indicates that an error occurred during optimization in the specified phase. The cause of this error is insufficient memory.

User Response: Increase the region parameter, specify REGION=0M, or use the information from previous messages to make other adjustments. Resubmit the job. If this fails to correct the problem, contact BMC Software Customer Support.

BMC51498I ***phase* OPTIMIZATION, RC = *rc*, #LOAD TASKS = *n*, #COPY TASKS = *c*, #INDEX TASKS = *i***

Explanation: LOADPLUS is in the process of determining the optimal number of load, copy, and index processes. Each iteration of the optimizer produces a message.

- *phase*—indicates the LOADPLUS processing phase in which the optimization is occurring
- *rc*—indicates the return code for the particular iteration
- *n*—indicates the number of load processes being tested for this iteration of the optimizer
- *r*—indicates the number of copy tasks that the load job will perform concurrently
- *i*—indicates the number of index tasks that the load job will perform concurrently

At least one return code in the series of messages should be 0. Table C-5 on page C-148 describes the possible return codes.

If no return code is 0, BMC51497E follows this message.

BMC Software, Inc., Confidential and Proprietary Information

Table C-5 Load, Copy, and Index Optimization Return Codes

Return Code	Tasks Affected	Meaning
0	any	valid solution found
1	LOAD	insufficient total storage
2		insufficient storage above the line
3		insufficient storage below the line
4	COPY	insufficient total storage
5		insufficient storage above the line
6		insufficient storage below the line
7	INDEX	insufficient total storage
8		insufficient storage above the line
9		insufficient storage below the line
90	any	insufficient total storage for selected load buffers
91		insufficient storage above the line for selected load buffers
92		insufficient storage below the line for minimum overheads
93		insufficient storage above the line for minimum overheads
94		insufficient region size for minimum overheads
95		insufficient storage for index tasks
96		insufficient storage for copy tasks
97		insufficient storage for load tasks
98		insufficient storage for copy buffers
99		insufficient storage for copy buffers

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51501E **INVALID NUMERIC DATA, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'**

NUMERIC VALUE OUT-OF-RANGE, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

STRING VALUE TOO LONG, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

DATE/TIME STRING TOO SHORT, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

DATE/TIME STRING TOO LONG, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

INVALID DATE/TIME DATA, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

DATE/TIME VALUE OUT-OF-RANGE, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

INVALID DOUBLE-BYTE STRING, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

DATA FAILED FIELDPROC ENCODE, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

DATA FAILED USER CONVERSION, COLUMN = '*colname*', RECORD NO. *n* OF *ddname*, DATA = '*data*'

Explanation: An error occurred during data conversion. If the error results in the record being discarded, there will be a corresponding line in the error summary report (which starts with message BMC51521I) that gives more detailed information. If you receive this message when loading CSV data, the data value that is displayed also contains the length of the field in the first four hexadecimal numbers.

User Response: Correct the data and resubmit the job.

BMC51502E **FIELD '*fldname*' EXTENDS BEYOND END-OF-RECORD, START POSITION = *s*, END POSITION = *e*, RECORD NO. *n* OF *ddname*, LENGTH = *l***

Explanation: For fields with a fixed displacement, a fixed length, and from a fixed length SYSREC, this is a fatal error and LOADPLUS terminates the job. For variable fields, the error is not fatal. LOADPLUS discards the row in error and continues processing.

User Response: Correct the field name or the data and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51503E **DUPLICATE KEY FOUND FOR CLUSTER INDEX '*creator.ixname*', RECORD NO. = *n* OF *ddname***

Explanation: LOADPLUS found a duplicate for the clustering index named. See message BMC51504E for more information.

User Response: Correct the clustering index and resubmit the job.

BMC51504E **KEY VALUE = 'EBCDIC *key_value_by_column*'**

Explanation: This message accompanies message BMC51503E. For columns with FIELDPROCS, the value displayed remains encoded.

User Response: Correct the problem that message BMC51503E refers to.

BMC51505E **DUPLICATE KEY FOUND FOR INDEX '*creator.ixname*', RECORD NO. = *n* OF *ddname***

Explanation: LOADPLUS found a duplicate for the named index. See message BMC51506E for more information.

User Response: Correct the information and resubmit the job.

BMC51506E **KEY VALUE = 'EBCDIC *key_value_by_column*', RID-1 = X'*r1*', RID-2 = X'*r2*'**

Explanation: This message accompanies message BMC51505E. RID-1 and RID-2 are the RIDs for the two key entries that conflict.

User Response: Correct the problem that message BMC51505E refers to.

BMC Software, Inc., Confidential and Proprietary Information

BMC51507I **XBLKS = *b*, XFERS = *t*, EMPTY WAITS = *w*, FULL WAITS = *f***

APBUFS = *b*, XFERS = *t*, EMPTY WAITS = *w*, FULL WAITS = 0

Explanation: This message provides statistics that LOADPLUS gathered during one of two different tasks, as follows.

XBLKS version

This version of the message provides statistics that LOADPLUS gathered during index work record creation:

- *b*—the number of transfer blocks used
- *t*—the number of block transfers
- *w*—the number of empty block wait counts
- *f*—the number of full block wait counts

See "Interpreting Performance-Related Messages" on page 6-39 for more information.

APBUFS version

This version of the message provides statistics that LOADPLUS gathered during the transfer of records from LOADPLUS to Apply Plus for a SQLAPPLY load (LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY):

- *b*—the number of Apply Plus buffers used
- *t*—the number of buffer transfers
- *w*—the number of empty buffer wait counts

Note: FULL WAITS is always 0 for this version of the message.

User Response: No action is required. However, you can use the information from the XBLKS version of this message to help improve LOADPLUS performance.

BMC51508I **MAX *type* TASKS = *t*, *object* PER TASK = *i*, SORTWKS PER TASK = *s*, MAX OPEN PARTITIONS PER TASK = *o***

Explanation: This message describes the information that LOADPLUS has determined that it will use to perform processing.

- *t*—indicates the maximum number of concurrent data or index tasks that LOADPLUS can multitask

  The number of tasks does not include other processes that always use a single task.

- *i*—indicates the maximum number of indexes or partitions that LOADPLUS can process per task based on the available resources

BMC Software, Inc., Confidential and Proprietary Information

- *s*—indicates the number of sort work files that are assigned to each task
- *o*—indicates the maximum number of partitions that can be open per task

When performing a two-phase load, LOADPLUS issues this message once for the index tasks and once for the data tasks. See “Interpreting Performance-Related Messages” on page 6-39 for more information.

User Response: No action is required. However, you can use this information to help improve performance. See “Interpreting Performance-Related Messages” on page 6-39 for more information.

BMC51509E **SQL -803 ENCOUNTERED FOR INDEX '*creator.ixname*', RECORD NO. *n* OF SYSREC *ddname* IS A DUPLICATE OF RID = *rowid***

Explanation: When running LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY, Apply Plus received an SQL -803 error from DB2, which indicates a duplicate record.

User Response: Check your discard file. Your response to this error depends on your requirements.

BMC51510I ***task_no*.: *task_name* TASK, *block/buffer* XFERS = *t*, EMPTY WAITS = *w*, FULL WAITS = *f***

Explanation: LOADPLUS gathered these performance statistics during processing.

- *task_name*—either READ, SORT, INDEX, or SQLAPPLY
- *block/buffer*—either XBLK (transfer blocks) or APBUF (Apply Plus buffers)
- *t*—the number of block or buffer transfers
- *w*—the number of empty block or buffer wait counts
- *f*—the number of full block wait counts, or 0 if this message applies to Apply Plus buffers

See “Interpreting Performance-Related Messages” on page 6-39 for more information.

User Response: No action is required.

BMC51511E **SQL -530 ENCOUNTERED FOR FOREIGN KEY '*constraint*', RECORD NO. *n* OF SYSREC *ddname***

Explanation: When running LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY, Apply Plus received an SQL -530 error from DB2, which indicates a referential integrity violation.

User Response: Correct the information and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51512E **WHEN SPECIFICATION (*start*:*end*) FOR TABLE '*creator.tbname*' EXTENDS BEYOND END-OF-RECORD, RECORD NO. *n* OF *ddname*, LENGTH = *l***

Explanation: The named (*start*:*end*) specification was not entirely contained in the input record.

User Response: Correct the (*start*:*end*) specification and resubmit the job.

BMC51513E **NULLIF/DEFAULTIF SPECIFICATION (*start*:*end*) FOR FIELD '*fldname*' EXTENDS BEYOND END-OF-RECORD, RECORD NO. *n* OF *ddname*, LENGTH = *l***

Explanation: The named (*start*:*end*) specification was not entirely contained in the input record.

User Response: Correct the (*start*:*end*) specification and resubmit the job.

BMC51514E **CHECK CONSTRAINT VIOLATION. TABLE ='*creator.tbname*', CONSTRAINT '*constraint*', RECORD NO. *n* OF *ddname***

Explanation: Record *n* from input from *ddname* violated the table check constraint *constraint* on table *creator.tbname*.

User Response: Correct the information and resubmit the job.

BMC51515E **RANGE VIOLATION FOR IDENTITY COLUMN '*colname*' OF '*creator.tbname*', RECORD NO. *record_number* OF *ddname***

Explanation: While loading identity column values, LOADPLUS determined that it would have to generate values that are outside the range that is defined on the identity column. The value for the IDERROR command or installation option is DISCARD, so LOADPLUS discards the specified record.

User Response: Correct your data and resubmit the job.

BMC51516E **RANGE EXCEEDED FOR IDENTITY COLUMN OF TABLE '*creator.tbname*'**

Explanation: While loading identity column values, LOADPLUS determined that it would have to generate a value that is outside the range that is defined on the identity column. The value for the IDERROR command or installation option is FAIL, so LOADPLUS terminates processing with return code 12.

User Response: Correct your data and resubmit the job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51517I **NO RECORDS WILL BE WRITTEN TO THE SYSDISC FILE BECAUSE AT LEAST ONE OF THE INPUT FILES IS A BATCHPIPE**

Explanation: This message displays when your input file is a batch pipe and either your JCL includes a SYSDISC DD statement or you are dynamically allocating a discard file.

User Response: No action is required.

BMC51519E **A PARTIAL SQLAPPLY LOAD IS NOT ALLOWED BECAUSE A GENERATED IDENTITY COLUMN DETERMINES PARTITION NUMBER**

Explanation: You attempted to run a SQLAPPLY load job for which the identity column is part of the limit key of a partition and for which identity column values are being generated. Under these circumstances, LOADPLUS cannot determine which partition to load.

User Response: Change your LOAD command and resubmit the job.

BMC51521I **LOADPLUS ERROR SUMMARY REPORT FOR *ddname*, ID# *nn***

Explanation: This is the heading for the report that contains the discarded record information for the specified input data set. The ID *nn* is used when there is a DUPLICATE KEY discard to which *ddname* has the related record.

User Response: No action is required.

BMC51522I **PHYSICAL LOGICAL DISCARD DISCARD RELATED TABLE FIELD, INDEX OR**

BMC51523I **RECORD RECORD RECORD TYPE ID: RECORD NAME CONSTRAINT NAME**

Explanation: These are the report headings for the specific discard record information.

- PHYSICAL RECORD—This is the number of the physical record of the discarded input record.
- LOGICAL RECORD—This is the number of the logical record of the discarded input record. The logical record number takes into account the concatenation of physical records if you specify CONTINUEIF. If you do not specify CONTINUEIF, the number of the logical record is the same as the number of the physical record.
- DISCARD RECORD—This is the number of the logical record of the associated discard record(s).

BMC Software, Inc., Confidential and Proprietary Information

- DISCARD TYPE—This is the reason that LOADPLUS discarded the record.

  — CONVERSION—A conversion error occurred while LOADPLUS was processing the record. The FIELD NAME identifies the input field that was being converted when LOADPLUS detected an error. If you specified MSGLEVEL(1), see also message BMC51505E.

  — CHECK CONSTRAINT—A table check constraint exists on this table and it would have been violated by this input record. The table check constraint name is given.

  — DUPLICATE KEY—The DB2 row that was created from the input record contained a duplicate key. If this is the first occurrence of a series of duplicate keys, the number in the RELATED DISCARD column matches the number of this record's physical record. Check the RELATED DISCARD to identify the conflicting input record. If you specified MSGLEVEL(1), see also messages BMC51503E through BMC51506E.

  — FIELD SPECIFICATION—LOADPLUS cannot process the field because it is not entirely contained in the input record. The FIELD NAME identifies the input field being processed when LOADPLUS detected an error. If you specified MSGLEVEL(1), see also message BMC51502E.

  — PARTITION—LOADPLUS discarded the DB2 row that was created from the input record because the partition to which it belongs is not being loaded (that is, the partition is not specified on INTO TABLE PART).

  — SQL-530—The DB2 row was not inserted because it received an SQL -530 error. This type of discard occurs only with a LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY load.

  — SQL-803—The DB2 row was not inserted because it received an SQL -803 error. This type of discard occurs only with a LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY load.

  — VALIDPROC—The table's VALIDPROC routine did not accept the DB2 row that was created from the input record.

  — WHEN—No INTO TABLE option selected the record based on selection criteria; or you specified FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD and LOADPLUS discarded the record due to a table OBID mismatch or a column definition mismatch.

- RELATED ID:RECORD—For a DUPLICATE KEY error, the number in the RELATED ID:RECORD column is normally the ID number and physical record number of the record that is in conflict with this record. But if this is the first occurrence of a series of duplicate keys, the number in the RELATED ID:RECORD column matches the number of this record's physical record.

BMC Software, Inc., Confidential and Proprietary Information

For a CONVERSION error, the RELATED RECORD number has the following meaning:

1 The numeric input value is invalid.

2 The numeric value is out of range for the column.

3 The string value is too long for the column.

4 The date/time string is too short.

5 The date/time string is too long.

6 The date/time string is invalid.

7 The date/time value is out of range.

8 The double-byte string has an odd number of bytes or an invalid shift-in or shift-out character

9 The input data is rejected by the column's FIELDPROC.

10 The input data is rejected by the user-written conversion routine.

- TABLE NAME—This is the table name of the table (specified in the INTO TABLE option) that is associated with the error.
- FIELD, INDEX, OR CONSTRAINT NAME—This is the field or index name that is associated with the error. The field name is used for CONVERSION and FIELD SPECIFICATION errors. The index name is used for DUPLICATE KEY errors.

User Response: No action is required.

BMC51524E ***discard_error_information***

Explanation: This message describes a discard record.

User Response: No action is required

BMC Software, Inc., Confidential and Proprietary Information

BMC51530I LOADPLUS DASD REQUIREMENT ESTIMATES

BMC51531I 3380 CYLS 3390 CYLS

BMC51532I DDNAME KBYTES PRI SEC PRI SEC INDEX

BMC51533I *ddname k p s p s creator.ixname*

Explanation: ANALYZE has estimated the data set sizes in cylinders (by device type), for the load (SORTOUT), work (SYSUT1), sort work (SORTWK), discard (SYSDISC), error (SYSERR), and copy data sets. LOADPLUS provides the estimated information in a table format. Messages BMC51530I, BMC51531I, and BMC51532I provide the heading information and multiple BMC51533I messages provide the estimated values. Since some types of load jobs allow you to specify single or multiple SORTOUT and SYSUT1 data sets, this message provides values for both.

If you specified ANALYZE ONLY, LOADPLUS provides values for all data sets, regardless of whether you need them for the type of load that you are running. This gives you the information that you need to run any type of load job.

LOADPLUS issues a separate BMC51533I message for each data set and this message provides the following information:

- DDNAME—data set name
- KBYTES—number of kilobytes
- 3380 CYLS—primary (PRI) and secondary (SEC) 3380 cylinder quantities
- 3390 CYLS—primary (PRI) and secondary (SEC) 3390 cylinder quantities
- INDEX—index name, where applicable

User Response: No action is required.

BMC51535I ESTIMATED CARDINALITY OF SPACE = *n* ESTIMATED AVG ROW LENGTH = *n*

Explanation: ANALYZE estimated the cardinality and average row length of the table space that you are loading. LOADPLUS uses this number when calculating the values in message BMC51533I.

User Response: No action is required.

BMC51536I CARDINALITY OF SPACE = *n* ESTIMATED AVG ROW LENGTH = *n*

Explanation: ANALYZE calculated the exact cardinality and estimated the average row length for the table space that you are loading. LOADPLUS uses this number when calculating the values in message BMC51533I.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51539I ***task_no*: SAMPLING STATISTICS: PART=*p*,TP=*n*,SP=*n*,SR=*n*,AVGR=*a*,SD=*n*, SE=*n*,AVGF=*f*,SD=*n*,SE=*n*,EP=*n***

Explanation: This message displays the sampling statistics that ANALYZE used to determine the cardinality and average row size for the data that you are loading. This information might be useful to BMC Software Customer Support when diagnosing a problem. It does not provide information that is useful for performance tuning. LOADPLUS produces this message only if you specified MSGLEVEL(1).

The following list describes each of the statistics displayed in this message:

- PART—partition number

  If the table space is not partitioned, this value is 0.

- TP—total number of pages in the data set

- SP—number of sampled pages

- SR—total number of rows on the sampled pages

- AVGR—average row length determined by this sample

  For LOADPLUS, this value is always 0.

- SD—standard deviation of AVGR

  For LOADPLUS, this value is always 0.

- SE—standard error of AVGR

  For LOADPLUS, this value is always 0.

- AVGF—number of rows or key RIDs per pages times 100, as determined by this sample

- SD—standard deviation of AVGF

- SE—standard error of AVGF

- EP—number of empty pages (pages having zero rows) in this sample

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51567E 'LOADDN' AND 'WORKDDN' SPECIFICATIONS ARE TOO SIMILAR

Explanation: For dynamic work file allocation, LOADPLUS must be able to determine whether ddnames that you specify in the JCL are intended to be LOAD data sets or WORK data sets. For this reason, ddname prefixes that you specify with the LOADDN and WORKDDN options must be different enough for LOADPLUS to make this distinction.

Different enough means that one prefix cannot be a substring of the other.

For additional information, see the syntax descriptions for these two options on pages 3-37 and 3-38.

User Response: Change your LOADDN or WORKDDN specification and resubmit your job.

BMC51576E/I/S/U/W *Apply_Plus_message*

Explanation: This message contains an Apply Plus message that Apply Plus passes to LOADPLUS during a LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY load. The message severity of the LOADPLUS message (BMC51576) reflects the severity of the Apply Plus message.

User Response: Refer to your Apply Plus documentation for information about this message.

BMC51577I ROWS DISCARDED

BMC51578I INSERTED SQLERRORS

BMC51579I *object_type object_name rows sql_discards*

Explanation: These messages display the following statistics for an online load that uses Apply Plus (LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY):

- *object_type*—the type of object that you are loading (table space, table, or part)
- *object_name*—the name of the object that you are loading
- ROWS INSERTED (*rows*)—the number of rows that LOADPLUS successfully loaded
- DISCARDED SQLERROR (*sql_discards*)—the number of rows that LOADPLUS discarded due to SQL errors

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51580E **APPLY PLUS FOR DB2 FOR 0S/390 VERSION *v.r* OR HIGHER IS NOT AVAILABLE**

BMC51581I **APPLY PLUS MUST BE AVAILABLE IN STEPLIB, JOBLIB, OR LINKLIST TO PERFORM A RESUME YES SHRLEVEL CHANGE (SQLAPPLY) LOAD. NO ADDITIONAL LICENSE IS REQUIRED TO USE THIS FEATURE.**

Explanation: You specified LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY, but the correct version of the Apply Plus component of the BMC Software Log Master for DB2 product is not available to LOADPLUS.

User Response: Install the correct version of Apply Plus, which shipped with the LOADPLUS product. If the correct version is already installed, make Apply Plus available to LOADPLUS in the STEPLIB, JOBLIB, or LINKLIST.

BMC51582W **THERE ARE RI RELATIONSHIPS BETWEEN TABLES TO BE LOADED AND LOADPLUS WAS UNABLE TO RESOLVE ALL PARENT-CHILD RELATIONSHIPS. -530 SQL ERRORS MAY RESULT.**

Explanation: You specified ORDER YES with LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY and there are referential integrity (RI) relationships between two or more of the tables you are loading. For this type of load, LOADPLUS attempts to resolve all RI relationships. However, LOADPLUS could not resolve all RI relationships for this particular job.

During the COMBINED phase, LOADPLUS uses DB2 to perform insert processing. As a part of this processing, DB2 looks at RI relationships. Therefore, if LOADPLUS could not resolve all RI relationships for this job, you might also obtain SQL -530 errors when processing passes to DB2.

User Response: No action is required. However, this warning might have occurred because you were loading tables that are self-referential or that have circular relationships. You can prevent this warning (and potential -530 SQL errors) in the future by ordering the data prior to your load job and specifying ORDER NO with your LOAD command.

BMC51586W **MAXASSIGNEDVAL FOR IDENTITY COLUMN OF TABLE '*creator.tbname*' HAS BEEN RESET TO *current_value*. PREVIOUS VALUE WAS *old_value*.**

Explanation: While generating identity column values, LOADPLUS updated the value of the MAXASSIGNEDVAL field of SYSIBM.SYSSEQUENCES. See “How LOADPLUS Updates MAXASSIGNEDVAL” on page 2-35 for information about the value with which LOADPLUS updates this field.

User Response: No action is required.

BMC Software, Inc., Confidential and Proprietary Information

BMC51587W **THE VALUE SPECIFIED FOR IDCACHE IS INVALID. IT HAS BEEN RESET TO *n*.**

Explanation: You specified a value for the IDCACHE option that is outside the valid range of 1 through 100000. LOADPLUS sets the value for this option to one of the following values and continues processing:

- If you specified a value greater than 100000, LOADPLUS sets this value to 100000.
- If you specified a value less than 1, LOADPLUS sets this value to the default that was specified at installation.

User Response: No action is required.

BMC51590E/W **FORMAT BMCUNLOAD VERIFICATION FAILED - *reason***

Explanation: You specified FORMAT BMCUNLOAD, but one of the following conditions occurred during verification processing:

- The verification records in the input data set (SYSREC) do not match the table that you are loading. Two instances of message BMC51591 follow this message.
- The identity column or ROWID column on the table that you are loading is defined as GENERATED ALWAYS. Two instances of message BMC51591 follow this message.
- The input data file does not contain header records that indicate that it was generated by UNLOAD PLUS.

If you receive the warning version of this message, LOADPLUS continues loading your table. If you receive the error version of this message, either the load job fails or LOADPLUS discards the rows as it attempts to find a valid verification record for the target tables specified in the load job. If LOADPLUS does not find a valid verification record, the job completes with a return code 0.

User Response: If the verification records do not match the table that you are loading, use the messages that follow this one to determine where the discrepancy lies between the two tables. If your identity column or ROWID column on the table being loaded is defined as GENERATED ALWAYS, you must ALTER the table definition and then resubmit your job. If you are not loading an input file that was generated by UNLOAD PLUS, correct the options on your LOAD command and resubmit your job.

BMC Software, Inc., Confidential and Proprietary Information

BMC51591E/W ***unloaded_or_loaded*** **- SSID='*subsystem_id*', TABLE='*creator.tbname*', COLUMN='*colname*', *attribute*='*definition*'**

Explanation: In certain cases, two instances of this message follow message BMC51590E. These messages describe the attributes of the table that was unloaded and the attributes of the table that you are loading. Use these messages to determine where the discrepancy lies between the two tables.

If you receive the warning version of this message, LOADPLUS continues loading your table. If you receive the error version of this message, LOADPLUS discards the rows as it attempts to find a valid verification record for this table.

User Response: See "Data from UNLOAD PLUS in Internal Format" on page 2-44 for information about allowable differences between the table that you unloaded and the table that you are loading. If necessary, change the definition of the table that you are loading to match the table that you unloaded and resubmit your job.

BMC51592I PRELOAD STATISTICS: *n* BMCUNLOAD VERIFICATION RECORDS READ FROM *ddname*

Explanation: You specified FORMAT BMCUNLOAD, and LOADPLUS read *n* verification records from the named input data set.

User Response: No action is required.

ICOPY Installation Messages

Messages that the ICOPY job issues are shown here in bold. When a message contains text that can vary, the type of variable is shown in lowercase letters. The actual message contains the specific text of the variable type shown.

BMC75000 MESSAGE TEXT IS MISSING = *msgid*

Explanation: The product has issued a message that was not coded as a BMC Software message. This is an internal error.

User Response: Contact BMC Software Customer Support and report the value of *msgid*.

BMC75001 NOT APF AUTHORIZED

Explanation: The product has detected that it does not have the authorization needed to run correctly. The product must run from an APF-authorized library.

User Response: Contact your system programming staff, if necessary, to have the product copied to an authorized library.

BMC Software, Inc., Confidential and Proprietary Information

BMC75003 **SUBSYSTEM *ssid* IS MEMBER *member_name* OF DATA SHARING GROUP *group_name***

Explanation: The product issues this message to identify the DB2 subsystem and data sharing group in a DB2 data sharing environment.

User Response: No action is required.

BMC75004 **A GROUP ATTACH WAS PERFORMED WITH *group_attach_name***

Explanation: The product issues this message when a group attachment name is used to connect to a DB2 data sharing member.

User Response: No action is required.

BMC75005 **UTILITY EXECUTION COMPLETE, RETURN CODE = 0**

Explanation: The execution of the product is complete.

User Response: No action is required.

BMC75006 **UTILITY EXECUTION TERMINATING *plan_name ssid*, RETURN CODE = *return_code***

Explanation: The product terminated with an error or warning.

User Response: If this return code is 4, indicating a warning, search for WARNING in your output. Note that the product may continue to process some objects after a failure on another object. Therefore, the error message may not occur immediately prior to this message. If you had an error, correct it, then restart the product.

BMC75007 **ATTEMPTING TO CONNECT TO DB2 SUBSYSTEM *ssid*, USING PLAN *plan_name***

Explanation: The product attempted to connect to the DB2 subsystem shown using the plan name shown.

User Response: No action is required if the connection is successful. If the connection is not successful, check the subsystem ID (*ssid*) and plan name (*plan_name*) given in the message to determine the problem.

BMC75008 ***text***

Explanation: The text shown is output from a facility used by this product (for example, DB2 SQL, DB2 Call Attachment Facility, or IDCAMS).

User Response: Review the messages and any previous messages to determine an appropriate response.

BMC Software, Inc., Confidential and Proprietary Information

BMC75009 **UNEXPECTED SQL ERROR. SQLCODE = *sqlcode*. RC = *return_code*. ATTEMPTING TO PROCESS THE ERROR. ANY AVAILABLE TEXT FOLLOWS:**

Explanation: The product received an SQL error, but an attempt to call DSNTIAR to explain the error failed.

User Response: Contact BMC Software Customer Support with the values of *sqlcode* and *return_code* shown in the message.

BMC75010 **SUCCESSFUL CONNECT TO *ssid* (RELEASE *n.n* OF DB2) USING PLAN *plan_name***

Explanation: This informational message gives the DB2 subsystem ID (*ssid*), version (*n.n*), and the plan name (*plan_name*) in use.

User Response: No action is required.

BMC75112 **INTERNAL MEMORY MANAGEMENT ERROR. CODE = *int***

Explanation: An internal error occurred.

User Response: Contact BMC Software Customer Support with the code shown in this message. You should also provide

- the job output that contains the error messages (as defined by the SYSPRINT DD statement)
- any job output that contains a snap dump (The product produces a snap dump if the JCL contains a SYSERR DD statement.)

BMC75122 **REBIND OF PLAN *plan_name* FAILED. AUTHID = *authid*. RC = *return_code***

Explanation: The job attempted to rebind the product plan, but was unsuccessful.

User Response: Verify that the authorization ID shown in the message has authority to rebind the plan. A rebind may fail if the DB2 synonyms or tables used in the plan do not exist. Try to rebind the plan directly. In addition, you can rerun the bind job (*product_code ssid* B) that was created in the control library downloaded from the installation tape, and then review the DB2 messages. Correct the problem and then submit the job again.

BMC75124 ***text***

Explanation: An error occurred as a result of the DB2 Call Attachment Facility (CAF), an SQL statement, or a DB2 command.

User Response: Take the appropriate action to correct the error identified.

BMC Software, Inc., Confidential and Proprietary Information

BMC75251 INSTALL CATALOG UPDATE EXECUTION STARTING *date time*

Explanation: This is the start message for the BMC Software install utility. The date and time of the start are given.

User Response: No action required.

BMC75252 DB2 RELEASE *release* IS NOT SUPPORTED BY THIS UTILITY

Explanation: The install utility for this product is being run on an unsupported release of DB2.

User Response: You cannot install or run the product on an unsupported version of DB2. Contact BMC Software Customer Support for supported version information for this product.

BMC75253 WARNING: THIS UTILITY HAS ALREADY BEEN RUN FOR THIS PLAN (*plan_name*) ON DB2 SYSTEM *ssid*

Explanation: This message should be preceded by message BMC75272. The data in the plan is consistent with the effects of a previous run of the install utility.

User Response: Check that the plan name in this message is correct for the copy of the product you are attempting to install. If it is incorrect, check the value for the PLANINST option in the install options job $C30DOPT and run the job again if necessary. If the plan name is correct and you are sure that the product install utility was not previously run for this copy of the product, contact BMC Software Customer Support.

BMC75254 NO SQL STATEMENTS FOUND FOR USERID *authid* AND PLAN *plan_name*, ON DB2 SUBSYSTEM *ssid*

Explanation: The install utility for this product has searched the SYSIBM.SYSSTMT table to find information about the product plan. No SQL statements of interest were found.

User Response: Verify the following items:

- Check that the second parameter (*authid*) specified to the install utility is the same user ID specified as the owner when the product execution plan was bound on the subsystem (*ssid*) identified by the first parameter of the install utility.
- Check that a valid synonym exists for all BMC-created tables under the authorization ID specified to the install utility. (For example, one BMC-created table is named BMC*xxx*_BMCCOPY, where *xxx* is the product code).
- Check that the plan name (*plan_name*) is the name of the product execution plan bound in this subsystem.

If all checks are satisfactory, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC75259 ******* INSTALL CATALOG SUPPORT *version* COPYRIGHT BMC SOFTWARE INC. *date***

Explanation: This message displays the version number and copyright for the install catalog support utility.

User Response: No action is required.

BMC75260 **THE INSTALL CATALOG SUPPORT UTILITY IS NOT APF AUTHORIZED - CONTACT YOUR SYSTEMS PROGRAMMING STAFF**

Explanation: The install catalog support utility has detected that it does not have the authorization needed to run correctly. This program must run from an APF-authorized library.

User Response: If necessary, contact your system programming staff to have the install utility copied to an authorized library.

BMC75264 **THE *parameter_description* PARAMETER IS MISSING FROM THE JCL**

Explanation: The *parameter description* is either SSID, AUTHID, or PRODUCT CODE and describes the missing JCL parameter.

User Response: If the product installation panels or the generated JCL have not been altered, contact BMC Software Customer Support.

BMC75265 **THE PRODUCT CODE ENTERED, '*product_code*', IS INVALID**

Explanation: *product_code* is not valid for this product. This is the third parameter passed to the install utility in the JCL generated by the installation panels. Valid product codes are

- ACP (COPY PLUS)
- AFR (RECOVER PLUS)
- ACA (R+/CHANGE ACCUM)
- ALP (Log Master)
- AMU (LOADPLUS)
- ARU (REORG PLUS).

User Response: If the installation panels or the generated JCL have not been altered, contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC75266 NO ROWS FOUND IN SYSIBM.SYSSTMT FOR STATEMENT '*statement_text*' IN PLAN *plan_name*

Explanation: A synonym was found for the authorization ID specified in the JCL and the table name in *statement_text*, but no statements were found in the SYSIBM.SYSSTMT table for the product execution plan (*plan_name*). This may indicate an unnecessary synonym, in which case the install utility will complete with return code 0 and this message can be ignored.

User Response: If the install utility completes with return code 0 and no other messages are issued, no action is required.

BMC75267 THE ATTACH OF THE DSNUTILB PROGRAM TO EXECUTE THE REPAIR UTILITY FAILED WITH RETURN CODE *return_code*

Explanation: The install utility attaches program DSNUTILB to execute the REPAIR utility. The attach has failed and a decimal return code from the attach macro is given. There may be additional system messages and a SYSUDUMP dump may have occurred.

User Response: Collect all output from the install utility job and contact BMC Software Customer Support.

BMC75268 THE REPAIR UTILITY ABENDED WITH CODE X'*hex_abend_code*' - THE SYSPRINT OUTPUT FOLLOWS:

Explanation: The install utility attaches program DSNUTILB to execute the REPAIR utility. DSNUTILB has terminated with the ABEND code shown. There may be additional system messages and a SYSUDUMP dump has probably occurred. Other messages following this one may be helpful.

User Response: Correct the cause of the ABEND code and submit the install utility again. If this is not possible, contact BMC Software Customer Support.

BMC75270 THE REPAIR UTILITY ENDED WITH RETURN CODE *return_code* - THE SYSPRINT OUTPUT FOLLOWS:

Explanation: The install utility attaches program DSNUTILB to execute the REPAIR utility. DSNUTILB has terminated with the decimal return code shown. The messages following this one are a copy of the SYSPRINT output from REPAIR and should include an explanation of the return code.

User Response: Collect all output from the install utility job and contact BMC Software Customer Support.

BMC Software, Inc., Confidential and Proprietary Information

BMC75273 **THE DATA X'*hex_data*' AT ROW '*sectno_value seqno_value*' OFFSET X'*hex_offset_value*' FAILS TO MATCH THE DBID AND OBID FOR ANY TABLE**

Explanation: Either the product execution plan has been corrupted, or an internal error occurred in the install utility.

User Response: Contact BMC Software Customer Support.

BMC75274 **CAF ERROR RC = *return_code*, REASON = *code*, PLAN = *plan_name*, SSID = *ssid***

Explanation: This message accompanies an earlier message to give more specific information on a DB2 Call Attachment Facility (CAF) error. Other messages may follow. If the message indicates a CAF code F30021, you may have a mismatch between the DB2 release level of the subsystem and the DB2 library in your STEPLIB.

User Response: Determine the cause of the problem and correct it. Then submit the job again. If necessary, refer to *IBM DB2 UDB for OS/390 Messages and Codes* document for further explanation of the codes.

BMC75275 ***text***

Explanation: This message accompanies a previous BMC75274 message and provides the text for a CAF error.

User Response: See message BMC75274.

BMC75276 **CANNOT CONNECT TO DB2**

Explanation: The product cannot connect to the DB2 subsystem specified. See message BMC75274 for more information.

User Response: Correct the problem and submit the job again.

BMC75277 **AUTHID PARAMETER *authid1* DOES NOT MATCH BIND AUTHID *authid2* OF PLAN *plan_name***

Explanation: The authid parameter of the product installation plan must match the authid (CREATOR column of SYSIBM.SYSPLANS) of the plan.

User Response: Change the parameter to match the creator of the plan, or check that you are using the correct plan name. The plan name is defined in the options module for the product. Valid option modules are:

- ACP$OPTS (COPY PLUS)
- AFR$OPTS (RECOVER PLUS)
- ACA$OPTS (R+/CHANGE ACCUM)
- ALP$OPTS (Log Master)
- AMU$OPTS (LOADPLUS)
- ARU$OPTS (REORG PLUS)

BMC Software, Inc., Confidential and Proprietary Information

BMC75278 WARNING: NO ZAPS WERE GENERATED FOR TABLE *dbname.spname*

Explanation: The product expected to generate zaps for the table shown but did not find any SQL statements requiring change.

User Response: If the product or the product's Installation Verification Procedure (IVP) fails with return code greater than four, check that all install jobs were run successfully, especially the ones that create BMC tables and indexes.

BMC75279 WARNING: AN INDEX MATCHING *dbname.ixname* IS NOT DEFINED OR IS DEFINED INCORRECTLY

Explanation: The product detected that the index indicated is not currently defined in DB2 or does not have the expected definition.

User Response: Verify that all install jobs ran successfully. If they did, contact BMC Software Customer Support.

BMC75280 ERROR: INDEX *creator.ixname* IS DEFINED INCORRECTLY

Explanation: The product expected to find an index defined on its control tables whose definition matches the DB2 catalog index named in the message.

User Response: Verify whether all product installation jobs ran successfully. If any of the jobs did not run successfully, correct the problem, then re-execute the unsuccessful job and all subsequent installation jobs. If all installation jobs ran successfully, contact BMC Software Customer Support.

BMC75300 INTERNAL ERROR: *routine*, CODE = *code*

Explanation: An internal error occurred.

User Response: Contact BMC Software Customer Support with the values of *routine* and *code* shown in this message. You should also provide

- the job output that contains the error messages (as defined by the SYSPRINT DD statement)
- any job output that contains a snap dump (The product produces a snap dump if the JCL contains a SYSERR DD statement.)

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Appendix D Calculating LOADPLUS Work Data Set Sizes

This appendix presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Overview

This appendix provides formulas to help you calculate the allocation requirements of the various work files for the LOADPLUS product. This appendix also includes several example calculations. If you run a load job with ANALYZE PAUSE or ANALYZE ONLY, LOADPLUS uses the formulas in this appendix to do the calculations for you and generates a report with the results.

As an alternative to specifying data sets in your JCL, you can have LOADPLUS dynamically allocate your SORTOUT, SYSUT1, SORTWK, SYSDISC, SYSERR, and copy data sets. If you enable dynamic allocation for each DDTYPE, LOADPLUS calculates the file sizes that it needs and dynamically allocates the files. You must specify the ENUMROWS command option to provide LOADPLUS with the information that it needs to make the calculations. For more information, see “Dynamic Work File Allocation” on page 2-13 and “Dynamic Work File Allocation Options” on page 3-133.

DASD Capacity

Table D-1 shows the different track capacities for various DASD models to assist you in determining space allocations.

Table D-1 DASD Capacity Conversion Chart

	Model	
	3390-1 3390-2 3390-3	**3380 3380-E 3380-K**
Track capacity (bytes)	56,664	47,476
Tracks per cylinder	15	15
Total cylinders per volume	1,113 2,226 3,339	885 1,770 2,655

Tracks

Use Table D-1 and the following formula to calculate the number of tracks required:

```
# DASD bytes required/track capacity
```

BMC Software, Inc., Confidential and Proprietary Information

Cylinders

Use Table D-1 on page D-2 and the following formula to calculate the number of cylinders required:

```
# tracks required/# tracks per cylinder
```

Note: Tracks are rarely fully utilized due to block overhead when there are multiple blocks per track (which are required for 3380 and 3390 devices). BMC Software strongly recommends that you increase the amount of primary DASD allocation by a minimum of 2 percent to account for the block overhead.

SORTOUT Data Set

LOADPLUS requires the SORTOUT data set for a two-phase load. For a single-phase load, the SORTOUT data set is required if you specify LOAD RESUME YES SHRLEVEL NONE without PART REPLACE, you specify ORDER YES, and you want to be able to restart. LOADPLUS does not use this data set for a single-phase LOAD REPLACE load or for a LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY load (also referred to as a SQLAPPLY load).

LOADPLUS populates the SORTOUT data set during the PRELOAD or the COMBINED phase. For a two-phase load job, the data set contains the DB2 row images to load into the tables that you specified on the LOAD command. For a single-phase load job, the data set contains the clustering index and the DB2 row images and LOADPLUS uses this data set only if you restart your job.

The space that LOADPLUS requires for the SORTOUT data set is the sum of the space that LOADPLUS requires for each table that you select to load. The space required for the SORTOUT data set depends on these factors:

- row length for each loaded table

  — If the rows are variable length, use the average row length.
  — If the rows are compressed, consider the compression ratio in your calculations.

- number of new rows selected for each loaded table

BMC Software, Inc., Confidential and Proprietary Information

- number of rows that already exist for each loaded table, if you specify RESUME

Note: Do not include IBM's row overhead when you determine the length of the row.

In the following formulas, the variable *n* represents the row overhead that LOADPLUS needs. The value of *n* is 9 bytes for a table space that is LARGE (either by definition or default) or defined with DSSIZE, and 8 bytes for other table spaces.

Start with the following basic calculation for each table:

```
# new rows selected for this table × (average row length
+ n bytes)
```

Add additional space to the basic calculation for each of the following statements that applies to the load that you are running.

1. If you specify ORDER YES or ORDER PRESORTED, add the value calculated by the following formula to the amount that you have calculated so far:

   ```
   # new rows selected for this table × (length of
   longest clustering index key associated with this
   table space)
   ```

2. If you specify ORDER YES, are adding data to an existing table space or partition, and you do not specify INDEX UPDATE, add the value calculated by the formula that follows to the amount that you have calculated so far:

   ```
   # keys in existing clustering index × (length of
   longest clustering key + n bytes)
   ```

If you are loading just one table, the primary space allocation equals the estimated table size (figured in the preceding basic calculation). If you are loading more than one table, add the estimated table size values for all tables to determine the primary space allocation.

Calculate the secondary space allocation by determining the space that LOADPLUS would need if all the rows were the maximum length (including the four-byte overhead for variable length records), less the primary space value. If this difference is zero (as it will be if the rows are fixed length), provide a secondary space allocation that is 25 percent of the primary allocation. If your rows are very small (less than 20 bytes), provide a secondary allocation that is 50 percent of the primary allocation.

BMC Software, Inc., Confidential and Proprietary Information

If you are specifying multiple SORTOUT data sets, divide your total allocation by the number of data sets to obtain the individual allocation for each data set. For performance considerations, see “Using Multiple SORTOUT Data Sets” on page 6-7.

SYSUT1 Data Set

LOADPLUS requires a SYSUT1 data set for a two-phase load. For a single-phase load, the SYSUT1 data set is required if you specify LOAD RESUME YES SHRLEVEL NONE and you want to be able to restart. LOADPLUS does not use this data set for a single-phase LOAD REPLACE load or for a SQLAPPLY load.

The SYSUT1 data set contains information that LOADPLUS needs to build participating nonclustering indexes. If you specify ORDER NO on the LOAD command, LOADPLUS treats clustering indexes like participating nonclustering indexes. For a single-phase load, LOADPLUS uses these data sets only if you restart your job.

During the PRELOAD or COMBINED phase, LOADPLUS places all index work records in one or more SYSUT1 data sets. Each work record contains a single key value for an index. These work records must be sorted before the index can be built. The space that LOADPLUS requires for the SYSUT1 data set depends on the following factors:

- whether you are using multiple or single SYSUT1 data sets
- size of the index keys
- number of rows selected for each loaded table
- number of indexes participating in the load
- number of rows that already exist for each loaded table, if you specify RESUME YES

Multiple SYSUT1*nn* Data Sets

The amount of space that you need is significantly less if you use multiple SYSUT1*nn* data sets instead of a single SYSUT1 data set. Because LOADPLUS places work records for each index in a separate data set and sorts each data set individually, LOADPLUS does not pad the key values to the length of the longest defined key when you use multiple SYSUT1*nn* data sets.

In the following formulas, the value of the variable *n* is 7 bytes for a table space that is LARGE (either by definition or default) or defined with DSSIZE, and 6 bytes for other table spaces.

BMC Software, Inc., Confidential and Proprietary Information

Each SYSUT1 data set holds one index. Calculate the space that LOADPLUS needs for each index by using one of the following formulas. Select the appropriate formula based on the type of index and the keywords that you specify on the command.

- If the index is nonpartitioned and you do not specify INDEX UPDATE, use the following formula:

  ```
  (total # rows expected to be in the associated table
  after the load) × (length of index key + n bytes)
  ```

- If the index is nonpartitioned and you specify INDEX UPDATE, use the following formula:

  ```
  (# new rows added to the associated table) × (length
  of index key + n bytes)
  ```

- If the clustering index is partitioned, you specify ORDER NO, and you do not specify INDEX UPDATE, use the following formula:

  ```
  (total # rows expected to be in participating
  partitions after the load) × (length of index key +
  n bytes)
  ```

- If the clustering index is partitioned, and you specify ORDER NO and INDEX UPDATE, use the following formula:

  ```
  (# new rows added to participating partitions) ×
  (length of index key + n bytes)
  ```

Note: LOADPLUS puts the clustering index keys for all participating partitions in a single SYSUT1 data set.

Allocate the space for the participating index with the largest key length in the first SYSUT1*nn* DD statement in your JCL, space for the index with the second-largest key length in the second SYSUT1*nn* DD statement, and so on. Use the amount that you calculated previously for each index's primary quantity. Provide a small secondary quantity allocation as a safety measure against out-of-space conditions.

Single SYSUT1 Data Set

Because the sort process requires that all key values be the same length, any key value shorter than the longest key defined is padded with zeros. This can dramatically increase the space requirement for the SYSUT1 data set.

BMC Software, Inc., Confidential and Proprietary Information

In the following formulas, the value of the variable *n* is 7 bytes for a table space that is LARGE (either by definition or default) or defined with DSSIZE, and 6 bytes for other table spaces.

Use the following procedure to calculate the space that LOADPLUS needs for all of the indexes on each table that you are loading. Select the appropriate formula based on the type of index and the keywords that you specify on the command. If you specify ORDER NO on the LOAD command, for the purposes of these calculations, the clustering index is a participating nonclustering index.

For each table that you are loading, calculate the space that LOADPLUS needs by using one of the following formulas:

- If you do not specify INDEX UPDATE, use the following formula:

  ```
  (total # of rows expected to be in this table after
  the load) × (length of longest participating
  nonclustering index key on any table involved in the
  load + n bytes) × (# of participating nonpartitioned,
  nonclustering indexes on this table)
  ```

- If you specify INDEX UPDATE, use the following formula:

  ```
  (# of new rows added to this table) × (length of
  longest participating nonclustering index key on any
  table involved in the load + n bytes) × (# of
  participating nonclustering indexes on this table)
  ```

In addition, if you specify ORDER NO, do not specify INDEX UPDATE, and have a partitioned index, use the following formula to account for the space that this index needs:

```
(# of rows expected to be in the participating partitions
after the load) × (length of the longest participating
index key on the partitioned table + n bytes)
```

Add this result to the amount that you already calculated.

If you are loading multiple tables, add the amounts that you have calculated for each table to find the amount of space that LOADPLUS needs for the primary allocation for the SYSUT1 data set. Provide a small secondary space allocation as a safety measure against out-of-space conditions.

BMC Software, Inc., Confidential and Proprietary Information

SYSDISC Data Set

The SYSDISC data set is optional unless you specify DISCARDS on your LOAD command. SYSDISC contains records from the SYSREC data set that LOADPLUS discards because of errors. The space that LOADPLUS requires for the SYSDISC data set depends on the following factors:

- record length of the SYSREC data set
- number of records that are discarded due to errors

Use the following formula to calculate the space that LOADPLUS needs for the SYSDISC data set:

```
(LRECL of SYSREC) × (# records expected to be discarded)
```

If you expect few or no errors, allocate a small amount of primary space and a large amount of secondary space. The maximum amount of space that you would need for SYSDISC is the amount of space that the SYSREC data set uses.

SYSERR Data Set

The SYSERR data set contains information about errors that are found in the input data and is for LOADPLUS internal use only. LOADPLUS uses the SYSERR information to produce the discard report and to prevent duplicate key violations.

LOADPLUS writes one 20-byte record to the SYSERR data set for each error detected. If an input record that causes an error is selected for multiple INTO clauses, the error can generate multiple SYSERR records.

To calculate the space that LOADPLUS requires for the SYSERR data set, use one of the following calculations:

- For a SQLAPPLY load, use the following formula:

  ```
  (estimated # errors × 520 bytes) × 2
  ```

- For all other load types, use the following formula:

  ```
  estimated # errors × 20 bytes
  ```

If you expect few or no errors, allocate a small amount of primary space and a large amount of secondary space. You can use the DISCARDS option to limit the number of SYSERR records that LOADPLUS generates.

BMC Software, Inc., Confidential and Proprietary Information

If you expect a specific number of errors, use the formula above for the primary space and provide a small secondary space allocation as a safety measure against out-of-space conditions.

Note: If you underestimate the size of your SYSERR data set and receive an *x*37 abend, which indicates an out-of-space condition, you must perform specific tasks before restarting your load job. For more information about these tasks, see "Inadequate Space Failure on SYSERR Data Set" on page 4-32.

SORTWK Data Set

LOADPLUS uses the SORTWK data sets to sort the types of data described in Table D-2.

Table D-2 Types of Data Sorted Using SORTWK

Type of Data	Corresponding File	Page Reference for Allocation Calculations
row image data	SORTOUT	D-10
index data	SYSUT1	D-11
error records	SYSERR	D-13

If you explicitly specify SORTWK data sets in your JCL, they will need enough space to handle the largest sort task. If you are allocating multiple SORTWK data sets, BMC Software recommends that you allocate 12 data sets.

LOADPLUS determines how multiple SORTWK data sets are used, so allocate the same amount of space for each data set. After you calculate the amount of space needed, allocate all SORTWK data sets with primary space equal to the largest amount needed.

Table D-3 on page D-10 describes the information that you need to include in your SORTWK calculation.

BMC Software, Inc., Confidential and Proprietary Information

Table D-3 SORTWK Data Set Total Allocation Guide

ORDER Option	LOAD Command	Calculate SORTWK Space for
YES	LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY	the greater of space needed to sort data or error information
	all other loads	the greater of space needed to sort data or index information
NO	LOAD RESUME YES SHRLEVEL CHANGE SQLAPPLY	space needed to sort error information only
	all other loads	space needed to sort index information only

If you have clustering indexes and specify ORDER NO, LOADPLUS sorts clustering indexes the same as participating nonclustering indexes.

Sort records that LOADPLUS creates carry overhead bytes. These are represented in the formulas that follow as variables *a* and *n*.

The value of *a* is either 0 (if no EDITPROCs are defined) or the maximum clustering key length (if EDITPROCs are defined).

The value of *n* is 9 bytes for a table space that is LARGE (either by definition or default) or defined with DSSIZE, and 8 bytes for other table spaces.

Sorting Row Image Data

The record that LOADPLUS uses to sort the row image data contains the information that is written to the SORTOUT data set plus a sort key. The sort key contains six additional overhead bytes.

For a SQLAPPLY load, use the following calculation to calculate the amount of space needed for sorting row image data. The number of new rows is the number of additional rows that you expect to be in the table space after the load completes.

```
# new rows × (average row length + clustering key length
+ n)
```

For all other load types, use the following calculation. The number of rows is the total number that you expect to be in the table space or in the participating partitions of the table space after the load completes.

```
size of SORTOUT data sets + (# rows × (6 + a))
```

BMC Software, Inc., Confidential and Proprietary Information

Sorting Index Information

LOADPLUS uses the index work records to sort the index work information. The index work records contain the index key value plus the sort overhead bytes. Use the same index key value length for this data set that you calculated for the SYSUT1 data set. (That is, if you are using a single SYSUT1 data set, include the padding for shorter index keys.)

Use the information in Table D-4 on page D-12 and Table D-5 on page D-12 in conjunction with the formulas that follow to determine the amount of SORTWK data set space that LOADPLUS requires. Use Table D-4 to determine which allocation method to use, then use Table D-5 to find a description of that method.

Use the single SYSUT1 data set columns in Table D-4 if you are determining the SORTWK data set size for a single-phase LOAD REPLACE job. Although this type of load does not use the SYSUT1 data set, LOADPLUS uses the SORTWK data set to sort the indexes.

After referencing Table D-4, use the following formula if you have a single index. The number of index entries is the total number of entries that will exist in the index after the load is complete.

```
# of index entries for the index × (index key length + a
+ n)
```

After referencing Table D-4, use the following formula if you have a group of indexes (such as nonclustering, unique nonclustering, or nonunique indexes) to consider in your calculations. The number of rows for a table is the total number of rows that will exist in the table space or participating partitions after the load is complete.

```
# rows for a given table × (maximum index key length of
all participating indexes in the given group associated
with this table space + a + n) × (# of all participating
indexes in the given group in this table)
```

Add the space needed for each table to arrive at the total amount of space needed.

BMC Software, Inc., Confidential and Proprietary Information

Table D-4 SORTWK Data Set Allocation Guide for Index Information

Index Type	Single SYSUT1 Data Set			Multiple SYSUT1 Data Sets		
	ORDER NO	ORDER YES	ORDER PRE-SORTED	ORDER NO	ORDER YES	ORDER PRE-SORTED
no clustering indexes and no unique nonclustering indexes	method 1	method 1	not applicable	method 2	method 2	not applicable
at least one clustering index and no unique nonclustering indexes	method 1	method 3	method 8	method 2	method 4	method 8
no clustering index and at least one participating unique nonclustering index	method 1	method 1	not applicable	method 5	method 5	not applicable
at least one clustering index and at least one participating unique nonclustering index	method 1	method 7	method 8	method 5	method 6	method 9

Table D-5 Methods for Allocating SORTWK Data Sets

Method	Description
1	Allocate enough space to sort all participating nonclustering indexes. Remember that LOADPLUS treats the clustering index as a participating nonclustering index if you specify ORDER NO.
2	*Minimum:* Allocate enough space to sort the largest index. *Maximum (best performance):* Allocate enough space to sort all participating nonclustering indexes. Remember that LOADPLUS treats the clustering index as a participating nonclustering index if you specify ORDER NO. **Note:** If you cannot allocate the maximum, you can improve performance by allocating as much as possible.
3	Allocate either enough space to sort the row images or enough space to sort all participating nonclustering indexes at once—whichever is greater.
4	Allocate either enough space to sort the row images or the value determined from method 2—whichever is greater.
5	Allocate either enough space to sort all participating unique nonclustering indexes or the value determined from method 2—whichever is greater.
6	Allocate enough space to sort the row images *plus* enough space to sort all participating unique nonclustering indexes at once. Alternatively, you can allocate enough space to sort all participating nonunique indexes.
7	Allocate enough space to sort all row images *plus* all participating nonclustering indexes.
8	Allocate enough space to sort all participating nonclustering indexes.
9	Allocate either enough space to sort all participating nonclustering, unique indexes at once or enough space to sort all participating nonclustering nonunique indexes—whichever is greater.

BMC Software, Inc., Confidential and Proprietary Information

Sorting Error Records

All types of loads sort error records. For most load types, LOADPLUS also uses the SORTWK data set for sorting data and index information. As a result, the allocation is sufficient to include error information.

However, when you specify or default to ORDER NO with a SQLAPPLY load, LOADPLUS sorts *only* error information. Therefore, you can allocate a much smaller SORTWK data set than for other types of loads.

Calculate the space that LOADPLUS needs for sorting error information for a SQLAPPLY load as follows:

```
(# errors expected × 20) × 2
```

Copy Data Sets

The copy data sets (BMCCPY*nn*, BMCCPZ*nn*, BMCRCY*nn*, and BMCRCZ*nn*) contain image copies of the loaded table space. The DB2 RECOVER utility uses these data sets when table space recovery is required.

In each case, the data set size depends on the number of pages required when the table space is loaded. For a two-phase load, LOADPLUS calculates this value during the PRELOAD phase and the value is included in message BMC51486I. Allocate the primary amount as the total amount calculated. No secondary amount is needed.

Note: You cannot change the amount allocated for a single-phase load. LOADPLUS issues message BMC50512 at the end of the COMBINED phase, indicating the number of pages it used for the data sets.

The space that LOADPLUS requires for these data sets depends on the following criteria:

- whether the table space is partitioned or nonpartitioned
- if partitioned, whether all partitions are copied to one data set, or each partition is copied to a separate data set

BMC Software, Inc., Confidential and Proprietary Information

Nonpartitioned Table Space

If the table space is not partitioned, all pages are copied to a single copy data set. Calculate the space that LOADPLUS requires for this data set by multiplying the number of pages required for loading by the table space page size.

Partitioned Table Space, Single Copy Data Set

If the table space is partitioned and all partitions are copied to a single data set, the space that LOADPLUS requires is the total number of pages for all loaded partitions times the page size of the table space.

Partitioned Table Space, Multiple Copy Data Sets

If the table space is partitioned and each partition is copied to a separate data set, the space that LOADPLUS requires for each data set is the number of pages loaded for that partition times the page size of the table space. The copy data sets are allocated with the partition number appended to the prefix (BMCCPY, BMCCPZ, BMCRCY, and BMCRCZ).

Examples of Space Calculations

The following examples illustrate ways to calculate the total space needed by using various combinations of the SORTOUT, SYSUT1, and SORTWK data sets.

Example 1

In this example, the space calculations are determined for a table with the following characteristics:

- 10-byte, unique clustering index
- 15-byte, nonunique nonclustering index
- 5-byte, nonunique nonclustering index
- 10 million rows (100-byte, fixed-length)
- ORDER NO and LOAD REPLACE options specified
- in a table space that is not LARGE or defined with DSSIZE

BMC Software, Inc., Confidential and Proprietary Information

The example uses DASD calculations for 3390 devices, allocates 12 SORTWK data sets, sets the secondary allocation to 25 percent of the primary, and rounds up all numbers in the formulas to whole numbers.

SORTOUT*n* Data Set

Calculate the total space needed for your SORTOUT data sets as follows

```
    # DASD bytes required = # new rows selected for this table
                            × (average row length + n)
                          = 10,000,000 × (100 + 8)
                          = 10,000,000 × 108
                          = 1,080,000,000
   # 3390 tracks required = 1,080,000,000/56,664
                          = 19,060
# 3390 cylinders required = 19,060/15
                          = 1271
```

Use the following DD statement for a single SORTOUT data set:

```
//SORTOUT DD UNIT=3390,SPACE=(CYL,(1271,318)),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA
```

Use the following DD statements for two SORTOUT data sets:

```
//SORTOUT1 DD UNIT=3390,SPACE=(CYL,(636,159)),
// DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA01
//SORTOUT2 DD UNIT=3390,SPACE=(CYL,(636,159)),
// DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA02
```

Multiple SYSUT1*nn* Data Sets

When using multiple SYSUT1*nn* data sets, calculate the space needed for each data set as follows:

```
    # DASD bytes required = # rows expected to be in the associated
                            table after the load × (length of index
                            key + n)
```

BMC Software, Inc., Confidential and Proprietary Information

For the 10-byte clustering index:

```
    # DASD bytes required = 10,000,000 × (10 + 6)
                          = 10,000,000 × (16)
                          = 160,000,000
   # 3390 tracks required = 160,000,000/56,664
                          = 2824
# 3390 cylinders required = 2824/15
                          = 189
```

For the 15-byte nonclustering index:

```
    # DASD bytes required = 10,000,000 × (15 + 6)
                          = 10,000,000 × (21)
                          = 210,000,000
   # 3390 tracks required = 210,000,000/56,664
                          = 3707
# 3390 cylinders required = 3707/15
                          = 248
```

For the 5-byte nonclustering index:

```
    # DASD bytes required = 10,000,000 × (5 + 6)
                          = 10,000,000 × (11)
                          = 110,000,000
   # 3390 tracks required = 110,000,000/56,664
                          = 1942
# 3390 cylinders required = 1942/15
                          = 130
```

Use the following DD statements for the SYSUT1*nn* data sets:

```
//SYSUT101 DD UNIT=3390,SPACE=(CYL,(248,60),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA01
//SYSUT102 DD UNIT=3390,SPACE=(CYL,(189,50),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA02
//SYSUT103 DD UNIT=3390,SPACE=(CYL,(130,35),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA03
```

The 15-byte index is assigned to the first SYSUT1*nn* DD statement in the JCL because it is the index with the largest key length. The 10-byte index is assigned next, followed by the 5-byte index. A secondary quantity of 25 percent of the primary allocation is provided as a safety measure against out-of-space conditions. BMC Software recommends specifying RLSE to release unused DASD space when the data set is closed.

BMC Software, Inc., Confidential and Proprietary Information

Single SYSUT1 Data Set

When using a single SYSUT1 data set, calculate the space needed as follows:

```
    # DASD bytes required = total # rows expected to be in
                            this table after the load ×
                            (length of longest participating nonclustering
                            index key on any table involved in the
                            load + n) × # of participating nonclustering,
                            nonpartitioned indexes on this table
                          = 10,000,000 × (15 + 6) × 3
                          = 10,000,000 × 21 × 3
                          = 630,000,000
   # 3390 tracks required = 630,000,000/56,664
                          = 11,119
# 3390 cylinders required = 11,119/15
                          = 742
```

As stated earlier, when you specify ORDER NO, LOADPLUS treats clustering indexes like participating nonclustering indexes. Therefore, the value for the number of nonclustering indexes in the preceding calculation is three instead of two.

Use the following DD statement for the SYSUT1 data set:

```
//SYSUT1 DD UNIT=3390,SPACE=(CYL,(742,185),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA01
```

The single SYSUT1 data set requires more DASD than multiple SYSUT1*nn* data sets because of the required padding of all keys to the length of the longest key.

SORTWK Data Set (Using Multiple SYSUT1*nn* Data Sets)

When using multiple SYSUT1*nn* data sets, you can calculate the minimum or maximum space needed for each SORTWK data set (see Table D-4 on page D-12.)

BMC Software, Inc., Confidential and Proprietary Information

The minimum space required is enough space to sort the largest index, calculated as follows:

```
    # DASD bytes required = # of index entries for the index
                            × (index key length + a + n)
                          = 10,000,000 × (15 + 0 + 14)
                          = 10,000,000 × (29)
                          = 290,000,000
   # 3390 tracks required = 290,000,000/56,664
                          = 5118
# 3390 cylinders required = 5118/15
                          = 342
 # cylinders required for = # 3390 cylinders required/
 each SORTWK allocation     # SORTWKs
                          = 342/12
                          = 29
```

Use the following DD statements for the SORTWK data sets:

```
//SORTWK01 DD DISP=(NEW,DELETE),SPACE=(CYL,(29,7)),...
//SORTWK02 DD DISP=(NEW,DELETE),SPACE=(CYL,(29,7)),...
  . . . . .
//SORTWK12 DD DISP=(NEW,DELETE),SPACE=(CYL,(29,7)),...
```

For optimum performance, use the maximum space needed, which is enough space to sort all of the participating nonclustering indexes. The maximum space is calculated as follows:

```
    # DASD bytes required = # rows for a given table ×
                            (maximum index key length of all participating
                            nonclustering indexes associated
                            with this table space + a + n)
                            × # of all participating nonclustering indexes
                          = 10,000,000 × (15 + 0 + 14) × 3
                          = 10,000,000 × (29) × 3
                          = 870,000,000
   # 3390 tracks required = 870,000,000/56,664
                          = 15,354
# 3390 cylinders required = 15,354/15
                          = 1024
 # cylinders required for = # 3390 cylinders required/
 each SORTWK allocation     # of SORTWKs
                          = 1024/12
                          = 86
```

BMC Software, Inc., Confidential and Proprietary Information

Use the following DD statements for the SORTWK data sets:

```
//SORTWK01 DD DISP=(NEW,DELETE),SPACE=(CYL,(86,22)),...
//SORTWK02 DD DISP=(NEW,DELETE),SPACE=(CYL,(86,22)),...
  . . . . .
//SORTWK12 DD DISP=(NEW,DELETE),SPACE=(CYL,(86,22)),...
```

SORTWK Data Set (Using a Single SYSUT1 Data Set)

When using a single SYSUT1 data set, calculate the space needed for each SORTWK data set the same as you would for the maximum space required when using multiple SYSUT1*nn* data sets (see Table D-4 on page D-12.) This calculation ensures sufficient space to sort all participating nonclustering indexes.

```
    # DASD bytes required = # rows for a given table ×
                            (maximum index key length of all participating
                            nonclustering indexes associated
                            with this table space + a + n)
                            × # of participating nonclustering indexes
                          = 10,000,000 × (15 + 0 + 14) × 3
                          = 10,000,000 × (29) × 3
                          = 870,000,000
   # 3390 tracks required = 870,000,000/56,664
                          = 15,354
# 3390 cylinders required = 15,354/15
                          = 1024
 # cylinders required for = # 3390 cylinders required/
 each SORTWK allocation     # of SORTWKs
                          = 1024/12
                          = 86
```

Use the following DD statements for the SORTWK data sets:

```
//SORTWK01 DD DISP=(NEW,DELETE),SPACE=(CYL,(86,22)),...
//SORTWK02 DD DISP=(NEW,DELETE),SPACE=(CYL,(86,22)),...
  . . . . .
//SORTWK12 DD DISP=(NEW,DELETE),SPACE=(CYL,(86,22)),...
```

BMC Software, Inc., Confidential and Proprietary Information

Example 2

In this example, the space calculations are determined for a table with the following characteristics:

- 8-byte, unique clustering index
- 13-byte, nonunique nonclustering index
- 9-byte, unique nonclustering index
- 3 million rows (80-byte, fixed-length)
- ORDER YES and LOAD REPLACE options specified
- in a table space that is not LARGE or defined with DSSIZE

The example uses DASD calculations for 3390 devices, allocates 12 SORTWK data sets, sets the secondary allocation to 25 percent of the primary, and rounds up all numbers in the formulas to whole numbers.

SORTOUT*n* Data Set

Calculate the total space needed for your SORTOUT data sets as follows:

```
  # DASD bytes required = (# new rows selected for this table
                          × (average row length + n)) +
                          (# new rows selected for this table
                          × length of longest clustering
                          index key)
                        = (3,000,000 × (80 + 8))+(3,000,000 × 8)
                        = 264,000,000 + 24,000,000
                        = 288,000,000
 # 3390 tracks required = 288,000,000/56,664
                        = 5083
3390 cylinders required = 5083/15
                        = 339
```

Use the following DD statement for a single SORTOUT data set:

```
//SORTOUT DD UNIT=3390,SPACE=(CYL,(339,85)),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA
```

Use the following DD statements for two SORTOUT data sets:

```
//SORTOUT1 DD UNIT=3390,SPACE=(CYL,(170,43)),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA01
//SORTOUT2 DD UNIT=3390,SPACE=(CYL,(170,43)),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA02
```

BMC Software, Inc., Confidential and Proprietary Information

Multiple SYSUT1*nn* Data Sets

When using multiple SYSUT1*nn* data sets with ORDER YES specified, you need to calculate only the space needed for each participating nonclustering index because the clustering index is sorted with the row image data in the SORTOUT data set.

```
    # DASD bytes required = total # rows expected to be in
                            associated table after the load
                            × (length of index key + n)
```

For the 13-byte nonclustering index:

```
    # DASD bytes required = 3,000,000 × (13 + 6)
                          = 3,000,000 × (19)
                          = 57,000,000
   # 3390 tracks required = 57,000,000/56,664
                          = 1006
# 3390 cylinders required = 1006/15
                          = 68
```

For the 9-byte nonclustering index:

```
    # DASD bytes required = 3,000,000 × (9 + 6)
                          = 3,000,000 × (15)
                          = 45,000,000
   # 3390 tracks required = 45,000,000/56,664
                          = 795
# 3390 cylinders required = 795/15
                          = 53
```

Use the following DD statements for the SYSUT1*nn* data sets:

```
//SYSUT101 DD UNIT=3390,SPACE=(CYL,(68,17),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA01
//SYSUT102 DD UNIT=3390,SPACE=(CYL,(53,14),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA02
```

The 13-byte index is assigned to the first SYSUT1*nn* DD statement in the JCL because it is the index with the largest key length. A secondary quantity that is 25 percent of the primary allocation is provided as a safety measure against out-of-space conditions. BMC Software recommends specifying RLSE to release unused DASD space when the data set is closed.

BMC Software, Inc., Confidential and Proprietary Information

Single SYSUT1 Data Set

When using a single SYSUT1 data set, calculate the space needed as follows:

```
    # DASD bytes required = total # rows expected to be in
                            this table after the load ×
                            (length of longest participating nonclustering
                            index key on any table involved in the
                            load + n) × # of participating nonclustering
                            nonpartitioned indexes on this
                            table
                          = 3,000,000 × (13 + 6) × 2
                          = 3,000,000 × 19 × 2
                          = 114,000,000
   # 3390 tracks required = 114,000,000/56,664
                          = 2012
# 3390 cylinders required = 2012/15
                          = 135
```

Use the following DD statements for the SYSUT1 data set:

```
//SYSUT1 DD UNIT=3390,SPACE=(CYL,(135,35),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA01
```

The single SYSUT1 data set requires more DASD than multiple SYSUT1*nn* data sets because of the required padding of all keys to the length of the longest key.

SORTWK Data Set (Using Multiple SYSUT1*nn* Data Sets)

When using multiple SYSUT1*nn* data sets, you can calculate the total of the space required to sort row images plus space to sort all participating nonclustering, unique indexes; or, the space required to sort all participating nonunique indexes—whichever is greater (see Table D-4 on page D-12.)

Calculate the space to sort row images as follows:

```
    # DASD bytes required = size of SORTOUT data set +
                            (total # of rows expected to be in
                            the table space after the load
                            × (6 + a))
                          = 288,000,000 + (3,000,000 × (6 + 0))
                          = 288,000,000 + 18,000,000
                          = 306,000,000
   # 3390 tracks required = 306,000,000/56,664
                          = 5401
# 3390 cylinders required = 5401/15
                          = 361
```

BMC Software, Inc., Confidential and Proprietary Information

Then add to that the space to sort all participating nonclustering, unique indexes:

```
    # DASD bytes required = # rows in table after load is
                            complete × (maximum key length of
                            all participating nonclustering unique indexes
                            + a + n) × # of participating nonclustering
                            unique indexes
                          = 3,000,000 × (9 + 0 + 14) × 1
                          = 3,000,000 × (23 × 1)
                          = 69,000,000
   # 3390 tracks required = 69,000,000/56,664
                          = 1218
# 3390 cylinders required = 1218/15
                          = 82

Total # 3390 cylinders    = space to sort row images + space to sort all
                            participating unique nonclustering indexes
                          = 361 + 82
                          = 443
```

Alternatively, you can calculate the space needed to sort all participating nonunique indexes as follows:

```
    # DASD bytes required = # rows in table after load is
                            complete × (maximum key length of all
                            participating nonunique nonclustering indexes
                            + a + n) × # of participating nonunique
                            nonclustering indexes
                          = 3,000,000 × (13 + 0 + 14)× 1
                          = 3,000,000 × (27) × 1
                          = 81,000,000
   # 3390 tracks required = 81,000,000/56,664
                          = 1430
# 3390 cylinders required = 1430/15
                          = 96
```

Based on the preceding calculations, you would use the space required to sort the row image data plus all participating nonclustering unique indexes because it is greater than the space required to sort all participating nonunique indexes.

```
 # cylinders required for = # 3390 cylinders required/# of
 each SORTWK allocation   SORTWKs
                          = 443/12
                          = 37
```

BMC Software, Inc., Confidential and Proprietary Information

Use the following DD statements for the SORTWK data sets:

```
//SORTWK01 DD DISP=(NEW,DELETE),SPACE=(CYL,(37,10)),...
//SORTWK02 DD DISP=(NEW,DELETE),SPACE=(CYL,(37,10)),...
  . . . . .
//SORTWK12 DD DISP=(NEW,DELETE),SPACE=(CYL,(37,10)),...
```

SORTWK Data Set (Using a Single SYSUT1 Data Set)

When using a single SYSUT1 data set, calculate the space needed to sort all row images plus the space needed to sort all participating nonclustering indexes (see Table D-4 on page D-12.)

Calculate the space to sort row images as follows:

```
    # DASD bytes required = size of SORTOUT data set +
                            (total # of rows expected to be in
                            the table space after the load
                            × (6 + a))
                          = 288,000,000 + (3,000,000 × (6 + 0))
                          = 288,000,000 + 18,000,000
                          = 306,000,000
   # 3390 tracks required = 306,000,000/56,664
                          = 5401
# 3390 cylinders required = 5401/15
                          = 361
```

BMC Software, Inc., Confidential and Proprietary Information

Then add to that the space to sort all participating nonclustering indexes:

```
    # DASD bytes required = # rows in table after load is
                            complete × (maximum key length of all
                            participating nonclustering indexes + a + n)
                            × # of participating nonclustering indexes
                          = 3,000,000 × (13 + 0 + 14) × 2
                          = 3,000,000 × 27 × 2
                          = 162,000,000
   # 3390 tracks required = 162,000,000/56,664
                          = 2859
# 3390 cylinders required = 2859/15
                          = 191

                    Total = space to sort row images + space to
                            sort all participating nonclustering indexes
                          = 361 + 191
                          = 552

 # cylinders required for = # 3390 cylinders required/# of
 each SORTWK allocation     SORTWKs
                          = 552/12
                          = 46
```

Use the following DD statements for the SORTWK data sets:

```
//SORTWK01 DD DISP=(NEW,DELETE),SPACE=(CYL,(46,12)),...
//SORTWK02 DD DISP=(NEW,DELETE),SPACE=(CYL,(46,12)),...
  . . . . .
//SORTWK12 DD DISP=(NEW,DELETE),SPACE=(CYL,(46,12)),...
```

Example 3

In this example, the space calculations are determined for a table with the following characteristics:

- 20-byte, unique clustering index
- 10-byte, nonunique nonclustering index
- 2 million rows (78-byte, fixed-length)
- Adding an additional 500,000 rows
- ORDER YES and LOAD RESUME YES options specified
- in a table space that is not LARGE or defined with DSSIZE

The example uses DASD calculations for 3390 devices, allocates 12 SORTWK data sets, sets the secondary allocation to 25 percent of the primary, and rounds up all numbers in the formulas to whole numbers.

BMC Software, Inc., Confidential and Proprietary Information

SORTOUT*n* Data Set

Calculate the total space needed for your SORTOUT data sets as follows:

```
    # DASD bytes required = (# new rows selected for this table
                            × (row length + n)) + (# new rows
                            selected for the table × length of
                            longest clustering index key )
                            + (# keys in existing clustering
                            index × (length of longest
                            clustering index key + n))
                          = (500,000 × (78 + 8)) +
                            (500,000 × 20) +
                            (2,000,000 × (20 + 8))
                          = 43,000,000 + 10,000,000 +
                            56,000,000
                          = 109,000,000
   # 3390 tracks required = 109,000,000/56,664
                          = 1924
# 3390 cylinders required = 1924/15
                          = 129
```

Use the following DD statement for a single SORTOUT data set:

```
//SORTOUT DD UNIT=3390,SPACE=(CYL,(129,32)),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA
```

Use the following DD statements for two SORTOUT data sets:

```
//SORTOUT1 DD UNIT=3390,SPACE=(CYL,(65,16)),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA01
//SORTOUT2 DD UNIT=3390,SPACE=(CYL,(65,16)),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.SORTOUT.DATA02
```

BMC Software, Inc., Confidential and Proprietary Information

Single SYSUT1 Data Set

When using a single SYSUT1 data set, calculate the space needed as follows:

```
    # DASD bytes required = total # rows expected to be in the
                            table after the load × (length of
                            longest participating nonclustering index key
                            in any table involved in the load
                            + n) × # participating nonclustering indexes
                          = 2,500,000 × (10 + 6) × 1
                          = 2,500,000 × 16 × 1
                          = 40,000,000
   # 3390 tracks required = 40,000,000/56,664
                          = 706
# 3390 cylinders required = 706/15
                          = 48
```

Use the following DD statement for the SYSUT1 data set:

```
//SYSUT1 DD UNIT=3390,SPACE=(CYL,(48,12),RLSE),
//  DISP=(MOD,CATLG,CATLG),DSN=YOUR.LOADPLUS.INDEX.DATA01
```

Because this example has only one nonclustering index, it is not meaningful to have multiple SYSUT1 data sets.

SORTWK Data Set (Using a Single SYSUT1 Data Set)

Calculate the space needed to sort all row images or the space needed to sort all participating nonclustering indexes at once—whichever is greater (see Table D-4 on page D-12.)

Calculate the space to sort all row images as follows:

```
    # DASD bytes required = size of SORTOUT data set +
                            (total # of rows expected to be in
                            the table space after the load
                            × (6 + a))
                          = 109,000,000 + (2,500,000 × (6 + 0))
                          = 109,000,000 + 15,000,000
                          = 124,000,000
   # 3390 tracks required = 124,000,000/56,664
                          = 2189
# 3390 cylinders required = 2189/15
                          = 146
```

BMC Software, Inc., Confidential and Proprietary Information

Alternatively, use the following formula to calculate the space needed to sort all the participating nonclustering indexes at once:

```
    # DASD bytes required = # rows expected to be in the table
                            space after the load × (longest participating
                            nonclustering index key + a + n) ×
                            # of participating nonclustering indexes
                          = 2,500,000 × (10 + 0 + 14) × 1
                          = 2,500,000 × 24 × 1
                          = 60,000,000
   # 3390 tracks required = 60,000,000/56,664
                          = 1059
# 3390 cylinders required = 1059/15
                          = 71

 # cylinders required for = # 3390 cylinders required/# SORTWKs
 each SORTWK allocation   = 146/12
                          = 13
```

Based on the preceding calculations, you would use the results of the first calculation because the amount of space is greater than the results of the second calculation.

Use the following DD statements for the SORTWK data sets:

```
//SORTWK01 DD DISP=(NEW,DELETE),SPACE=(CYL,(13,4)),...
//SORTWK02 DD DISP=(NEW,DELETE),SPACE=(CYL,(13,4)),...
  . . . . .
//SORTWK12 DD DISP=(NEW,DELETE),SPACE=(CYL,(13,4)),...
```

BMC Software, Inc., Confidential and Proprietary Information

Appendix E RULES Installation Option Examples

This appendix presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Overview

The RULES installation option is set during installation and you cannot override it at runtime. You can specify either RULES=STANDARD or the more restrictive RULES=BMC.

Note: If you specify FORMAT UNLOAD, FORMAT BMC, or FORMAT BMCUNLOAD, the LOADPLUS product changes the value of this option to BMC, regardless of the value specified during installation. If you specify FORMAT CSV, LOADPLUS changes the value of this option to STANDARD, regardless of the value specified during installation.

The information in this appendix uses a single sample table to illustrate the RULES option. The following statement creates this sample table.

```
CREATE TABLE SAMPLE_TABLE
       (CHAR_COL    CHAR(1)      NOT NULL WITH DEFAULT,
        DEC_COL     DECIMAL(5,0),
        DATE_COL    DATE)
IN DATABASE.TABLESPACE
```

Error Conditions with RULES=BMC

With RULES=BMC (the more restrictive option), the following control statements produce errors:

```
LOAD DATA REPLACE INTO SAMPLE_TABLE
WHEN DEC_COL= ' '

BMC50122E FOR 'WHEN' COLUMN 'DEC_COL', CONSTANT IS
          WRONG TYPE OR COLUMN IS NOT NULLABLE:
```

```
LOAD DATA REPLACE INTO TABLE SAMPLE_TABLE
(CHAR_COL POS(1:1) CHAR(1),
 DEC_COL  POS(2:4) DEC(5,0),
 DATE_COL POS(5:8) DATE NULLIF DATE_COL=' ')

BMC50123E FOR 'NULLIF/DEFAULTIF' COLUMN 'DATE_COL',
          CONSTANT IS NOT IN RANGE OF COLUMN DATA
          TYPE: ' '
```

BMC Software, Inc., Confidential and Proprietary Information

Using the RULES=STANDARD installation option prevents these errors. You can also code the statements with starting and ending (*start:end*) column positions as follows:

```
LOAD DATA REPLACE INTO TABLE SAMPLE_TABLE
(CHAR_COL POS(1:1) CHAR(1),
 DEC_COL  POS(2:4) DEC(5,0),
 DATE_COL POS(5:8) DATE NULLIF (5:8)=' ')
```

```
LOAD DATA REPLACE INTO TABLE SAMPLE_TABLE
WHEN CHAR_COL='X'
(CHAR_COL POS(1:1) CHAR(1) DEFAULTIF CHAR_COL='X',
 DEC_COL  POS(2:4) DEC(5,0),
 DATE_COL POS(5:8) DATE NULLIF (5:8)=' ')
```

WHEN, NULLIF, and DEFAULTIF Processing Order

RULES=BMC and RULES=STANDARD affect the order in which LOADPLUS performs WHEN, NULLIF, and DEFAULTIF processing.

```
 LOAD DATA REPLACE INTO TABLE SAMPLE_TABLE
 WHEN CHAR_COL = 'X'
 (CHAR_COL POS(1:1) DEFAULTIF CHAR_COL='X'
  DEC_COL  POS(2:4) DEC(5,0),
  DATE_COL POS(5:8) DATE NULLIF DATE_COL=' ')
```

With RULES=BMC in effect, LOADPLUS performs the NULLIF or DEFAULTIF processing first to determine whether to set CHAR_COL to a blank because it is defined as NOT NULL WITH DEFAULT. Next, LOADPLUS performs the WHEN comparison to determine whether to load the record into the table. For any record with an 'X' in column 1, LOADPLUS changes the 'X' to a blank and discards the record.

With RULES=STANDARD in effect, the opposite process occurs. LOADPLUS first performs the WHEN comparison to determine whether to load the record into the table, and, if so, continues with the NULLIF or DEFAULTIF data conversion. For any record with an 'X' in column 1, LOADPLUS changes the 'X' to a blank and loads the record into the table.

BMC Software, Inc., Confidential and Proprietary Information

Comparison Operators Allowed

RULES=STANDARD has more restrictions on comparison operators than RULES=BMC. Each option permits different comparison operators for use in WHEN, NULLIF, and DEFAULTIF predicates, as shown in the following table:

Operators	Description	RULES= STANDARD	RULES= BMC
<	less than		X
<=	less than or equal		X
=	equal	X	X
<>	not equal	X	X
¬ =	not equal	X	X
> =	greater than or equal		X
>	greater than		X
IN	equal to any	X	X
NOT IN	not equal to all	X	X

For more information, see “WHEN” on page 3-93.

The following statement creates an error condition when RULES=STANDARD is the installation option, because you cannot use the *less than* operator (<).

```
LOAD DATA REPLACE INTO TABLE SAMPLE_TABLE
(CHAR_COL POS(1:1) CHAR(1)
 DEC_COL  POS(2:4) DEC(5,0)NULLIF DEC_COL<'5',
 DATE_COL POS(5:8) DATE)
```

```
BMC51415E FOR 'NULLIF/DEFAULTIF' FIELD 'DEC_COL',
          ONLY  '=', '¬ =','<>', 'IN', AND 'NOT IN'
          COMPARISONS ARE ALLOWED.
```

BMC Software, Inc., Confidential and Proprietary Information

WHEN, NULLIF, or DEFAULTIF Predicates

RULES=STANDARD requires that all constants used in WHEN, NULLIF, or DEFAULTIF predicates be character strings or hex strings enclosed in quotes. The following statement produces an error because it does not meet that requirement:

```
 LOAD DATA REPLACE INTO TABLE SAMPLE_TABLE
(CHAR_COL POS(1:1) CHAR(1)
 DEC_COL  POS(2:4) DEC(5,0)NULLIF DEC_COL='1',
 DATE_COL POS(5:8) DATE)
```

```
BMC51416E FOR 'NULLIF/DEFAULTIF' FIELD 'DEC_COL',
          CONSTANT MUST BE A CHARACTER STRING OR A HEX
          STRING
```

RULES=BMC does not have this requirement.

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Appendix F LOADPLUS User Exits

This appendix presents the following topics:

BMC Software, Inc., Confidential and Proprietary Information

Overview

The LOADPLUS product allows you to further customize your load processing through user-written exit routines. These user exits provide LOADPLUS with user-defined variables that you can use to construct data set name patterns with the DSNPAT installation option or command option. See page 3-145 for details about the DSNPAT option.

LOADPLUS supports exit routines that are written in assembler, COBOL II, IBM's Language Environment COBOL (LE COBOL), C, and IBM's Language Environment C (LE C).

You must have dynamic work file allocation active to direct LOADPLUS to invoke your exit routine. In addition, you can either:

- supply the module name and program language in the DSNUEXIT installation option or

- supply the module name and program language in the DSNUEXIT option of the LOAD command.

Using the DSNUEXIT option of the LOAD command overrides any value in the installation options. See page 3-135 for details about the DSNUEXIT option of the LOAD command. See Appendix A, "LOADPLUS Installation Options," for additional information about the installation options.

Depending on the type of load that you are running, LOADPLUS invokes your user-written exit routine at the beginning of either the PRELOAD phase or the COMBINED phase.

Sample User Exits

Source code for the sample LOADPLUS user exits is distributed in the LOADPLUS *HLQ*.CNTL library (where *HLQ* is the high-level qualifier set during installation). The following list shows the name of each sample exit and its associated programming language:

- AMUEDSNA—assembler
- AMUEDSN2—COBOL II and LE COBOL
- AMUEDSC—C
- AMUEDSL—LE C

BMC Software, Inc., Confidential and Proprietary Information

Copy the appropriate member as the base code for your exit routine, edit it, then compile, or assemble and link your resulting exit into an authorized library so that it can be loaded during execution.

If you already have user exits for REORG PLUS for DB2 and are not using the XPRTYPE field, you can use those same exits for LOADPLUS by creating an alias of AMUDSNXP for ARUDSNXP.

The library in which your exit routine resides must be included in your system's LINKLIST or in your JOBLIB or STEPLIB. For COBOL II and C programs, the appropriate runtime libraries must be authorized and in your LINKLIST, JOBLIB, or STEPLIB. For LE COBOL and LE C programs, the appropriate language environment runtime libraries must be authorized and in your LINKLIST, JOBLIB, or STEPLIB.

LOADPLUS communicates with your exit by passing parameters *to* the exit and receiving a return code and user-defined variables *from* the exit. The following list describes the location of the return code for each type of exit:

- For an assembler exit, the return code is the contents of register 15.
- For a COBOL exit, the return code is set in the RETURN-CODE variable.
- For a C exit, the return code is the value that is returned by the return function.

Valid return codes and their results are:

- 0—indicates a good return

  Processing continues.

- other—indicates an error

  The utility terminates.

BMC Software, Inc., Confidential and Proprietary Information

User-Defined Variables

These user exits return user-defined variables to LOADPLUS. You can use these variables with the DSNPAT installation or command option to construct data set name patterns. LOADPLUS uses these patterns and the values that are returned for any user-defined variables to create data set names during dynamic work file allocation. The following rules apply when using a user-defined exit routine to create your variables:

- Variable names must comply with the following rules:
  - must begin with an underscore character
  - must begin in the leftmost byte of the XPUVNAME field
  - must be padded on the right with blanks if the variable name is shorter than 9 bytes
  - must not contain embedded blanks
  - non-blank characters after the underscore must consist of the following national characters:
    - A through Z,
    - 0 through 9,
    - #, @, $
- Variable data must comply with the following rules:
  - must begin in the leftmost byte of the XPUVDATA field
  - must be padded on the right with blanks if the variable data is shorter than 8 bytes
  - must not contain embedded blanks
  - non-blank characters must consist of the following characters:
    - A through Z,
    - 0 through 9,
    - #, @, $, –, .

BMC Software, Inc., Confidential and Proprietary Information

LOADPLUS Assembler User Exit

This section provides a sample exit parameter block, variable mapping block, and exit that you can use to write a user exit in assembler.

Exit Parameter Block DSECT

This DSECT (Figure F-1) contains both input and output fields. Input fields pass vital information to the user exit, such as the database name and user ID. If you make any modifications to these input fields, LOADPLUS disregards them on return. The output fields pass information about your user variables back to LOADPLUS.

Figure F-1 Assembler Exit Parameter Block

```
*-------------------------------------------------------------------*
*                                                                   *
* AMUDSNXP DEFINES THE USER VARIABLE USER EXIT PARM BLOCK.          *
*                                                                   *
* NOTE:                                                             *
*                                                                   *
*   YOU MAY NOT MODIFY THE FIELDS IN FRONT OF THE USER AREA.        *
*                                                                   *
*   YOU MAY NOT MODIFY THE 'STRUCTURE' OF THIS DSECT.               *
*                                                                   *
*-------------------------------------------------------------------*
AMUDSNXP DSECT ,                  PARMS PASSED TO EXIT
*
*  INPUT AREA
*
XPJOBN   DS    CL8                JOBNAME
XPSTEP   DS    CL8                STEPNAME
XPDBNAME DS    CL8                DATABASE NAME
XPSPNAME DS    CL8                SPACE NAME
XPRESUME DS    CL1                RESUME (Y/N)
XPREPLAC DS    CL1                REPLACE (Y/N)
         DS    CL2                RESERVED FOR LOADPLUS
XPUSER   DS    CL8                USER ID
XPSSID   DS    CL4                DB2 SUBSYSTEM ID
XPDATE   DS    CL6                UTILITY EXECUTION DATE MMDDYY
XPTIME   DS    CL6                UTILITY EXECUTION TIME HHMMSS
XPUTILID DS    CL16               UTILITY ID
XPDATE8  DS    CL8                UTILITY EXECUTION DATE MMDDYYYY
XPGRPNM  DS    CL4                DATA SHARING GROUP NAME
XPVCAT   DS    CL8                VCAT NAME (FROM 1ST PART IF PARTITND)
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
XPDATEJ  DS    CL7                UTILITY EXECUTION DATE CCYYDDD
         DS    CL13               RESERVED FOR LOADPLUS
*
*  USER WORK AREA
*
XPUSRWD1 DS    F                  USER WORD 1
XPUSRWD2 DS    F                  USER WORD 2
XPUSRWD3 DS    F                  USER WORD 3
XPUSRWD4 DS    F                  USER WORD 4
XPUSRWD5 DS    F                  USER WORD 5
XPUSRWD6 DS    F                  USER WORD 6
XPUSRWD7 DS    F                  USER WORD 7
XPUSRWD8 DS    F                  USER WORD 8
*
*  OUTPUT AREA
*
XPUVAREA DS    100CL17            USER VARIABLE AREA
XPUVENT# EQU   100                NUMBER OF ENTRIES IN OUTPUT AREA
XP$      EQU   *-AMUDSNXP
```

DSECT Fields

The major DSECT fields and their uses are as follows:

XPJOBN contains the job name, up to 8 bytes

XPSTEP contains the step name, up to 8 bytes

XPDBNAME contains the database name, up to 8 bytes

XPSPNAME contains the name of the table space or index space from the LOAD command, up to 8 bytes

XPRESUME contains an indication of whether a LOAD RESUME is being performed—Y (yes) or N (no)

XPREPLAC contains an indication of whether a LOAD REPLACE is being performed—Y (yes) or N (no)

XPUSER contains the user ID of the user running the LOADPLUS utility, up to 8 bytes

XPSSID contains the DB2 subsystem ID, 4 bytes

XPDATE contains the date of the execution of the utility, in the format MMDDYY, 6 bytes

BMC Software, Inc., Confidential and Proprietary Information

XPTIME contains the time of the execution of the utility, in the format HHMMSS, 6 bytes

XPUTILID contains the utility ID, up to 16 bytes

XPDATE8 contains the date of the execution of the utility, in the format MMDDYYYY, 8 bytes

XPGRPNM contains the DB2 data-sharing group name.

In a nondata-sharing environment, the field contains the DB2 subsystem ID.

XPVCAT contains the VCATNAME specified in the DB2 catalog for the table space being loaded, or for the first partition if the table space is partitioned

XPDATEJ contains the Julian date of the execution of the utility, in the format CCYYDDD, 7 bytes

XPUSRWD1...XPUSRWD8
provides work space, 8 parameters, up to 4 bytes each

XPUVAREA defines the area that contains user-defined variable information

See page F-4 for details about establishing user-defined variables.

XPUVENT# equates to the maximum number of entries in the output variable area

There is a limit of 100 entries.

Variable Mapping Block DSECT

This DSECT (Figure F-2) defines the output area of the exit parameter DSECT. Using this DSECT allows you to easily address the variable definition table. The number of entries in this table (each entry consists of both a variable name and its value) must not exceed 100.

Figure F-2 Assembler Variable Mapping Block

```
*-----------------------------------------------------------------------*
*                                                                       *
* XPUVARS DEFINES THE OUTPUT AREA OF THE AMUDSNXP DSECT AT LABEL        *
* XPUVAREA.  YOU MAY USE THIS DSECT TO EASILY ADDRESS THE TABLE         *
* DEFINED AT XPUVAREA.                                                  *
*                                                                       *
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
* NOTE:                                                                *
*                                                                      *
*   THE NUMBER OF ENTRIES IN THE XPUVAREA TABLE MUST NOT EXCEED 100    *
*   OR YOU WILL ADDRESS BEYOND THE END OF THE GETMAINED STORAGE        *
*   PROVIDED FOR THIS ROUTINE.                                         *
*                                                                      *
* VARIABLE NAME:                                                       *
*                                                                      *
*   1.  MUST BEGIN WITH AN UNDERSCORE CHARACTER C'_'  X'6D'            *
*   2.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVNAME FIELD              *
*   3.  IF LESS THAN 9 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS  *
*   4.  MAY NOT CONTAIN EMBEDDED BLANKS                                *
*   5.  NON-BLANK CHARACTERS AFTER THE LEFTMOST BYTE MUST BE NATIONAL  *
*       CHARACTERS WHICH ARE A THROUGH Z, 0 THROUGH 9, #, @, AND $     *
*                                                                      *
* VARIABLE DATA:                                                       *
*                                                                      *
*   1.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVDATA FIELD              *
*   2.  IF LESS THAN 8 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS  *
*   3.  MAY NOT CONTAIN EMBEDDED BLANKS                                *
*   4.  NON-BLANK CHARACTERS MUST BE 'NATIONAL' CHARACTERS WHICH ARE   *
*       A THROUGH Z, 0 THROUGH 9, #, @, $, -, AND .                    *
*                                                                      *
*----------------------------------------------------------------------*
*
XPUVARS  DSECT ,                  MAP A VARIABLE ENTRY
XPUVNAME DS    CL9                VARIABLE NAME
XPUVDATA DS    CL8                VARIABLE DATA
XPUVENT$ EQU   *-XPUVNAME         LENGTH OF EACH ENTRY
```

DSECT Fields

The major DSECT fields and their uses are as follows:

XPUVNAME the name of the variable that you are defining

XPUVDATA the value of the variable that you are defining

XPUVENT$ the length of each of these variable entries in the table

BMC Software, Inc., Confidential and Proprietary Information

Sample Assembler User Exit

Figure F-3 is a sample user exit written in assembler.

Figure F-3 Sample Assembler User Exit

```
AMUEDSNA TITLE 'AMUEDSNA - LOADPLUS USER EXIT USER EXAMPLE - V51'
AMUEDSNA CSECT
AMUEDSNA AMODE 31
AMUEDSNA RMODE 24
*-------------------------------------------------------------------*
*                        D I S C L A I M E R                        *
*-------------------------------------------------------------------*
*                                                                   *
*  THIS IS A SAMPLE LOADPLUS USER EXIT.  THIS EXIT WOULD BE USED    *
*  IN ORDER TO DEFINE USER VARIABLES FOR BUILDING DATASET NAMES FOR *
*  DYNAMIC WORKFILE ALLOCATION WITH THE DSNPAT KEYWORD.             *
*                                                                   *
*  THIS EXIT WILL ONLY BE INVOKED WHEN IT IS SPECIFICALLY NAMED     *
*  IN THE AMU$OPTS DSNUEXIT PARAMETER OR IN THE DSNUEXIT PARAMETER OF *
*  THE LOAD COMMAND.  THE SPECIFICATION IN THE LOAD COMMAND WILL    *
*  OVERRIDE THE SPECIFICATION IN THE AMU$OPTS MACRO.  IF THE NAME   *
*  SPECIFIED IS 'NONE' WITHOUT THE SINGLE QUOTE MARKS, THE EXIT WILL *
*  NOT BE INVOKED.                                                  *
*                                                                   *
*  NOTE: PLEASE REVIEW THE DOCUMENTATION IN THE REFERENCE MANUAL,   *
*  AND THE FOLLOWING USAGE NOTES PRIOR TO IMPLEMENTING THIS EXIT.   *
*                                                                   *
*  PLEASE CALL BMC SOFTWARE TECHNICAL SUPPORT WITH ANY QUESTIONS YOU *
*  MAY HAVE IN THIS AREA.                                           *
*                                                                   *
*                PHONE: 1-800-537-1813                              *
*                                                                   *
*-------------------------------------------------------------------*
        SPACE
*-------------------------------------------------------------------*
*                            N O T E S                              *
*-------------------------------------------------------------------*
*                                                                   *
* AMUEDSNA IS CALLED ONCE AT THE BEGINNING OF EITHER THE PRELOAD    *
* PHASE OR THE COMBINED PHASE.                                      *
*                                                                   *
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
* WHEN INVOKED, R1  CONTAINS THE ADDRESS OF A USER EXIT BLOCK         *
*                   DESCRIBED BY THE AMUDSNXP DSECT.                  *
*                                                                     *
*               R13 CONTAINS THE ADDRESS OF THE CALLER'S SAVE AREA    *
*                                                                     *
*               R14 CONTAINS THE CALLER'S RETURN ADDRESS              *
*                                                                     *
*               R15 CONTAINS THE ENTRY POINT ADDRESS OF THIS EXIT     *
*                                                                     *
* UPON EXIT     R15 CONTAINS THE RETURN CODE FROM THIS EXIT           *
*                                                                     *
*                   R15=0     GOOD RETURN                           *
*                                                                     *
*                   R15=OTHER ERROR RETURN, TERMINATE UTILITY         *
*                                                                     *
* AMUDSNXP DSECT    THE AREA MAPPED BY THIS DSECT CONTAINS BOTH INPUT *
*                   AND OUTPUT FIELDS.  INPUT FIELDS ARE PROVIDED TO  *
*                   PASS VITAL INFORMATION TO THIS EXIT.  ANY MODIFI- *
*                   CATIONS TO THESE FIELDS WILL BE DISREGARDED UPON  *
*                   RETURN TO THE CALLER.  THE OUTPUT FIELDS ARE      *
*                   PROVIDED TO ALLOW THE EXIT TO PASS INFORMATION    *
*                   BACK TO THE CALLER.                               *
*                                                                     *
*---------------------------------------------------------------------*
         EJECT
*---------------------------------------------------------------------*
* INTERNAL REGISTER USAGE                                             *
*---------------------------------------------------------------------*
*                                                                     *
*  0 -                                                                *
*  1 - ON ENTRY TO MODULE = A(AMUDSNXP BLOCK) COPIED TO R10           *
*  2 -                                                                *
*  3 -                                                                *
*  4 -                                                                *
*  5 -                                                                *
*  6 -                                                                *
*  7 -                                                                *
*  8 -                                                                *
*  9 -                                                                *
* 10 -                                                                *
* 11 -                                                                *
* 12 - BASEREG                                                        *
* 13 - SAVEAREA                                                       *
* 14 -                                                                *
* 15 -                                                                *
*                                                                     *
*---------------------------------------------------------------------*
*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
R0       EQU   0                  R
R1       EQU   1                   E
R2       EQU   2                    G
R3       EQU   3                     I
R4       EQU   4                      S
R5       EQU   5                       T
R6       EQU   6                        E
R7       EQU   7                         R
R8       EQU   8
R9       EQU   9                            E
R10      EQU   10                            Q
R11      EQU   11                             U
R12      EQU   12                              A
R13      EQU   13                               T
R14      EQU   14                                E
R15      EQU   15                                 S
*
*--------------------------------------------------------------------*
*                                                                    *
* AMUDSNXP DEFINES THE USER VARIABLE USER EXIT PARM BLOCK.           *
*                                                                    *
* NOTE:                                                              *
*                                                                    *
*   YOU MAY NOT MODIFY THE FIELDS IN FRONT OF THE USER AREA.         *
*                                                                    *
*   YOU MAY NOT MODIFY THE 'STRUCTURE' OF THIS DSECT.                *
*                                                                    *
*--------------------------------------------------------------------*
*
AMUDSNXP DSECT ,                  PARMS PASSED TO EXIT
*
*  INPUT AREA
*
XPJOBN   DS    CL8                JOBNAME
XPSTEP   DS    CL8                STEPNAME
XPDBNAME DS    CL8                DATABASE NAME
XPSPNAME DS    CL8                SPACE NAME
XPRESUME DS    CL1                RESUME (Y/N)
XPREPLAC DS    CL1                REPLACE (Y/N)
         DS    CL2                RESERVED FOR LOADPLUS
XPUSER   DS    CL8                USER ID
XPSSID   DS    CL4                DB2 SUBSYSTEM ID
XPDATE   DS    CL6                UTILITY EXECUTION DATE MMDDYY
XPTIME   DS    CL6                UTILITY EXECUTION TIME HHMMSS
XPUTILID DS    CL16               UTILITY ID
XPDATE8  DS    CL8                UTILITY EXECUTION DATE MMDDYYYY
XPGRPNM  DS    CL4                DATA SHARING GROUP NAME
XPVCAT   DS    CL8                VCAT NAME (FROM 1ST PART IF PARTITND)
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
XPDATEJ  DS    CL7               UTILITY EXECUTION DATE CCYYDDD
         DS    CL13              RESERVED FOR LOADPLUS
*
*  USER WORK AREA
*
XPUSRWD1 DS    F                 USER WORD 1
XPUSRWD2 DS    F                 USER WORD 2
XPUSRWD3 DS    F                 USER WORD 3
XPUSRWD4 DS    F                 USER WORD 4
XPUSRWD5 DS    F                 USER WORD 5
XPUSRWD6 DS    F                 USER WORD 6
XPUSRWD7 DS    F                 USER WORD 7
XPUSRWD8 DS    F                 USER WORD 8
*
*  OUTPUT AREA
*
XPUVAREA DS    100CL17           USER VARIABLE AREA
XPUVENT# EQU   100               NUMBER OF ENTRIES IN OUTPUT AREA
XP$      EQU   *-AMUDSNXP
*
*-------------------------------------------------------------------*
*                                                                   *
* XPUVARS DEFINES THE OUTPUT AREA OF THE AMUDSNXP DSECT AT LABEL    *
* XPUVAREA.  YOU MAY USE THIS DSECT TO EASILY ADDRESS THE TABLE     *
* DEFINED AT XPUVAREA.                                              *
*                                                                   *
* NOTE:                                                             *
*                                                                   *
*   THE NUMBER OF ENTRIES IN THE XPUVAREA TABLE MUST NOT EXCEED 100 *
*   OR YOU WILL ADDRESS BEYOND THE END OF THE GETMAINED STORAGE     *
*   PROVIDED FOR THIS ROUTINE.                                      *
*                                                                   *
* VARIABLE NAME:                                                    *
*                                                                   *
*   1.  MUST BEGIN WITH AN UNDERSCORE CHARACTER C'_'  X'6D'         *
*   2.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVNAME FIELD           *
*   3.  IF LESS THAN 9 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS *
*   4.  MAY NOT CONTAIN EMBEDDED BLANKS                             *
*   5.  NON-BLANK CHARACTERS AFTER THE LEFTMOST BYTE MUST BE NATIONAL *
*       CHARACTERS WHICH ARE A THROUGH Z, 0 THROUGH 9, #, @, AND $  *
*                                                                   *
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
* VARIABLE DATA:                                                        *
*                                                                       *
*   1.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVDATA FIELD               *
*   2.  IF LESS THAN 8 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS   *
*   3.  MAY NOT CONTAIN EMBEDDED BLANKS                                 *
*   4.  NON-BLANK CHARACTERS MUST BE 'NATIONAL' CHARACTERS WHICH ARE    *
*       A THROUGH Z, 0 THROUGH 9, #, @, $, -, AND .                     *
*                                                                       *
*-----------------------------------------------------------------------*
*
XPUVARS  DSECT ,                  MAP A VARIABLE ENTRY
XPUVNAME DS    CL9                VARIABLE NAME
XPUVDATA DS    CL8                VARIABLE DATA
XPUVENT$ EQU   *-XPUVNAME         LENGTH OF EACH ENTRY
         EJECT
*
*-----------------------------------------------------------------------*
*                                                                       *
* LOCAL PROGRAM WORK AREA DSECT POINTED TO BY R11                       *
*                                                                       *
*-----------------------------------------------------------------------*
*
WORKAREA DSECT ,
SAVE     DS    18F                LOCAL SAVE AREA
DWORK    DS    D                  DOUBLEWORD WORK AREA
CWORK    DS    CL8                CHARACTER WORK AREA
WRKAREA$ EQU   *-WORKAREA
         EJECT
*
*-----------------------------------------------------------------------*
*                                                                       *
*                    P R O G R A M   S T A R T                          *
*                                                                       *
*-----------------------------------------------------------------------*
*
AMUEDSNA CSECT
*
*>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>*
*                                                                       *
* UNCOMMENT THE NEXT TWO INSTRUCTIONS TO DISABLE THIS EXIT.             *
*                                                                       *
*>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>*
*
*        LA    R15,0                  RC=0, GOOD RETURN
*        BSM   0,R14                  JUST RETURN TO CALLER
*
```

(continued on following page)

(continued from preceding page)

```
*>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>*
*                                                                       *
* UNCOMMENT THE PREVIOUS TWO INSTRUCTIONS TO DISABLE THIS EXIT.         *
*                                                                       *
*>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>*
*
         STM   R14,R12,12(R13)          SAVE CALLERS REGS
         LR    R12,R15                  R12 IS MY BASEREG
         USING AMUEDSNA,R12             ESTABLISH ADDRESSABLITY
*
         LR    R10,R1                   GET A(PASSED BLOCK)
         USING AMUDSNXP,R10             MAP IT
*
*-----------------------------------------------------------------------*
*                                                                       *
* GET SOME WORKING STORAGE                                              *
*                                                                       *
*-----------------------------------------------------------------------*
*
         XR    R11,R11                  ZERO R11
         GETMAIN RC,LV=WRKAREA$,LOC=BELOW  GET WORK AREA
         LTR   R15,R15                  OK?
         BNZ   DSNXRC08                 ..NO, ERROR
         LR    R11,R1                   R11 = A(USER WORK AREA)
         ST    R11,XPUSRWD1             SAVE A(USER WORK AREA)
         USING WORKAREA,R11             ADDRESS WORKAREA DSECT
         LA    R15,SAVE                 MY SAVE AREA@
         ST    R15,8(,R13)              SAVE IN CALLERS SAVE AREA
         ST    R13,4(,R15)              SAVE CALLERS SAVE AREA@ IN MINE
         LR    R13,R15                  SET OURS CURRENT
         EJECT
*
*-----------------------------------------------------------------------*
*                                                                       *
* CREATE USER-DEFINED VARIABLES                                         *
*                                                                       *
* THE FOLLOWING USER VARIABLES WILL BE CREATED:                         *
*                                                                       *
*   _JDATE     - JULIAN DATE IN THE FORM OF Dyyddd BUILT FROM THE      *
*                LOADPLUS SUPPLIED VARIABLE &DATE8                      *
*                                                                       *
*   _JCDATE    - JULIAN DATE IN THE FORM OF Dyyyyddd BUILT FROM THE    *
*                LOADPLUS SUPPLIED VARIABLE &DATE8                      *
*                                                                       *
*   _UTILPFX   - UP TO 8 BYTES OF THE FIRST NODE OF THE UTILITY ID      *
*                                                                       *
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
*   _UTILSFX   - UP TO 8 BYTES OF THE REMAINING PORTION OF THE        *
*                UTILITY ID                                           *
*                                                                     *
*---------------------------------------------------------------------*
*
         LA    R3,XPUVAREA              @(OUTPUT VARIABLE AREA)
         USING XPUVARS,R3               ESTABLISH BASE
*
*---------------------------------------------------------------------*
*                                                                     *
*  BUILD _JDATE AND _JCDATE USER VARIABLES                            *
*                                                                     *
*---------------------------------------------------------------------*
*
         MVC   XPUVNAME(XPUVENT$),BLANKS CLEAR ENTRY
*
         MVC   XPUVNAME,=CL9'_JDATE'    SET VAR NAME
*
         LA    R2,MONTHTAB              @(MONTH TABLE - NO LEAP YEAR)
         PACK  DWORK(8),XPDATE8+4(4)    GET YEARS
         CVB   R4,DWORK                  INTO R4
         LR    R7,R4                    SAVE A COPY IN R7
         SRDL  R4,32                    PREPARE FOR DIVIDE
         D     R4,=F'4'                 DIVIDE BY 4
         LTR   R4,R4                    LEAP YEAR?
         BNZ   DSNX0110                  NO ->
         LR    R4,R7                    GET YEAR AGAIN
         SRDL  R4,32                    PREPARE FOR DIVIDE
         D     R4,=F'100'               DIVIDE BY 100
         LTR   R4,R4                    LEAP YEAR?
         BNZ   DSNX0100                   YES ->
         LR    R4,R7                    GET YEAR AGAIN
         SRDL  R4,32                    PREPARE FOR DIVIDE
         D     R4,=F'400'               DIVIDE BY 400
         LTR   R4,R4                    LEAP YEAR?
         BNZ   DSNX0110                   NO ->
*
DSNX0100 DS    0H
*
         LA    R2,LEAPTAB               @(MONTH TABLE - LEAP YEAR)
*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
DSNX0110 DS    0H
*
         PACK  DWORK(8),XPDATE8(2)      GET MONTH
         CVB   R4,DWORK                  INTO R4
         BCTR  R4,0                     SUBTRACT 1
         SLL   R4,1                     MULTIPLY BY 2 FOR OFFSET
         LH    R6,0(R2,R4)              GET DAYS FOR PRIOR MONTHS
         PACK  DWORK(8),XPDATE8+2(2)    GET DAYS
         CVB   R4,DWORK                  INTO R4
         AR    R6,R4                    TOTAL DAYS IN R6
         LR    R5,R7                    GET YEAR
         M     R4,=F'1000'              SHIFT THE YEAR OVER
         AR    R5,R6                    MAKE IT YYDDD FORMAT
         CVD   R5,DWORK                 MAKE IT PACKED
         MVC   CWORK(L'PL4PAT),PL4PAT   GET EDIT PATTERN
         ED    CWORK(L'PL4PAT),DWORK+4  EDIT YYDDD
*
         MVI   XPUVDATA,C'D'            SET VAR DATA
         MVC   XPUVDATA+1(5),CWORK+3    SET VAR DATA
*
         LA    R3,XPUVENT$(,R3)         @(NEXT ENTRY)
         MVC   XPUVNAME(XPUVENT$),BLANKS CLEAR ENTRY
*
         MVC   XPUVNAME,=CL9'_JCDATE'   SET VAR NAME
*
         CVD   R5,DWORK                 MAKE IT PACKED
         MVC   CWORK(L'PL4PAT),PL4PAT   GET EDIT PATTERN
         ED    CWORK(L'PL4PAT),DWORK+4  EDIT YYYYDDD
*
         MVI   XPUVDATA,C'D'            SET VAR DATA
         MVC   XPUVDATA+1(7),CWORK+1    SET VAR DATA
*
*--------------------------------------------------------------------*
*                                                                    *
*  BUILD _UTILPFX AND _UTILSFX USER VARIABLES                        *
*                                                                    *
*--------------------------------------------------------------------*
*
         LA    R3,XPUVENT$(,R3)         @(NEXT ENTRY)
         MVC   XPUVNAME(XPUVENT$),BLANKS CLEAR ENTRY
*
         MVC   XPUVNAME,=CL9'_UTILPFX'  SET VAR NAME
*
         TRT   XPUTILID,TRTAB           LOOK FOR DELIMETER
         BZ    DSNX0200                  NONE ->
*
```

(continued on following page)

(continued from preceding page)

```
         LA    R2,XPUTILID              @(UTIL ID)
         LR    R4,R1                    SAVE @(DELIMETER)
         SR    R1,R2                    LENGTH OF FIRST PART
         BZ    DSNX0200                  NONE ->
*
         C     R1,=F'8'                 MORE THAN 8 BYTES LONG
         BNH   DSNX0130                  NO ->
         LA    R1,8                     MAKE IT 8 BYTES
*
DSNX0130 DS    0H
*
         BCTR  R1,0                     SUBTRACT 1 FOR EXECUTED MVC
         LA    R5,XPUTILID              @(UTILID)
         EX    R1,DSNXMVCU              MOVE UTILID PREFIX
*
         LA    R3,XPUVENT$(,R3)         @(NEXT ENTRY)
         MVC   XPUVNAME(XPUVENT$),BLANKS CLEAR ENTRY
*
         MVC   XPUVNAME,=CL9'_UTILSFX'  SET VAR NAME
         LA    R5,1(,R4)                @(1ST CHAR PAST DELIMETER)
         LA    R1,XPUTILID+L'XPUTILID   @(END OF UTILID)
         SR    R1,R5                    LENGTH OF SECOND PART OF UTILID
         BZ    DSNXRC00                  NONE ->
         C     R1,=F'8'                 MORE THAN 8 BYTES LONG
         BNH   DSNX0140                  NO ->
         LA    R1,8                     MAKE IT 8 BYTES
*
DSNX0140 DS    0H
*
         BCTR  R1,0                     SUBTRACT 1 FOR EXECUTED MVC
         EX    R1,DSNXMVCU              MOVE UTILID PREFIX
*
         B     DSNXRC00                 RETURN RC=0
*
DSNX0200 DS    0H
*
         MVC   XPUVDATA,XPUTILID        GET FIRST 8 BYTES
*
         LA    R3,XPUVENT$(,R3)         @(NEXT ENTRY)
         MVC   XPUVNAME(XPUVENT$),BLANKS CLEAR ENTRY
*
         MVC   XPUVNAME,=CL9'_UTILSFX'  SET VAR NAME
         MVC   XPUVDATA,XPUTILID+8      GET NEXT 8 BYTES
*
         B     DSNXRC00                 RETURN RC=0
*
```

(continued on following page)

(continued from preceding page)

```
DSNXMVCU MVC   XPUVDATA(0),0(R5)        *** EXECUTED INSTRUCTION ***
*
         DROP  R3
*
         EJECT
*
*--------------------------------------------------------------------*
*                                                                    *
* SET RETURN CODE AND EXIT                                           *
*                                                                    *
*--------------------------------------------------------------------*
*
DSNXRC00 DS    0H
*
         LA    R3,0                    SAVE RETURN CODE
         B     DSNX9000                  GOOD RETURN
*
DSNXRC08 DS    0H
*
         LA    R3,8                    SAVE RETURN CODE
*                                       TERMINATE LOADPLUS EXECUTION
DSNX9000 DS    0H
*
         LTR   R11,R11                 DID WE GET STORAGE?
         BZ    DSNX9900                 NO ->
         L     R13,SAVE+4
         FREEMAIN RC,LV=WRKAREA$,A=(R11) FREE LOCAL WORK AREA
*
DSNX9900 DS    0H
*
         LR    R15,R3                  RESTORE RETURN CODE
         L     R14,12(,R13)
         LM    R0,R12,20(R13)
         BSM   0,R14                   RETURN
         EJECT
*
*--------------------------------------------------------------------*
*                                                                    *
*                        C O N S T A N T S                           *
*                                                                    *
*--------------------------------------------------------------------*
*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
*                0 1 2 3 4 5 6 7 8 9 A B C D E F
*
TRTAB    DC    X'00000000000000000000000000000000' 0
         DC    X'00000000000000000000000000000000' 1
         DC    X'00000000000000000000000000000000' 2
         DC    X'00000000000000000000000000000000' 3
         DC    X'FF000000000000000000FF0000FFFF' 4 (SP) . + |
         DC    X'00000000000000000000000000000FF00' 5 ;
         DC    X'FFFF000000000000000FFFF00FF0000' 6 - / ≠ , _
         DC    X'0000000000000000000FF000000FF00' 7 : =
         DC    X'00000000000000000000000000000000' 8
         DC    X'00000000000000000000000000000000' 9
         DC    X'00000000000000000000000000000000' A
         DC    X'00000000000000000000000000000000' B
         DC    X'00000000000000000000000000000000' C
         DC    X'00000000000000000000000000000000' D
         DC    X'FF000000000000000000000000000000' E \
         DC    X'00000000000000000000000000000000' F
*
*                0 1 2 3 4 5 6 7 8 9 A B C D E F
*
MONTHTAB DC    H'0'                    JANUARY
         DC    H'31'                   FEBRUARY
         DC    H'59'                   MARCH
         DC    H'90'                   APRIL
         DC    H'120'                  MAY
         DC    H'151'                  JUNE
         DC    H'181'                  JULY
         DC    H'212'                  AUGUST
         DC    H'243'                  SEPTEMBER
         DC    H'273'                  OCTOBER
         DC    H'304'                  NOVEMBER
         DC    H'334'                  DECEMBER
*
LEAPTAB  DC    H'0'                    JANUARY
         DC    H'31'                   FEBRUARY
         DC    H'60'                   MARCH
         DC    H'91'                   APRIL
         DC    H'121'                  MAY
         DC    H'152'                  JUNE
         DC    H'182'                  JULY
         DC    H'213'                  AUGUST
         DC    H'244'                  SEPTEMBER
         DC    H'274'                  OCTOBER
         DC    H'305'                  NOVEMBER
         DC    H'335'                  DECEMBER
*
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
PL4PAT   DC    X'4021202020202020'     EDIT PATTERN
*
BLANKS   DC    CL17' '                 SOME BLANKS
*
         EJECT
         LTORG ,
         SPACE
         END   AMUEDSNA
```

LOADPLUS COBOL II and LE COBOL User Exit

This section provides a sample exit parameter record, variable mapping record, and exit that you can use to write a user exit in COBOL or LE COBOL.

COBOL II and LE COBOL Exit Parameter Record

This record (Figure F-4) contains both input and output fields. Input fields pass vital information to the user exit, such as the database name and user ID. If you make any modifications to these input fields, LOADPLUS disregards them on return. The output fields pass information about your user variables back to LOADPLUS.

Figure F-4 COBOL II and LE COBOL Exit Parameter Record

```
      *---------------------------------------------------------------
       01  LOAD-EXIT-PARMS.
           05  FIXED-PARM-VALUES.
               10  EXIT-JOBNAME              PIC X(8).
               10  EXIT-STEPNAME             PIC X(8).
               10  EXIT-DBNAME               PIC X(8).
               10  EXIT-TSNAME               PIC X(8).
               10  EXIT-RESUME               PIC X(1).
               10  EXIT-REPLACE              PIC X(1).
               10  EXIT-FILLER1              PIC X(2).
               10  EXIT-USERID               PIC X(8).
               10  EXIT-DB2-SSID             PIC X(4).
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
    10  EXIT-DATE.
        15  EXIT-MM               PIC 9(2).
        15  EXIT-DD               PIC 9(2).
        15  EXIT-YY               PIC 9(2).
    10  EXIT-TIME                 PIC X(6).
    10  EXIT-UTILID-PARM          PIC X(16).
    10  FILLER REDEFINES EXIT-UTILID-PARM.
        15  EXIT-PREFIX           PIC X(8).
        15  EXIT-SUFFIX           PIC X(8).
    10  EXIT-DATE8.
        15  EXIT-DATE8-MM         PIC 9(2).
        15  EXIT-DATE8-DD         PIC 9(2).
        15  EXIT-DATE8-YEAR       PIC 9(4).
        15  FILLER REDEFINES EXIT-DATE8-YEAR.
            20  EXIT-DATE8-CC     PIC 9(2).
            20  EXIT-DATE8-YY     PIC 9(2).
    10  EXIT-GRPNM                PIC X(4).
    10  EXIT-VCAT                 PIC X(8).
    10  EXIT-DATEJ.
        15  EXIT-DATEJ-YEAR       PIC 9(4).
        15  FILLER REDEFINES EXIT-DATEJ-YEAR.
            20  EXIT-DATEJ-CC     PIC 9(2).
            20  EXIT-DATEJ-CC     PIC 9(2).
        15  EXIT-DATEJ-DDD        PIC 9(3).
    10  EXIT-FILLER2              PIC X(13).

05  WORK-AREA-ADDRESSES.
    10  WORK-AREA-1               PIC 9(4).
    10  WORK-AREA-2               PIC 9(4).
    10  WORK-AREA-3               PIC 9(4).
    10  WORK-AREA-4               PIC 9(4).
    10  WORK-AREA-5               PIC 9(4).
    10  WORK-AREA-6               PIC 9(4).
    10  WORK-AREA-7               PIC 9(4).
    10  WORK-AREA-8               PIC 9(4).
```

Parameter Record Fields

The major parameter record fields and their uses are as follows:

EXIT-JOBNAME contains the job name, up to 8 bytes

EXIT-STEPNAME contains the step name, up to 8 bytes

EXIT-DBNAME contains the database name, up to 8 bytes

BMC Software, Inc., Confidential and Proprietary Information

EXIT-TSNAME contains the name of the table space or index space from the LOAD command, up to 8 bytes

EXIT-RESUME contains an indication of the whether a LOAD RESUME is being performed—Y (yes) or N (no)

EXIT-REPLACE contains an indication of the whether a LOAD REPLACE is being performed—Y (yes) or N (no)

EXIT-USERID contains the user ID of the user running the LOADPLUS utility, up to 8 bytes

EXIT-DB2-SSID contains the DB2 subsystem ID, 4 bytes

EXIT-DATE contains the date of the execution of the utility, in the format MMDDYY, 6 bytes

EXIT-TIME contains the time of the execution of the utility, in the format HHMMSS, 6 bytes

EXIT-UTILID-PARM contains the utility ID, up to 16 bytes

EXIT-DATE8 contains the date of the execution of the utility, in the format MMDDYYYY, 8 bytes

EXIT-GRPNM contains the DB2 data-sharing group name

In a nondata-sharing environment, the field contains the DB2 subsystem ID.

EXIT-VCAT contains the VCATNAME specified in the DB2 catalog for the table space being loaded, or for the first partition if the table space is partitioned

EXIT-DATEJ contains the Julian date of the execution of the utility, in the format CCYYDDD, 7 bytes

WORK-AREA-1...WORK-AREA-8
provides work space, 8 parameters, up to 4 bytes each

COBOL II and LE COBOL Variable Mapping Record

This record (Figure F-5 on page F-23) defines the output area of the exit parameter record. Using this record allows you to easily address the variable definition table. The number of entries in this table (each entry consists of both a variable name and its value) must not exceed 100.

BMC Software, Inc., Confidential and Proprietary Information

Figure F-5 COBOL II and LE COBOL Variable Mapping Record

```
      05  USER-DEFINED-VARILABLE-TABLE OCCURS 100 TIMES.
          10  VARIABLE-NAME             PIC X(9).
          10  VARIABLE-VALUE            PIC X(8).
          10  FILLER REDEFINES VARIABLE-VALUE.
              15  VARIABLE-PREFIX       PIC X(1).
              15  VARIABLE-JUL-DATE     PIC X(7).
```

Variable Mapping Record Fields

The major variable mapping record fields and their uses are as follows:

USER-DEFINED-VARIABLE-TABLE
the table containing user-defined variable information

See page F-4 for details about establishing user-defined variables.

VARIABLE-NAME the name of the variable that you are defining

VARIABLE-VALUE the value of the variable that you are defining

Sample COBOL II and LE COBOL User Exit

Figure F-6 is a sample user exit written in COBOL.

Figure F-6 Sample COBOL II and LE COBOL User Exit

```
      *-----------------------------------------------------------------
      * ALL COBOL MODULES MUST BE COMPILED WITH DATA(31) AND DYNAM TO
      * EXECUTE PROPERLY!!!!!!!!
      *-----------------------------------------------------------------
       IDENTIFICATION  DIVISION.
      *-----------------------------------------------------------------
       PROGRAM-ID.     AMUEDSN2.
       AUTHOR.         BMC SOFTWARE
       DATE-WRITTEN.   AUGUST 1995.
       DATE-COMPILED.
      *-----------------------------------------------------------------
      * AMUEDSN2 IS A SAMPLE DB2 COBOL II USER EXIT.
      *
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
* THIS IS A SAMPLE LOADPLUS USER EXIT.   THIS EXIT WOULD BE USED
* IN ORDER TO DEFINE USER VARIABLES FOR BUILDING DATASET NAMES
* FOR DYNAMIC WORKFILE ALLOCATION WITH THE DSNPAT KEYWORD.
*
* AMUEDSN2 IS CALLED ONLY ONCE PER EXECUTION OF AMUUMAIN
* THE MODULE IS CALLED AT THE BEGINNING OF EITHER THE PRELOAD
* PHASE OR THE COMBINED PHASE.
*
* THIS EXIT WILL ONLY BE INVOKED WHEN IT IS THE VALUE SET FOR
* DSNUEXIT PARM IN AMU$OPTS DURING INSTALLATION OR IN THE
* DSNUEXIT PARAMETER OF THE LOAD COMMAND.  THE LOAD COMMAND
* OPTION WILL OVERRIDE THE PARM IN THE AMU$OPTS MACRO.
*
* PLEASE REVIEW ADDITIONAL DOCUMENTION IN THE REFERENCE MANUAL
*----------------------------------------------------------------

 ENVIRONMENT     DIVISION.
 INPUT-OUTPUT    SECTION.
 FILE-CONTROL.
 DATA            DIVISION.
 FILE            SECTION.
 EJECT
*----------------------------------------------------------------
 WORKING-STORAGE SECTION.
*----------------------------------------------------------------

 01  FILLER                 PIC X(16) VALUE 'WORKING STORAGE '.

*----------------------------------------------------------------
*    MISCELLANEOUS LITERALS, TABLE SUBSCRIPTS, NUMERIC VALUES
*    FOR JULIAN-DATE OR JULIAN/CENTURY-DATE CONVERSION, ETC.
*----------------------------------------------------------------

 01  MISCELLANEOUS.
     05  SAVE-DATA          PIC X(8)          VALUE SPACES.
     05  DAYS-SUBX          PIC S9(3) COMP    VALUE ZERO.
     05  SUBX               PIC S9(3) COMP    VALUE ZERO.
     05  ONE                PIC S9(1) COMP-3  VALUE +1.
     05  TWO                PIC S9(1) COMP-3  VALUE +2.
     05  FOUR               PIC S9(1) COMP-3  VALUE +4.
     05  MAX-LENGTH         PIC S9(3) COMP-3  VALUE +16.
     05  NINETEEN           PIC S9(3) COMP-3  VALUE +19.
     05  TWENTY             PIC S9(3) COMP-3  VALUE +20.
     05  NINETY-FIVE        PIC S9(3) COMP-3  VALUE +95.
     05  ONE-HUNDRED        PIC S9(3) COMP-3  VALUE +100.
     05  FOUR-HUNDRED       PIC S9(3) COMP-3  VALUE +400.
     05  JULIAN-DATE-DESC   PIC X(9)          VALUE '_JDATE'.
     05  JULIAN-CDATE-DESC  PIC X(9)          VALUE '_JCDATE'.
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
    05  UTILITY-PREFIX     PIC X(9)          VALUE '_UTILPFX'.
    05  UTILITY-SUFFIX     PIC X(9)          VALUE '_UTILSFX'.
    05  UTILID-PREFIX      PIC X(8)          VALUE SPACES.
    05  UTILID-SUFFIX      PIC X(8)          VALUE SPACES.
    05  UTILID-POINTER     PIC S9(3)         VALUE ZERO.
    05  UTILID-COUNTER     PIC S9(3)         VALUE ZERO.
    05  UTILID-TALLY       PIC S9(3)         VALUE ZERO.

01  DATE-WORK-AREA.
    05  CONVERTED-DATE.
        10  DATE-PREFIX            PIC X(1)     VALUE 'D'.
        10  JULIAN-CDATE           PIC 9(7)     VALUE ZERO.
        10  FILLER REDEFINES JULIAN-CDATE.
            15  JULIAN-CC          PIC 9(2).
            15  JULIAN-DATE        PIC 9(5).
            15  FILLER REDEFINES JULIAN-DATE.
                20  JULIAN-YY      PIC 9(2).
                20  JULIAN-DAYS    PIC 9(3).
    05  WORK-YEAR                  PIC S9(3) COMP-3 VALUE ZERO.
    05  YEAR-ANSWER                PIC S9(3) COMP-3 VALUE ZERO.
    05  YEAR-REMAINDER             PIC S9(9) COMP-3 VALUE ZERO.
    05  FILLER REDEFINES YEAR-REMAINDER.
        10  YEAR-X                 PIC X(5).

*-----------------------------------------------------------------
*    THE FOLLOWING TWO TABLES ARE USED TO CALCULATE THE JULIAN
*    DAY DEPENDING ON WHETHER THE YEAR IS A LEAP OR NOT.
*-----------------------------------------------------------------

01  NO-LEAP-MONTHS.
    05  JANUARY              PIC S9(3)  COMP-3  VALUE 00.
    05  FEBUARY              PIC S9(3)  COMP-3  VALUE 31.
    05  MARCH                PIC S9(3)  COMP-3  VALUE 59.
    05  APRIL                PIC S9(3)  COMP-3  VALUE 90.
    05  MAY                  PIC S9(3)  COMP-3  VALUE 120.
    05  JUNE                 PIC S9(3)  COMP-3  VALUE 151.
    05  JULY                 PIC S9(3)  COMP-3  VALUE 181.
    05  AUGUST               PIC S9(3)  COMP-3  VALUE 212.
    05  SEPTEMBER            PIC S9(3)  COMP-3  VALUE 243.
    05  OCTOBER              PIC S9(3)  COMP-3  VALUE 273.
    05  NOVEMBER             PIC S9(3)  COMP-3  VALUE 304.
    05  DECEMBER             PIC S9(3)  COMP-3  VALUE 334.
01  FILLER REDEFINES NO-LEAP-MONTHS.
    05  MONTH-DAYS           PIC S9(3)  COMP-3  OCCURS 12 TIMES.
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
 01  LEAP-MONTHS.
     05  LEAP-JANUARY        PIC S9(3)  COMP-3  VALUE 00.
     05  LEAP-FEBRUARY       PIC S9(3)  COMP-3  VALUE 31.
     05  LEAP-MARCH          PIC S9(3)  COMP-3  VALUE 60.
     05  LEAP-APRIL          PIC S9(3)  COMP-3  VALUE 91.
     05  LEAP-MAY            PIC S9(3)  COMP-3  VALUE 121.
     05  LEAP-JUNE           PIC S9(3)  COMP-3  VALUE 152.
     05  LEAP-JULY           PIC S9(3)  COMP-3  VALUE 182.
     05  LEAP-AUGUST         PIC S9(3)  COMP-3  VALUE 213.
     05  LEAP-SEPTEMBER      PIC S9(3)  COMP-3  VALUE 244.
     05  LEAP-OCTOBER        PIC S9(3)  COMP-3  VALUE 274.
     05  LEAP-NOVEMBER       PIC S9(3)  COMP-3  VALUE 305.
     05  LEAP-DECEMBER       PIC S9(3)  COMP-3  VALUE 335.
 01  FILLER REDEFINES LEAP-MONTHS.
     05  LEAP-MONTH-DAYS     PIC S9(3)  COMP-3  OCCURS 12 TIMES.

*-----------------------------------------------------------------
 LINKAGE SECTION.
 *----------------------------------------------------------------
 *    THE TABLE CAN NOT OCCUR MORE THAN 100 TIME OR A STORAGE
 *    OVERLAY WILL OCCUR.
 *----------------------------------------------------------------
 01  LOAD-EXIT-PARMS.
     05  FIXED-PARM-VALUES.
         10  EXIT-JOBNAME              PIC X(8).
         10  EXIT-STEPNAME             PIC X(8).
         10  EXIT-DBNAME               PIC X(8).
         10  EXIT-TSNAME               PIC X(8).
         10  EXIT-RESUME               PIC X(1).
         10  EXIT-REPLACE              PIC X(1).
         10  EXIT-FILLER1              PIC X(2).
         10  EXIT-USERID               PIC X(8).
         10  EXIT-DB2-SSID             PIC X(4).
         10  EXIT-DATE.
             15  EXIT-MM               PIC 9(2).
             15  EXIT-DD               PIC 9(2).
             15  EXIT-YY               PIC 9(2).
         10  EXIT-TIME                 PIC X(6).
         10  EXIT-UTILID-PARM          PIC X(16).
         10  FILLER REDEFINES EXIT-UTILID-PARM.
             15  EXIT-PREFIX           PIC X(8).
             15  EXIT-SUFFIX           PIC X(8).
         10  EXIT-DATE8.
             15  EXIT-DATE8-MM         PIC 9(2).
             15  EXIT-DATE8-DD         PIC 9(2).
             15  EXIT-DATE8-YEAR       PIC 9(4).
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
            15  FILLER REDEFINES EXIT-DATE8-YEAR.
                20  EXIT-DATE8-CC     PIC 9(2).
                20  EXIT-DATE8-YY     PIC 9(2).
        10  EXIT-GRPNM                PIC X(4).
        10  EXIT-VCAT                 PIC X(8).
        10  EXIT-DATEJ.
            15  EXIT-DATEJ-YEAR       PIC 9(4).
            15  FILLER REDEFINES EXIT-DATEJ-YEAR.
                20  EXIT-DATEJ-CC     PIC 9(2).
                20  EXIT-DATEJ-CC     PIC 9(2).
            15  EXIT-DATEJ-DDD        PIC 9(3).
        10  EXIT-FILLER2              PIC X(13).

    05  WORK-AREA-ADDRESSES.
        10  WORK-AREA-1               PIC 9(4).
        10  WORK-AREA-2               PIC 9(4).
        10  WORK-AREA-3               PIC 9(4).
        10  WORK-AREA-4               PIC 9(4).
        10  WORK-AREA-5               PIC 9(4).
        10  WORK-AREA-6               PIC 9(4).
        10  WORK-AREA-7               PIC 9(4).
        10  WORK-AREA-8               PIC 9(4).

    05  USER-DEFINED-VARILABLE-TABLE OCCURS 100 TIMES.
        10  VARIABLE-NAME             PIC X(9).
        10  VARIABLE-VALUE            PIC X(8).
        10  FILLER REDEFINES VARIABLE-VALUE.
            15  VARIABLE-PREFIX       PIC X(1).
            15  VARIABLE-JUL-DATE     PIC X(7).

*-----------------------------------------------------------------
 PROCEDURE DIVISION USING LOAD-EXIT-PARMS.
*-----------------------------------------------------------------

 0000-MAIN.
     MOVE ZERO TO RETURN-CODE.
     PERFORM 1000-PROCESS-DATE.
     GOBACK.

*-----------------------------------------------------------------
*    THE DATE IS PASSED IN A MMDDYYYY FORMAT AND CONVERTED TO
*    A JULIAN-DATE(WITH NO CENTURY) FORMAT OR A JULIAN-DATE
*    (WITH THE CENTURY) FORMAT.
*-----------------------------------------------------------------
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
1000-PROCESS-DATE.
    MOVE EXIT-DATE8-DD   TO JULIAN-DAYS.
    MOVE EXIT-DATE8-MM   TO DAYS-SUBX.
    MOVE EXIT-DATE8-CC   TO JULIAN-CC.
    MOVE EXIT-DATE8-YY   TO JULIAN-YY.
    MOVE EXIT-DATE8-YEAR TO WORK-YEAR.

    DIVIDE WORK-YEAR BY FOUR
              GIVING YEAR-ANSWER
              REMAINDER YEAR-REMAINDER.

    IF YEAR-REMAINDER > ZERO
        THEN
            PERFORM 1100-NO-LEAP-YEAR
        ELSE
            DIVIDE WORK-YEAR BY ONE-HUNDRED
                     GIVING YEAR-ANSWER
                     REMAINDER YEAR-REMAINDER
            IF YEAR-REMAINDER > ZERO
                THEN
                    PERFORM 1200-LEAP-YEAR
                ELSE
                    DIVIDE WORK-YEAR BY FOUR-HUNDRED
                              GIVING YEAR-ANSWER
                              REMAINDER YEAR-REMAINDER
                    IF YEAR-REMAINDER > ZERO
                        THEN
                            PERFORM 1100-NO-LEAP-YEAR
                        ELSE
                            ADD LEAP-MONTH-DAYS(DAYS-SUBX)
                                            TO JULIAN-DAYS
                            PERFORM 1200-LEAP-YEAR
                    END-IF
            END-IF
    END-IF.

1000-PROCESS-DATE-EXIT.
    EXIT.

1100-NO-LEAP-YEAR.
    ADD MONTH-DAYS(DAYS-SUBX) TO JULIAN-DAYS.
    PERFORM 1300-CENTURY.

1100-NO-LEAP-YEAR-EXIT.
    EXIT.
```

(continued on following page)

(continued from preceding page)

```
 1200-LEAP-YEAR.
     ADD LEAP-MONTH-DAYS(DAYS-SUBX) TO JULIAN-DAYS.
     PERFORM 1300-CENTURY.

 1200-LEAP-YEAR-EXIT.
     EXIT.

 1300-CENTURY.
     PERFORM 1400-EDIT-UTILID.

 1300-CENTURY-EXIT.
     EXIT.

*----------------------------------------------------------------
*    THE FULL 16 BYTES OF THE UTILITY ID PARM IS CHECKED FOR
*    A DELIMITER.  IF ONE IS FOUND AFTER THE FIRST 8 BYTES, IT
*    IS INCLUDED IN THE SUFFIX.  IF A DELIMITER IS FOUND IN THE
*    FIRST 8 BYTES, ONLY THOSE CHARACTERS/NUMBERS UP TO THE
*    DELIMITER, WILL BE MOVED INTO THE PREFIX FIELD.
*----------------------------------------------------------------
*    THE DELIMITER CHARACTER IMMEDIATELY FOLLOWING THE FORWARD
*    SLASH ('/') IS A BROKEN VERTICAL BAR, HEX 6A, AND IS NOT
*    DISPLAYABLE IN BOOKMANAGER
*----------------------------------------------------------------

 1400-EDIT-UTILID.
     MOVE ONE TO UTILID-POINTER, UTILID-TALLY.
     UNSTRING EXIT-UTILID-PARM
              DELIMITED BY ' ' OR '.' OR '+' OR '|' OR ';'
                 OR '-' OR '/' OR '¦'OR ',' OR '_' OR ':'
                 OR '=' OR '\'
             INTO UTILID-PREFIX
                  COUNT IN UTILID-COUNTER
                  WITH POINTER UTILID-POINTER.
     IF UTILID-COUNTER = MAX-LENGTH
         THEN
             MOVE EXIT-SUFFIX TO UTILID-SUFFIX
         ELSE
             MOVE UTILID-COUNTER TO UTILID-POINTER
             ADD TWO TO UTILID-POINTER
             UNSTRING EXIT-UTILID-PARM
                 INTO UTILID-SUFFIX
                 WITH POINTER UTILID-POINTER
     END-IF.
     PERFORM 1500-UTILID-PARMS.
```

(continued on following page)

(continued from preceding page)

```
1400-EDIT-UTILID-EXIT.
    EXIT.

1500-UTILID-PARMS.
    MOVE ONE TO SUBX.
    MOVE UTILITY-PREFIX TO VARIABLE-NAME(SUBX).
    MOVE UTILID-PREFIX TO VARIABLE-VALUE(SUBX).

    ADD ONE TO SUBX.
    MOVE UTILITY-SUFFIX TO VARIABLE-NAME(SUBX).
    MOVE UTILID-SUFFIX TO VARIABLE-VALUE(SUBX).

    ADD ONE TO SUBX.
    MOVE JULIAN-DATE-DESC TO VARIABLE-NAME(SUBX).
    MOVE DATE-PREFIX TO VARIABLE-PREFIX(SUBX).
    MOVE JULIAN-DATE TO VARIABLE-JUL-DATE(SUBX).

    ADD ONE TO SUBX.
    MOVE JULIAN-CDATE-DESC TO VARIABLE-NAME(SUBX).
    MOVE CONVERTED-DATE TO VARIABLE-VALUE(SUBX).

1500-UTILID-PARMS-EXIT.
    EXIT.
```

LOADPLUS C User Exit

This section provides a sample exit parameter structure, variable mapping structure, and exit that you can use to write a user exit in C.

C Exit Parameter Structure

This structure (Figure F-7 on page F-31) contains both input and output fields. Input fields pass vital information to the user exit, such as the database name and user ID. If you make any modifications to these input fields, LOADPLUS disregards them on return. The output fields pass information about your user variables back to LOADPLUS.

BMC Software, Inc., Confidential and Proprietary Information

Figure F-7 C Exit Parameter Structure

```
struct amudsnpx {
     char       xpjobn(|8|);     /* jobname                          */
     char       xpstep(|8|);     /* stepname                         */
     char       xpdbname(|8|);   /* database name                    */
     char       xpspname(|8|);   /* space name                       */
     char       xpresume(|1|);   /* resume   (Y or N)                */
     char       xpreplace(|1|);  /* replace  (Y or N)                */
     char       xpresrv1(|2|);   /* reserved for loadplus            */
     char       xpuser(|8|);     /* user id                          */
     char       xpssid(|4|);     /* db2 subsystem id                 */
     char       xpdate(|6|);     /* utility execution date mmddyy    */
     char       xptime(|6|);     /* utility execution time hhmmss    */
     char       xputilid(|16|);  /* utility id                       */
     char       xpdate8(|8|);    /* utility execution date mmddyyyy */
     char       xpgrpnm(|4|);    /* data sharing group name          */
     char       xpvcat(|8|); /* vcat name (from 1st part if partitnd)*/
     char       xpdatej(|7|);    /* utility execution date ccyyddd   */
     char       xpresrv2(|13|);  /* reserved for loadplus            */
     int           xpusrwd1;     /* user word 1*/
     int           xpusrwd2;     /* user word 2*/
     int           xpusrwd3;     /* user word 3*/
     int           xpusrwd4;     /* user word 4*/
     int           xpusrwd5;     /* user word 5*/
     int           xpusrwd6;     /* user word 6*/
     int           xpusrwd7;     /* user word 7*/
     int           xpusrwd8;     /* user word 8*/
     ...
```

Structure Fields

The major structure fields and their uses are as follows:

xpjobn contains the job name, up to 8 bytes

xpstep contains the step name, up to 8 bytes

xpdbname contains the database name, up to 8 bytes

xpspname contains the name of the table space or index space from the LOAD command, up to 8 bytes

xpresume contains an indication of whether a LOAD RESUME is being performed—Y (yes) or N (no)

xpreplace contains an indication of whether a LOAD REPLACE is being performed—Y (yes) or N (no)

BMC Software, Inc., Confidential and Proprietary Information

xpuser contains the user ID of the user running the LOADPLUS utility, up to 8 bytes

xpssid contains the DB2 subsystem ID, 4 bytes

xpdate contains the date of the execution of the utility, in the format MMDDYY, 6 bytes

xptime contains the time of the execution of the utility, in the format HHMMSS, 6 bytes

xputilid contains the utility ID, up to 16 bytes

xpdate8 contains the date of the execution of the utility, in the format MMDDYYYY, 8 bytes

xpgrpnm contains the DB2 data-sharing group name

In a nondata-sharing environment, the field contains the DB2 subsystem ID.

xpvcat contains the VCATNAME specified in the DB2 catalog for the table space being loaded, or for the first partition if the table space is partitioned

xpdatej contains the Julian date of the execution of the utility, in the format CCYYDDD, 7 bytes

xpusrwd1...xpusrwd8 provides work space, 8 parameters, up to 4 bytes each

C Exit Variable Mapping Structure

This structure (Figure F-8) defines the output area of the exit parameter structure. Using this structure allows you to easily address the variable definition table. The number of entries in this table (each entry consists of both a variable name and its value) must not exceed 100.

Figure F-8 C Exit Variable Mapping Structure

```
struct XPUVAREA {
    char xpuvname(|9|);
    char xpuvdata(|8|);
} xpuvars(|XPUVENT|) ;
```

BMC Software, Inc., Confidential and Proprietary Information

Structure Fields

The major structure fields and their uses are as follows:

XPUVAREA defines the area that contains user-defined variable information

See page F-4 for details about establishing user-defined variables.

xpuvname the name of the variable that you are defining

xpuvdata the value of the variable that you are defining

XPUVENT equates to the maximum number of entries in the output variable area

There is a limit of 100 entries.

Sample C User Exit

Figure F-9 is a sample user exit written in C.

Figure F-9 Sample C User Exit

```
/*--------------------------------------------------------------------*/
/*                       D I S C L A I M E R                          */
/*--------------------------------------------------------------------*/
/*                                                                    */
/* THIS IS A SAMPLE LOADPLUS USER EXIT.  THIS EXIT WOULD BE USED      */
/* IN ORDER TO DEFINE USER VARIABLES FOR BUILDING DATASET NAMES FOR   */
/* DYNAMIC WORKFILE ALLOCATION WITH THE DSNPAT KEYWORD.               */
/*                                                                    */
/* THIS EXIT WILL ONLY BE INVOKED WHEN IT IS SPECIFICALLY NAMED       */
/* IN THE AMU$OPTS DSNUEXIT PARAMETER OR IN THE DSNUEXIT PARAMETER OF */
/* THE LOAD COMMAND.  THE SPECIFICATION IN THE LOAD COMMAND WILL      */
/* OVERRIDE THE SPECIFICATION IN THE AMU$OPTS MACRO.  IF THE NAME     */
/* SPECIFIED IS 'NONE' WITHOUT THE SINGLE QUOTE MARKS, THE EXIT WILL  */
/* NOT BE INVOKED.                                                    */
/*                                                                    */
/* NOTE: PLEASE REVIEW THE DOCUMENTATION IN THE REFERENCE MANUAL,     */
/* AND THE FOLLOWING USAGE NOTES PRIOR TO IMPLEMENTING THIS EXIT.     */
/*                                                                    */
/* PLEASE CALL BMC SOFTWARE PRODUCT SUPPORT WITH ANY QUESTIONS YOU    */
/* MAY HAVE IN THIS AREA.                                             */
/*                                                                    */
/*                PHONE: 1-800-537-1813                               */
/*                                                                    */
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
/*--------------------------------------------------------------------*/
/*                            N O T E S                               */
/*--------------------------------------------------------------------*/
/*                                                                    */
/* AMUEDSC  IS CALLED ONCE AT THE BEGINNING OF EITHER THE PRELOAD     */
/* PHASE OR THE COMBINED PHASE.                                       */
/*                                                                    */
/* WHEN INVOKED IT IS PASSED THE ADDRESS OF A USER EXIT BLOCK         */
/*                  DESCRIBED BY THE AMUDSNXP STRUCT.                 */
/*                  THIS CALL IS FROM ASEMBLER MODULE DYNWDSNX         */
/*                                                                    */
/* UPON EXIT    RETRUN CODE FROM THIS EXIT                            */
/*                                                                    */
/*                      0     GOOD RETURN                             */
/*                                                                    */
/*                      OTHER ERROR RETURN, TERMINATE UTILITY         */
/*                                                                    */
/* AMUDSNXP STRUCT  THIS STRUCT CONTAINS BOTH INPUT                   */
/*                  AND OUTPUT FIELDS.  INPUT FIELDS ARE PROVIDED TO  */
/*                  PASS VITAL INFORMATION TO THIS EXIT.  ANY MODIFI- */
/*                  CATIONS TO THESE FIELDS WILL BE DISREGARDED UPON  */
/*                  RETURN TO THE CALLER.  THE OUTPUT FIELDS ARE      */
/*                  PROVIDED TO ALLOW THE EXIT TO PASS INFORMATION    */
/*                  BACK TO THE CALLER.                               */
/*                                                                    */
/*--------------------------------------------------------------------*/
/*                                                                    */
/*--------------------------------------------------------------------*/
/*                                                                    */
/* XPUVARS DEFINES THE OUTPUT AREA OF THE AMUDSNXP STRUCT AT TAG      */
/* XPUVAREA. YOU MAY USE THIS STRUCT TO EASILY REFERENCE THE ARRAY    */
/* DEFINED AT XPUVAREA.                                               */
/*                                                                    */
/* NOTE:                                                              */
/*                                                                    */
/*  THE NUMBER OF ENTRIES IN THE XPUVAREA ARRAY MUST NOT EXCEED 100   */
/*  OR YOU WILL ADDRESS BEYOND THE END OF THE STRUCTURE STORAGE       */
/*  PROVIDED FOR THIS ROUTINE.                                        */
/*                                                                    */
/* VARIABLE NAME:                                                     */
/*                                                                    */
/*  1.  MUST BEGIN WITH AN UNDERSCORE CHARACTER C'_'  X'6D'           */
/*  2.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVNAME FIELD             */
/*  3.  IF LESS THAN 9 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS */
/*  4.  MAY NOT CONTAIN EMBEDDED BLANKS                               */
/*  5.  NON-BLANK CHARACTERS AFTER THE LEFTMOST BYTE MUST BE NATIONAL */
/*      CHARACTERS WHICH ARE A THROUGH Z, 0 THROUGH 9, #, @, AND $    */
/*  6.  MUST BE UPPER CASE.                                           */
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
/*                                                                    */
/* VARIABLE DATA:                                                     */
/*                                                                    */
/*  1.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVDATA FIELD             */
/*  2.  IF LESS THAN 8 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS */
/*  3.  MAY NOT CONTAIN EMBEDDED BLANKS                               */
/*  4.  NON-BLANK CHARACTERS MUST BE 'NATIONAL' CHARACTERS WHICH ARE  */
/*      A THROUGH Z, 0 THROUGH 9, #, @, $, -, AND .                   */
/*                                                                    */
/*--------------------------------------------------------------------*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define XPUVENT 100
/* void dynwdsnx(struct XPVAREA*); */
static int monthtab(||) = {
            0,                  /*   january   */
            31,                 /*   february  */
            59,                 /*   march     */
            90,                 /*   april     */
            120,                /*   may       */
            151,                /*   june      */
            181,                /*   july      */
            212,                /*   august    */
            243,                /*   september */
            273,                /*   october   */
            304,                /*   november  */
            334                 /*   december  */
} ;
static int leaptab(||) = {
            0,                   /*   january   */
            31,                  /*   february  */
            60,                  /*   march     */
            91,                  /*   april     */
            121,                 /*   may       */
            152,                 /*   june      */
            182,                 /*   july      */
            213,                 /*   august    */
            244,                 /*   september */
            274,                 /*   october   */
            305,                 /*   november  */
            335                  /*   december  */
} ;
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
struct amudsnpx {
     char       xpjobn(|8|);     /* jobname                          */
     char       xpstep(|8|);     /* stepname                         */
     char       xpdbname(|8|);   /* database name                    */
     char       xpspname(|8|);   /* space name                       */
     char       xpresume(|1|);   /* resume   (Y or N)                */
     char       xpreplace(|1|);  /* replace  (Y or N)                */
     char       xpresrv1(|2|);   /* reserved for loadplus            */
     char       xpuser(|8|);     /* user id                          */
     char       xpssid(|4|);     /* db2 subsystem id                 */
     char       xpdate(|6|);     /* utility execution date mmddyy    */
     char       xptime(|6|);     /* utility execution time hhmmss    */
     char       xputilid(|16|);  /* utility id                       */
     char       xpdate8(|8|);    /* utility execution date mmddyyyy */
     char       xpgrpnm(|4|);    /* data sharing group name          */
     char       xpvcat(|8|); /* vcat name (from 1st part if partitnd)*/
     char       xpdatej(|7|);    /* utility execution date ccyyddd   */
     char       xpresrv2(|13|);  /* reserved for loadplus            */
     int           xpusrwd1;     /* user word 1*/
     int           xpusrwd2;     /* user word 2*/
     int           xpusrwd3;     /* user word 3*/
     int           xpusrwd4;     /* user word 4*/
     int           xpusrwd5;     /* user word 5*/
     int           xpusrwd6;     /* user word 6*/
     int           xpusrwd7;     /* user word 7*/
     int           xpusrwd8;     /* user word 8*/
     struct XPUVAREA {
         char xpuvname(|9|);
         char xpuvdata(|8|);
     } xpuvars(|XPUVENT|) ;
} ;
int amuedsc (struct amudsnpx *xparm)
{
    char yr(|5|);
    char day(|4|);
    char month(|4|);
    char wuid(|17|);
    int i;
    int x;
    int l;
    int wday;
    int wyear;
    int iyear;
    int imonth;
    int totdays;
    int xdisable;
    int pfxlen;
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
    int sfxstrt;
    int sfxlen;
    struct XPUVAREA *tp;
    tp = xparm->xpuvars;
    x = l = wday = wyear = iyear = imonth = totdays = 0;
    xdisable = i = pfxlen = sfxstrt = sfxlen = 0;
/*--------------------------------------------------------------------*/
/*                                                                    */
/* to disable this exit set    xdisable = 1                           */
/*                                                                    */
/*--------------------------------------------------------------------*/
    if (xdisable)
       return(0);
/*--------------------------------------------------------------------*/
/*                                                                    */
/* CREATE USER-DEFINED VARIABLES                                      */
/*                                                                    */
/* THE FOLLOWING USER VARIABLES WILL BE CREATED:                      */
/*                                                                    */
/*  _JDATE     - JULIAN DATE IN THE FORM OF DYYDDD BUILT FROM THE     */
/*               LOADPLUS SUPPLIED VARIABLE &DATE8                    */
/*                                                                    */
/*  _JCDATE    - JULIAN DATE IN THE FORM OF DYYYYDDD BUILT FROM THE   */
/*               LOADPLUS SUPPLIED VARIABLE &DATE8                    */
/*                                                                    */
/*  _UTILPFX   - UP TO 8 BYTES OF THE FIRST NODE OF THE UTILITY ID    */
/*                                                                    */
/*  _UTILSFX   - UP TO 8 BYTES OF THE REMAINING PORTION OF THE        */
/*               UTILITY ID                                           */
/*                                                                    */
/*--------------------------------------------------------------------*/

/*--------------------------------------------------------------------*/
/*                                                                    */
/* BUILD _JDATE AND _JCDATE USER VARIABLES                            */
/*                                                                    */
/*--------------------------------------------------------------------*/

    memcpy(month,xparm->xpdate8,2);
    memcpy(day,xparm->xpdate8+2,2);
    memcpy(yr,xparm->xpdate8+4,4);
    sscanf(yr,"%d",&wyear);
    sscanf(day,"%d",&wday);
    sscanf(month,"%d",&imonth);
    imonth = imonth - 1;
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
    if (wyear % 4 == 0 && wyear % 100 != 0 || wyear % 400 == 0)
       totdays = wday + leaptab(|imonth|);
    else
       totdays = wday + monthtab(|imonth|);
    sprintf(yr,"%d",wyear);
    sprintf(day,"%03.03d",totdays);
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memcpy(tp->xpuvname,"_JDATE", 6);
    memcpy(tp->xpuvdata,"D",1);
    memcpy(tp->xpuvdata+1,xparm->xpdate8+6,2);
    memcpy(tp->xpuvdata+3,day,3);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"D%-2.2s%3.3s",xparm->xpdate8+6,day); */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_JDATE' HAS INVALID DATA                */
/*-------------------------------------------------------------------*/
    tp++;
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
    memcpy(tp->xpuvname,"_JCDATE", 7);
    memcpy(tp->xpuvdata,"D",1);
    memcpy(tp->xpuvdata+1,yr,4);
    memcpy(tp->xpuvdata+5,day,3);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"D%-4.4s%-3.3s",yr,day);                    */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_JCDATE' HAS INVALID DATA               */
/*-------------------------------------------------------------------*/
/*                                                                   */
/* build _utilpfx and _utilsfx user variables                        */
/*                                                                   */
/*-------------------------------------------------------------------*/
    tp++;
    sprintf(wuid,"%16.16s",xparm->xputilid);
/*-------------------------------------------------------------------*/
/*  get the length of xputilid                                       */
/*  find the first deliminator character                             */
/*  limit _UTILPFX and UTILSFX to 8 characters -DSN node limit       */
/*-------------------------------------------------------------------*/
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
    for (i=0;wuid(|i|) != '\0'; i++)
    {
        switch (wuid(|i|))
        {
            case ' ':
            case '.':
            case '+':
            case '|':
            case ';':
            case '-':
            case '/':
            case '≠':
            case ',':
            case '_':
            case ':':
            case '=':
            case '\\':
                if (pfxlen == 0)
                {
                    pfxlen = i;
                    sfxstrt = i + 1;
                }
                break;
            default:
                break;
       }
    }
    if (pfxlen == 0)             /* no delimiter found */
    {
        pfxlen = 8;
        sfxstrt = 8;
    }
    sfxlen = i - sfxstrt;
    if (pfxlen > 8)
        pfxlen = 8;
    if (sfxlen > 8)
        sfxlen = 8;
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
    memcpy(tp->xpuvname, "_UTILPFX", 8);
    memcpy(tp->xpuvdata, xparm->xputilid,  pfxlen);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"%-8.8s",xparm->xputilid);                  */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_UTILPFX' HAS INVALID DATA              */
/*-------------------------------------------------------------------*/
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
    tp++;
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
    memcpy(tp->xpuvname, "_UTILSFX", 8);
    memcpy(tp->xpuvdata, xparm->xputilid+sfxstrt, sfxlen);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"%-8.8s",xparm->xputilid+8);                */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_UTILSPX' HAS INVALID DATA              */
/*-------------------------------------------------------------------*/
    return (0);
}
```

LOADPLUS LE C User Exit

This section provides a sample exit parameter structure, variable mapping structure, and exit that you can use to write a user exit in LE C.

LE C Exit Parameter Structure

This structure (Figure F-10) contains both input and output fields. Input fields pass vital information to the user exit, such as the database name and user ID. If you make any modifications to these input fields, LOADPLUS disregards them on return. The output fields pass information about your user variables back to LOADPLUS.

Figure F-10 LE C Exit Parameter Structure

```
struct amudsnxp {
     char       xpjobn[8];       /* jobname                        */
     char       xpstep[8];       /* stepname                       */
     char       xpdbname[8];     /* database name                  */
     char       xpspname[8];     /* space name                     */
     char       xpresume[1];     /* resume  (Y or N)               */
     char       xpreplace[1];    /* replace (Y or N)               */
     char       xpresrv1[2];     /* reserved for loadplus          */
     char       xpuser[8];       /* user id                        */
     char       xpssid[4];       /* db2 subsystem id               */
```

(continued on following page)

(continued from preceding page)

```
char       xpdate[6];       /* utility execution date mmddyy   */
char       xptime[6];       /* utility execution time hhmmss   */
char       xputilid[16];    /* utility id                      */
char       xpdate8[8];      /* utility execution date mmddyyyy */
char       xpgrpnm[4];      /* data sharing group name         */
char       xpvcat[8];  /* vcat name (from 1st part if partitnd)*/
char       xpdatej[7];      /* utility execution date ccyyddd  */
char       xpresrv2[13];    /* reserved for load plus          */
int           xpusrwd1;     /* user word 1*/
int           xpusrwd2;     /* user word 2*/
int           xpusrwd3;     /* user word 3*/
int           xpusrwd4;     /* user word 4*/
int           xpusrwd5;     /* user word 5*/
int           xpusrwd6;     /* user word 6*/
int           xpusrwd7;     /* user word 7*/
int           xpusrwd8;     /* user word 8*/
...
```

Structure Fields

The major structure fields and their uses are as follows:

xpjobn contains the job name, up to 8 bytes

xpstep contains the step name, up to 8 bytes

xpdbname contains the database name, up to 8 bytes

xpspname contains the name of the table space or index space from the LOAD command, up to 8 bytes

xpresume contains an indication of whether a LOAD RESUME is being performed—Y (yes) or N (no)

xpreplace contains an indication of whether a LOAD REPLACE is being performed—Y (yes) or N (no)

xpuser contains the user ID of the user running the LOADPLUS utility, up to 8 bytes

xpssid contains the DB2 subsystem ID, 4 bytes

xpdate contains the date of the execution of the utility, in the format MMDDYY, 6 bytes

xptime contains the time of the execution of the utility, in the format HHMMSS, 6 bytes

BMC Software, Inc., Confidential and Proprietary Information

xputilid contains the utility ID, up to 16 bytes

xpdate8 contains the date of the execution of the utility, in the format MMDDYYYY, 8 bytes

xpgrpnm contains the DB2 data-sharing group name

In a nondata-sharing environment, the field contains the DB2 subsystem ID.

xpvcat contains the VCATNAME specified in the DB2 catalog for the table space being loaded, or for the first partition if the table space is partitioned

xpdatej contains the Julian date of the execution of the utility, in the format CCYYDDD, 7 bytes

xpusrwd1...xpusrwd8 provides work space, 8 parameters, up to 4 bytes each

LE C Exit Variable Mapping Structure

This structure (Figure F-11) defines the output area of the exit parameter structure. Using this structure allows you to easily address the variable definition table. The number of entries in this table (each entry consists of both a variable name and its value) must not exceed 100.

Figure F-11 LE C Exit Variable Mapping Structure

```
struct XPUVAREA {
    char xpuvname[9];
    char xpuvdata[8];
} xpuvars[XPUVENT] ;
```

Structure Fields

The major structure fields and their uses are as follows:

XPUVAREA defines the area that contains user-defined variable information

See page F-4 for details about establishing user-defined variables.

xpuvname the name of the variable that you are defining

xpuvdata the value of the variable that you are defining

BMC Software, Inc., Confidential and Proprietary Information

XPUVENT equates to the maximum number of entries in the output variable area

There is a limit of 100 entries.

Sample LE C User Exit

Figure F-12 is a sample user exit written in LE C.

Figure F-12 Sample LE C User Exit

```
/*-------------------------------------------------------------------*/
/*                       D I S C L A I M E R                         */
/*-------------------------------------------------------------------*/
/*                                                                   */
/* THIS IS A SAMPLE LOADPLUS USER EXIT.  THIS EXIT WOULD BE USED     */
/* IN ORDER TO DEFINE USER VARIABLES FOR BUILDING DATASET NAMES FOR  */
/* DYNAMIC WORKFILE ALLOCATION WITH THE DSNPAT KEYWORD AND FOR       */
/* SHRLEVEL CHANGE WITH THE SPILLDSNPAT KEYWORD.                     */
/*                                                                   */
/* THIS EXIT WILL ONLY BE INVOKED WHEN IT IS SPECIFICALLY NAMED      */
/* IN THE AMU$OPTS DSNUEXIT PARAMETER OR IN THE DSNUEXIT PARAMETER OF */
/* THE LOAD COMMAND.  THE SPECIFICATION IN THE LOAD COMMAND WILL     */
/* OVERRIDE THE SPECIFICATION IN THE AMU$OPTS MACRO.  IF THE NAME    */
/* SPECIFIED IS 'NONE' WITHOUT THE SINGLE QUOTE MARKS, THE EXIT WILL */
/* NOT BE INVOKED.                                                   */
/*                                                                   */
/* NOTE: PLEASE REVIEW THE DOCUMENTATION IN THE REFERENCE MANUAL,    */
/* AND THE FOLLOWING USAGE NOTES PRIOR TO IMPLEMENTING THIS EXIT.    */
/*                                                                   */
/* PLEASE CALL BMC SOFTWARE PRODUCT SUPPORT WITH ANY QUESTIONS YOU   */
/* MAY HAVE IN THIS AREA.                                            */
/*                                                                   */
/*               PHONE: 1-800-537-1813                               */
/*                                                                   */
/*-------------------------------------------------------------------*/
/*                           N O T E S                               */
/*-------------------------------------------------------------------*/
/*                                                                   */
/* AMUEDSL IS CALLED ONCE AT THE BEGINNING OF EITHER THE PRELOAD     */
/* PHASE OR THE COMBINED PHASE.                                      */
/*                                                                   */
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
/* WHEN INVOKED IT IS PASSED THE ADDRESS OF A USER EXIT BLOCK         */
/*                  DESCRIBED BY THE AMUDSNXP STRUCT.                 */
/*                  THIS CALL IS FROM ASEMBLER MODULE DYNWDSNX         */
/*                                                                    */
/* UPON EXIT    RETRUN CODE FROM THIS EXIT                            */
/*                                                                    */
/*                      0     GOOD RETURN                             */
/*                                                                    */
/*                      OTHER ERROR RETURN, TERMINATE UTILITY         */
/*                                                                    */
/* AMUDSNXP STRUCT  THIS STRUCT CONTAINS BOTH INPUT                   */
/*                  AND OUTPUT FIELDS.  INPUT FIELDS ARE PROVIDED TO  */
/*                  PASS VITAL INFORMATION TO THIS EXIT.  ANY MODIFI- */
/*                  CATIONS TO THESE FIELDS WILL BE DISREGARDED UPON  */
/*                  RETURN TO THE CALLER.  THE OUTPUT FIELDS ARE      */
/*                  PROVIDED TO ALLOW THE EXIT TO PASS INFORMATION    */
/*                  BACK TO THE CALLER.                               */
/*                                                                    */
/*--------------------------------------------------------------------*/
/*                                                                    */
/*--------------------------------------------------------------------*/
/*                                                                    */
/* XPUVARS DEFINES THE OUTPUT AREA OF THE AMUDSNXP STRUCT AT TAG      */
/* XPUVAREA. YOU MAY USE THIS STRUCT TO EASILY REFERENCE THE ARRAY    */
/* DEFINED AT XPUVAREA.                                               */
/*                                                                    */
/* NOTE:                                                              */
/*                                                                    */
/*  THE NUMBER OF ENTRIES IN THE XPUVAREA ARRAY MUST NOT EXCEED 100   */
/*  OR YOU WILL ADDRESS BEYOND THE END OF THE STRUCTURE STORAGE       */
/*  PROVIDED FOR THIS ROUTINE.                                        */
/*                                                                    */
/* VARIABLE NAME:                                                     */
/*                                                                    */
/*  1.  MUST BEGIN WITH AN UNDERSCORE CHARACTER C'_'  X'6D'           */
/*  2.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVNAME FIELD             */
/*  3.  IF LESS THAN 9 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS */
/*  4.  MAY NOT CONTAIN EMBEDDED BLANKS                               */
/*  5.  NON-BLANK CHARACTERS AFTER THE LEFTMOST BYTE MUST BE NATIONAL */
/*      CHARACTERS WHICH ARE A THROUGH Z, 0 THROUGH 9, #, @, AND $    */
/*  6.  MUST BE UPPER CASE.                                           */
/*                                                                    */
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
/* VARIABLE DATA:                                                    */
/*                                                                   */
/*  1.  MUST BEGIN IN THE LEFTMOST BYTE OF XPUVDATA FIELD            */
/*  2.  IF LESS THAN 8 BYTES, MUST BE PADDED ON THE RIGHT WITH BLANKS */
/*  3.  MAY NOT CONTAIN EMBEDDED BLANKS                              */
/*  4.  NON-BLANK CHARACTERS MUST BE 'NATIONAL' CHARACTERS WHICH ARE  */
/*      A THROUGH Z, 0 THROUGH 9, #, @, $, -, AND .                  */
/*                                                                   */
/*-------------------------------------------------------------------*/
#pragma runopts(PLIST(HOST))
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define XPUVENT 100
static int monthtab[] = {
             0,                  /*   january   */
             31,                 /*   february  */
             59,                 /*   march     */
             90,                 /*   april     */
             120,                /*   may       */
             151,                /*   june      */
             181,                /*   july      */
             212,                /*   august    */
             243,                /*   september */
             273,                /*   october   */
             304,                /*   november  */
             334                 /*   december  */
} ;
static int leaptab[] = {
             0,                   /*   january   */
             31,                  /*   february  */
             60,                  /*   march     */
             91,                  /*   april     */
             121,                 /*   may       */
             152,                 /*   june      */
             182,                 /*   july      */
             213,                 /*   august    */
             244,                 /*   september */
             274,                 /*   october   */
             305,                 /*   november  */
             335                  /*   december  */
} ;
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
struct amudsnxp {
     char       xpjobn[8];       /* jobname                          */
     char       xpstep[8];       /* stepname                         */
     char       xpdbname[8];     /* database name                    */
     char       xpspname[8];     /* space name                       */
     char       xpresume[1];     /* resume  (Y or N)                 */
     char       xpreplace[1];    /* replace (Y or N)                 */
     char       xpresrv1[2];     /* reserved for loadplus            */
     char       xpuser[8];       /* user id                          */
     char       xpssid[4];       /* db2 subsystem id                 */
     char       xpdate[6];       /* utility execution date mmddyy    */
     char       xptime[6];       /* utility execution time hhmmss    */
     char       xputilid[16];    /* utility id                       */
     char       xpdate8[8];      /* utility execution date mmddyyyy */
     char       xpgrpnm[4];      /* data sharing group name          */
     char       xpvcat[8];  /* vcat name (from 1st part if partitnd)*/
     char       xpdatej[7];      /* utility execution date ccyyddd  */
     char       xpresrv2[13];    /* reserved for load plus           */
     int           xpusrwd1;     /* user word 1*/
     int           xpusrwd2;     /* user word 2*/
     int           xpusrwd3;     /* user word 3*/
     int           xpusrwd4;     /* user word 4*/
     int           xpusrwd5;     /* user word 5*/
     int           xpusrwd6;     /* user word 6*/
     int           xpusrwd7;     /* user word 7*/
     int           xpusrwd8;     /* user word 8*/
     struct XPUVAREA {
         char xpuvname[9];
         char xpuvdata[8];
     } xpuvars[XPUVENT] ;
} ;
int main    (int argc, char *argv[])
{
    char yr[5];
    char day[4];
    char month[4];
    char wuid[17];
    int i;
    int x;
    int l;
    int wday;
    int wyear;
    int iyear;
    int imonth;
    int totdays;
    int xdisable;
    int pfxlen;
```

(continued on following page)

(continued from preceding page)

```
    int sfxstrt;
    int sfxlen;
    struct XPUVAREA *tp;
    struct amudsnxp *xparm;
    xparm = (void *)argv[1];
    tp = xparm->xpuvars;
    x = l = wday = wyear = iyear = imonth = totdays = 0;
    xdisable = i = pfxlen = sfxstrt = sfxlen = 0;
/*--------------------------------------------------------------------*/
/*                                                                    */
/* to disable this exit set    xdisable = 1                           */
/*                                                                    */
/*--------------------------------------------------------------------*/
    if (xdisable)
       return(0);
/*--------------------------------------------------------------------*/
/*                                                                    */
/* CREATE USER-DEFINED VARIABLES                                      */
/*                                                                    */
/* THE FOLLOWING USER VARIABLES WILL BE CREATED:                      */
/*                                                                    */
/*  _JDATE     - JULIAN DATE IN THE FORM OF DYYDDD BUILT FROM THE     */
/*               LOADPLUS SUPPLIED VARIABLE &DATE8                    */
/*                                                                    */
/*  _JCDATE    - JULIAN DATE IN THE FORM OF DYYYYDDD BUILT FROM THE   */
/*               LOADPLUS SUPPLIED VARIABLE &DATE8                    */
/*                                                                    */
/*  _UTILPFX   - UP TO 8 BYTES OF THE FIRST NODE OF THE UTILITY ID    */
/*                                                                    */
/*  _UTILSFX   - UP TO 8 BYTES OF THE REMAINING PORTION OF THE        */
/*               UTILITY ID                                           */
/*                                                                    */
/*--------------------------------------------------------------------*/

/*--------------------------------------------------------------------*/
/*                                                                    */
/* BUILD _JDATE AND _JCDATE USER VARIABLES                            */
/*                                                                    */
/*--------------------------------------------------------------------*/

    memcpy(month,xparm->xpdate8,2);
    memcpy(day,xparm->xpdate8+2,2);
    memcpy(yr,xparm->xpdate8+4,4);
    sscanf(yr,"%d",&wyear);
    sscanf(day,"%d",&wday);
    sscanf(month,"%d",&imonth);
    imonth = imonth - 1;
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
    if (wyear % 4 == 0 && wyear % 100 != 0 || wyear % 400 == 0)
       totdays = wday + leaptab[imonth];
    else
       totdays = wday + monthtab[imonth];
    sprintf(yr,"%d",wyear);
    sprintf(day,"%03.03d",totdays);
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memcpy(tp->xpuvname,"_JDATE", 6);
    memcpy(tp->xpuvdata,"D",1);
    memcpy(tp->xpuvdata+1,xparm->xpdate8+6,2);
    memcpy(tp->xpuvdata+3,day,3);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"D%-2.2s%3.3s",xparm->xpdate8+6,day); */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_JDATE' HAS INVALID DATA                */
/*-------------------------------------------------------------------*/
    tp++;
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
    memcpy(tp->xpuvname,"_JCDATE", 7);
    memcpy(tp->xpuvdata,"D",1);
    memcpy(tp->xpuvdata+1,yr,4);
    memcpy(tp->xpuvdata+5,day,3);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"D%-4.4s%-3.3s",yr,day);                    */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_JCDATE' HAS INVALID DATA               */
/*-------------------------------------------------------------------*/
/*                                                                   */
/* build _utilpfx and _utilsfx user variables                        */
/*                                                                   */
/*-------------------------------------------------------------------*/
    tp++;
    sprintf(wuid,"%16.16s",xparm->xputilid);
/*-------------------------------------------------------------------*/
/*  get the length of xputilid                                       */
/*  find the first deliminator character                             */
/*  limit _UTILPFX and UTILSFX to 8 characters -DSN node limit       */
/*-------------------------------------------------------------------*/
```

(continued on following page)

(continued from preceding page)

```
    for (i=0;wuid[i] != '\0'; i++)
    {
        switch (wuid[i])
        {
            case ' ':
            case '.':
            case '+':
            case '|':
            case ';':
            case '-':
            case '/':
            case '≠':
            case ',':
            case '_':
            case ':':
            case '=':
            case '\\':
                if (pfxlen == 0)
                {
                    pfxlen = i;
                    sfxstrt = i + 1;
                }
                break;
            default:
                break;
       }
    }
    if (pfxlen == 0)             /* no delimiter found */
    {
        pfxlen = 8;
        sfxstrt = 8;
    }
    sfxlen = i - sfxstrt;
    if (pfxlen > 8)
        pfxlen = 8;
    if (sfxlen > 8)
        sfxlen = 8;
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
    memcpy(tp->xpuvname, "_UTILPFX", 8);
    memcpy(tp->xpuvdata, xparm->xputilid,  pfxlen);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"%-8.8s",xparm->xputilid);                  */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_UTILPFX' HAS INVALID DATA              */
```

(continued on following page)

BMC Software, Inc., Confidential and Proprietary Information

(continued from preceding page)

```
/*-------------------------------------------------------------------*/
    tp++;
/*-------------------------------------------------------------------*/
/*   clear the user variable area to blanks                          */
/*-------------------------------------------------------------------*/
    memset(tp->xpuvname,' ', sizeof(struct XPUVAREA));
    memcpy(tp->xpuvname, "_UTILSFX", 8);
    memcpy(tp->xpuvdata, xparm->xputilid+sfxstrt, sfxlen);
/*-------------------------------------------------------------------*/
/*  sprintf(tp->xpuvdata,"%-8.8s",xparm->xputilid+8);                */
/*  do not use this format as the '\0' used by sprintf will cause    */
/*  BMC51239I USER VARIABLE '_UTILSPX' HAS INVALID DATA              */
/*-------------------------------------------------------------------*/
    return (0);
}
```

BMC Software, Inc., Confidential and Proprietary Information

Index

Symbols

Numerics

A

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

B

C

BMC Software, Inc., Confidential and Proprietary Information

D

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

E

F

G

H

BMC Software, Inc., Confidential and Proprietary Information

I

BMC Software, Inc., Confidential and Proprietary Information

J

K

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

L

BMC Software, Inc., Confidential and Proprietary Information

M

N

BMC Software, Inc., Confidential and Proprietary Information

O

P

BMC Software, Inc., Confidential and Proprietary Information

BMC Software, Inc., Confidential and Proprietary Information

Q

R

BMC Software, Inc., Confidential and Proprietary Information

S

BMC Software, Inc., Confidential and Proprietary Information

T

BMC Software, Inc., Confidential and Proprietary Information

U

BMC Software, Inc., Confidential and Proprietary Information

V

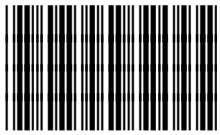
W

BMC Software, Inc., Confidential and Proprietary Information

X

Z

Notes



41072